Министерство науки и высшего образования Российской Федерации

ФГБОУ ВО «Кубанский государственный технологический университет»

Кафедра информационных систем и программирования

**Развертывание и интеграция программных систем**

Отчет по лабораторной работе №8

**Классические методы интеграции. Вариант интеграции на уровне платформ (платформы интеграции)**

Выполнил:

студент 4 курса

группы 21-КБ-ПР1

Агниев С.В.

Проверил:

проф.кафедры ИСП

Зайков В.П.

Краснодар

2025

**Цель лабораторной работы:** Рассмотреть применение возможных методов интеграции информационных подсистем, ПО которых работает на различных ОС.

**Вариант** **№01:** Информационная система предприятия оптово-розничной торговли комплектующими монтажных систем и отопления.

**Задание:** Общая постановка задачи: необходимо интегрировать N информационных подсистем систем, которые характеризуются следующим:

* имеются программные продукты с несовместимыми лицензиями;
* наблюдается существенная технологическая и концептуальная разница в ПО;
* продукты в составе ПО работают на различных платформах.

**Выполнение работы**

1. В соответствии со своим вариантом предприятия:

* поставить цель и задачи интеграции (увеличение прибыли, снижение издержек, сокращение трудозатрат для выполнения задач и др.);
* рассмотреть применение возможных методов интеграции информационных подсистем (программных продуктов) в единую систему;
* выделить основные факторы, влияющие (положительно и отрицательно) на процесс интеграции в информационной системе предприятия.

Привести по одному-двум примерам реализации того или иного метода.

**Цели и задачи интеграции**

**Цели интеграции:**

* **Увеличение прибыли:** Оптимизация процессов может привести к росту доходов.
* **Снижение издержек:** Автоматизация рутинных задач сокращает затраты.
* **Сокращение трудозатрат:** Упрощение взаимодействия между системами позволяет снизить нагрузку на сотрудников.

**Задачи интеграции:**

* Объединение данных из различных источников.
* Обеспечение совместимости программных продуктов.
* Оптимизация бизнес-процессов через автоматизацию.

**Методы интеграции информационных подсистем**

Существуют различные методы интеграции, каждый из которых имеет свои преимущества и недостатки:

1. **Горизонтальная интеграция:** Для горизонтальной интеграции необходимо подключить все системы (1С Бухгалтерия, 1С Управление торговлей, Складская система, Bitrix) к корпоративной сервисной шине (например, Apache Camel или MuleSoft). Разрабатываются адаптеры для каждой системы, обеспечивающие преобразование данных в унифицированный формат (JSON или XML). Настраиваются маршруты передачи данных через REST API или SOAP, а также правила обработки событий. После этого проводится тестирование корректности обмена данными, настройка мониторинга через инструменты шины и обучение персонала работе с новой системой.
2. **Вертикальная интеграция:** Для вертикальной интеграции настраивается взаимодействие между уровнями систем (например, передача данных о заказах из 1С Управление торговлей в 1С Бухгалтерию). Используются API для прямого обмена данными между подсистемами. Разрабатывается архитектура решения с учетом бизнес-логики и требований к данным. Реализуются процессы синхронизации: данные преобразуются в нужный формат и передаются по заданным сценариям. После настройки проводится тестирование всех уровней взаимодействия, настройка мониторинга и оптимизация производительности.

**Основные факторы, влияющие на процесс интеграции**

**Положительные факторы:**

* Стандартизация бизнес-процессов: Наличие четко прописанных и стандартизированных бизнес-процессов в компании способствует успешной интеграции. Например, если все отделы (продаж, логистики, бухгалтерии) следуют единым стандартам обработки заказов и учета товаров, это упрощает обмен данными между системами и снижает вероятность ошибок. Стандартизация позволяет легко настроить интеграцию и обеспечить согласованность данных.

**Отрицательные факторы:**

* Разнообразие бизнес-процессов: Если в разных подразделениях компании существуют различные подходы к выполнению одних и тех же задач, это может затруднить интеграцию. Например, если отдел продаж использует одну систему для учета клиентов, а склад — другую для управления запасами, несоответствие процессов может привести к путанице и ошибкам при передаче данных между системами.
* Нехватка четких процедур: Отсутствие четко прописанных процедур может стать серьезным препятствием для успешной интеграции. Например, если в компании не установлены ясные правила по обработке возвратов или обмену товара, это может вызвать сложности при попытке интегрировать системы учета и CRM. Непонимание того, как должны выполняться процессы, может привести к неэффективности и конфликтам между системами.

2. Определить назначение и основные характеристики интеграционной платформы.

**Назначение и характеристики интеграционной платформы**

Интеграционная платформа служит для объединения различных информационных систем в единую экосистему. Основные характеристики:

* **Масштабируемость:** Возможность добавления новых систем без значительных затрат.
* **Безопасность:** Защита данных при передаче между системами.
* **Гибкость:** Поддержка различных форматов данных и протоколов.

3. Описать работу интеграционной платформы, которую можно применить на рассматриваемом предприятии (например, платформы интеграции API, Azure или другой). Описание произвести на основе пяти этапов создания и работы интеграционной платформы (приведенных выше), с иллюстрацией примерами.

**Этап 1. Идентификация источников данных**

На этом этапе определяются все источники данных, которые будут участвовать в интеграции:

1. **1С Бухгалтерия**: данные о счетах, платежах, налоговых документах.
2. **1С Управление торговлей**: информация о заказах, продажах, клиентах.
3. **Складская система (1С Логистика)**: данные об остатках на складах, перемещениях товаров.
4. **Bitrix**: CRM-данные о клиентах, сделках, маркетинговых активностях.

**Действия:**

* Подключение к базам данных 1С через REST API или OData.
* Настройка доступа к Bitrix через стандартный модуль интеграции и API Bitrix24.
* Определение форматов данных (JSON, XML) для передачи между системами.

**Этап 2. Проектирование и картирование рабочего процесса**

На этом этапе разрабатывается логика взаимодействия между системами:

1. **Картирование данных**:
   * Поля клиентов из Bitrix сопоставляются с контрагентами в 1С.
   * Номенклатура товаров из 1С Управление торговлей синхронизируется с каталогом в Bitrix.
   * Остатки товаров из Складской системы передаются в CRM для отображения доступности.
2. **Рабочие процессы**:
   * При создании заказа в Bitrix информация автоматически передается в 1С Управление торговлей для формирования счета.
   * Обновления остатков на складе из 1С Логистика синхронизируются с CRM и системой управления торговлей.

**Инструменты Azure:**

* Использование **Azure Logic Apps** для проектирования рабочих процессов и настройки триггеров (например, обновление данных при изменении статуса заказа).
* Применение **Azure Data Factory** для ETL-процессов (извлечение, преобразование и загрузка данных).

**Этап 3. Преобразование данных и маршрутизация**

На данном этапе данные преобразуются в унифицированный формат для обеспечения совместимости между системами:

1. **Преобразования:**
   * Конвертация форматов (например, XML из 1С в JSON для Bitrix).
   * Агрегация данных (например, объединение остатков по нескольким складам).
2. **Маршрутизация:**
   * Данные о клиентах из Bitrix направляются в 1С Бухгалтерию и Управление торговлей.
   * Информация о заказах из 1С Управление торговлей передается в Складскую систему для резервирования товаров.

**Инструменты Azure:**

* Использование **Azure Functions** для выполнения преобразований данных в реальном времени.
* Применение **Event Grid** для маршрутизации событий между системами.

**Этап 4. Интеграция, выполнение и мониторинг**

На этом этапе запускаются интеграционные процессы и обеспечивается их контроль:

1. **Интеграция:**
   * Реализация обмена данными через REST API и OData-соединения.
   * Настройка периодической синхронизации через расписания в Azure Logic Apps.
2. **Мониторинг:**
   * Отслеживание успешности выполнения процессов с помощью **Azure Monitor**.
   * Настройка оповещений об ошибках (например, недоступность API или сбой передачи данных).

Пример: если при передаче заказа из Bitrix в 1С Управление торговлей возникает ошибка (например, несоответствие формата), администратор получает уведомление через Email или Microsoft Teams.

**Этап 5. Синхронизация данных и обновления**

На финальном этапе обеспечивается поддержание актуальности данных между всеми системами:

1. **Реализация синхронизации:**
   * Данные о статусах заказов обновляются одновременно в Bitrix и 1С Управление торговлей.
   * Изменения остатков на складе автоматически отображаются в CRM.
2. **Режимы синхронизации:**
   * Реальное время: обновления статусов заказов и остатков товаров.
   * Периодическое обновление: ночная синхронизация больших объемов данных (например, полная выгрузка каталога).

**Инструменты Azure:**

* Использование **Azure SQL Database** как промежуточного хранилища для временного хранения данных перед синхронизацией.
* Настройка автоматической репликации изменений через **Azure Service Bus**.

**Пример реализации**

**Сценарий:**

Менеджер создает заказ клиента в CRM Bitrix24. После этого:

* Заказ автоматически передается в 1С Управление торговлей для выставления счета.
* Складская система проверяет наличие товара и резервирует его.
* В CRM обновляется статус заказа («Ожидание оплаты»).
* После оплаты данные поступают в 1С Бухгалтерию для учета.

4. Пояснить примерами позитивное влияние интеграционной платформы на бизнес рассматриваемого предприятия.

Интеграционная платформа может значительно улучшить эффективность бизнеса:

* **Сокращение расходов на выполнение регуляторных обязательств:** Автоматизация процессов учета и отчетности позволяет значительно снизить затраты на соблюдение регуляторных норм. Например, в компании, занимающейся оптово-розничной торговлей комплектующими для монтажа систем отопления, интеграция 1С Бухгалтерия с другими системами позволяет автоматически генерировать необходимые отчеты для налоговых органов, что исключает необходимость ручного ввода данных и снижает вероятность ошибок.
* **Снижение рисков:** Централизация данных на интеграционной платформе уменьшает вероятность ошибок, связанных с дублированием информации. Например, если все данные о клиентах и заказах хранятся в единой системе, это позволяет избежать ситуаций, когда информация о заказе в CRM не совпадает с данными на складе. Это минимизирует риски потерь и недовольства клиентов из-за неверной информации.
* **Обеспечение организационных слияний и поглощений:** Интеграционная платформа облегчает процесс объединения систем при слиянии компаний. Например, если компания по продаже комплектующих для монтажа систем отопления приобретает другую фирму, наличие интеграционной платформы позволяет быстро объединить базы данных и системы учета, что упрощает управление и минимизирует время на интеграцию.
* **Создание целостного представления клиентской базы:** Интеграция различных систем позволяет создать единую базу данных клиентов, что улучшает анализ данных о покупателях. Например, если информация о клиентах из CRM Bitrix и 1С Управление торговлей объединена, компания может более точно сегментировать клиентов и разрабатывать персонализированные предложения, что повышает уровень обслуживания и увеличивает продажи.
* **Обеспечение разумных инвестиций в технологии модернизации:** Интеграционная платформа позволяет оптимизировать затраты на обновление технологий. Например, вместо того чтобы обновлять каждую систему отдельно, компания может инвестировать в единую интеграционную платформу, которая будет поддерживать все существующие системы. Это позволяет сократить общие расходы на ИТ-инфраструктуру и повысить эффективность работы всех подразделений.

5. Рассчитать стоимость платформенной интеграции. В качестве обязательных статей калькуляции использовать затраты на оплату труда, приобретение (дополнительно) аппаратных средств, программных ресурсов, амортизационные отчисления. Обосновать выбор значений статей калькуляции.

**1. Затраты на оплату труда**

Для выполнения интеграционного проекта потребуется команда специалистов, включая разработчиков, аналитиков и проектных менеджеров. Средняя зарплата ИТ-специалиста в России составляет около 100 000 рублей в месяц. Предположим, что проект займет 3 месяца и потребуется 5 специалистов.

100000×5×3=1500000

**2. Приобретение аппаратных средств**

При использовании облачной платформы Azure затраты на аппаратные средства могут быть минимальными, но все же предполагается необходимость в серверах для локального хранения данных и резервного копирования. Ориентировочная стоимость оборудования составит около 300 000 рублей.

**3. Приобретение программных ресурсов**

Лицензии на программное обеспечение (например, для использования API или дополнительных модулей для 1С) могут составлять значительную часть бюджета. Ориентировочная стоимость лицензий на ПО может составить около 200 000 рублей.

**4. Амортизационные отчисления**

Амортизация оборудования и программного обеспечения может составлять примерно 20% от общей стоимости активов в год. Если общая стоимость аппаратных средств и ПО составляет 500 000 рублей, то амортизационные отчисления за первый год составят:

500000×0.2=100000

**Общая стоимость интеграции**

Суммируя все статьи калькуляции, получаем:

1500000+300000+200000+100000=2100000

**Итоговая сумма**

Таким образом, общая стоимость платформенной интеграции информационных систем на базе Microsoft Azure составляет **2,1 миллиона рублей**.

6. Привести примеры (написать) интерфейсов для решения задачи интеграции в информационной системе рассматриваемого предприятия.

Примеры интерфейсов для решения задачи интеграции

Интеграция информационных систем, таких как 1С Бухгалтерия, 1С Управление торговлей, Складская система (1С Логистика) и Bitrix, может быть реализована с использованием различных интерфейсов. Ниже представлены примеры интерфейсов, которые могут быть использованы для успешной интеграции этих систем.

1. **REST API**

**Описание:** REST (Representational State Transfer) — это архитектурный стиль, который использует HTTP-протокол для обмена данными. REST API позволяет системам взаимодействовать друг с другом через стандартные HTTP-запросы (GET, POST, PUT, DELETE).

**Пример использования:**

* **1С Бухгалтерия и Bitrix:** С помощью REST API можно передавать данные о платежах и счетах из 1С Бухгалтерия в Bitrix. Например, при создании нового платежа в 1С автоматически отправляется запрос на создание соответствующей записи в CRM Bitrix.

2. **SOAP API**

**Описание:** SOAP (Simple Object Access Protocol) — это протокол обмена сообщениями, который позволяет обмениваться структурированными данными между системами. Он использует XML для форматирования сообщений и может работать через различные протоколы передачи данных.

**Пример использования:**

* **Интеграция 1С Управление торговлей и Складская система:** SOAP API может быть использован для передачи данных о заказах между этими системами. Например, при создании нового заказа в 1С Управление торговлей информация о товаре и количестве может быть отправлена в Складскую систему для резервирования.

3. **Webhook**

**Описание:** Webhook — это механизм, позволяющий одной системе отправлять данные другой системе в реальном времени по мере возникновения событий. Это асинхронный способ интеграции, который позволяет избежать постоянных запросов к серверу.

**Пример использования:**

* **Bitrix и 1С Бухгалтерия:** При изменении статуса заказа в Bitrix (например, переход в статус «Оплачен») Webhook может отправить уведомление в 1С Бухгалтерия для автоматического создания соответствующей записи о приходе платежа.