

Е. Д. Вишнякова,
студент,
Волгоградский государственный аграрный университет
О. В. Кочеткова,
доктор технических наук, профессор
Волгоградский государственный аграрный университет

ИССЛЕДОВАНИЕ ПРОЦЕССА КОНТРОЛЯ РАБОТОСПОСОБНОСТИ ИНТЕГРАЦИИ ДАННЫХ В ДЕЯТЕЛЬНОСТИ СИТУАЦИОННОГО ЦЕНТРА ГУБЕРНАТОРА ВОЛГОГРАДСКОЙ ОБЛАСТИ

Аннотация. Исследование процесса контроля работоспособности интеграции данных из смежных систем в информационно-аналитическую подсистему автоматизированной системы Ситуационного центра Губернатора Волгоградской области для автоматизации процесса. Цель работы определить необходимость автоматизации процесса и выбрать способ реализации проекта. Данная работа является описанием в рамках написания выпускной квалифицированной работы по оптимизации деятельности работников отдела функционирования Ситуационного центра.

Ключевые слова: исследование, цифровые технологии, интеграция данных, ситуационный центр, автоматизация, база данных, аналитика данных, бизнес-аналитика, информационные системы

INVESTIGATION OF THE PROCESS OF MONITORING THE OPERABILITY OF DATA INTEGRATIONS IN THE ACTIVITIES OF THE SITUATIONAL CENTER OF THE GOVERNOR OF THE VOLGOGRAD REGION

Abstract. The study of the process of monitoring the operability of integrating data from related systems into the information and analytical subsystem of the automated system of the Situational Center of the Governor of the Volgograd region to automate the process. The purpose of the research is to determine the need to automate the process and choose a way to implement the project. This research is a description within the framework of writing a graduate qualified work on optimizing the activities of employees of the department of functioning of the Situational Center.

Keywords: research, digital technologies, data integration, situation center, automation, database, data analytics, business analytics, information systems

Введение. На территории Волгоградской области функционирует Ситуационный центр Губернатора Волгоградской области. Его целью является обеспечение своевременного реагирования Губернатора на деятельность региональных и муниципальных структур. Для этого появилась потребность в своде информации в единую базу и представлении аналитики.

Основная часть. В рамках действия Ситуационного центра в 2018 году была разработана автоматизированная система Ситуационного центра Губернатора Волгоградской области.

ИАП АС СЦ ГВО (Подсистема) – это информационно-аналитическая подсистема автоматизированной системы Ситуационного центра Губернатора Волгоградской области.

Главной целью создания Подсистемы была создание инструмента для принятия управленческих решений на уровне региона. Пользователями Подсистемы являются участники межведомственного взаимодействия: главы и сотрудники муниципальных районов региона, руководство региональных государственных структур, глава региона – губернатор Волгоградской области.

Для подключения к Подсистеме необходимо подать запрос в Ситуационный центр. Совместно с сотрудниками СЦ определяются требования к Личному кабинету и данные, которые необходимо мониторить. Более подробно показано на диаграмме eEPC (рис. 1).

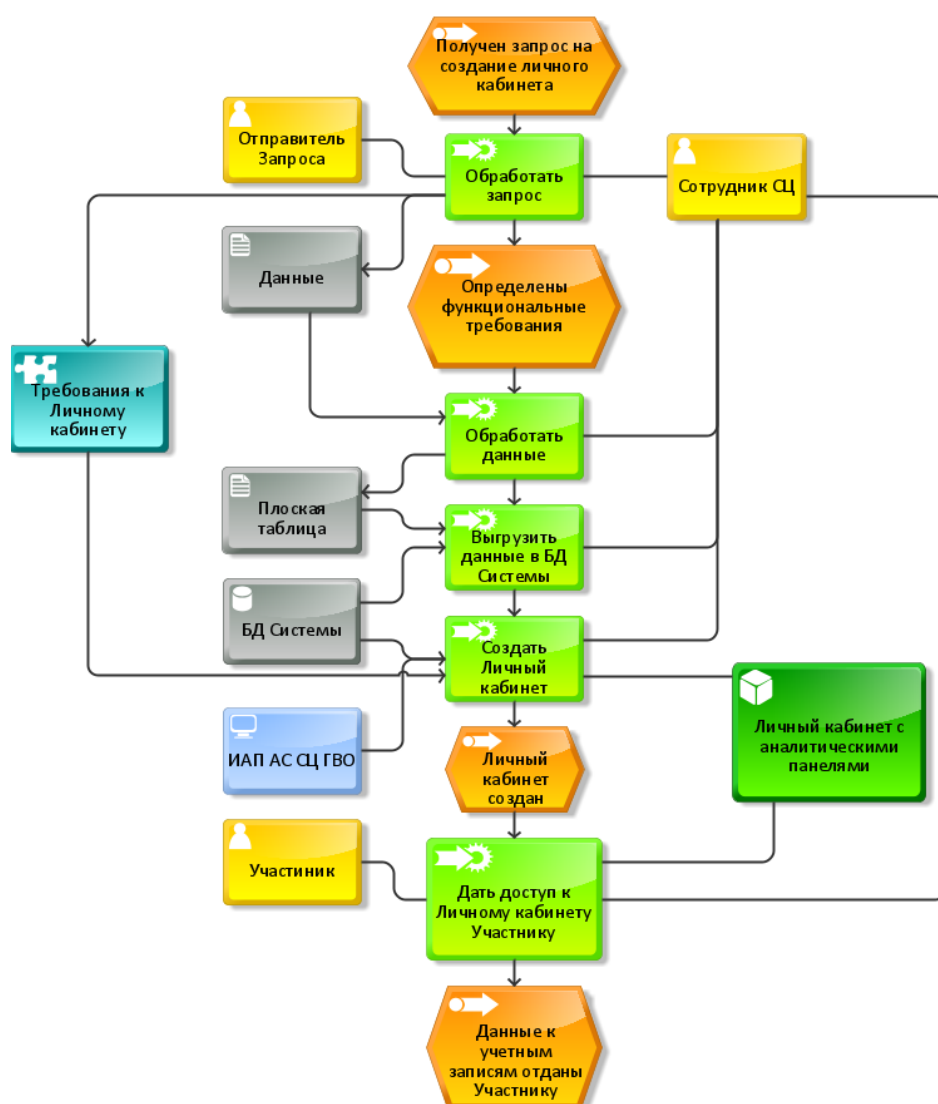


Рис. 1. eEPC-диаграмма создания Личного кабинета Участника

Подсистема включает в себя несколько функциональных модулей (компонентов), каждый из которых решает свои задачи.

Одним из важных модулей для ИАП АС СЦ ГВО является «Компонент сбора, хранения и управления данными». С его помощью загружают данные в БД Подсистемы. В Подсистеме используется три вида ввода данных: «ручной», с помощью ETL и интеграция. При «ручном» виде загрузки, при создании виджета. ETL загрузчик выгружает данные из файла с ПК пользователя. Интеграцию используют при выгрузке данных напрямую из другой системы.

Участники могут запросить настройку интеграции данных из систем региональных и федеральных уровней. На текущий момент реализовано 22 интеграции со сторонними системами.

В работе будет изучен способ получения данных через автоматизированное интегрирование их прямо в Подсистему. Поэтому рассмотрим процесс получения данных и его достоинства и недостатки (рис. 2).

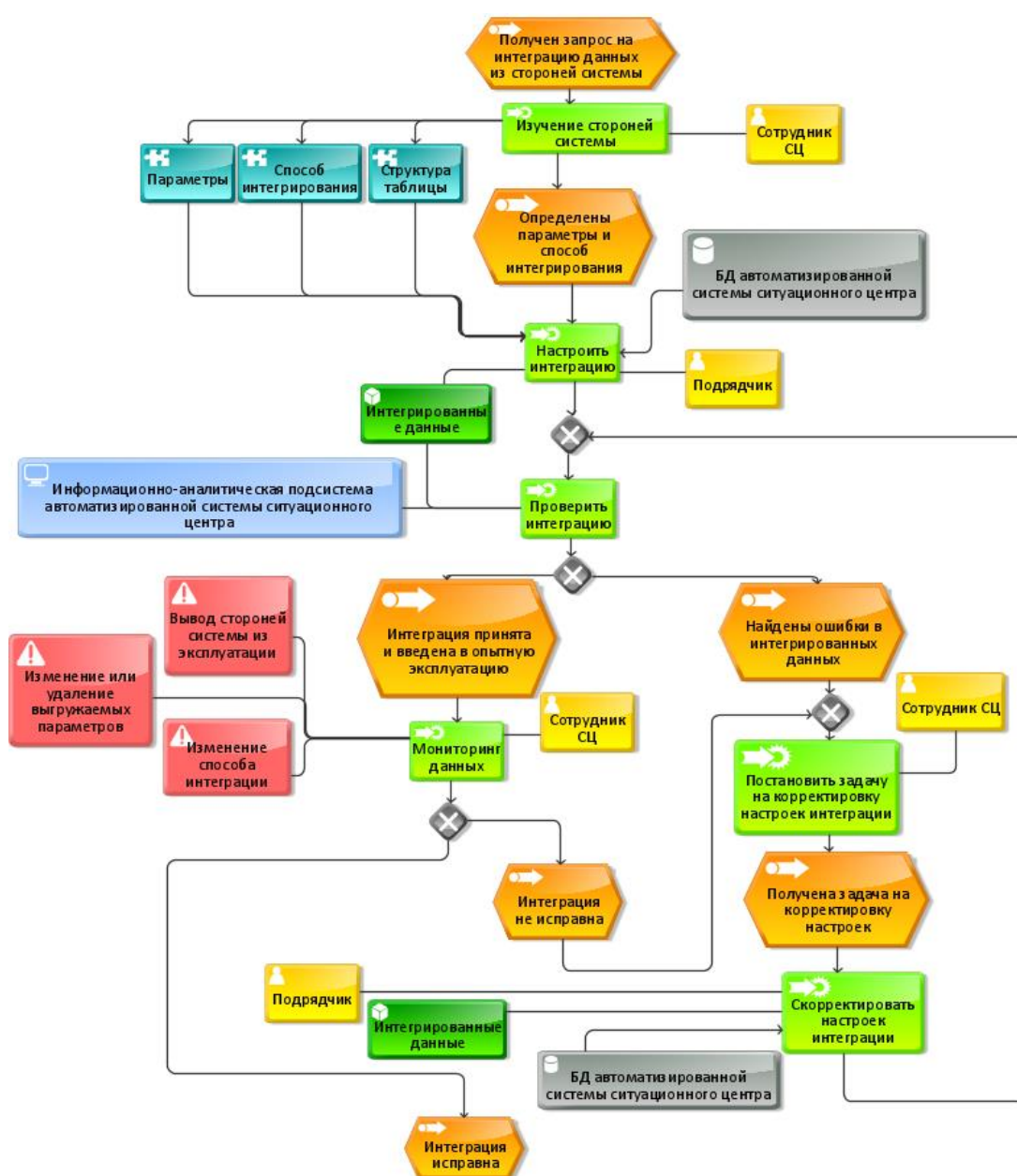


Рис. 2. eEPC-диаграмма процесса настройки интеграции

Настройка интеграции – сложный процесс, при котором сотрудники ситуационного центра привлекают сторонних специалистов (подрядчиков). При составлении заявки на настройку интеграции сотрудники ситуационного центра в адрес подрядчика должны направить:

- параметры, которые необходимо выгружать из сторонней системы. Зависит от запросов пользователей;
- структуры таблиц (таблицы). Зависит от параметров;
- способ интегрирования. Зависит от возможностей сторонней системы и ИАП АС СЦ ГВО.

После определения параметров, способа интеграции и структур таблиц, сотрудники СЦ передают их подрядчикам, для настройки. В среднем весь процесс настройки интеграции занимает от нескольких дней до месяца. Сама проверка работоспособности интеграций занимает от 1 до 3 дней.

Из-за постоянного развития и усовершенствования технологий передачи данные риски могут возникнуть как во время эксплуатации интеграции, так и при первой ее настройке. Что негативно сказывается на уровне удовлетворенности пользователей и их доверии Подсистеме.

Поэтому для сокращения времени проверки и реагирования на появившиеся риски, а также исключения человеческого фактора при проверке, что увеличивает качество, было принято решение использовать инструмент для проверки работоспособности интеграций. Чтобы подобрать подходящий инструмент, необходимо определить и разобрать критерии, по которым необходимо сделать выбор, изучим подробнее процесс.

Исходя из этого, выдвигаем гипотезу: если разработать инструмент для проверки работоспособности интеграций, то количество времени на проверку и риски сбоя сократятся.

Чтоб автоматизировать механизм контроля необходимо:

- изучить данных уже действующих интеграций;
- разработать метод проверки;
- создать скрипт;
- проанализировать эффективность автоматизации.

Все эти этапы можно реализовать с минимумом затрат:

1) для выгрузки данных использовать SQL-запросы к БД. ПО можно использовать любое, подходящее для этого;

2) изучить подробно часто возникающие ошибки и составить правила проверки. Для этого необходимо изучить соответствующую литературу;

3) разработать скрипт, который будет проверять данные, выявлять ошибки и оповещать о них. Использовать можно любой текстовый редактор и язык программирования;

4) изучить показатели до применения разработок и после. Основные показатели будут время и качество проверки.

Целью исследования является повышение качества контроля за работоспособностью интеграций, а также сокращение временных издержек на проведение проверки данных.

Объектом работы будет информационно-аналитическая подсистема автоматизированной системы Ситуационного центра Губернатора Волгоградской области.

Предмет работы: механизм контроля работоспособности интеграции.

Для достижения цели необходимо поставить и решить следующие задачи:

- 1) изучить предметную область;
- 2) определить инструменты для реализации проекта;
- 3) разработать автоматизированный механизм контроля;
- 4) оценить эффективность использования инструмента для проверки работоспособности интеграций деятельности ИАП АС СЦ ГВО.

Использование инструмента должно максимизировать эффективность использования автоматизированных интеграций и снизить риски сбоев для принимающей стороны процесса.

Сотрудники СЦ при использовании данной разработки смогут ставить детализированные задачи на решение проблем, возникших при использовании интеграций, так как будут знать причину сбоя или ошибки.

Выбор объекта исследования и сбор материалов об объекте исследования. Чтобы достичь цели работы, необходимо изучить принцип проверки интеграций. Из этого следует, что процесс передачи данных будет объектом исследования.

Интеграция данных – это набор методов, инструментов и архитектурных процедур, позволяющих компаниям комбинировать и использовать все типы данных. Процесс интеграции обеспечивает не только консолидацию, но и чистоту данных, а также отсутствие ошибок, что позволяет бизнесу использовать данные максимально эффективно [8].

Изучим историю интеграций данных: предпосылки их создания, актуальность использования и их проблематику.

Преимущества и риски использования интегрирования данных. Объединение данных из разных источников стоит на повестке дня с тех пор, как бизнес-системы начали собирать данные. Лишь в начале 1980-х годов ученые-информатики начали проектировать системы, поддерживающие совместимость неоднородных и отдельных баз данных [13]

В 1991 году была запущена первая система интеграции данных в Университете штата Миннесота. Перед университетом стояла задача обеспечить совместимость тысяч баз данных о населении. Система собирала и организовывала данные из разрозненных источников, преобразовывала их и загружала в единую схему ракурсов.

При долговременном использовании системы возникали различные проблемы, связанные с качеством данных, управлением, моделированием, с изоляцией и разрозненностью данных.

Проблема интеграции данных является неотъемлемым аспектом проблематики развития информационной инфраструктуры предприятия. То есть, если развивается инфраструктура источников интеграции, то и процесс интегрирования данных должен развиваться.

Если рассмотреть процесс интеграции со стороны бизнеса-процесса, как в статье электронного журнала DataRevie, можно выделить три преимущества использования интеграции данных [9]:

- стандартизация. Интеграция данных позволяет выбрать и установить единый стандарт представления и работы с данными, что значительно упрощает взаимодействие и ускоряет процесс передачи данных;
- автоматизация. Интеграция данных предполагает автоматический сбор информации из разрозненных источников в режиме реального времени и их моментальное преобразование в предустановленный формат;
- эффективность. Интеграция данных значительно повышает эффективность принятия управленческих решений. Так как данные собираются и преобразовываются в режиме реального времени, то пользователь может принимать управленческие решения в любое время, без ожидания.

То есть использование интеграция уменьшает время на сбор данных и повышает эффективность их использования в бизнес-процессах. Для деятельности СЦ это важно для достижения ее главной цели: своевременного реагирования Губернатора на деятельность региональных и муниципальных структур.

Для определения важности интегрирования данных для Подсистемы, рассмотрим причины ее использования. IT-компания Deco System, в своем блоге выделяют четыре причины использования интеграции данных в ИС [11]:

- Обеспечение единого доступа к данным. Интеграция позволяет обеспечить согласованность данных, делая их доступными.
- Улучшение взаимодействия между подразделениями или организациями. Когда системы интегрированы, команды могут быстрее и эффективнее сотрудничать.
- Повышение эффективности и экономии ресурсов. Упрощение процессов, уменьшение ручного труда. Это, в свою очередь, ведет к сокращению затрат и ошибок.
- Гибкость и адаптивность к изменениям. Легкая модификация в ответ на изменение требований и технологические инновации.

Из этого следует, что интегрирование данных наилучший способ для загрузки данных в базу данных ВІ-системы, которое необходимо развивать.

Типы интеграций и технологии реализации. Существует три типа интеграций: консолидация, федерализация и распространение (рис. 3) [10].

Консолидация данных подразумевает сбор данных из нескольких первичных источников в одно постоянное место хранения.

Консолидация имеет задержку между обновлением информации в источнике, и обновленные данные появляются в конечном месте хранения. Задержка может занимать от нескольких секунд до нескольких дней. Время задержки зависит от настроек обновления данных в источнике и конечном месте хранения и параметров передачи данных.

Федерализация данных преобразует данные из одного или нескольких источников в единую виртуальную картину. Преобразование происходит по запросу к виртуальной картине, который запускает процесс извлечение данных из источников.

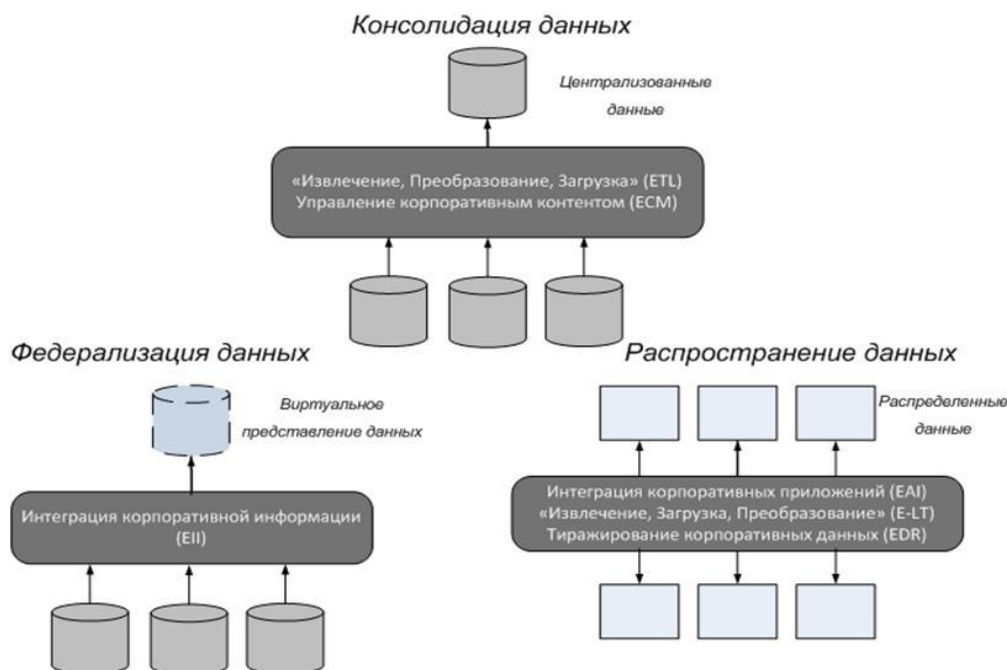


Рис. 3. Типы интеграций

Метод преобразования данных из одного или нескольких источников по запросу называется федерализацией данных. Запрос, который преобразует данные, запускает процесс передачи данных.

Необходимо отметить, что в англоязычной литературе термин *federated data warehouse* сейчас используется в двух разных значениях. Часть специалистов подразумевает под федеративным хранилищем создание виртуальной структуры, оперирующей с выборками данных. Другие называют федеративным хранилищем единый физический репозиторий, работающий с копиями данных, который, другими словами, может быть назван распределенным хранилищем.

Метод копирования данных из источников в несколько мест, называется распространением данных. Главными преимуществами распространения данных являются передача данных в режиме реального времени и гарантированная доставка данных.

Существует несколько техник, которые используются в настройке интеграций [12]:

1. ETL (*extract, transform, load* – «извлечение, трансформация, загрузка») – заключается в извлечении данных, преобразовании их в нужный формат и загрузке в целевую систему.

2. ELT (*extract, load, transform* – «извлечение, загрузка, трансформация») – заключается в извлечении данных и преобразовании их в нужный формат только после загрузки в целевую систему.

3. *Data virtualization* – данные остаются у источников, но представлены в виде единой виртуальной таблицы.

При интегрировании данных в базу данных Подсистемы СЦ используют консолидацию данных в технике ETL. При этом данные копируются из сторонней системы в ИАП, а не удаляются. Но копируются не все данные, а только не-

обходимые, которые указаны сотрудниками СЦ. По их же требованиям и происходит трансформация или преобразование данных до выгрузки их в базу данных

Изучим преимущества консолидации данных и технологии ETL. Консолидация данных может повысить качество передачи данных из других систем. Данные сразу копируются в базу Подсистемы, что обеспечивает доступ к ним и упрощает их управление. При необходимости внесение корректировок или запроса дополнительных манипуляций с данными нет необходимости повторного обращения к персоналу сторонней системы. Вместо этого можно просто расширить диапазон запрашиваемых данных через кодирование. За счет закрытости базы и единого контроля обеспечивается безопасность данных. Так Подсистема исключает распространение данных между Участниками без их согласия.

Рассмотрим этапы процесса ETL: извлечение, трансформация и загрузка. Извлечение: система собирает данные из целевых источников, например, API-интерфейсов, датчиков, баз данных, серверных журналов JSON, маркетинговых инструментов и так далее.

Какие могут быть источники:

- базы данных и системы;
- различные среды;
- приложения;
- CRM-системы;
- платформы хранения данных;
- хранилища данных;
- инструменты аналитики.

Для преобразования данные проходят через анализ, разделение/объединение и фильтрацию. После преобразования отправляются в конечную точку и проходят очередную обработку. Поэтому чаще используют форматы JSON, Parquet или CSV для хранения данных.

Запрашиваемые данные не всегда подходят для хранения в целевой системе в том же виде, что и в источнике, поэтому некоторые из них требуют преобразования. Преобразование включает в себя:

- очистку данных, которая убирает дубликаты, заменяет или удаляет нулевых значений, ищет и исправляет ошибки;
- разделение помогает делить данные из одной ячейки на несколько;
- извлечение позволяет достать отдельную информацию из поля данных;
- объединение данных из нескольких ячеек в одну.

После преобразованная информация из промежуточной области отправляется в целевую базу данных, озеро данных или хранилище данных. При этом можно использовать полную загрузку или добавочную. Полная загрузка – выгрузка всех данных из источника чаще используется для исследований. Полная загрузка может длиться долгое время и нуждается в больших вычислительных ресурсах. Добавочная загрузка сравнивает входные данные с уже имеющимися и создает новые записи только в случае появления новых данных в источниках.

Методы проверки данных. Наличие рисков предусматривает управление ими. Так как ИАП является объектом, на который направленно исследование, то необходимо проводить проверку данных, лежащих в базе данных Подсистемы.

Проверка данных – это процесс обеспечения точности, полноты и согласованности данных, собранных, хранящихся и анализируемых предприятием. Существует несколько методов проверки данных. Проверка типа данных проверяет, находятся ли данные в ячейке и имеют ли соответствующий тип. Проверка кода гарантирует, что поля выбраны из представленного списка и соответствуют стандартам форматирования. Подходит данным, у которых есть определенный список, например, индекс адреса или наименование субъектов.

Проверка диапазона позволяет оценить, соответствуют ли входные данные диапазону. Например, для геоданных, ограниченных областью или субъектом РФ.

Проверка формата. Проверка соответствия необходимого формата данных в выбранном типе данных. Если формат даты определен в виде ДД.ММ.ГГГГ, то он должен соответствовать ему.

Проверка присутствия гарантирует наличие данных в ячейках.

Проверка длины проверяет, что в ячейке находятся необходимое количество символов.

Так как данные требующие проверки являются разнотипными и разноформатными, то при разработке инструмента проверки необходимо применение комплекса методов. Из представленных методов выделяют основные факторы проверки: формат; диапазон; завершенность; согласованность; зависимость атрибута; недействительные значения; отсутствующие значения; орфографические ошибки.

Проверка данных можно провести одним из трех способов: сценарии, корпоративные инструменты и инструменты с открытым исходным кодом. Выбор и использование правильных инструментов и инфраструктур проверки данных может оказать существенное влияние на потенциал Подсистемы, потому что могут помочь достичь следующих преимуществ.

Повышение качества и надежности данных. Инструменты и системы проверки могут гарантировать точность данных и их полноту. Это в свою очередь сказывается на качестве работы всей Подсистемы и удовлетворенности пользователей.

Повышение безопасности и соответствия требованиям. Использование специализированного инструментария в процессе проверки защищает данные от нежелательных действий персонала, а также обеспечивает соответствия данных требованиям сотрудников СЦ.

Оптимизация производительности и эффективности данных. Оптимизация скорости, масштабируемость и экономическая эффективность операций с данными сказывается на работоспособности всей Подсистемы и отдела СЦ в частности.

При выборе инструментария для проверки данных стоит обратить внимание на некоторые аспекты:

1. Тип и сложность данных. Для разных типов данных и источников могут потребоваться разные методы и сценарии проверки.

2. Объем и частота проверки данных могут повлиять на выбор и использование инструментов и инфраструктур проверки данных, поскольку некоторые инструменты могут быть более подходящими для пакетной проверки, тогда как другие могут быть более подходящими для проверки в режиме реального времени или потоковой проверки.

3. Уровень и режим проверки данных определяют, насколько глубокой и интерактивной является проверка данных. Это может зависеть от целей проверки данных, критериев проверки данных и методов проверки данных.

Проверка данных – это не только технический процесс, но и стратегический, который обеспечивает качество, точность и согласованность данных. Получение ценной информации способствует своевременной оптимизации операций и улучшает качество использования Подсистемы.

Процесс проверки включает в себя проверку данных на соответствие заранее определенным правилам или критериям, а также выявление и исправление любых ошибок или несоответствий. Проверка данных может помочь избежать распространенных ошибок, таких как пропущенные значения, опечатки, дубликаты, выбросы или недопустимые форматы. Это также может помочь оптимизировать процесс интеграции данных и принятия решений, предоставляя чистые, последовательные и актуальные данные.

В зависимости от типа, источника и цели ваших данных вам может потребоваться применить различные методы в процессе проверки данных. А для улучшения процесса проверки данных необходимо:

1. Четкое и последовательное определение правил и критериев проверки данных.

2. Выбор подходящего уровня и метода проверки данных. Проверка данных может выполняться на разных уровнях, таких как уровень поля, записи, файла или базы данных. Это также может быть сделано с использованием различных методов, таких как ручной, автоматический или полуавтоматический.

3. Использование подходящих инструментов и методов для проверки данных. Для проверки данных доступно множество инструментов и методов, таких как программное обеспечение для обеспечения качества данных, профилирование данных, очистка данных, преобразование данных, проверка данных и сверка данных.

4. Тестирование и контроль процесса проверки данных. Перед внедрением использования инструментария на постоянную основу необходимо протестировать и оценить его эффективность.

Анализ предметной области в рамках поставленной задачи по материалам научно-технических зарубежных публикаций и информации в сети Интернет. Инструменты по проверке данных внедрены в некоторые платформы по управлению данными. Но их использование невозможно из-за условий контрактов о развитии Системы.

При составлении контракта о создании и контрактов о развитие и сопровождении Подсистемы прописано условие, исключающее подключение сторонних ПО для функционирования Подсистемы. Этот пункт обеспечивает целостность автоматизированной системы, ее самостоятельность и независимость

в функционировании. А также данный пункт обеспечивает повышенную конфиденциальность данных, которые выгружают в базу данных Подсистемы.

Из этого следует, что сотрудникам СЦ подойдет способ реализации проверки «сценарий». То есть код, который будет проводить проверку данных.

При поиске в сети Интернет, представляются отрывки и примеры анализа небольших данных, а также теория о тестировании баз данных. Но в предложенных множествах ссылок на различные сайты, не было найдено какого-либо готового решения подходящего данных Подсистемы. Например, в статье о скриптах описаны основы тестирования данных и примеры сценариев кода для проверки небольших данных. Также Интернете можно найти и теорию по тестированию базы данных. Но вся информация дана для изучения, и практических работ, которые можно сразу применить, нет.

Обратимся к зарубежным источникам. Так в поисковой системе Google будем искать «data validation» и «the process of data validation», но в зарубежных источниках тоже приведена только теории. При этом информация, которая представлена в зарубежных источниках, дублируется и в российских. Все источники располагают общей информацией и общими примерами. В списке используемых источников они под номерами 2–6 и 8, 9, 12. Из этого следует, что, используя собранную информацию, мы можем разработать продукт, подходящий под характеристики данных Подсистемы.

Работа по исследованию данных в деятельности ИАП является уникальной за счет специфики объекта исследования. Системы подобные ИАП АС СЦ ГВО невозможно встретить, так как она разрабатывалась по запросам и требованиям Волгоградской области и является интеллектуальной собственностью. Поэтому тема исследования является уникальной и актуальной.

Создание схемы бизнес-процесса. Для более детального изучения проблематики работы необходимо рассмотреть подробно сам бизнес-процесс. Схема бизнес-процессов будет основана на eEPC-диаграмме процесса настройки интеграции (рис. 2). Действующие роли для схемы: Сотрудник СЦ, Пользователь и Подрядчик. А переменными будут: требования и корректность работоспособности интеграции. Формат для требований определим как файл, а для определения корректности – как строка.

После определения ролей и переменных – создание диаграммы в нотации BPWN. В итоге получаем схему, изображенную на рис. 4.

Составим формы для процессов «Сформировать требования к интеграции», «Настройка интеграции», «Проверка интеграции» и «Мониторинг работоспособности интеграции».

Результат процесса «Сформировать требования к процессу интеграции» – это сформированный файл требований. Поэтому в форме необходимо предусмотреть отправку файла от Сотрудника СЦ Подрядчику (рис. 5).

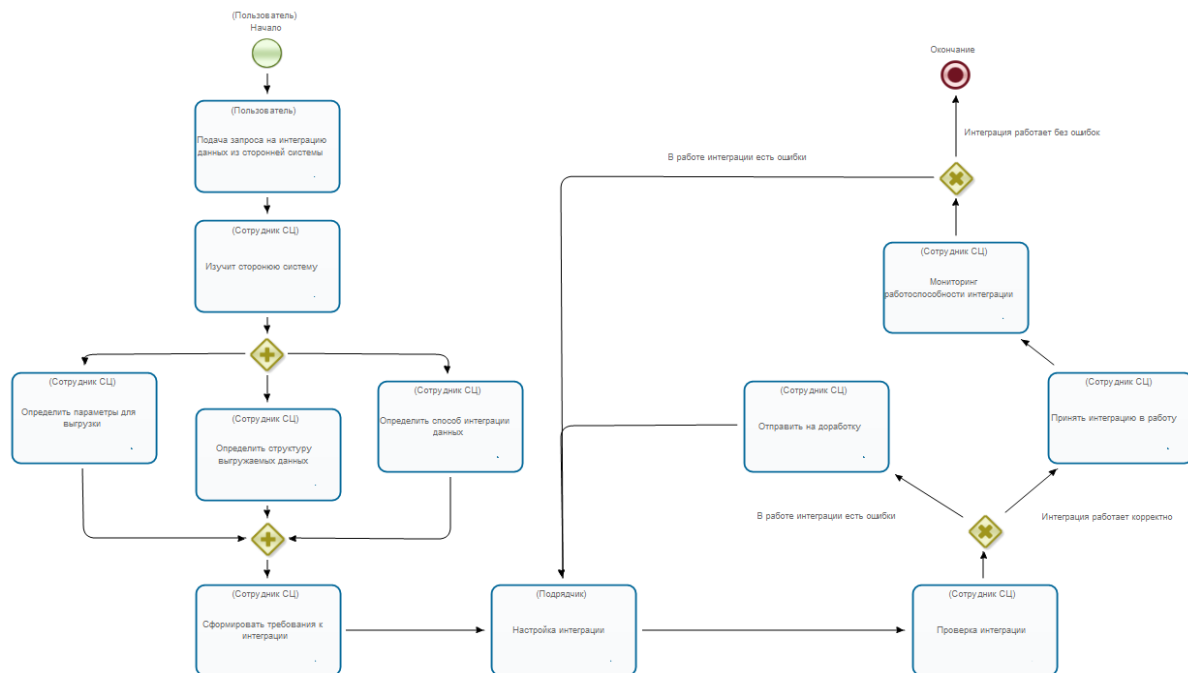


Рис. 4. BPWN-диаграмма процесса настройки интеграции

Рис. 5. Форма «Сформировать требования к интеграции»

Чтобы подрядчик смог получить информацию, необходимо в форме предусмотреть возможность скачать файл с требованиями. Поэтому форма показана на рис. 6.

Рис. 6. Форма «Настройка интеграции»

Форма для «Проверка интеграции» (рис. 7) должна показать корректность работы интеграции. Поэтому используя переменную «корректность работоспособности интеграции» и в форме назначим ей значение «да» или «нет». От ответа будет зависеть следующий процесс. Если интеграция работает корректно, то Сотрудник СЦ принимает интеграцию в работу или же отправляет на доработку.

Рис. 7. Форма «Проверка интеграции»

Соответствующую форму необходимо настроить для процесса «Мониторинг работоспособности интеграции» (рис. 8). Только если интеграция работает без ошибок, весь процесс проверки работоспособности завершается или интеграция снова отправляется на доработку.

Рис. 8. Форма «Мониторинг работоспособности интеграции»

Заведение учетных записей и создание отношений. Для создания учетных записей для соответствующих ролей, откроем веб-интерфейс RupaWFE и зайдем под администратором. Перейдя во вкладку «Исполнители» и заводим пользователей «Сотрудник СЦ», «Пользователь» и «Подрядчик». Для удобства соберем их в одну группу «Проверка работоспособности интеграций» (рис. 9).

Для загрузки созданного процесса (рис. 4) в веб-интерфейс переходим во вкладку «Запустить процесс» и разворачиваем «Загрузить определение процесса». Выбрав выгруженный процесс, нажимаем на «Загрузить определение процесса» (рис. 10).

Свойства исполнителя		Задачи сотрудников группы	Обладатели полномочий
Имя	Проверка работоспособности интеграций *		
Описание			
Группа LDAP			
Применить			

Группы исполнителя	
Вид	Без фильтра
Добавить	
☐	Имя / Описание
Удалить	

Члены группы			
Вид: Без фильтра			
Добавить			
☐	Имя /	Полное Имя	Описание
☐	Подрядчик		
☐	Пользователь		
☐	Сотрудник СЦ		
			Всего:3
			Удалить

Отношения, в которые исполнитель входит в правой части	
Имя отношения	Описание

Отношения, в которые исполнитель входит в левой части	
Имя отношения	Описание

Рис. 9. Исполнители по процессу «Проверка работоспособности интеграций»

Меню	Определения процессов																								
- Список заданий Запустить процесс - Запущенные процессы - Архивные процессы - Исполнители - Отчёты - Отношения - Бот станции - Источники данных - Внутреннее хранилище - Система - Ошибки - Настройки - Логи сервера - Задания сотрудников - Отправить сигнал	Вид: Без фильтра Загрузить определение процесса Определение процесса Настройка интеграции.par - 44 Kb Тип процесса Создать новый тип Дней до архивации (пусто = по умолчанию) Применить тип ☒ Только к новым процессам ☐ Ко всем процессам Загрузить определение процесса История <table border="1"> <thead> <tr> <th>☐</th> <th>Запустить</th> <th>Имя /</th> <th>Описание</th> <th>Тип процесса</th> <th>Дата загр</th> </tr> </thead> <tbody> <tr> <td>☐</td> <td>@</td> <td>AjaxFormDemo</td> <td>AJAX support demo</td> <td>Other</td> <td>14.11.2022</td> </tr> <tr> <td>☐</td> <td>👤</td> <td>BotDemo</td> <td>Bot demo process (no start form provided)</td> <td>Bots</td> <td>14.11.2022</td> </tr> <tr> <td>☐</td> <td>👤</td> <td>Business...</td> <td>Participants of this process are members of manager...</td> <td>Company</td> <td>14.11.2022</td> </tr> </tbody> </table>	☐	Запустить	Имя /	Описание	Тип процесса	Дата загр	☐	@	AjaxFormDemo	AJAX support demo	Other	14.11.2022	☐	👤	BotDemo	Bot demo process (no start form provided)	Bots	14.11.2022	☐	👤	Business...	Participants of this process are members of manager...	Company	14.11.2022
☐	Запустить	Имя /	Описание	Тип процесса	Дата загр																				
☐	@	AjaxFormDemo	AJAX support demo	Other	14.11.2022																				
☐	👤	BotDemo	Bot demo process (no start form provided)	Bots	14.11.2022																				
☐	👤	Business...	Participants of this process are members of manager...	Company	14.11.2022																				

Рис. 10. Загрузка процесса в веб-интерфейс RunaWFE

Потом нажимаем на новый процесс и переходим во вкладку «Запущенные процессы» и открываем процесс для назначения исполнителей. В блоке «Роли процесса» назначаем ролям соответствующих исполнителей (рис. 11).

Роли процесса			
Изменить исполнителя			
Имя	Глобальная	Исполнитель	О
Сотрудник СЦ		Сотрудник СЦ	3+
Пользователь		Пользователь	3+
Подрядчик		Подрядчик	3+
Перманентные процессы			

Рис. 11. Назначенные исполнители на соответствующие роли

Организация тестирования и реализация цикла тестирования. Для тестирования заходим под учетной записью Пользователя и завершаем задачу «Поддача запроса на интеграцию данных из сторонней системы» (рис. 12).

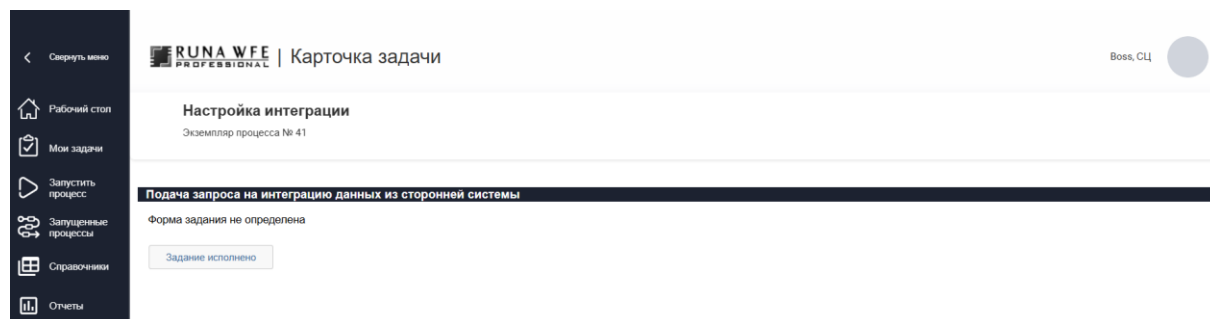


Рис. 12. Выполнение задачи «Поддача запроса на интеграцию данных из сторонней системы»

Дальше заходим под Сотрудником СЦ и выполняем поочередно задачи:

- 1) изучить стороннюю систему;
- 2) определить способ интеграции данных;
- 3) определить параметры для выгрузки;
- 4) определить структуру выгружаемых данных;

Определение проблематики. Как видно из моделирования процесса настройки: процесс в данном виде несовершенен. Отсутствие контроля над данными приводят к их потере и отказу передачи, что приводит к недействительным показателям для показа пользователю. Такие показатели влияют на принятие решений на уровне регионального управления.

Выбор методов и инструментария исследования. Перед созданием самого инструмента, который будет контролировать потоки данных, необходимо определить на каком языке будет он создан. Для этого был проведен анализ нескольких вариантов.

Существует еще множество языков, которые можно включить в этот список. Но наш выбор был сделан исходя из функциональных возможностей оборудования и специализации экспертов.

В СЦ уже используется несколько разработок на Python, поэтому для его использования уже все подключено к местам сотрудников. SQL – уникальный язык, который тоже используется в работе сотрудников, поэтому был включен в список. Java использовался в создании и развитии многих модулей Подсистемы, поэтому подрядчики рекомендовали рассмотреть его вариант. Matlab используется в многих научных сферах, а также позволяет использовать сложные математические вычисления.

Python – популярный язык программирования общего назначения и широко используется в сообществе специалистов по данным. SQL (язык структурированных запросов) определяет, управляет и запрашивает реляционные базы данных. Со временем язык потерпел множество видоизменений, но основные его принципы остаются неизменными. Java – популярный язык общего назначения,

который работает на виртуальной машине Java Virtual Machine. Это абстрактная вычислительная система, которая обеспечивает плавную переносимость между платформами. Matlab – это признанный язык для численных расчетов, используемый как в научных целях, так и в индустрии. Он был разработан и лицензирован MathWorks, компанией, созданной в 1984 году, основной целью которой являлось коммерциализация программного обеспечения.

Изучим подробнее варианты перед проведением исследования. Для этого составим таблицу (табл. 1) с описанием их преимуществ, недостатков, доступности и комментарии специалистов и экспертов.

Таблица 1

Характеристики вариантов для проведения исследования

Вариант	Преимущества	Недостатки	Доступность	Комментарий
Python	Python – универсальный инструмент. Он был задуман как язык, который можно легко расширять, дописывая собственные модули и функции. Он может выполнять одни и те же действия на различных операционных системах без переписывания программ под них. Если раньше для работы с графикой, разными форматами файлов, системными и сторонними библиотеками требовалось изменение кода и модели программирования, то с Python эта необходимость отпадает. Одно из главных преимуществ языка – низкий порог входа. Отсюда и большое сообщество питонистов, которое, если что, всегда поможет с вопросами. Один из главных недостатков – это высокоуровневый язык, поэтому у программирования на нем есть ограничения, например, по	Производительность. Здесь Python проигрывает таким языкам, как С или С++, которые обычно используются для системного программирования и создания высокопроизводительных приложений. Мобильные приложения. В разработке мобильных приложений Python используется не так часто, как Java, Swift или Kotlin. Существуют библиотеки для создания мобильных приложений на Python, например, Kivy, но широкого распространения в разработке они не нашли. Не компилируется в машинный код. Это делает Python более универсальным, но в то же время снижает производительность по сравнению с кодом, написанным на компилируемом языке [16]	Бесплатная лицензия Для использования подходят множества редакторов	Python является хорошим вариантом для целей науки о данных (data science), и это утверждение справедливо как для начального, так и для продвинутого уровней работы в данной области. Большая часть науки о данных сосредоточена вокруг процесса ETL (извлечение-преобразование-загрузка). Эта особенность делает Python идеально подходящим для таких целей языком программирования. Библиотеки, такие как Tensorflow от Google, делают Python очень интересным языком для работы в области машинного обучения [23]

Вариант	Преимущества	Недостатки	Доступность	Комментарий
	доступу к оперативной памяти или процессору [16]			
SQL	Стандартизация. Разработанные международные нормы позволяют унифицировать базы данных, построенные на реляционной модели Реляционная основа. Табличная структура интерфейса облегчает пользователям работу с БД Можно создавать интерактивные запросы. Пользователю доступно все содержимое массива в реальном времени Простая интеграция на программном уровне. Приложения, которые используют доступ к БД, легко программируются с помощью языка SQL Гибкость настроек. Разработчик может динамически изменять, расширять структуру базы, настраивая ее под конкретные параметры Универсальность. Работать с языком могут специалисты разного профиля: разработчики БД, администраторы, программисты и конечные пользователи [9]	Не является на 100% реляционным. Эдгар Франк Кодд и Крис Дейт, создавшие реляционную модель, нашли несколько системных дефектов в синтаксисе языка. Это выражается в повторяющихся строках, неопределенных значениях, высокой избыточности и других отступлениях от канонов Сложность в освоении. Изначально среда создавалась как простой и понятный инструмент. Со временем язык эволюционировал и стал настолько сложным, что перешел в категорию инструментов для программистов Проблемы с совместимостью. Старые версии не поддерживают древовидные иерархические структуры, которые есть во многих современных БД [9]	Лицензия платная и бесплатная	SQL более полезен в качестве языка для обработки данных, чем в качестве передового аналитического инструмента. Тем не менее, так много процессов в области науки о данных зависит от ETL, а долговечность и эффективность SQL лишней раз свидетельствуют о том, что такой ЯП должен знать каждый специалист по данным (data scientist)

Вариант	Преимущества	Недостатки	Доступность	Комментарий
Java	С-подобный синтаксис – негласный стандарт среди программистов. Синтаксис Java основан на C++, но упрощен, что позволяет новичкам быстрее учиться и эффективнее использовать код. Возможности объектно-ориентированного программирования. ООП позволяет создавать расширяемые системы (extensible systems), которые заставляют работать систему с новыми компонентами без внесения изменений. Статическая типизация – заблаговременное объявление типа переменных, которое не может быть изменено впоследствии. Благодаря ей ошибок можно избежать уже на первых этапах. Java ориентирована на JVM (Java Virtual Machine). Эта программа позволяет запускать написанный на Java код на любом поддерживаемом устройстве или операционной системе. Помогает и обратная совместимость: при переходе на новую версию JVM разработчикам не нужно менять старый код. Большое сообщество. На GitHub свыше 128 тысяч публичных репозиторий с Java-проектами [15]	Кроссплатформенность. С одной стороны, она позволяет разрабатывать приложения одновременно под различные системы: Windows, Linux, MacOS, – но при этом приходится подстраиваться под ранние версии Java, чтобы сохранить работоспособность на устройствах предыдущих поколений. Строгость языка. Она позволяет четко понимать, что выполняет написанный код, но при этом может стать препятствием в изучении языка новичками [15]		Много чего можно сказать в пользу изучения Java как языка для работы в области науки о данных. Многие компании оценят возможность беспрепятственной интеграции готового кода программного продукта в собственную кодовую базу, а производительность и типобезопасность Java являются его неоспоримыми преимуществами. Тем не менее, к недостаткам такого языка можно отнести тот факт, что у него отсутствуют наборы специфических пакетов, которые доступны для других языков. Несмотря на такой недостаток, Java является языком программирования, которому обязательно стоит уделить внимание, особенно если вы уже знаете R или Python.

Вариант	Преимущества	Недостатки	Доступность	Комментарий
MATLAB	Широкий функционал. Язык MatLab включает продвинутую библиотеку для обработки и построения графиков, интегрированные функции линейной алгебры (LAPACK, BLAS), быстрого преобразования Фурье (FFTW), работы с полиномами, базовой статистики, решения дифференциальных уравнений и пр. Обновляемость. Основные обновления языка и всей среды программирования происходят два раза в год. Преобразования в «быстрый» код. Программы, написанные на MatLab, можно перевести на C/C++ с помощью MatLab Coder. Интегрированность. Большой выбор прикладных приложений от официальных и сторонних разработчиков расширяет возможности языка. Большое научное сообщество. Язык MatLab используется во многих университетах и исследовательских институтах [4]	Перегруженность. Большое число команд и операторов существенно замедляют работу программ, написанных на MatLab (в последних версиях этот недостаток практически устранен). Данные хранятся в оперативной памяти в виде векторов, поэтому на быстрое действие также сильно влияет скорость векторизации – преобразования данных в векторы и обратно, операции с ними. Узкая направленность. Использовать MatLab можно только в его замкнутой экосистеме, на других программных платформах этот язык неэффективен. Высокая стоимость. Программная среда распространяется платно. Для покупки по минимальной стоимости нужно быть студентом образовательного учреждения. Также алгоритмы являются проприетарными, поэтому просмотр кода большей их части недоступен [4]	Цены варьируются в зависимости от выбранного вами варианта языка	Благодаря своему широкому использованию в различных количественных вычислениях как для научных целей, так и для целей индустрии, MATLAB стал достойным вариантом для применения в области науки о данных. Он придется вам как нельзя кстати, если для ваших ежедневных целей необходима интенсивная, продвинутая математическая функциональность, собственно, для чего MATLAB и был разработан.

Для проведения сравнительного анализа будет использоваться метод идеальной точки. Для проведения расчетов используем Excel MS, так как он доступен и удобен. Экспертами будут сотрудники ЦЦ и персонал со стороны подрядчика, занимающийся настройками интеграций. Оценку будем проводить по основным критериям:

- 1) функциональность – оценивание обширность функциональных возможностей языка, относительно всех вариантов;
- 2) доступность – оценивание сложность установки языка для текстовых редакторов;
- 3) производительность – оценивание результативности и эффективности языка в предметной области;

- 4) надежность – оценивание влияния команд языка программирования на данные;
- 5) направленность на предметную область;
- 6) поддержка разработчиков и сообщества.

Метод идеальной точки. Рассматривается m -мерное пространство (где m – число локальных критериев), в котором априори выбирается вектор, отображающий идеальное решение (или, что-то же самое, идеальная точка, координатами которой являются идеальные значения (например, минимальные или максимальные значения) локальных критериев). В этом пространстве вводится некоторая метрика, с целью вычисления расстояния между вектором, отображающим рассматриваемое решение, и идеальным. В качестве наилучшего выбирается такое решение, векторная оценка которого наиболее близка к идеальной точке.

Были выделены шесть критериев для сравнения, по которым эксперты выставили баллы оценки. Далее по каждому критерию находится наилучшее значение и заносится в столбец условного эталонного объекта. Затем необходимо посчитать стандартизированные показатели по формуле (1):

$$x_{ij} = \frac{a_{ij}}{\max a_{ij}}, \quad (1)$$

где x_{ij} – стандартизированные показатели, a_{ij} – исходные показатели.

В результате нужно определить рейтинговую оценку каждого редактора по формуле (2):

$$R_j = \sqrt{K_1(1 - X_{1j}^2)^2 + \dots + K_n(1 - X_{nj}^2)^2}, \quad (2)$$

где $K_1, K_2, \dots, K_n$ – весовые коэффициенты показателей, назначаемые экспертом.

В табл. 2 представлено сравнение языков программирования с помощью метода идеальной

Таблица 2

Сравнение языков программирования по методу идеальной точки

Показатели		Анализируемые языки программирования								Условный эталонный инструментарий	Весовые показатели
		Python		SQL		Java		Matlab			
		Баллы	Стандартизированные показатели	Баллы	Стандартизированные показатели	Баллы	Стандартизированные показатели	Баллы	Стандартизированные показатели		
1	Функциональность	9	0,90	7	0,70	8	0,80	10	1,00	10	0,20
2	Доступность	9	1,00	9	1,00	8	0,89	4	0,44	9	0,20
3	Производительность	9	0,90	8	0,80	10	1,00	9	0,90	10	0,10
4	Надежность	8	0,80	7	0,70	9	0,90	10	1,00	10	0,20
5	Направленность на предметную область	10	1,00	9	0,90	9	0,90	10	1,00	10	0,20
6	Поддержка разработчиков и сообщества	9	1,00	8	0,89	9	1,00	9	1,00	9	0,10
Рейтинговая оценка		0,19		0,36		0,22		0,36		Итого:	1
Рейтинговое место		1		3		2		4			

На основе проведенных расчетов составим для каждого языка программирования лепестковые диаграммы для наглядной демонстрации рейтинговой оценки (рис. 13).

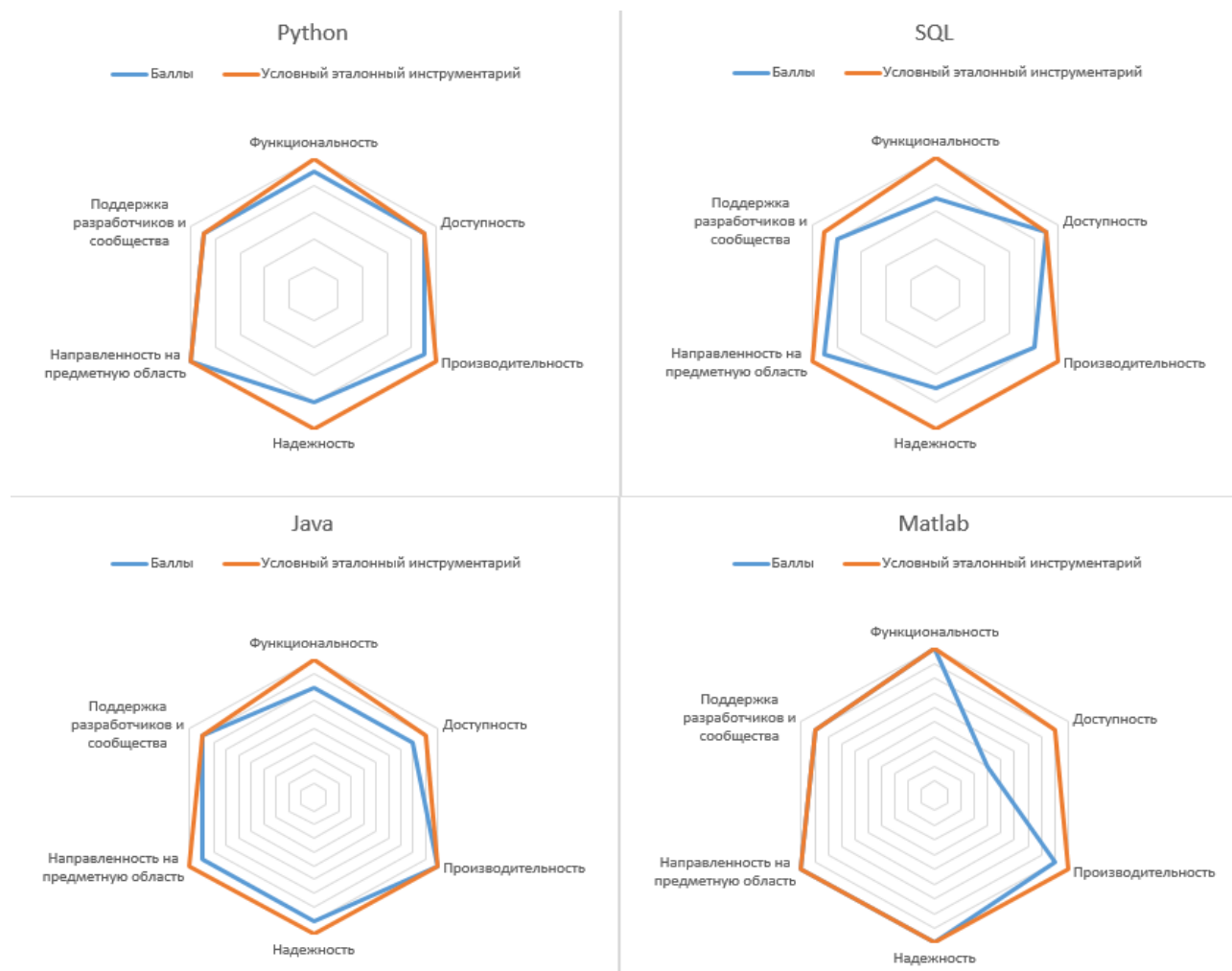


Рис. 13. Лепестковые диаграммы отклонение оценок вариантов от эталонных

На диаграммах отклонение у Python, наибольшее у SQL. На основе оценок было принято решение использовать для разработки инструмента Python, для контроля его точности SQL в БД Подсистемы.

Обоснование метода проектирования объекта исследования

Метод проектирования определяет работу по разработке инструмента и внедрению его в деятельность Ситуационного центра. При выборе метода стоит определить требования к инструменту и самому методу.

Ранее было определено, что инструмент должен быть разработан вручную, а также необходимо его корректировать снова и снова на протяжении пользования интегрировании данных в Подсистему.

Модели жизненного цикла информационных систем разделяют на каскадную, итерационную и спиральную. Из них под наш проект подойдет спиральная, так как подразумевает несколько версий одного продукта. На рис. 14 показана схема спиральной модели.

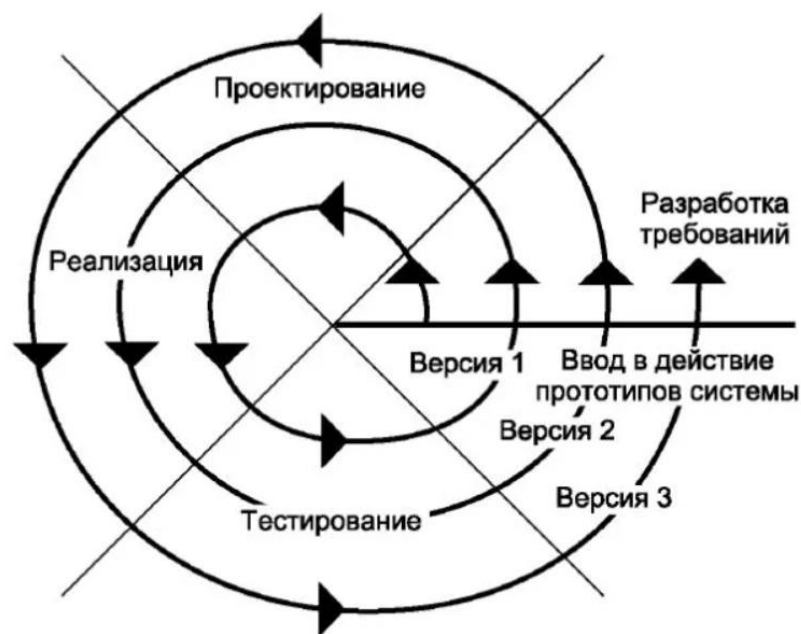


Рис. 14. Схема спиральной модели ЖК

Под такую модель подходят методы XP и RAD.

Технология XP связана со стремлением авторов (К. Бек, У. Каннингем, М. Фаулер) поднять существующие методы разработки ИС на новый уровень. Для оценки проектов применяются два показателя – критичность и масштаб. Критичность определяется последствиями, масштаб – количеством работников.

Технология RAD (англ. Rapid Application Development) основана на спиральной модели жизненного цикла и обеспечивает ускорение разработки ИС благодаря широкому привлечению к процессу проектирования будущих пользователей. Для данной технологии характерно перенесение основных работ с предпроектной стадии на проектирование.

Применение технологии RAD – наиболее подходящее для разработки инструмента по проверки работоспособности. Для проекта не так важны масштаб или количество разработчиков, поэтому технологию XP использовать не целесообразно.

В технологии RAD разделяют несколько видов прототипов: горизонтальный, вертикальный, одноразовый и эволюционный.

Для горизонтального важен интерфейс приложения и ему уделяется наибольшее внимание при разработке. В вертикальном прототипе важны все срезы приложения. Одноразовый применяется для решения проблемы единой, без дальнейшего пользования прототипом.

Эволюционный прототип подходит для проекта разработки, так как этот прототип подразумевает совершенствование продукта на следующих итерациях.

Для описания проектной деятельности необходимо определить методологию. Использовать будем методологию RUP, так как его используют при спиральной модели ЖК. В качестве языка моделирования используется Unified Modelling Language (UML).

Наибольшее внимание RUP уделяет начальным стадиям разработки проекта – анализу и моделированию – что призвано снизить риски проекта за счет раннего обнаружения возможных ошибок. Последовательный выпуск версий организован таким образом, чтобы наиболее существенные риски устранялись в первую очередь.

Итак, для разработки инструмента проверки целостности данных были определены:

- Модель ЖК: Спиральная;
- Технология: RAD;
- Методология: RUP;

Заключение. На момент написания статьи уже разработана основа для инструмента. С помощью сценария, написанного на Python, можно выгружать таблицы из БД Подсистемы. Далее необходимо изучить методы исследования данных и применить их на действующие интеграции. После определения методов будет доработан основной сценарий.

Список литературы

1. Методические рекомендации по созданию и вводу в эксплуатацию ситуационных центров, работающих по единому регламенту взаимодействия, утвержденные решением Межведомственной комиссии по координации деятельности федеральных органов исполнительной власти по созданию системы распределенных ситуационных центров, работающих по единому регламенту взаимодействия (протокол № 2 от 7 мая 2015 г.)

2. 13 лучших инструментов для анализа данных, 11.07.2021 // New-science [Электронный ресурс]. – URL: <https://new-science.ru/13-luchshih-instrumentov-dlya-analiza-dannyh/?ysclid=lx097naokh203391547> (дата обращения: 01.09.2024).

3. Data Validation: Types, Benefits, and Accuracy Process, 21.12.2023 // Atlan [Электронный ресурс]. – URL: <https://atlan.com/what-is-data-validation> (дата обращения: 01.09.2024).

4. MatLab // Skillfactory media [Электронный ресурс]. – URL: <https://blog.skillfactory.ru/glossary/matlab> (дата обращения: 01.09.2024).

5. Mona Rakibe, Why Data Validation Matters: 8 Steps to Boost Data Quality & Value // Telmai [Электронный ресурс]. – URL: <https://www.telm.ai/blog/8-must-follow-steps-to-ensure-successful-data-validation> (дата обращения: 01.09.2024).

6. Sebastian Taylor, Data Validation // CFI [Электронный ресурс]. – URL: <https://corporatefinanceinstitute.com/resources/data-science/data-validation> (дата обращения: 01.09.2024).

7. Top 15 Automation Tools for Data Analytics, 10.01.2024 // Geeksforgeeks [Электронный ресурс]. – URL: <https://www.geeksforgeeks.org/top-15-automation-tools-for-data-analytics> (дата обращения: 01.09.2024).

8. What is data validation? // TIBCO [Электронный ресурс]. – URL: <https://www.tibco.com/glossary/what-is-data-validation#:~:text=Data%20validation%20is%20the%20process,validation%20to%20ensure%20accurate%20results> (дата обращения: 01.09.2024).

9. Громов В. SQL: что это и зачем // Курсы.ру [Электронный ресурс]. – URL: <https://kursy.ru/znaniya/programmirovaniye/chto-takoe-sql-i-zachem/> (дата обращения: 01.09.2024).
10. Дэвид Тейлор. 25 лучших бесплатных инструментов для интеллектуального анализа данных (2024 г.), 05.01.2024 // Guru [Электронный ресурс]. – URL: <https://www.guru99.com/ru/best-data-mining-tools.html> (дата обращения: 01.09.2024).
11. Единый регламент взаимодействия распределенных ситуационных центров, утвержденный решением Межведомственной комиссии по координации деятельности федеральных органов исполнительной власти по созданию системы распределенных ситуационных центров, работающих по единому регламенту взаимодействия (протокол №2 от 07.05.2015).
12. Интеграция данных // Tadviser [Электронный ресурс]. – URL: https://www.tadviser.ru/index.php/Статья:Интеграция_данных (дата обращения: 01.09.2024).
13. Интеграция данных в компании: Что это? Как? И зачем? // DataReview [Электронный ресурс]. – URL: <https://datareview.info/article/integraciya-dannyx-v-kompanii-chto-eto-kak-i-zachem/> (дата обращения: 01.09.2024).
14. Интеграция информационных систем: виды и применение // Deco system [Электронный ресурс]. – URL: <https://www.decosystems.ru/typy-integratsii-informatsionnykh-sistem/?ysclid=lwyofgu26n497266366> (дата обращения: 01.09.2024).
15. Кому и для чего нужен Java? // Skillfactory media [Электронный ресурс]. – URL: <https://blog.skillfactory.ru/komu-i-dlya-chego-nuzhen-python/>
16. Кому и для чего нужен Python? // Skillfactory media [Электронный ресурс]. – URL: <https://blog.skillfactory.ru/komu-i-dlya-chego-nuzhen-python/> (дата обращения: 01.09.2024).
17. Малков А. В. Создание интегрированных систем менеджмента – один из механизмов реализации целей устойчивого развития // Менеджмент в России и за рубежом. – 2019. – № 3. – С. 36–40.
18. Анвар М. Рейтинг инструментов проверки данных, которые стоит изучить 2024. 20.03.2024 // Astera [Электронный ресурс]. – URL: <https://www.astera.com/ru/type/blog/data-validation-tools/> (дата обращения: 01.09.2024).
19. Методы исследований в менеджменте : учеб. пособие / А. Е. Плахин, Е. С. Огородникова, М. С. Хохолуш [и др.]. – Екатеринбург : Изд-во УрГЭУ, 2020. – 248 с.
20. Методы проверки данных: раскрытие бизнес потенциала: стратегии проверки данных для стартапов, 23.03.2024 // FasterCapital [Электронный ресурс]. – URL: <https://fastercapital.com/ru/content/Методы-проверки-данных--раскрытие-бизнес-потенциала--стратегии-проверки-данных-для-стартапов.html> (дата обращения: 01.09.2024).
21. Постановление Администрации Волгоградской области от 23 мая 2011 года № 244-п «Об организации межведомственного информационного взаимодействия в Волгоградской области».

22. Распоряжение комитета информационных технологий и коммуникаций Волгоградской области от 1 июня 2011 года № 03-о/д «О мерах по реализации постановления Администрации Волгоградской области от 23 мая 2011 года № 244-п «Об организации межведомственного информационного взаимодействия в Волгоградской области».

23. Тюфякова В. 9 языков программирования для работы с Big Data // Skillfactory media [Электронный ресурс]. – URL: <https://blog.skillfactory.ru/9-yazykov-dlya-raboty-s-big-data/> (дата обращения: 01.09.2024).

24. Управление качеством и интегрированные системы менеджмента : учебник / А. В. Аверин, Т. В. Барт, С. А. Бука [и др.] ; под ред. А. В. Аверина. – М.: КноРус, 2020. – 357 с.

25. Федюкин В. К. Управление качеством производственных процессов : учеб. пособие. – М.: КноРус, 2018. – 229 с.

26. Что такое интеграция данных? // SAP [Электронный ресурс]. – URL: <https://www.sap.com/central-asia-caucasus/products/technology-platform/what-is-enterprise-integration/data-integration.html> (дата обращения: 01.09.2024).

27. Шарафутдинова Е. Н. Системы инструментов управления качеством : учеб. пособие / Е. Н. Шарафутдинова, О. В. Плинка. – Екатеринбург: Издательство УрГЭУ, 2021. – 180 с.

Е. М. Дубовикова,

кандидат педагогических наук, доцент,
Волгоградский государственный университет
(Волжский филиал)

Г. Г. Егоров,

кандидат юридических наук, доцент,
Волгоградский государственный университет
(Волжский филиал)

ИСПОЛЬЗОВАНИЕ СОВРЕМЕННЫХ НЕЙРОСЕТЕЙ ПРИ ИЗУЧЕНИИ АНГЛИЙСКОГО ЯЗЫКА

Аннотация. В данной статье рассматривается применение нейросетей в обучении иностранным языкам. Проанализировано, как задания и тексты, созданные нейросетями, могут быть использованы в учебном процессе. Кроме того, в статье уточняется понятие «искусственный интеллект» в контексте обучения языкам. Исследуется практическое применение искусственного интеллекта в образовании, особенно в обучении иностранным языкам. Утверждается, что программы искусственного интеллекта могут успешно персонализировать обучение, повысить мотивацию учащихся, существенно сократить время преподавателей на подготовку материалов и позволить им сосредоточиться на более творческих аспектах своей работы.

Ключевые слова: изучение английского языка, нейросети, машинный контроль, контроль освоения курса, современные технологии