



**Описание функциональных характеристик
Продукта Platform V Dictionaries (SDT)**

ОГЛАВЛЕНИЕ

Описание программного решения	4
Термины и определения	4
Назначение и задачи программного решения	4
Структура программного решения.....	4
Функции	4
Функции безопасности	5
Нефункциональные особенности.....	5
Описание функциональных характеристик компонента LNSE (LNSE).....	6
Цель создания	6
Основные функции.....	6
Сценарии использования	7
Предоставление данных или метаинформации справочников	7
Прием уведомлений о бизнес-ошибке.....	7
Импорт файлов с моделями, маршрутизацией или первичным наполнением справочников.....	8
Создание и актуализация реплик внешних справочников.....	8
Предоставление данных или метаинформации реплик справочников	9
Удаление версий данных и/или версий структуры реплик справочников	9
Отображение метаинформации и данных справочников	10
Создание структур справочников.....	11
Актуализация данных внутренних справочников.....	12
Экспорт данных справочников	14
Управление маршрутами справочников	15
Синхронизация внутренних синхронизируемых справочников.....	16
Управление разрешениями на доступ к справочникам.....	16
Описание функциональных характеристик компонента Сервис управления справочными данными (NSIX).....	18
Цель создания	18
Основные функции.....	18
Сценарии использования	21
Описание функциональных характеристик компонента Сервис интеграции справочных данных (DDIS).....	24
Цель создания	24

Основные функции.....	24
Сценарии использования	25
Настройка системы распространения (use-case: Настроить систему)	25
Мониторинг работы системы (use-case: Проконтролировать распространение)	26
Устранение проблем (use-case: Устранить проблему).....	27
Чтение справочников (use-case: Получить изменения справочников)	27
Запись справочников (use-case: Передать справочники)	27
Распространение (use-case: Распространить справочники)	28

Описание программного решения

Термины и определения

Термин/Аббревиатура	Определение
БД	База данных
Платформа	Набор продуктов Platform V, правообладателем которых является АО "СберТех". Перечень таких продуктов обозначен в документации на конкретный продукт
Продукт	Platform V Dictionaries (SDT)
API	Application programming interface. Набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом)

Назначение и задачи программного решения

Platform V Dictionaries – продукт класса MDM (Master Data Management), предназначенный для централизованной работы с мастер-данными (нетранзакционными данными), критичными для бизнеса компании. Продукт Platform V Dictionaries предназначен для хранения мастер-данных, управления ими, а также распространения их конечным потребителям.

Структура программного решения

Продукт включает в себя следующие компоненты:

- Сервис управления справочными данными (NSIX);
- Сервис предоставления справочных данных (LNSE);
- Сервис интеграции справочных данных (DDIS).

Функции

Продукт Platform V Dictionaries предоставляет следующие функциональные возможности:

- Создание и ведение мастер-данных с помощью интерфейса продукта.
- Версионирование данных (как отдельных записей, так и полного массива).
- Поддержка иерархических структур данных – обеспечивается возможность хранения и предоставления данных в древовидной структуре.
- Импорт/экспорт мастер-данных.

- Стандартный бизнес-процесс ведения мастер-данных (включающий согласование и утверждение новой версии данных).
- Централизованное хранение мастер-данных/нормативно-справочной информации.
- Возможность гибкой настройки модели мастер-данных, в том числе "на лету", то есть без необходимости перезапуска и выпуска новых поставок продукта.
- Форматно-логический контроль модели, гарантия ссылочной целостности, проверка дублей при загрузке.
- Обеспечение потребителей достоверной, актуальной и непротиворечивой информацией.
- API загрузки моделей мастер-данных и контента.
- API чтения данных с возможностью фильтрации.
- Мультиотенантность.
- Управление доступом к справочникам на основе ролей.
- Аудит изменений.
- Горизонтальное масштабирование.

Функции безопасности

Для обеспечения безопасности передаваемых данных используется протокол mTLS.

Обеспечение безопасной работы в среде контейнеризации достигается:

- Ролевой моделью доступа к ресурсам в данной среде, определенной на уровне безопасности единицы поставки.
- Istio, выполняющим функцию маршрутизации и создания защищенного соединения.

Действия, производимые компонентами продукта, логируются и записываются в локальный лог контейнера.

Используемые в компонентах механизмы безопасности описаны в документе "Руководство по безопасности" по каждому из компонентов продукта.

Нефункциональные особенности

Продукт не имеет нефункциональных особенностей.

Нефункциональные особенности каждого компонента продукта описаны в разделе "Нефункциональные требования к программному компоненту" документа "Детальная архитектура" каждого из компонентов.

Описание функциональных характеристик компонента LNSE (LNSE)

Цель создания

Компонент Сервис предоставления справочных данных продукта Platform V Dictionaries (далее — LNSE) предназначен для обеспечения работы со справочниками из различных приложений, реализованных на Platform V.

Компонент обеспечивает хранение, визуализацию, актуализацию и получение справочных данных, необходимых для оперативного использования в Platform V.

Компонент предоставляет возможность работы с двумя типами справочников:

- внутренние — справочники, владельцами которых являются приложения, реализованные на Platform V. Первичное наполнение осуществляется с использованием инструментов DevOps, изменения вносятся в административном интерфейсе компонента LNSE.
- внешние — справочники, владельцами которых являются либо реализованные на Platform V приложения, либо внешние АС. Загрузка справочников осуществляется интеграционными адаптерами загрузки.

Доступ к справочникам компонент LNSE предоставляет посредством версионизируемого REST API, исключая таким образом прямое взаимодействие приложений, использующих справочники, с БД справочников.

В рамках компонента LNSE предусмотрены версионирование структур справочников, версионирование данных справочников, логическая изоляция данных и независимое обслуживание запросов приложений, функционирующих в рамках различных экземпляров продуктов различных проектов Platform V.

Основные функции

Описание функций:

Название функции	Потребитель функции	Результат
Предоставление программного интерфейса для получения справочных данных	Приложение, реализованное на Platform V	Выдача структурированных данных справочника
Создание и актуализация реплик справочников, управляемых смежными компонентами и/или внешними АС	Интеграционный адаптер	Загрузка справочника из внешнего источника
Ведение хранилища локальных справочников	Системный администратор справочников, Бизнес-администратор справочников	Создание и актуализация локального справочника
Версионирование структур (моделей) справочников	Бизнес-администратор справочников, Системный	Хранение истории версий структуры справочника

Название функции	Потребитель функции	Результат
	администратор справочников, Интеграционный адаптер	
Версионирование данных справочников	Бизнес-администратор справочников, Системный администратор справочников, Интеграционный адаптер	Хранение истории версий данных справочника
Изоляция данных справочников	Бизнес-администратор справочников, Системный администратор справочников, Интеграционный адаптер	Логическая изоляция справочных данных при мультитенантной инсталляции компонента LNSE

Сценарии использования

Предоставление данных или метаданных справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Приложение запрашивает у компонента LNSE данные справочника либо метаданные справочника. Запрос может содержать:
 - условия фильтрации данных;
 - величину смещения относительно начала выборки.
2. Компонент LNSE возвращает запрошенные данные.
3. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

2А Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Прием уведомлений о бизнес-ошибке

1. Приложение передает компоненту LNSE уведомление о выявлении бизнес-ошибки.
2. Компонент LNSE сохраняет данные и возвращает сообщение об успешном завершении операции.
3. Завершение сценария.

2А Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Импорт файлов с моделями, маршрутизацией или первичным наполнением справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Jenkins передает Компоненту LNSE один из следующих файлов:
 - XML-файл с моделями справочников;
 - JSON-файл с данными справочника;
 - JSON-файл с маршрутизацией справочника;
 - JSON-файл с разрешениями на доступ к справочникам;
 - ZIP-архив набором перечисленных файлов.
2. Компонент LNSE импортирует файлы — в зависимости от содержания файла или архива:
 - создает и регистрирует версию структуры справочника;
 - создает версию данных справочника;
 - сохраняет данные справочника;
 - активирует версию данных и/или версию структуры справочника;
 - регистрирует разрешения на доступ к справочникам.
3. Компонент LNSE возвращает сообщение об успешном завершении операции.
4. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

2A Получена ошибка или предупреждение:

1. Компонент LNSE возвращает сообщение об ошибке или предупреждение.
2. Завершение сценария.

Создание и актуализация реплик внешних справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Адаптер загрузки передает Компоненту LNSE данные для загрузки.
2. Компонент LNSE загружает и сохраняет данные — в зависимости от содержания запроса:
 - создает и регистрирует версию структуры справочника;
 - создает версию данных справочника;
 - сохраняет данные справочника;
 - активирует версию данных и/или версию структуры справочника;
 - создает задачу на отложенную активацию.
3. Компонент LNSE возвращает сообщение об успешном завершении операции.

4. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

2А Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Предоставление данных или метаинформации реплик справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Адаптер загрузки запрашивает у Компонента LNSE данные справочника либо метаинформацию справочника.
2. Компонент LNSE возвращает запрошенные данные.
3. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

2А Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Удаление версий данных и/или версий структуры реплик справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Адаптер загрузки передает Компоненту LNSE запрос на удаление версии структуры или версии данных справочника или справочника.
2. Компонент LNSE удаляет данные.
3. Компонент LNSE возвращает сообщение об успешном завершении операции.
4. Завершение сценария

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

2А Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Отображение метаинформации и данных справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе Компонента LNSE инициирует отображение списка экземпляров справочников.
2. Компонент LNSE отображает список экземпляров справочников.
3. Пользователь выбирает справочник и инициирует отображение метаинформации и данных экземпляра справочника.
4. Компонент LNSE отображает вкладки с метаинформацией и данными экземпляра справочника.
5. Пользователь переходит на вкладку с данными экземпляра справочника.
6. Компонент LNSE отображает записи версии экземпляра справочника.
7. Пользователь, при необходимости, выбирает версию структуры и/или версию данных экземпляра справочника.
8. Компонент LNSE отображает записи выбранной версии экземпляра справочника.
9. Пользователь переходит на вкладку с версиями структуры экземпляра справочника.
10. Компонент LNSE отображает список версий структуры экземпляра справочника.
11. Пользователь выбирает версию структуры и инициирует отображение ее описания.
12. Компонент LNSE отображает окно с описанием выбранной версии структуры.
13. Пользователь инициирует скрытие окна с описанием версии структуры.
14. Компонент LNSE отображает список версий структуры экземпляра справочника.
15. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

5А Переход к списку записей из вкладки с версиями данных:

1. Пользователь переходит на вкладку с версиями данных экземпляра справочника.
2. Компонент LNSE отображает список версий данных версии структуры экземпляра справочника.

3. Пользователь, при необходимости, выбирает версию структуры экземпляра справочника.
4. Компонент LNSE отображает список версий данных выбранной версии структуры.
5. Пользователь выбирает версию данных и инициирует отображение записей.
6. Компонент LNSE отображает вкладку с данными экземпляра справочника. На вкладке отображаются записи выбранной версии.
7. Переход к шагу 7 основного сценария.

5Б Просмотр описания версии структуры:

1. Переход к шагу 7 основного сценария.

10А Отказ от просмотра описания версии структуры:

1. Завершение сценария.

15А Переход к списку записей из вкладки с версиями структуры:

1. Пользователь выбирает версию структуры и инициирует отображение списка версий данных.
2. Компонент LNSE отображает вкладку с версиями данных для выбранной версии структуры экземпляра справочника.
3. Переход к шагу 3А3 альтернативного сценария.

Создание структур справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе компонента LNSE инициирует импорт XML-файла с моделью справочника.
2. Компонент LNSE отображает форму выбора файла.
3. Пользователь указывает путь к локальному файлу с моделью справочника и подтверждает старт импорта.
4. Компонент LNSE импортирует модель справочника — создает и регистрирует версию структуры справочника.
5. Компонент LNSE отображает обновленный список версий структуры справочника и сообщение об успешном завершении операции.
6. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

4А. Получена ошибка:

1. Компонент LNSE возвращает сообщение об ошибке.
2. Завершение сценария.

Актуализация данных внутренних справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе компонента LNSE переходит на вкладку с версиями данных экземпляра справочника.
2. Компонент LNSE отображает список версий данных экземпляра справочника.
3. Пользователь, при необходимости, выбирает версию структуры экземпляра справочника.
4. Компонент LNSE отображает список версий данных выбранной версии структуры.
5. Пользователь инициирует создание версии данных.
6. Компонент LNSE отображает форму создания версии данных.
7. Пользователь заполняет доступные поля и инициирует сохранение.
8. Компонент LNSE создает версию данных, отображает обновленный список версий данных экземпляра справочника и сообщение об успешном завершении операции.
9. Пользователь переходит на вкладку с данными экземпляра справочника.
10. Компонент LNSE отображает данные версии экземпляра справочника.
11. Пользователь выбирает версию структуры, версию данных.
12. Компонент LNSE отображает записи выбранной версии экземпляра справочника.
13. Пользователь инициирует импорт CSV-файла с наполнением справочника.
14. Компонент LNSE отображает форму выбора файла.
15. Пользователь указывает путь к локальному файлу с наполнением справочника и подтверждает старт импорта.
16. Компонент LNSE импортирует данные и отображает обновленный список записей версии данных экземпляра справочника.

17. Пользователь переходит на вкладку с версиями данных экземпляра справочника.
18. Компонент отображает список версий данных экземпляра справочника.
19. Пользователь выбирает версию данных и инициирует ее активацию.
20. Компонент LNSE отображает окно активации версии данных.
21. Пользователь подтверждает выполнение операции.
22. Компонент LNSE активирует версию данных и отображает обновленный список версий данных экземпляра справочника.
23. Пользователь переходит на вкладку с версиями структуры экземпляра справочника.
24. Компонент LNSE отображает список версий структуры экземпляра справочника.
25. Пользователь выбирает версию структуры и инициирует ее активацию.
26. Компонент LNSE отображает окно активации версии структуры.
27. Пользователь подтверждает выполнение операции.
28. Компонент LNSE активирует версию структуры и отображает обновленный список версий структуры экземпляра справочника.
29. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

13А. Добавление записи вручную:

1. Пользователь инициирует добавление записи справочника.
2. Компонент LNSE отображает форму добавления записи.
3. Пользователь заполняет доступные поля и инициирует сохранение.
4. Компонент LNSE сохраняет запись и отображает обновленный список записей версии данных экземпляра справочника.
5. Если необходимо добавить другие записи, то переход к шагу 1 настоящего альтернативного сценария, иначе переход к шагу 17 основного сценария.

17А. Изменение записей вручную:

1. Пользователь инициирует изменение записи справочника.
2. Компонент LNSE отображает форму редактирования записи.

3. Пользователь заполняет доступные поля и инициирует сохранение.
4. Компонент LNSE сохраняет запись и отображает обновленный список записей версии данных экземпляра справочника.
5. Если необходимо изменить другие записи, то переход к шагу 1 настоящего альтернативного сценария, иначе переход к шагу 17 основного сценария.

13B. Ошибка импорта CSV-файла:

1. Компонент LNSE возвращает сообщения об ошибке.
2. Переход к шагу 13 основного сценария.

21A. Отложенная активация:

1. Пользователь указывает дату отложенной активации и инициирует сохранение.
2. Компонент LNSE создает задачу на отложенную активацию версии справочника.
3. Пользователь переходит на вкладку с запланированными активациями.
4. Компонент LNSE отображает список задач на отложенную активацию версий справочника.
5. Завершение сценария.

24A Версия структуры уже активирована:

1. Завершение сценария.

Экспорт данных справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе компонента LNSE переходит на вкладку с версиями данных экземпляра справочника.
2. Компонент LNSE отображает список версий данных экземпляра справочника.
3. Пользователь выбирает версию данных и инициирует экспорт.
4. Компонент LNSE отображает форму со списком ссылок для скачивания.
5. Пользователь переходит по ссылке, указывает путь для сохранения файла и скачивает файл.
6. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

ЗА1. Справочники с категорией конфиденциальности данных К-2:

1. Экспорт недоступен. Завершение сценария.

Управление маршрутами справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе компонента LNSE переходит на вкладку с маршрутами справочника.
2. Компонент LNSE отображает список маршрутов справочника.
3. Пользователь инициирует добавление нового маршрута.
4. Компонент LNSE отображает окно добавления маршрута.
5. Пользователь заполняет доступные поля и инициирует сохранение.
6. Компонент LNSE добавляет маршрут и отображает обновленный список маршрутов справочника.
7. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

ЗА Изменение маршрута:

1. Пользователь выбирает маршрут и инициирует его изменение.
2. Компонент LNSE отображает окно изменения маршрута.
3. Пользователь заполняет и/или изменяет значения доступных полей и инициирует сохранение.
4. Компонент LNSE изменяет маршрут и отображает обновленный список маршрутов справочника.
5. Завершение сценария.

ЗБ Удаление маршрута:

1. Пользователь выбирает маршрут и инициирует его удаление.
2. Компонент LNSE отображает окно удаления маршрута.
3. Пользователь подтверждает удаление маршрута.
4. Компонент LNSE удаляет маршрут и отображает обновленный список маршрутов справочника.

5. Завершение сценария.

Синхронизация внутренних синхронизируемых справочников

ОСНОВНОЙ СЦЕНАРИЙ:

1. Пользователь в административном интерфейсе компонента LNSE переходит на вкладку “Синхронизация”.
2. Компонент LNSE отображает информацию о статусах синхронизации активной версии справочника в смежных блоках.
3. Пользователь отмечает блоки, в которые необходимо передать версию справочника, и инициирует старт синхронизации.
4. Компонент LNSE выполняет синхронизацию.
5. Пользователь запрашивает статус синхронизации.
6. Компонент LNSE отображает статус синхронизации.
7. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

7А Повторный запрос статуса синхронизации

1. переход к шагу 5 основного сценария.

Управление разрешениями на доступ к справочникам

1. Пользователь в административном интерфейсе Компонента LNSE переходит по ссылке “Разрешения”.
2. Компонент LNSE отображает список зарегистрированных разрешений.
3. Завершение сценария.

АЛЬТЕРНАТИВНЫЕ СЦЕНАРИИ:

3А Добавление разрешения:

1. Пользователь инициирует добавление нового разрешения.
2. Компонент LNSE отображает окно добавления разрешения.
3. Пользователь заполняет доступные поля и инициирует сохранение.
4. Компонент LNSE добавляет разрешение и отображает обновленный список разрешений.
5. Завершение сценария.

3Б Удаление разрешения:

1. Пользователь выбирает запись и инициирует ее удаление.
2. Компонент LNSE отображает окно удаления разрешения.
3. Пользователь подтверждает удаление разрешения.
4. Компонент LNSE удаляет разрешение и отображает обновленный список разрешений.
5. Завершение сценария.

Описание функциональных характеристик компонента Сервис управления справочными данными (NSIX)

Цель создания

Компонент Сервис управления справочными данными (NSIX) продукта Platform V Dictionaries (SDT) предназначен для автоматизации работы организации со справочниками общего назначения.

Компонент NSIX обеспечивает:

- централизованное хранение справочных данных организации;
- возможность гибкой настройки модели справочников;
- версионирование моделей и данных справочников;
- нормализацию данных в рамках проектов реализации справочников;
- бизнес-процессы редактирования и создания справочных данных с возможностью двойного ввода, согласования и многоролевого редактирования записей справочников;
- разграничение доступа к данным на уровне справочников и записей справочников;
- аудит изменений справочных данных.

Принципы и описание работы

Компонент построен как классическая MDM-система:

- функциональный тип — Collaboration MDM, система совместного использования данных;
- обработка данных — консолидация с допустимыми типами связи master-slave;
- модель данных — гибридная, т.е. отсутствие преднастроенной модели данных;
- типы распространения данных — публикация/подписка.

Основные функции

Описание основных функций:

Название функции	Потребители функции	Аргументы функции	Результат
Навигация по справочникам	Оператор , Куратор, Аудитор	Список справочников	Просмотр списка справочников в разрезе их версий
Табличная форма списка записей справочника	Оператор ,	1. Справочник. 2. Атрибуты	Отображение списка записей справочника в табличном виде с постраничным разбиением,

Название функции	Потребители функции	Аргументы функции	Результат
	Куратор, Аудитор	для отображения	атрибутным и «быстрым» поиском. Форма конфигурируется списком атрибутов для отображения
Иерархическое дерево записей справочников	Оператор, Куратор, Аудитор	Справочник	Отображение дерева записей иерархических справочников, с возможностью переноса записей между уровнями иерархии
Иерархическое дерево атрибутов записей	Оператор, Куратор, Аудитор	1. Запись справочника. 2. Вкладки. 3. Атрибуты записи	Просмотр карточки записи справочника в виде дерева атрибутов разных типов: композитные, ссылочные, множественные, целочисленные и т.д. Форма настраивается посредством задания последовательности атрибутов, видимости атрибутов, разбиения атрибутов по вкладкам
Редактирование/создание/удаление/изменение статуса записи справочника	Оператор, Куратор	1. Запись справочника. 2. Вкладки. 3. Атрибуты записи	Создана или отредактирована запись справочника посредством создания новой версии. Шаги процесса редактирования определяются бизнес-процессами, специфичными для справочника. Существуют базовые процессы ведения версий записи справочника: редактирование с подтверждением куратором и редактирование по принципу «двойной руки»
Быстрый поиск	Оператор, Куратор, Аудитор	1. Ключ/наименование. 2. Записи справочника	Поиск записей справочника в табличной форме представления записей справочника. Поиск проводится по ключу/наименованию
Атрибутный поиск	Оператор, Куратор, Аудитор	1. Атрибуты модели справочника. 2. Записи справочника	Конструктор поисковых запросов по значениям атрибутов модели справочника
Управление версиями справочника	Оператор, Куратор	1. Справочник. 2. Версия	Создана/отредактирована/согласована/выведена в архив/переведена в действие версия справочника
Управление задачами импорта/экспорта данных	Оператор, Куратор, Аудитор	1. Задача импорта/экспорта. 2. Конфигурация	Старт, остановка задачи импорта/экспорта данных. Просмотр журналов загрузки/выгрузки данных

Название функции	Потребители функции	Аргументы функции	Результат
		задачи импорта/экспорта	
Импорт справочников из внешнего мастера через компонент Сервис интеграции справочных данных (DDIS) продукта Platform V Dictionaries (SDT)	Оператор , Куратор	1. Задача импорта из компонента Сервис интеграции справочных данных (DDIS) продукта Platform V Dictionaries (SDT). 2. Конфигурация задачи импорта	Получение справочников через сервис Dictionary_Transfer_ByArchive
Загрузка справочников с АРМ пользователя	Оператор , Куратор	1 . Справочники. 2. Конфигурация задачи импорта с АРМ пользователя	Загрузка справочников из файла с АРМ пользователя с формированием новых версий справочников или версий записей, в зависимости от типа версионирования справочника. Выполняется требование экспертов кибербезопасности, что при импорте новая версия справочника переходит в статус на согласование и необходимо подтверждение данных загрузки ответственным лицом — пользователь с ролью Куратора
Управление статусами версий справочника	Оператор , Куратор	1. Справочник. 2. Конфигурация ведения версий справочника	Предоставляется два базовых бизнес-процесса по ведению версий справочника: ведение версий по справочнику, ведение версий по записям. Возможна реализация пользовательских процессов, отличных от базовых
Управление статусами версий записи	Оператор , Куратор	1. Справочник. 2. Конфигурация ведения версий записей	Предоставляется два базовых бизнес-процесса по ведению версий записей: ведение версий записей по принципу двойной руки, ведение версий записей с подтверждением куратором
Создание новых моделей справочников	Оператор , Куратор	1. Справочник. 2. Атрибуты справочника	На контурах тестирования. Создание новых типов записей и справочников, формирование атрибутного состава записей

Название функции	Потребители функции	Аргументы функции	Результат
Настройка представления справочников	Оператор, Куратор	1.Справочник. 2. Вкладки. 3. Параметры отображения атрибутов. 4. Подсказки. 5. Табличные формы	На контурах тестирования. Настройка вкладок, последовательности отображения атрибутов, подсказок, табличных форм, формы краткого просмотра
Публикация справочника в витрине	Куратор	Справочник	Справочник опубликован в витрине. Внешние пользователи API получают к нему доступ
Чтение данных справочников	Внешний пользователь API	Справочник	Выгрузка данных справочника для внешних пользователей API
Ведение аудита по данным	Аудитор	1. Запись справочника. 2. Атрибуты записи. 3. Пользователь	Фиксация действий в системе (на основе Аудит (AUDT))
Ведение аудита по действиям	Аудитор	Запись справочника. 2. История действий. 3. Пользователь	Фиксация действий в системе (на основе Аудит (AUDT))

Сценарии использования

№ п/п	Наименование сценария	Основной сценарий	Альтернативный сценарий
1	Создание/редактирование/удаление записей в справочниках	Оператор выбирает запись справочника. Редактирует/удаляет/изменяет статус записи справочника	Отсутствует
2	Создание новых версий справочников	Оператор создает новую версию справочника. Уточняет отдельные записи справочника. Сохраняет изменения	Внешний пользователь API создает новую версию справочника. Уточняет отдельные записи справочника. Сохраняет изменения
3	Связывание версий справочников	Оператор выбирает справочник. Выбирает версию справочника. Оператор выбирает справочник для связи. Выбирает версию	Отсутствует

№ п/п	Наименование сценария	Основной сценарий	Альтернативный сценарий
		справочника для связи. Сохраняет связь между справочниками	
4	Экспорт данных в виде Excel файла	Оператор создает новую задачу экспорта данных. Уточняет параметры экспорта данных. Запускает задачу экспорта на выполнение. Проверяет журнал исполнения задачи экспорта	Отсутствует
5	Создание структуры нового справочника	Куратор создает структуру нового справочника. Задает необходимые параметры новой структуры. Задает поля и другие характеристики структуры. Сохраняет новую структуру справочника	Внешний пользователь API создает структуру нового справочника. Задает необходимые параметры новой структуры. Задает поля и другие характеристики структуры. Сохраняет новую структуру справочника
6	Изменение структуры существующего справочника	Куратор выбирает справочник для изменений. Вносит изменения в структуру полей, меняет их тип и размерность. Сохраняет измененную структуру справочника	Отсутствует
7	Ввод в действие справочников	Куратор выбирает справочник для ввода в действие. Осуществляет ввод справочника в действие	Внешний пользователь API выбирает справочник для ввода в действие. Осуществляет ввод справочника в действие
8	Публикация справочников в витрине	Куратор выбирает справочник для публикации. Осуществляет публикацию справочника	Внешний пользователь API выбирает справочник для публикации. Осуществляет публикацию справочника
9	Импорт данных в справочники	Куратор выбирает справочник для импорта данных. Уточняет параметры импорта данных. Запускает задачу импорта на	Отсутствует

№ п/п	Наименование сценария	Основной сценарий	Альтернативный сценарий
		выполнение. Проверяет журнал исполнения задачи импорта	
10	Просмотр данных справочников	Аудитор выбирает справочник для просмотра. Анализирует данные выбранного справочника	Отсутствует
11	Просмотр истории изменений данных	Аудитор выбирает справочник для просмотра. Анализирует историю изменений данных справочника	Отсутствует
12	Просмотр истории работы регламентных заданий	Аудитор открывает журнал работы регламентных заданий. Анализирует историю работы регламентных заданий	Отсутствует
13	Чтение/поиск данных из справочников	Внешний пользователь API выполняет запрос к витрине данных. На основе результатов запроса выполняется бизнес-логика клиента	Отсутствует
14	Загрузка данных в справочник	Внешний пользователь API выполняет запрос на загрузку данных в справочник. В результате выполнения запроса обновляются данные выбранного справочника	Отсутствует

Описание функциональных характеристик компонента Сервис интеграции справочных данных (DDIS)

Цель создания

Продукт предназначен для:

1. Реализации единой модели данных, используемых для построения интеграционных сценариев.
2. Определения базовых абстракций для реализации этапов интеграционного сценария.
3. Реализации общего механизма исполнения интеграционного сценария на основе реализованных абстракций.
4. Реализации на основе абстракций компонент-содержащих логику типовых шагов интеграционного сценария:
 - инициирование интеграционного сценария;
 - извлечение данных из источника;
 - компоненты форматно-логического контроля;
 - компоненты проверки целостности;
 - компоненты, реализующие парсинг XML, JSON, CSV;
 - компоненты, реализующие загрузку данных в систему-получателя.

Основные функции

Основные функции Продукта:

Название функции	Потребитель функции	Аргументы функции	Результат	Описание
Получение справочников	Сервис интеграции	Версии справочников	Распространение	Создается распространение, содержащее ресурсы поступивших из внешней системы источника, версий справочников
Валидация данных	Микросервисы обработки и публикации	Распространение	Распространение	Проверка целостности данных, проверка ЭП, проверка необходимости загрузки по белым спискам, проверка доступности

Название функции	Потребитель функции	Аргументы функции	Результат	Описание
				загрузки инкремента
Трансформация данных	Микросервисы публикации	Распространение	Распространение	Преобразование формата данных, маппинг имен справочников, полей и формата данных
Распространение справочников	Внешняя система потребитель	Распространение	Состояние системы потребителя	В систему потребитель загружены версии справочников
Конфигурирование сервиса	Сервис интеграции	Параметры конфигурации	Измененные значения параметров	Перечень справочников, доменов, адреса назначения, роадмап распространения, ведение перечня БД для публикации
Контроль распространения	Оператор сервиса интеграции	Распространение	Статус распространения и его шагов	Контроль обработки распространения с отображением статуса прохождения по шагам и возникающих ошибок

Сценарии использования

Настройка системы распространения (use-case: Настроить систему)

Настройка системы распространения включает в себя один из специфичных сценариев:

- настройка дорожной карты распространения;
- изменение перечня и параметров потребителей;
- настройка маппингов и белых списков.

Настройка дорожной карты распространения (use-case: Настроить дорожную карту распространения)

1. Для настройки нужно перейти в UI системы на вкладку Маршруты.
2. Выбрать необходимый маршрут.
3. Выбрать действие добавить, удалить или редактировать маршрут.

4. В режиме добавления маршрута необходимо:
 1. Начать маршрут со шлюза (например, data-mart-gate).
 2. После выбора шлюза или процессора, к маршруту в этой точке можно добавить следующий процессор (например, distribution-database, distribution-dictionary).
5. В режиме редактирования маршрута допускается:
 1. Добавлять и удалять процессоры.
 2. Разветвлять маршрут.

Изменение перечня и параметров потребителей (use-case: Изменить перечень и параметры потребителей)

1. Для изменения потребителей нужно зайти в UI системы на вкладку Процессоры.
2. Выбрать процессор (например, distribution-database).
3. Для процессора выбрать действие Перейти к настройкам.
4. Перейти на вкладку Потребителей.
5. Выбрать действие Добавить, Удалить или редактировать Потребителя.

Настройка маппингов и белых списков (use-case: Настроить маппинги и белые списки)

Настройка маппингов и белых списков проводится в json-файле конфигурации процессора. Для настройки можно указать конкретный перечень справочников, либо загрузку всех поступающих справочников.

Мониторинг работы системы (use-case: Проконтролировать распространение)

Для диагностики работы системы распространения нужно войти в UI системы и выполнить один из следующих специфичных сценариев:

- просмотр поступивших версий справочников;
- проверка состояния распространения.

Просмотр поступивших версий справочников (use-case: Получить изменения справочников)

Выполняется в UI системы в соответствии с документом Руководство оператора.

Проверка состояния распространения (use-case: Проверить состояние шагов распространения)

1. В UI системы выбрать раздел Распространения.
2. Выбрать или отфильтровать нужное распространение.

3. Нажать на просмотр распространения.
4. Проверить этапы распространения на наличие ошибок.

Устранение проблем (use-case: Устранить проблему)

Для устранения проблем с несколькими потребителями, можно воспользоваться сценарием «Инициализация шлюза витрины».

Инициализация шлюза витрины (use-case: Запустить инициализирующую выгрузку)

1. В разделе UI шлюза выбрать шлюз витрины.
2. Выполнить действие Запустить шлюз.

Чтение справочников (use-case: Получить изменения справочников)

Порядок поиска справочника по актуальной версии:

1. Получаем справочник по идентификатору GET /dictionary/{dictionaryId}, либо перечень справочников по фильтру GET /dictionary (наименование, домен, ...).
2. В составе данных о справочнике есть поле actualDictionaryVersion, которое содержит сведения об актуальной версии справочника.
3. По идентификатору актуальной версии справочника, получаем контент версии данных справочника GET /dictionaryVersion/{dictionaryVersionId}/content.

Порядок поиска справочника по конкретной версии:

1. Ищем версии справочника по фильтру GET /dictionaryVersion (наименование справочника, наименование домена, признак актуальности версии).
2. По идентификатору актуальной версии справочника получаем контент версии данных справочника GET /dictionaryVersion/{dictionaryVersionId}/content.

Порядок поиска справочника актуальной версии по цепочке инкрементальных версий:

1. По идентификатору имеющейся версии справочника получаем цепочку инкрементов до актуальной версии справочника GET /dictionaryVersion/{dictionaryVersionId}/chain.
2. Для каждой инкрементальной версии из цепочки скачиваем контент GET /dictionaryVersion/{dictionaryVersionId}/content.

Запись справочников (use-case: Передать справочники)

1. Перебираем все справочники набора.

2. Получаем справочник из витрины GET /dictionary фильтр name, domain.name. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
3. Если справочник не найден — создаем его POST /dictionary. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
4. Берем у справочника актуальную (полную/инкрементальную) версию и проверяем нужно ли загружать новые версии, формируем список версий, которые нужно загрузить в витрину.
5. Создаем поступление в статусе INITIAL POST /input. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
6. Создаем версию(и) справочника POST /dictionaryVersion в статусе DRAFT. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
7. Заливаем контент версии справочника POST (PATCH) /dictionaryVersion/{dictionaryVersionId}/content. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
8. При необходимости обновляем актуальную полную или инкрементальную версию в справочнике(ках) PUT /dictionary/{id}. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.
9. Переводим поступление в статус SUCCESS PUT /input/{id}. В случае неуспеха повторяем шаг n-раз, далее переводим поступление в ERROR.

Распространение (use-case: Распространить справочники)

1. Шлюз (data-mart-gate), в рамках заданного в конфигурации расписания запускает проверку Витрины (data-mart) на наличие обновленных данных (новых Input, справочников и версий справочников).
2. Выкачивает эти данные.
3. Сохраняет в свое хранилище CEPH.
4. Создает распространение (Distribution).
5. Передает этот Distribution Контроллеру.
6. Контроллер получает Distribution от Шлюза и создает цепочку Stage согласно Дорожной карте.
7. После чего Контроллер передает Distribution следующим компонентам, контролирует, чтобы распространение проходило строго по установленной Дорожной карте и в случае отказа одного из компонент, Контроллер пытается добиться восстановления этого компонента, либо при невозможности дальнейшей работы с этим компонентом уведомление пользователей об

ошибке и остановка распространения до принятия мер по устранению причины.

8. Контроллер передает Distribution Публикатору.
9. Публикатор на основе подписок выбирает нужные справочники из Distribution и записывает данные в БД Потребителя.
10. После успешной записи в БД Публикатор помечает свой Stage как успешно обработанный.
11. После завершения всей цепочки и все Stage перешли в статус SUCCESS все распространение переходит в SUCCESS.