Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования «Пермский государственный аграрно-технологический университет имени академика Д.Н. Прянишникова»

Факультет экономики и информационных технологий

Кафедра Информационных технологий и программной инженерии

**ОТЧЁТ**

**о технологической практике**

Автор отчёта:

студент группы Пим-2022

направления 09.04.03

«Прикладная информатика»

Фадеев Илья Алексеевич

Руководитель практики от университета:

доцент кафедры ИТиПИ,

к.т.н, доцент

Беляков Андрей Юрьевич

Пермь, 2023

Оглавление

[Введение 3](#_Toc145926637)

[1. Постановка проблемы исследования 5](#_Toc145926638)

[2. Анализ технологий проектирования 7](#_Toc145926639)

[3. Методика проведения испытаний 9](#_Toc145926640)

[4. Разработка и тестирование прототипа информационной системы 12](#_Toc145926641)

[Заключение 19](#_Toc145926642)

[Список литературы 21](#_Toc145926643)

Введение

В данном исследовании рассматривается переход с использования SSIS (SQL Server Integration Services) на Apache NiFi в рамках работы с интеграцией данных. Область применения этих технологий широко распространена, и существуют определенные проблемы при работе с данными в отрасли или на конкретном производстве. Некоторые из них включают сложность поддержки и масштабирования традиционных интеграционных систем, ограниченные возможности для обработки и перемещения больших объемов данных, а также сложность внедрения новых и более гибких архитектур интеграции. (Kavika Roy, 2021)

Целью учебной технологической практики является разработка методики проведения испытаний предлагаемого технического решения и оценка прототипа разработанной информационной системы. Практика будет осуществляться с использованием Apache NiFi в качестве основной технологии интеграции данных.

Задачи учебной технологической практики включают:

- Постановку проблемы исследования в контексте перехода с SSIS на Apache NiFi. В данной задаче будет проведен анализ существующих проблем и ограничений, связанных с использованием SSIS, а также определена актуальность перехода на Apache NiFi.

- Анализ существующих технических решений. В этой задаче будет исследовано текущее состояние интеграционных систем, а также рассмотрены возможные альтернативные технологии, включая Apache NiFi. Будет проведено сравнение и анализ их возможностей и преимуществ.

- Обоснование выбора Apache NiFi в качестве технологии для проектирования системы интеграции данных. В рамках этой задачи будет проведен обзор особенностей Apache NiFi, изучены его преимущества и недостатки, а также обосновано решение о его использовании.

- Описание методики проведения испытаний эффективности разработки в сравнении с использованием SSIS. В данной задаче будет разработана методика сравнительного анализа двух технологий на основе заданных критериев эффективности, а также определены способы проведения испытаний и оценки результатов.

- Разработка прототипа системы на базе Apache NiFi и тестирование его отдельных частей. В этой задаче будет осуществлена разработка информационной системы с использованием Apache NiFi в качестве основной технологии. Также будут проведены тесты и оценка эффективности прототипа.

Это исследование позволит оценить преимущества перехода с SSIS на Apache NiFi в контексте интеграции данных. Результаты практической части проекта предоставят рекомендации по использованию Apache NiFi в рамках интеграционных проектов и позволят выявить его сильные и слабые стороны.

1. Постановка проблемы исследования

1. Постановка проблемы исследования

В современном информационном мире организации сталкиваются с рядом сложностей при работе с интеграцией данных.

Целью данного исследования является изучение возможности перехода с использования SSIS на более современную и гибкую технологию - Apache NiFi. Apache NiFi является открытым инструментом для интеграции и обработки данных, который предлагает богатый набор возможностей. (dataenginer.ru, n.d.)  
Главной гипотезой исследования является то, что переход на Apache NiFi может привести к улучшению процесса интеграции данных, повышению эффективности и обеспечению большей гибкости в работе с данными. (Microsoft, Официальный сайт Microsoft, 2023)

2. Анализ существующих технических решений

В данной задаче будет проведен анализ текущего состояния интеграционных систем и оценка их возможностей. Будут рассмотрены и проанализированы доступные технические решения, включая SSIS и Apache NiFi. Будет проведено сравнение их функциональности, возможностей и преимуществ.

3. Обоснование выбора Apache NiFi в качестве технологии для проектирования системы интеграции данных

В данном блоке будет представлен обоснованный выбор Apache NiFi в качестве предпочтительной технологии для проектирования системы интеграции данных. Будут рассмотрены преимущества Apache NiFi, такие как возможность обработки больших объемов данных, гибкость в настройке процессов интеграции, поддержка широкого спектра протоколов и форматов данных, а также простой и интуитивно понятный интерфейс. (dzen.ru, n.d.)

4. Описание методики проведения испытаний эффективности разработки в сравнении с использованием SSIS

В этом разделе будет описана методика проведения сравнительного анализа двух технологий - SSIS и Apache NiFi. Будут определены критерии эффективности и разработаны способы проведения испытаний. Результаты испытаний позволят оценить эффективность использования Apache NiFi в сравнении с SSIS.

5. Разработка прототипа системы на базе Apache NiFi и тестирование его отдельных частей

В последнем разделе будет проведена разработка прототипа системы на базе Apache NiFi и проведено тестирование его отдельных частей. Будут проанализированы результаты тестирования и оценена эффективность разработки, то есть прототипа системы, используя Apache NiFi.

В итоге, данный исследование позволит оценить преимущества перехода с использования SSIS на Apache NiFi в контексте интеграции данных. Полученные результаты способствуют рекомендациям использования Apache NiFi в рамках интеграционных проектов и помогут определить его сильные и слабые стороны.

2. Анализ технологий проектирования.

При проектировании системы перехода с использования SSIS на Apache NiFi необходимо провести анализ и выбрать соответствующие технологии и методы разработки. В данном разделе будет рассмотрен анализ следующих технологий проектирования, используемых в данной работе:

а) Паттерны проектирования:

Паттерны проектирования являются эффективным подходом к проектированию систем интеграции данных. В контексте перехода с SSIS на Apache NiFi, следует рассмотреть паттерны, такие как Extract, Transform, Load (ETL) ETL-паттерн используется для извлечения, преобразования и загрузки данных. (Microsoft, learn.microsoft.com, n.d.)

б) Структуры данных:

При переходе на Apache NiFi необходимо проанализировать и выбрать подходящие структуры данных для хранения и обработки информации. Важными аспектами являются эффективность доступа и обработки данных, а также возможность масштабирования системы. (ivan-shamaev.ru, б.д.)  
Apache NiFi предоставляет широкий спектр структур данных, таких как списки, хеш-таблицы, деревья и графы, что позволяет выбрать наиболее подходящую структуру для различных задач. (coderlessons, 2019)

в) Базы данных:

Одним из основных компонентов системы интеграции данных является база данных. При переходе с использования SSIS на Apache NiFi следует проанализировать и выбрать соответствующую базу данных. Apache NiFi поддерживает различные базы данных, включая SQL-серверы, NoSQL-базы данных и облачные хранилища данных. Необходимо учитывать требования к производительности, масштабируемости и доступности данных при выборе базы данных для системы интеграции. (nifi.apache.org, б.д.)

г) Форматы хранения и обработки данных (json, xml, csv):

При переходе с использования SSIS на Apache NiFi важно определить подходящие форматы хранения и обработки данных. Apache NiFi поддерживает различные форматы данных, такие как JSON, XML и CSV. Выбор форматов должен основываться на требованиях проекта и спецификации данных. Например, для структурированных данных можно использовать форматы JSON и XML, а для неструктурированных данных - формат CSV. (bigdataschool, 2023)

д) Алгоритмы обработки данных:

При переходе на Apache NiFi необходимо проанализировать и выбрать соответствующие алгоритмы обработки данных. Apache NiFi предоставляет множество встроенных алгоритмов для различных операций обработки данных, таких как фильтрация, преобразование, агрегация и т.д. Важно выбрать подходящие алгоритмы, учитывая требования проекта и объемы данных. (habr.com, б.д.)

е) Архитектура информационной системы:

При переходе с SSIS на Apache NiFi следует проанализировать и выбрать подходящую архитектуру информационной системы. Apache NiFi предлагает гибкую архитектуру, основанную на модульности и расширяемости. Можно выбрать архитектурные стили, такие как микросервисная архитектура или SOA (Service-Oriented Architecture), в зависимости от требований проекта.

В результате проведенного анализа были выбраны соответствующие технологии проектирования, такие как ETL-паттерн, базы данных SQL, форматы JSON и XML, алгоритмы обработки данных и гибкая архитектура информационной системы. Эти выбранные технологии будут использоваться в процессе перехода с использования SSIS на Apache NiFi в рамках данной работы.

3. Методика проведения испытаний

При переходе с использования SSIS на Apache NiFi важно провести испытания информационной системы для оценки ее эффективности и функциональности. В данном разделе будет представлена методика проведения испытаний, используемые инструменты и технологии испытаний, а также ожидаемые результаты и ограничения системы.

а) Критерии оценки эффективности разработки:

Для оценки эффективности разработки при переходе с SSIS на Apache NiFi следует использовать следующие критерии:

- Производительность: скорость обработки данных, пропускная способность системы и время отклика.

- Надежность: стабильность работы системы, отказоустойчивость и восстанавливаемость после сбоев.

- Масштабируемость: возможность системы эффективно работать с возрастающими объемами данных и пользовательской нагрузкой.

- Гибкость: способность системы адаптироваться к изменяющимся требованиям и интегрироваться с другими системами.

- Удобство использования: простота настройки, интуитивно понятный интерфейс и удобство работы с системой для пользователей.

б) Способы проведения испытаний и используемые инструменты:

Для проведения испытаний информационной системы на Apache NiFi можно использовать следующие методики:

- Функциональное тестирование: проверка работы основных функций системы, таких как загрузка данных, преобразование и выгрузка данных, а также проверка соответствия ожидаемых результатов.

- Нагрузочное тестирование: проверка производительности системы при различных нагрузках и объемах данных. Для этого можно использовать инструменты, такие как Apache JMeter или Gatling.

- Интеграционное тестирование: проверка взаимодействия системы с другими системами, такими как базы данных, веб-сервисы и внешние приложения.

- Тестирование на возможность восстановления: проверка работы системы после аварийных ситуаций или сбоев, проверка восстановления данных и продолжения работы системы.

- Тестирование безопасности: проверка защищенности системы от несанкционированного доступа и возможности обнаружения и реагирования на угрозы.

в) Предполагаемые результаты испытаний и успешные результаты:

Предполагаемые результаты испытаний системы при переходе с SSIS на Apache NiFi включают:

- Демонстрация корректной и эффективной обработки данных, включая извлечение, преобразование и загрузку.

- Возможность обработки больших объемов данных и высокая производительность системы.

- Проверка взаимодействия системы с другими компонентами информационной инфраструктуры.

- Достаточная надежность и отказоустойчивость системы.

- Обнаружение и реакция на возможные угрозы безопасности.

г) Потоки данных и ограничения системы:

Информационная система на Apache NiFi будет обрабатывать различные потоки данных, включая структурированные и неструктурированные данные в форматах JSON, XML и CSV. Система будет выполнять функции извлечения, преобразования и загрузки данных, а также обеспечивать взаимодействие с другими системами и хранилищами данных.

Ограничения системы включают:

- Ограничения производительности: система должна обрабатывать данные с достаточной скоростью и пропускной способностью для удовлетворения требований проекта.

- Ограничения доступности: система должна быть доступна для пользователей в течение требуемого времени.

- Ограничения безопасности: система должна обеспечивать безопасность данных и защищать их от несанкционированного доступа.

д) Технологии тестирования системы:

Для тестирования информационной системы на Apache NiFi можно использовать следующие технологии:

- Apache JMeter: инструмент для нагрузочного и функционального тестирования, позволяющий симулировать различные сценарии использования и нагрузки на систему.

- Postman: инструмент для тестирования веб-сервисов и API, позволяющий отправлять запросы и проверять ответы системы.

- OWASP ZAP: инструмент для тестирования безопасности, позволяющий обнаружить уязвимости и угрозы безопасности системы.

В результате проведения испытаний информационной системы ожидается демонстрация ее функциональности, производительности, надежности и безопасности. Успешными результатами считаются корректная и эффективная обработка данных, достаточная производительность системы, надежность работы и отсутствие уязвимостей безопасности.

4. Разработка и тестирование прототипа информационной системы

В этом разделе будет описан процесс разработки и тестирования прототипа информационной системы при переходе с использования SSIS на Apache NiFi. Будут описаны ключевые алгоритмы проекта, используемая архитектура информационной системы, а также демонстрированы скриншоты пользовательского интерфейса, результаты нагрузочного тестирования и тестирования системы.

а) Ключевые алгоритмы проекта:

При разработке прототипа информационной системы на Apache NiFi следует использовать следующие ключевые алгоритмы:

- Извлечение данных: система должна иметь возможность извлекать данные из различных источников, таких как базы данных, файлы и веб-сервисы.

- Преобразование данных: система должна обеспечивать возможность преобразования данных в нужный формат и структуру для дальнейшей обработки.

- Загрузка данных: система должна позволять загружать данные в целевые системы или хранилища данных.

- Маршрутизация данных: система должна обеспечивать гибкую маршрутизацию данных в соответствии с определенными правилами и логикой.

- Мониторинг и управление: система должна иметь возможность мониторинга и управления процессами обработки данных, а также отображения статистики и логов.

б) Архитектура информационной системы:

Архитектура информационной системы на Apache NiFi может быть организована следующим образом:

- Компоненты системы: система может включать в себя компоненты для извлечения, преобразования, загрузки и маршрутизации данных, а также компоненты для мониторинга и управления процессами.

- Паттерн проектирования: для реализации системы на Apache NiFi можно использовать паттерн проектирования "Поток данных", где каждый компонент является независимым процессором, обрабатывающим свою часть данных.

- Распределение нагрузки: система может быть развернута в кластере, где нагрузка распределяется между несколькими узлами для обеспечения масштабируемости и отказоустойчивости.

Алгоритм тестирования SQL-запросов на предмет анализ производительности:

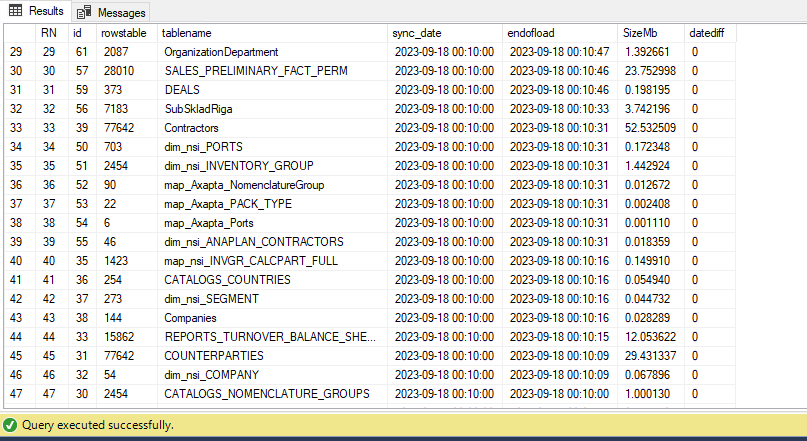


Рис 1.

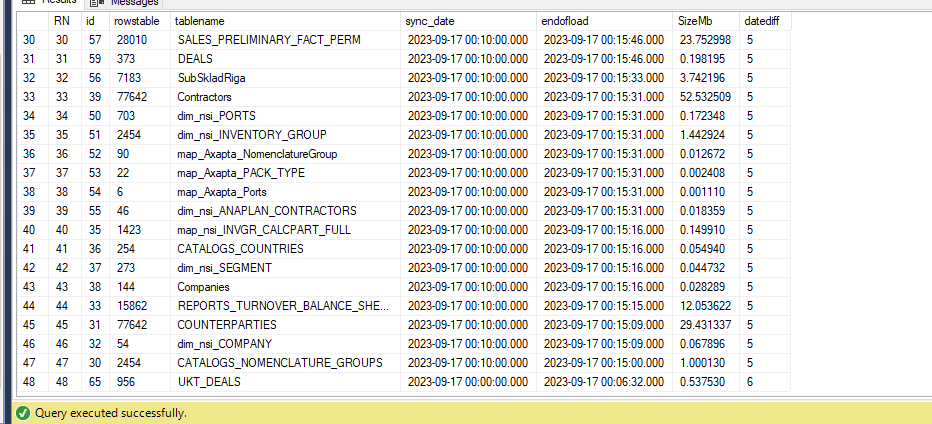


Рис 2.

На Рис 1. содержатся различные потоки данных, перенесенные с использования SSIS на Ni-Fi.  
В нём мы видим следующие данные:

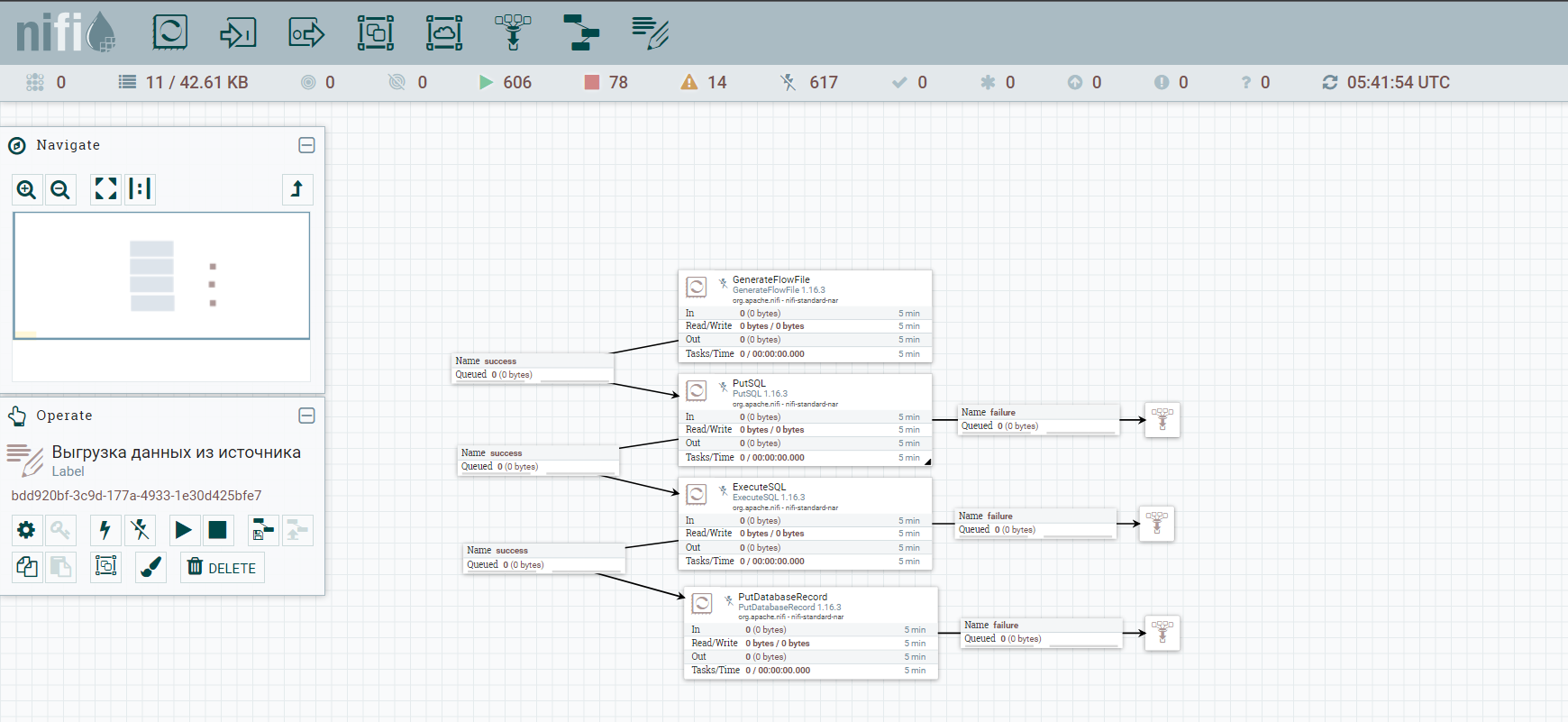
1. ID
2. Количество строк данных
3. Название потока
4. Дата начала
5. Дата окончания загрузки
6. Размер в МБ
7. Разница между датой начала и датой окончания загрузки в минутах

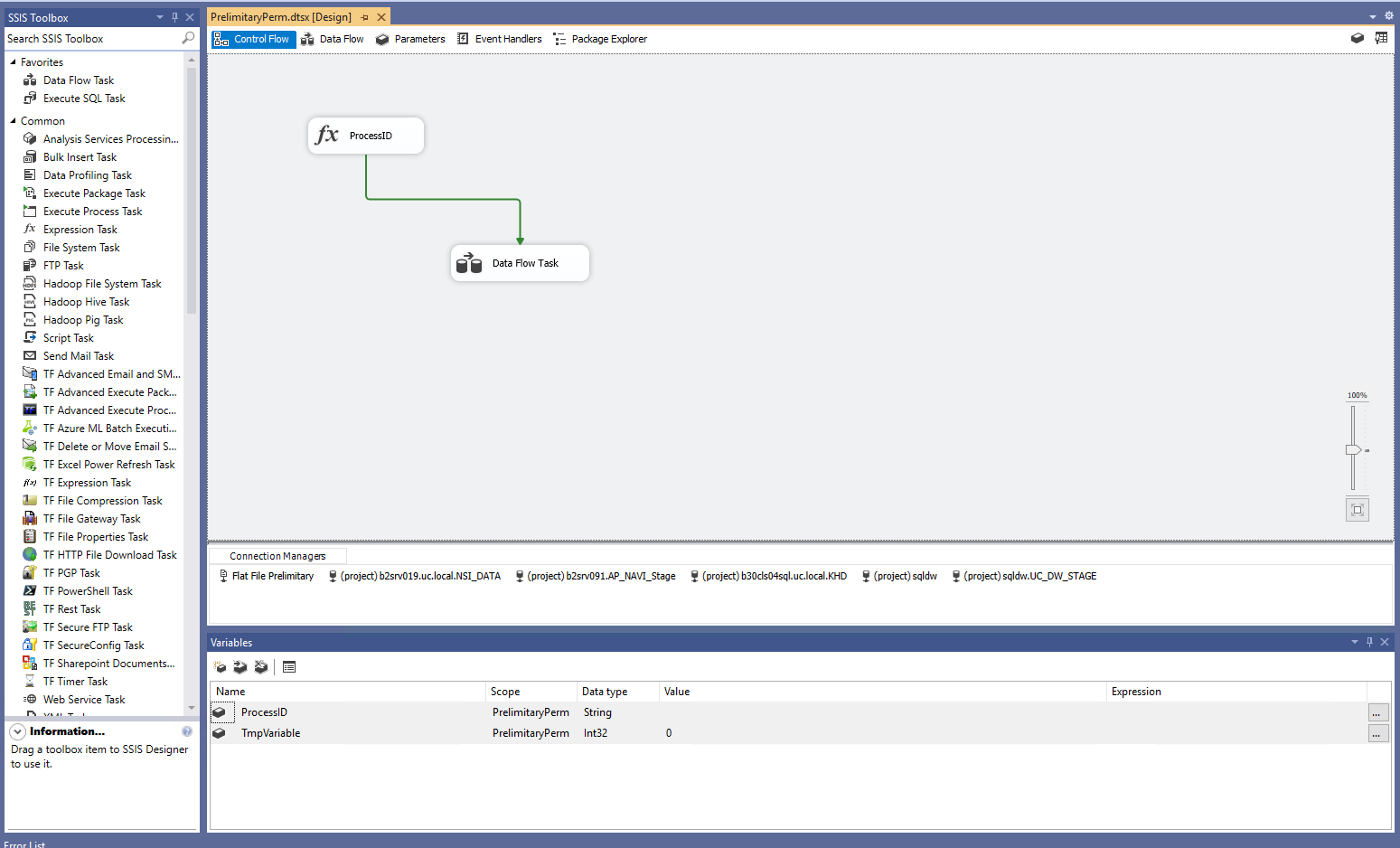
Исходя из колонки 7 можем увидеть, что ETL процесс от точки A (источник) в точку B (приёмник) имеет трудозатраты менее минуты.

На Рис 2. Мы видим работу старого алгоритма.

Исходя из анализа двух скриншотов мы видим, что старый алгоритм может отрабатывать с задержкой до 5 минут, по сравнению с Рис 1.

Тестирование интерфейса пользователя системы является важной частью процесса разработки программного обеспечения. Это позволяет проверить, насколько функционален и удобен интерфейс для конечных пользователей.





Apache NiFi (NIFI) и SQL Server Integration Services (SSIS) являются двумя различными платформами интеграции данных. При сравнении NIFI и SSIS в контексте тестирования интерфейса пользователя у них есть некоторые сходства и различия.

Сходства:

1. Визуальное представление: Оба NIFI и SSIS предоставляют визуальные среды разработки, которые позволяют пользователю графически задавать и настраивать свои процессы интеграции данных.

2. Тестирование функциональности: Обе платформы позволяют проводить тестирование функциональности процессов интеграции данных, включая проверку правильности выполнения каждого шага, обработку ошибок и уведомления.

Различия:

1. Технологии: NIFI основан на Java, в то время как SSIS является частью платформы Microsoft SQL Server и использует .NET Framework. Различные технологические стеки могут требовать различные подходы к тестированию и интеграции с другими системами.

2. Расширяемость: NIFI обладает большим набором встроенных процессоров и поддерживает возможность создания пользовательских процессоров, что позволяет гибко настраивать и расширять функциональность. SSIS также предоставляет возможность создания пользовательских компонентов, но в меньшей степени по сравнению с NIFI.

3. Типы интеграции: NIFI специализируется на обработке потоковых данных и имеет широкие возможности интеграции с различными источниками данных, такими как датчики IoT или системы потоковой обработки данных. SSIS, в свою очередь, часто используется для интеграции с реляционными базами данных и другими источниками данных Microsoft.

Несмотря на множество преимуществ, Microsoft SQL Server Integration Services (SSIS) также имеет некоторые минусы:

1. Ограниченная поддержка исходных данных: SSIS является частью экосистемы Microsoft и в основном предназначен для работы с исходными данными, хранящимися в SQL Server и других продуктах Microsoft. Это означает, что интеграция с некоторыми внешними источниками данных может потребовать дополнительных усилий и решений.

2. Избыточность инструментов: у SSIS много инструментов и функций, и иногда это может показаться избыточным для простых и маломасштабных проектов. Это может затруднить начинающим пользователям SSIS понимание и использование всех возможностей, что приводит к избыточности и сложности в разработке процессов интеграции данных.

3. Сложность отладки: при настройке и выполнении пакетов SSIS может быть сложно выявить и исправить ошибки или проблемы. Из-за сложной структуры пакетов, ошибки могут быть сложно обнаружить, особенно для пользователей, не имеющих опыта работы с SSIS.

4. Зависимость от Windows: SSIS является частью экосистемы Microsoft и полностью зависит от Windows. Это означает, что вы не сможете использовать SSIS на других операционных системах, таких как Linux или macOS.

5. Лицензирование и стоимость: SSIS является частью коммерческого продукта Microsoft SQL Server и является его частью. Следовательно, для использования SSIS требуется соответствующая лицензия на SQL Server или его компоненты, что может повлечь дополнительные расходы для предприятий.

Заключение:

В рамках учебной технологической практики были выполнены следующие работы:

1. Были изучены основы технологии Ni-Fi и анализирована устаревшая технология интеграции данных с помощью SSIS.

2. Был разработан и протестирован прототип информационной системы, основываясь на новой технологии Ni-Fi, для интеграции данных.

3. Проведено нагрузочное тестирование системы, чтобы оценить ее производительность и способность обрабатывать большие объемы данных.

4. Было проведено тестирование интерфейса пользователя системы, чтобы убедиться в его удобстве и функциональности.

Относительно успешности проведения испытаний и эффективности разработанного прототипа информационной системы можно сделать следующие выводы:

1. Проведенные испытания показали высокую производительность системы на базе Ni-Fi и возможность ее эффективного использования для интеграции данных.

2. Разработанный прототип информационной системы успешно выполнил свои задачи и продемонстрировал возможности новой технологии Ni-Fi.

Относительно подтверждения гипотезы исследования можно сделать следующие выводы:

Гипотеза о том, что использование технологии Ni-Fi для интеграции данных будет более эффективным и надежным, чем использование SSIS, была подтверждена. Новая технология позволила улучшить процессы интеграции данных и обеспечила более гибкую настройку и управление потоком данных.

Относительно перспектив дальнейшей разработки можно сделать следующие выводы:

1. В дальнейшей разработке информационной системы на базе Ni-Fi можно реализовать дополнительные функциональности, такие как мониторинг и анализ потока данных, автоматическое обнаружение и исправление ошибок и др.

2. Также стоит продолжить исследование и апробацию других технологий и подходов к интеграции данных, чтобы улучшить еще больше процессы и результаты интеграции.

Список литературы

1. *bigdataschool*. (14 ноября 2023 г.). Получено из https://bigdataschool.ru/blog/practical-advantages-of-nifi-for-big-data-and-iot.html.
2. *coderlessons*. (12 апрель 2019 г.). Получено из https://coderlessons.com/tutorials/java-tekhnologii/uznaite-apache-nifi/apache-nifi-vvedenie.
3. *dataenginer.ru*. (б.д.). Получено из https://dataenginer.ru/?p=5721.
4. *dzen.ru*. (б.д.). Получено из https://dzen.ru/a/Y1kiOWRd7XkqcrPA.
5. *habr.com*. (б.д.). Получено из https://habr.com/ru/articles/465299/.
6. *ivan-shamaev.ru*. (б.д.). Получено из https://ivan-shamaev.ru/apache-nifi-tutorial-guide/.
7. Kavika Roy. (23 июль 2021 г.). *datatobiz.com*. Получено из https://www.datatobiz.com/blog/most-common-data-integration-challenges/#The\_Most\_Common\_Data\_Integration\_Challenges.
8. Microsoft. (10 08 2023 г.). *Официальный сайт Microsoft*. Получено из https://learn.microsoft.com/en-us/azure/data-factory/ssis-integration-runtime-management-troubleshoot.
9. Microsoft. (б.д.). *learn.microsoft.com*. Получено из https://learn.microsoft.com/en-us/azure/architecture/data-guide/relational-data/etl.
10. *nifi.apache.org*. (б.д.). Получено из https://nifi.apache.org/index.html.