

LSFUSION WMS API

ОПИСАНИЕ API ДЛЯ ИНТЕГРАЦИИ КИС – WMS



СОДЕРЖАНИЕ

1	ВВЕДЕНИЕ	6
1.1	REST API	6
1.1.1	Модель клиент-сервер	6
1.1.2	Отсутствие состояния	6
1.1.3	Кэширование	6
1.1.4	Единообразие интерфейса	6
1.1.5	Слои	6
1.2	ФОРМАТ JSON	7
1.2.1	Особенности использования спецсимволов	7
2	LSFUSION WMS API – ОБЩИЕ ПРИНЦИПЫ	8
2.1	МЕТОДЫ ЗАПРОСОВ	8
2.2	POST – общее описание метода	8
2.2.1	POST – общие правила формирования JSON	9
2.2.2	POST – JSON для одиночного объекта	10
2.2.3	POST – JSON для пакетной передачи данных	10
2.2.4	POST – порядок и состав запросов	11
2.3	GET – общее описание методов	12
2.3.1	GET – получение данных одного объекта	12
2.3.2	GET – получение данных по всем объектам одного класса	13
2.3.3	GET – специальные методы	13
2.4	Базовая аутентификация	13
3	ЛОГИКА ОБМЕНА. ОПИСАНИЕ КЛАССОВ И СВОЙСТВ JSON LSFUSION WMS API	14
3.1	Принципиальные подходы к интеграции КИС – WMS	14
3.2	Идентификатор объекта externalId	15
3.3	Создание, изменение и удаление объектов и значений их свойств	15
3.4	Статусы документов WMS и фактическое количество	16
3.5	Определение соответствий и точек ввода объектов КИС – WMS	17
3.6	Справочник WMS «Контрагенты». Описание свойств класса legalEntity	18
3.6.1	Логика обмена «Контрагенты»	18
3.6.2	POST – метод для передачи данных по Контрагентам из КИС в WMS	18
3.6.3	POST – тело JSON для передачи данных по Контрагенту из КИС в WMS	18
3.6.4	GET – метод для получения в КИС данных по конкретному Контрагенту из WMS	19
3.6.5	GET – метод для получения в КИС данных по всем Контрагентам из WMS	19
3.6.6	GET – тело ответа JSON с данными по Контрагенту из WMS	19
3.6.7	JSON – описание свойств класса legalEntity	19
3.7	Справочник WMS «Единицы измерения». Описание свойств класса uom	20
3.7.1	Логика обмена «Единицы измерения»	20
3.7.2	POST – метод для передачи данных по ЕИ из КИС в WMS	20
3.7.3	POST – тело JSON для передачи данных по ЕИ из КИС в WMS	20
3.7.4	GET – метод для получения в КИС данных по конкретной ЕИ из WMS	21
3.7.5	GET – метод для получения в КИС данных по всем ЕИ из WMS	21
3.7.6	GET – тело ответа JSON с данными по ЕИ из WMS	21
3.7.7	JSON – описание свойств класса uom	21
3.8	Справочник WMS «Категории». Описание свойств класса category	22
3.8.1	Логика обмена «Категории»	22
3.8.2	POST – метод для передачи данных по Категории из КИС в WMS	22
3.8.3	POST – тело JSON для передачи данных по Категориям из КИС в WMS	22
3.8.4	GET – метод для получения в КИС данных по конкретной Категории из WMS	23
3.8.5	GET – метод для получения в КИС данных по всем Категориям из WMS	23
3.8.6	GET – тело ответа JSON с данными по Категории из WMS	23
3.8.7	JSON – описание свойств класса category	24
3.9	Справочник WMS «Категории 3PL». Описание свойств класса billingCategory	26
3.9.1	Логика обмена «Категории 3PL»	26
3.9.2	POST – метод для передачи данных по Категории 3PL из КИС в WMS	26

3.9.3	POST – тело JSON для передачи данных по Категориям 3PL из КИС в WMS	26
3.9.4	GET – метод для получения в КИС данных по конкретной Категории 3PL из WMS	27
3.9.5	GET – метод для получения в КИС данных по всем Категориям 3PL из WMS	27
3.9.6	GET – тело ответа JSON с данными по Категории 3PL из WMS	27
3.9.7	JSON – описание свойств класса <code>billingCategory</code>	27
3.10	Справочник WMS «Товары». Описание свойств класса <code>item</code>	28
3.10.1	Логика обмена «Товары»	28
3.10.2	POST – метод для передачи данных по Товару из КИС в WMS	28
3.10.3	POST – тело JSON для передачи данных по Товару из КИС в WMS	28
3.10.4	GET – метод для получения в КИС данных по конкретному Товару из WMS	29
3.10.5	GET – метод для получения в КИС данных по всем Товарам из WMS	29
3.10.6	GET – тело ответа JSON с данными по Товару из WMS	29
3.10.7	JSON – описание свойств класса <code>item</code>	30
3.11	Объект WMS «Упаковка товара». Описание свойств класса <code>userPack</code>	32
3.11.1	Логика обмена «Упаковка товара»	32
3.11.2	POST – метод для передачи данных по Упаковке товара из КИС в WMS	32
3.11.3	POST – тело JSON для передачи данных по Упаковкам товара из КИС в WMS	32
3.11.4	GET – метод для получения в КИС данных по конкретной Упаковке товара из WMS ...	33
3.11.5	GET – метод для получения в КИС данных по всем Упаковкам товара из WMS	33
3.11.6	GET – тело ответа JSON с данными по Упаковке товара из WMS	33
3.11.7	JSON – описание свойств класса <code>userPack</code>	34
3.12	Структура зависимостей классов <code>uom</code> , <code>item</code> , <code>userPack</code> , <code>lot</code> , <code>lotPack</code> , <code>lotPackBarCode</code>	35
3.13	Справочник WMS «Партии». Описание свойств класса <code>lot</code>	37
3.13.1	Логика обмена «Партии»	37
3.13.2	POST – метод для передачи данных по Партии из КИС в WMS	37
3.13.3	POST – тело JSON для передачи данных по Партии из КИС в WMS	37
3.13.4	GET – метод для получения в КИС данных по конкретной Партии из WMS	38
3.13.5	GET – метод для получения в КИС данных по всем Партиям из WMS	38
3.13.6	GET – тело ответа JSON с данными по Партии из WMS	38
3.13.7	JSON – описание свойств класса <code>lot</code>	39
3.14	Справочник WMS «Штрихкоды». Описание свойств классов <code>LotPack</code> , <code>lotPackBarCode</code>	40
3.14.1	Логика обмена «Штрихкоды»	40
3.14.2	POST – метод для передачи данных по Штрихкодам из КИС в WMS	41
3.14.3	POST – тело JSON для передачи данных по Штрихкодам из КИС в WMS	41
3.14.4	GET – метод для получения в КИС данных по всем штрихкодам из WMS	41
3.14.5	GET – метод для получения в КИС данных <code>lotPack</code> и Штрихкодам из WMS по <code>externalId</code> товара или упаковки и партии	41
3.14.6	GET – метод для получения в КИС данных по штрихкоду конкретного товара или конкретной упаковки товара с конкретной партией (<code>lotPack</code>) из WMS	41
3.14.7	GET – тело ответа JSON с данными по <code>lotPack</code> из WMS	42
3.14.8	JSON – описание свойств класса <code>lotPackBarCode</code> , <code>LotPack</code>	42
3.15	Документ WMS «Поступление». Описание свойств класса <code>userReceipt</code>	43
3.15.1	Логика обмена «Поступление»	43
3.15.2	POST – метод для передачи данных по Поступлению из КИС в WMS	44
3.15.3	POST – тело JSON для передачи данных по Поступлению из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»	44
3.15.4	POST – тело JSON для передачи данных по Поступлению и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»	44
3.15.5	GET – метод для получения в КИС статуса по конкретному Поступлению из WMS	46
3.15.6	GET – тело ответа JSON с данными по статусу конкретного Поступления из WMS	46
3.15.7	GET – метод для получения в КИС статусов по Поступлениям за указанный период из WMS	47
3.15.8	GET – тело ответа JSON с данными по статусу Поступлений за указанный период из WMS	47
3.15.9	GET – метод для получения в КИС полных данных по конкретному Поступлению из WMS	47

3.15.10	GET – метод для получения в КИС полных данных по всем Поступлениям из WMS	48
3.15.11	GET – тело ответа JSON с полными данными по Поступлению из WMS	48
3.15.12	JSON – описание свойств класса <code>userReceipt</code>	48
3.16	Документ WMS «Заказ (отгрузка)». Описание свойств класса <code>order</code>	51
3.16.1	Логика обмена «Заказ (отгрузка)»	51
3.16.2	POST – метод для передачи данных по заказу из КИС в WMS	52
3.16.3	POST – тело JSON для передачи данных по Заказу из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»	52
3.16.4	POST – тело JSON для передачи данных по Заказу и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»	52
3.16.5	GET – метод для получения в КИС статуса по конкретному Заказу из WMS	53
3.16.6	GET – тело ответа JSON с данными по статусу конкретного Заказа из WMS	53
3.16.7	GET – метод для получения в КИС статусов по Заказам за указанный период из WMS	53
3.16.8	GET – тело ответа JSON с данными по статусу Заказов за указанный период из WMS	54
3.16.9	GET – метод для получения в КИС полных данных по конкретному Заказу из WMS	54
3.16.10	GET – метод для получения в КИС полных данных по всем Заказам из WMS	54
3.16.11	GET – тело ответа JSON с полными данными по Заказу из WMS	55
3.16.12	JSON – описание свойств класса <code>order</code>	55
3.17	Документ WMS «отгрузка». Описание свойств класса <code>shipment</code>	58
3.17.1	Логика обмена «Отгрузка»	58
3.17.2	POST – метод для передачи данных по Отгрузке из КИС в WMS	58
3.17.3	POST – тело JSON для передачи данных по Отгрузке из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»	59
3.17.4	POST – тело JSON для передачи данных по Отгрузке и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»	59
3.17.5	GET – метод для получения в КИС статуса по конкретной Отгрузке из WMS	59
3.17.6	GET – тело ответа JSON с данными по статусу конкретной Отгрузки из WMS	60
3.17.7	GET – метод для получения в КИС статусов по Отгрузкам за указанный период из WMS	60
3.17.8	GET – тело ответа JSON с данными по статусу Отгрузок за указанный период из WMS	61
3.17.9	GET – метод для получения в КИС полных данных по конкретной Отгрузке из WMS	61
3.17.10	GET – метод для получения в КИС полных данных по всем Отгрузкам из WMS	61
3.17.11	GET – тело ответа JSON с полными данными по Отгрузке из WMS	61
3.17.12	JSON – описание свойств класса <code>shipment</code>	62
3.18	Документ WMS «Перемещение между складами». Описание свойств класса <code>transfer</code>	65
3.18.1	Логика обмена «Перемещение между складами»	65
3.18.2	POST – ответ сервера 409 на запрос, в котором указан склад КИС без настроенного соотношения со складом WMS	66
3.18.3	POST – метод для передачи данных по Перемещению из КИС в WMS	66
3.18.4	POST – тело JSON для передачи данных по Перемещению из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»	66
3.18.5	POST – тело JSON для передачи данных по Перемещению и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»	67
3.18.6	GET – метод для получения в КИС полных данных по конкретному Перемещению из WMS	67
3.18.7	GET – метод для получения в КИС полных данных по всем Перемещениям из WMS	67
3.18.8	GET – тело ответа JSON с полными данными по Перемещению из WMS	67
3.18.9	JSON – описание свойств класса <code>transfer</code>	68
3.18.10	Соотношение складов КИС – WMS	69
3.19	Документ WMS «Инвентаризация (по товару)». Описание свойств класса <code>inventoryItem</code>	70
3.19.1	Логика обмена «Инвентаризация (по товару)»	70

3.19.2	POST – метод для передачи данных по Инвентаризации из КИС в WMS	70
3.19.3	POST – тело JSON для передачи данных по Инвентаризации из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»	71
3.19.4	POST – тело JSON для передачи данных по Инвентаризации и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»	71
3.19.5	GET – метод для получения в КИС полных данных по конкретной Инвентаризации из WMS по Коду документа (vector)	71
3.19.6	GET – метод для получения в КИС полных данных по конкретной Инвентаризации из WMS по <code>externalId</code>	71
3.19.7	GET – метод для получения в КИС полных данных по Инвентаризациям из WMS за период (Вектор)	72
3.19.8	GET – метод для получения в КИС полных данных по всем Инвентаризациям (по товару) из WMS	72
3.19.9	GET – тело ответа JSON с полными данными по Инвентаризации из WMS	72
3.19.10	JSON – описание свойств класса <code>inventoryItem</code>	72
4	ЛОГИКА ОБМЕНА. ОПИСАНИЕ СПЕЦИАЛЬНЫХ МЕТОДОВ POST И GET	74
4.1	Остатки из WMS. Описание свойств МЕТОДА <code>remainingItems</code>	74
4.1.1	GET – метод для получения в КИС данных по остаткам товаров из WMS	74
4.1.2	GET – тело ответа JSON с данными по остаткам из WMS	74
4.1.3	JSON – описание свойств класса <code>remainingItems</code>	74
4.2	Изображение товара. Описание свойств класса <code>ItemImage</code>	76
4.2.1	Логика обмена «Изображение товара»	76
4.2.2	POST – метод для передачи изображения товара из КИС в WMS	76
4.2.3	POST – тело JSON для передачи изображения товара из КИС в WMS	76
4.2.4	JSON – описание свойств класса <code>ItemImage</code>	76
4.3	Серийные номера из документа WMS. Описание свойств класса <code>SeriesNumbers</code>	77
4.3.1	Логика обмена «Серийные номера из документа WMS»	77
4.3.2	GET – метод для получения в КИС данных по серийным номерам из документов WMS ..	77
4.3.3	GET – тело ответа JSON с данными по серийным номерам из документов WMS	77
4.3.4	JSON – описание свойств класса <code>SeriesNumbers</code>	77
4.4	Коды маркировки (Честный знак). Описание свойств класса <code>marks</code>	78
4.4.1	Логика обмена «Коды маркировки»	78

1 ВВЕДЕНИЕ

1.1 REST API

REST (REpresentational State Transfer) – популярный архитектурный подход для создания API в современном мире. Подробнее о REST: <https://ru.wikipedia.org/wiki/REST>. В IsFUSION WMS API реализован RESTful подход и соблюдены обязательные условия-ограничения:

1.1.1 Модель клиент-сервер

Данное ограничение указывает, что взаимодействие в рамках интеграции должно быть построено между клиентами и сервером. Сервер – это система, которая имеет требуемые ресурсы (WMS), а клиент – это система, которой нужно взаимодействовать с ресурсами, хранящимися на сервере:

- ⊖ Сервер: IsFUSION WMS;
- ⊖ Клиенты: КИС (ERP, 1C) и другие смежные системы Заказчика.

1.1.2 Отсутствие состояния

В период между запросами клиента никакая информация о состоянии клиента на сервере не хранится (Stateless protocol или «протокол без сохранения состояния»). Все запросы от клиента должны быть составлены так, чтобы сервер получил всю необходимую информацию для выполнения запроса. Состояние сессии при этом сохраняется на стороне клиента. Клиент инициирует отправку запросов, когда он готов (возникает необходимость) перейти в новое состояние.

1.1.3 Кэширование

Клиент должен выполнять кэширование ответов сервера: все ответы сервера должны записываться в базу данных клиента с целью предотвращения получения клиентом устаревших или неверных данных в ответ на последующие запросы.

1.1.4 Единообразие интерфейса

- ⊖ Идентификация ресурсов – все ресурсы идентифицируются в запросах с использованием идентификатора `externalId`
- ⊖ Манипуляция ресурсами через представление – если клиент хранит представление ресурса, включая метаданные – он обладает достаточной информацией для модификации или удаления ресурса.
- ⊖ Самоописываемые запросы – каждый запрос содержит достаточно информации, чтобы понять, каким образом его обрабатывать.

1.1.5 Слои

IsFUSION WMS API имеет эндпоинты (endpoints), к которым нужно обращаться, отправляя HTTP запросы.

Эндпоинты IsFUSION WMS API:

- ⊖ POST /exes – для передачи данных в WMS
- ⊖ GET /exes – для получения данных из WMS

Доступ к эндпоинту внутри корпоративной сети передачи данных (КСПД) или извне обеспечивается Заказчиком, при этом могут использоваться промежуточные узлы и/или vpn, которые подчиняются политике безопасности предприятия с целью обеспечения конфиденциальности информации.

1.2 ФОРМАТ JSON

JSON (англ. JavaScript Object Notation) – текстовый формат обмена данными, основанный на JavaScript:

- ⊖ Подробнее о формате JSON: <https://ru.wikipedia.org/wiki/JSON>
- ⊖ Спецификация JSON: <https://www.json.org/json-ru.html>

1.2.1 Особенности использования спецсимволов

В значениях свойств, передаваемых в теле запроса в формате JSON, могут присутствовать специальные символы. Такие символы нужно экранировать с помощью обратного слеша \. Например, \" для двойных кавычек " или \\ для обратного слеша \. Подробнее о других спецсимволах <https://w.wiki/8xFb>

Пример описания запроса со следующими полями:

- ⊖ Наименование организации `name` – Компания Zima\Leto
- ⊖ Комментарии к организации `comments` – Территория поселка "Беляево". От шлагбаума позвонить.

```
{
  "legalEntity": [
    {
      "externalId": "da861e9b-1a0a-11e8-bc4b-d8cb8a5022a01",
      "id": "012345",
      "name": "Компания Zima\\Leto",
      "comments": "Территория поселка \"Беляево\". От шлагбаума позвонить."
    }
  ]
}
```

2 LSFUSION WMS API – ОБЩИЕ ПРИНЦИПЫ

2.1 МЕТОДЫ ЗАПРОСОВ

IsFUSION WMS API – программный интерфейс для интеграции КИС – корпоративной информационной системы (ERP, 1C) с WMS. Он даёт возможность обмениваться информацией между КИС и WMS.

Методы IsFUSION WMS API позволяют изменять данные в WMS, например, передавать справочники товаров, контрагентов, приходные и расходные документы, и получать данные из WMS, такие как информация о фактически принятых или отгруженных товарах или статусы приходных и расходных документов в WMS.

Работа с API состоит из отправки запроса и получения ответа. Для этого необходимо настроить интеграцию КИС (1C, ERP) с WMS.

В IsFUSION WMS API используется два основных метода запросов:

- ⊖ POST – для передачи данных из КИС в WMS в теле запроса в формате JSON
- ⊖ GET – для получения данных в КИС из WMS в теле ответа в формате JSON

2.2 POST – ОБЩЕЕ ОПИСАНИЕ МЕТОДА

Метод POST предназначен для направления запроса, при котором сервер принимает данные, заключённые в тело сообщения в формате JSON. В рамках POST запроса произвольное количество данных любого типа может быть отправлено на сервер в теле сообщения запроса. Поля заголовка в POST запросе указывают на тип содержимого.

В IsFUSION WMS API POST-запрос – это HTTP-запрос к эндпоинту `POST /exec?action=IncomeApi.insertUpdate`

Запрос должен быть передан с заголовками: `Content-Type`, `Content-Length`, `Host`

Пример запроса POST:

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

Где `http://localhost:8080` – это адрес сервера WMS.

В теле запроса должен быть передан JSON файл, содержащий данные о добавляемых/изменяемых объектах (классы и свойства), с указанием уникального в рамках системы WMS идентификатора объекта `externalId`.

Пример тела запроса POST в формате JSON:

```
{
  "legalEntity": [
    {
      "externalId": "da861e9b-1a0a-11e8-bc4b-d8cb8a5022a02",
      "id": "CA",
      "name": "Центральный Аппарат",
      "fullName": "Центральный Аппарат",
      "address": "Москва",
      "email": "example1@email.com",
      "archived": false
    }
  ]
}
```


В результате получения этого запроса в IsFUSION WMS по данным из тела запроса JSON будут созданы отсутствующие объекты и обновлены существующие.

В случае успешного получения запроса сервер возвращает статус: 200

Если запрос не принят (содержит ошибки синтаксиса запросе или в теле запроса) – сервер возвращает статус: 500

В случае, если запрос принят, но при записи полученных в запросе данных в БД WMS срабатывает системное ограничение, то в теле ответа возвращается сообщение со статусом: 409

Пример сообщения для статуса 409:

```
Нарушена уникальность свойства (p11961
'IncomeApi.externalId[IncomeApi.IncomeAPIObject] 'Внешний ID' [IncomeApi(141:27)]')
для объектов
-----
|Строка (ИД)|
-----
|      00001|
|      00002|
-----
```

2.2.1 POST – общие правила формирования JSON

Пример тела запроса POST в формате JSON:

```
{
  "order": [
    {
      "externalId": "85a07281-395e-11e8-827d-34080432333d1",
      "legalEntity": "6543197c-8a35-11ec-94fd-00155d012212",
      "number": "Test-046",
      "dateTime": "2024-03-25 12:43:08",
      "typeDocument": "",
      "inStock": true,
      "orderLine": [
        {
          "externalId": "8ba75645-395e-11e8-827d-34080432333d1",
          "item": "85a072d3-395e-11e8-827d-34080432333d2",
          "quantity": 15
        }, {
          "externalId": "85a07350-395e-11e8-827d-34080432333d2",
          "item": "68aca555-0242-11ec-ba90-00155d01220e",
          "quantity": 20
        }
      ]
    }
  ]
}
```

- ⊖ **order** – название класса в теле запроса всегда со строчного символа;
- ⊖ **externalId** – всегда обязательное свойство;
- ⊖ **dateTime** – дата/время в формате "ГГГГ-ММ-ДД ЧЧ:ММ:СС";
- ⊖ **typeDocument** – строковое значение всегда в кавычках "Значение123". Свойство не должно передаваться, если такого свойства у объекта в КИС нет. При передаче в значении свойства "", как в примере выше, в WMS будет записано значение <пустое значение>;
- ⊖ **inStock** – булевый тип значения, обязательно **true**, если передается «истина». В случае, когда передается «ложь», может быть **false** или любое другое.

☞ `quantity` – числовое значение может быть передано без кавычек.

2.2.2 POST – JSON для одиночного объекта

По стандартной практике запрос POST может быть создан и направлен при создании или изменении в КИС объекта, по которому производится обмен данными. Этот запрос может содержать все данные по созданному объекту или только измененные данные.

Пример JSON для одиночного объекта:

```
{
  "uom": [
    {
      "externalId": "a281d921-5b3d-11ed-80f5-5ef3fcdad76b",
      "id": "PCE",
      "name": "Штука",
      "shortName": "шт.",
      "okei": "796",
      "denySplit": true
    }
  ]
}
```

Добавленные или измененные объекты одного класса, например, при первичной передаче элементов справочников из КИС в WMS, могут быть переданы одним POST запросом в одном массиве для класса.

Пример JSON для нескольких объектов одного класса:

```
{
  "uom": [
    {
      "externalId": "a281d921-5b3d-11ed-80f5-5ef3fcdad76b",
      "id": "PCE",
      "name": "Штука",
      "shortName": "шт.",
      "okei": "796",
      "denySplit": true
    }, {
      "externalId": "00c01eaa-fdbb-43b9-86e7-e69d33d8050b",
      "id": "KGM",
      "name": "Килограмм",
      "shortName": "кг",
      "okei": "166"
    }, {
      "externalId": "9d648d22-98af-4f21-a108-c9fe56dc340f",
      "id": "MTR",
      "name": "Метр",
      "shortName": "м",
      "okei": "006"
    }
  ]
}
```

2.2.3 POST – JSON для пакетной передачи данных

Для исключения коллизий связанных с отсутствием объектов на сервере (в WMS) при передаче клиентами (ERP, 1C) других связанных с ними объектов и гарантированной записи в БД WMS данных по всем объектам из КИС может быть создан и направлен «пакетный» POST запрос, включающий все связанные объекты (классы).

Например, элемент справочника «Единицы измерения» не связан с элементами других справочников, а элемент справочника «Товары» содержит атрибуты «Единица измерения» и «Категория» и ссылки на элементы этих справочников. Соответственно в момент передачи POST запроса для товара его единица измерения и категория, в которую он входит, должны быть переданы в более ранних запросах либо включены в «пакетный» JSON, включающий несколько классов.

Пример JSON для пакетной передачи данных:

```
{
  "uom": [
    {
      "externalId": "a281d921-5b3d-11ed-80f5-5ef3fcdad76b",
      "id": "PCE",
      "name": "штука",
      "shortName": "шт.",
      "okei": "796",
      "denySplit": true
    }
  ],
  "category": [
    {
      "externalId": "6d3e5375-0c21-47a6-af5d-a2f3fb8f5dc6",
      "name": "IT-оборудование"
    }
  ],
  "item": [
    {
      "externalId": "660671b6-87e2-4d70-a7c3-fd6300c55358",
      "id": "001",
      "name": "Товар_1 наименование",
      "fullName": "Товар_1 полное наименование",
      "article": "12345",
      "uom": "a281d921-5b3d-11ed-80f5-5ef3fcdad76b",
      "category": "6d3e5375-0c21-47a6-af5d-a2f3fb8f5dc6"
    }
  ]
}
```

2.2.4 POST – порядок и состав запросов

Как описано в разделах выше тело запроса POST может содержать как отдельный объект, так и пакет объектов в одном теле.

При отправке запросов POST, в теле которых содержится один объект, должен соблюдаться следующий порядок отправки запроса с определенным классом из КИС в WMS:

- ⊖ uom всегда перед item
- ⊖ category всегда перед item
- ⊖ item всегда перед userPack
- ⊖ item всегда перед userReceipt
- ⊖ item всегда перед shipment
- ⊖ legalEntity всегда перед userReceipt
- ⊖ legalEntity всегда перед shipment

В системе WMS для запросов POST, в теле которых содержится пакет объектов, гарантирован такой порядок инициализации модулей API, чтобы при пакетной загрузке в независимости от порядка следования объектов в файле JSON соблюдался следующий порядок создания объектов:

- ⊖ uom всегда перед item

- ⊖ category **всегда перед** item
- ⊖ item **всегда перед** userPack
- ⊖ item **всегда перед** userReceipt
- ⊖ item **всегда перед** shipment
- ⊖ legalEntity **всегда перед** userReceipt
- ⊖ legalEntity **всегда перед** shipment

2.3 GET – ОБЩЕЕ ОПИСАНИЕ МЕТОДОВ

Методы GET предназначены для получения клиентами информации от сервера. В рамках GET-запроса должен быть использован один из методов с указанием идентификатора объекта `externalId`, названия класса или других параметров обозначенных ниже. Сервер возвращает ответ в формате JSON.

В IsFUSION WMS API GET-запрос – это HTTP-запрос к эндпоинту `GET /exec?action=` с различными действиями и параметрами.

Запрос должен быть передан с заголовком `Host` и пустым телом запроса.

Примеры запросов GET:

```
http://localhost:8080/exec?action=getObject&p=6d3e5375-0c21-47a6-af5d-a2f3fb8f5dc6
```

```
http://localhost:8080/exec?action=IncomeApi.getObjectsUom
```

```
http://localhost:8080/exec?action=IncomeApi.getUserReceiptStatusesPeriod&p=21.03.2023 7:10&p=21.05.2023 7:11
```

Где `http://localhost:8080` – это адрес сервера WMS.

Пример ответа на GET:

```
{
  "externalId": "6884cee5-62b4-49b7-9a9d-387de56c0813",
  "barcodes": [
    "6971926016496"
  ],
  "name": "LED телевизор Haier 75 Smart TV S3",
  "fullName": "LED телевизор Haier 75 Smart TV S3",
  "article": "0476459",
  "uomExternalId": "846771",
  "categoryExternalId": "18244802",
  "length": 230,
  "width": 1110,
  "height": 1810,
  "grossWeight": 31.5
}
```

2.3.1 GET – получение данных одного объекта

Для получения данных из WMS по любому одному объекту должен быть использован метод `GET /exec?action=getObject` с параметром `&p=externalId` объекта

Пример запроса GET для получения данных одного объекта:

```
http://localhost:8080/exec?action=getObject&p=6d3e5375-0c21-47a6-af5d-a2f3fb8f5dc6
```

В результате успешного выполнения запроса сервер возвращает статус 200. При этом тело ответа содержит JSON-файл, описывающий конкретный объект, или, если объект не найден, тело ответа будет пустым.

Если запрос не принят (содержит ошибки синтаксиса в запросе) – сервер возвращает статус: 500

2.3.2 GET – получение данных по всем объектам одного класса

Для получения данных из WMS по всем объектам одного класса должен быть использован метод GET /exec?action=IncomeApi.getObjectНазваниеКласса

Пример запроса GET для получения данных по всем объектам одного класса:

```
http://localhost:8080/exec?action=IncomeApi.getObjectsLegalEntity
```

В результате успешного выполнения запроса сервер возвращает статус 200. При этом тело ответа содержит JSON-файл, описывающий все объекты запрошенного класса.

Если запрос не принят – сервер возвращает статус: 500

2.3.3 GET – специальные методы

Для получения из WMS специализированных или отчетных данных, например данные только по статусу документов или данные по остаткам, с определенными параметрами, например за какой-либо период, должны быть использованы специальные методы GET /exec?action=IncomeApi.Специализированный_запрос + Параметры

Пример запроса GET для получения данных по выполнению задач по конкретному документу:

```
http://localhost:8080/exec?action=IncomeApi.taskExecutesDoc&p=a169e29c-da8a-4f64-8586-8ca18177ff00
```

Пример запроса GET для получения статусов всех документов «Заказ» за период:

```
http://localhost:8080/exec?action=IncomeApi.getOrderStatusesPeriod&p=21.03.2023 7:10&p=21.05.2023 7:11
```

Детальное описание специальных методов GET представлено в разделах ниже.

2.4 БАЗОВАЯ АУТЕНТИФИКАЦИЯ

В IsFUSION WMS API для некоторых методов реализована обязательная базовая аутентификация. Если базовая аутентификация обязательна для метода, то это явно обозначено в описании этого метода в разделах ниже, иначе метод должен быть передан без базовой аутентификации.

При базовой аутентификации клиент вместе с запросом отправляет серверу логин и пароль. Эти данные отправляются в заголовке запроса Authorization в виде base64 кода.

```
Authorization: Basic base64_encode(login:password)
```

В случае если авторизация обязательна, но при отправке метода данные для авторизации не были отправлены в заголовке запроса или логин и пароль некорректные, то сервер возвращает ответ 401 (Unauthorized).

3 ЛОГИКА ОБМЕНА. ОПИСАНИЕ КЛАССОВ И СВОЙСТВ JSON LSFUSION WMS API

3.1 ПРИНЦИПИАЛЬНЫЕ ПОДХОДЫ К ИНТЕГРАЦИИ КИС – WMS

При настройке интеграции КИС – WMS на объекте автоматизации должны быть обеспечены следующие принципиальные подходы:

- 3.1.1.1.1 В качестве основного идентификатора объектов обмена служит свойство (атрибут) объекта WMS – `externalId` – это уникальный в рамках системы WMS идентификатор объекта (см. раздел «3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID»).
- 3.1.1.1.2 КИС является ведущей системой, формирование всех объектов обмена (Контрагентов, Единиц измерения, Номенклатуры, Партий, Приходных и Расходных документов, Перемещений и др.) производится в КИС. То есть КИС является точкой ввода для всех объектов обмена.
- 3.1.1.1.3 WMS в общем случае является ведомой системой и получает исходные данные из КИС. При этом получение данных служит для создания или изменения объекта, но не для его удаления (см. раздел «3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ»).
- 3.1.1.1.4 В КИС должны быть созданы специальные интерфейсы (сценарии для автоматической выгрузки, либо кнопки для ручного управления выгрузкой) для передачи созданных объектов в WMS.
- 3.1.1.1.5 В КИС должны быть созданы специальные интерфейсы (сценарии для автоматической загрузки, либо кнопки для ручного управления загрузкой) для получения данных по объектам обмена из WMS, например, статусы документов из WMS и фактическое количество по складским операциям из WMS.
- 3.1.1.1.6 В КИС должны быть созданы инструменты управления запросами (методами) и их кэшированием.
- 3.1.1.1.7 Факт произведенной в WMS операции должен быть получен в КИС в соответствующий документ (см. раздел «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО»). Таким образом синхронизируется регламентированный учет и складской (физический) учет в интегрируемых системах. Исходя из этого также следует, что проведение регламентированного документа, который отвечает за движение остатков в КИС, например, «Поступление товаров и услуг», должно производиться только после получения финального статуса документа из WMS и фактического количества по строкам документа.
- 3.1.1.1.8 В отдельных согласованных в данном документе случаях точкой ввода объектов обмена может быть WMS или КИС и WMS одновременно. Точки ввода объектов обмена представлены в разделе «3.5 ОПРЕДЕЛЕНИЕ СООТВЕТСТВИЙ И ТОЧЕК ВВОДА ОБЪЕКТОВ КИС – WMS».
- 3.1.1.1.9 Передача из КИС и получение в КИС данных из WMS производится по классам объектов представленных в таблице 1 (общий перечень) и определенных в разделе «3.5 ОПРЕДЕЛЕНИЕ СООТВЕТСТВИЙ И ТОЧЕК ВВОДА ОБЪЕКТОВ КИС – WMS» (обязательный перечень).

Таблица 1. Перечень и описание классов объектов IsFUSION WMS API

Название класса	Описание класса	Модуль WMS
-----------------	-----------------	------------

Название класса	Описание класса	Модуль WMS
legalEntity	Элемент справочника WMS «Контрагенты»	Основной
uom	Элемент справочника WMS «Единицы измерения»	Основной
category	Элемент справочника WMS «Категории»	Основной
billingCategory	Элемент справочника WMS «Категории ЗПЛ»	Биллинг
item	Элемент справочника WMS «Товары»	Основной
userPack	Объект WMS «Упаковка»	Основной
lot	Элемент справочника WMS «Партии»	Основной
lotPackBarCode, lotPack	Объект WMS lotPack связанный с элементом справочника WMS «Штрихкоды»	Основной
userReceipt	Документ WMS «Поступление»	Основной
order	Документ WMS «Заказ (отгрузка)»	Основной
shipment	Документ WMS «Отгрузка»	Основной
transfer	Документ WMS «Перемещение между складами»	Основной
inventoryItem	Документ WMS «Инвентаризация (по товару)»	Основной
itemImage	Изображение товара	Основной
address	Элемент справочника WMS «Адреса доставки»	Транспорт
driver	Элемент справочника WMS «Водители»	Транспорт
truck	Элемент справочника WMS «Автомобили»	Транспорт
route	Элемент справочника WMS «Маршруты»	Транспорт
trip	Документ WMS «Рейс»	Транспорт

3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID

- 3.2.1.1.1 Идентификатор объекта `externalId` – это уникальный в рамках системы WMS идентификатор объекта.
- 3.2.1.1.2 Для объектов, созданных в КИС, `externalId` должен поступать в WMS вместе с объектом из этой системы. Соответственно на стороне КИС требуется обеспечить уникальность идентификатора объекта в рамках всей системы. Самым близким аналогом `externalId` является GUID объекта из КИС или любой другой первичный или суррогатный ключ, который обеспечивает такую уникальность. GUID объекта из КИС может выступать в роли `externalId` для идентификации объектов в WMS.
- 3.2.1.1.3 Для объектов, созданных внутри WMS, `externalId` формируется системой WMS в момент создания объекта. Соответственно в КИС для объектов, точкой ввода которых будет определена WMS, должна быть обеспечена запись значения `externalId` объекта из WMS, и использование именно его в запросах API.

3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ

- 3.3.1.1.1 WMS в общем случае является ведомой системой и получает исходные данные из КИС. При этом получение данных служит для создания или изменения объекта в WMS, но не для его удаления.
- 3.3.1.1.2 Удаление переданного из КИС объекта в WMS не может быть произведено по API. В случаях, когда объект требуется удалить в WMS, это действие производится Оператором WMS вручную.
- 3.3.1.1.3 В целях деактивации удаленного в КИС объекта и для запрета его дальнейшего использования в WMS для некоторых объектов предусмотрены свойства `archived` (неактивный) или `isLocked` (заблокирован).
- 3.3.1.1.4 Переданное для объекта значение любого свойства (кроме `externalId`) можно изменить или «удалить», передав новое значение или пустое значение, как обозначено в разделе «2.2.1 POST – общие правила формирования JSON».

- 3.3.1.1.5 Если какое-либо свойство объекта не передано, то его значение не будет записано или изменено или удалено в WMS.
- 3.3.1.1.6 Обозначенные выше в данном разделе правила относятся в том числе и к строкам документов, которые участвуют в обмене. При настройке интеграции КИС – WMS следует принимать во внимание следующее:
- Строки документов – это один из объектов WMS;
 - Если документ был передан в WMS, а потом в нем в КИС были удалены строки, то удаление строки из табличной части (ТЧ) документа в КИС и передача данных по документу без этой строки не означает, что эта строка будет удалена в документе WMS. В этом случае данные по этой строке просто не будут изменены в WMS;
 - При удалении в КИС строки из ТЧ документа требуется передать в WMS данные по этой строке с нулевым плановым количеством.

3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО

- 3.4.1.1.1 В WMS отсутствует такое действие для документа, как «проведение документа», и для целей определения состояния документа служат статусы документа.
- 3.4.1.1.2 Статусы документа в WMS – это автоматически изменяемая последовательность определенных статусов, обозначающих состояние документа в каждый момент времени. Финальные статусы документа устанавливаются вручную Оператором WMS. В КИС сначала должен быть получен финальный статус документа, который означает, что работа с документом завершена. После получения финального статуса документа в КИС должны быть получены данные по фактическому количеству по строкам документа. Статусы документов WMS представлены в разделах с описанием логики обмена по каждому документу.
- 3.4.1.1.3 Фактическое количество (например, «принято», «подобрано», «отгружено») – это фактически обработанное количество товаров в складской операции (например, «Приемка», «Подбор», «Отгрузка») по каждой строке соответствующего иницирующего документа WMS. Заполняется значением по факту производства складской операции на ТСД. Именно это количество в WMS является действующим остатком товара (в т.ч. определенной партии) и именно с ним производятся складские операции.
- 3.4.1.1.4 Обозначенные выше в данном разделе определения означают, что для документа, участвующего в обмене данными, в КИС должны быть созданы дополнительные атрибуты:
- Атрибут документа для записи значения текущего статуса документа из WMS, например «Статус WMS». На основании значения этого атрибута следует разрешать или запрещать проведение документа в КИС, а также возможно автоматизировать такое проведение в случае получения из WMS финального статуса без расхождений.
 - Атрибут строки табличной части документа для записи значений по фактическому количеству, получаемых из WMS. Эти значения позволяют производить сверку планового и фактического количества, а также автоматизировать создание документов на основании, например, для недостачи при приемке создавать документ «Возврат товаров поставщику» на основании «Поступления товаров и услуг».

3.5 ОПРЕДЕЛЕНИЕ СООТВЕТСТВИЙ И ТОЧЕК ВВОДА ОБЪЕКТОВ КИС – WMS

Таблица 2. Определение соответствий и точек ввода объектов КИС – WMS

Название класса WMS	Описание класса WMS	Объект КИС	Описание объекта КИС	Точка ввода	
				WMS	КИС
legalEntity	Элемент справочника WMS «Контрагенты»				Да
uom	Элемент справочника WMS «Единицы измерения»				Да
category	Элемент справочника WMS «Категории»				Да
item	Элемент справочника WMS «Товары»				Да
userPack	Объект WMS «Упаковка»				Да
lot	Элемент справочника WMS «Партии»				Да
lotPackBarCode, lotPack	Внутренний объект системы lotPack + Элемент справочника WMS «Штрихкоды»				Да
userReceipt	Документ WMS «Поступление»				Да
					Да
					Да
					Да
order	Документ WMS «Заказ (отгрузка)»				Да
shipment	Документ WMS «Отгрузка»				Да
					Да
					Да
transfer	Документ WMS «Перемещение между складами»				Да
inventoryItem	Документ WMS «Инвентаризация (по товару)»				Да
itemImage	Изображение товара				Да

3.6 СПРАВОЧНИК WMS «КОНТРАГЕНТЫ». ОПИСАНИЕ СВОЙСТВ КЛАССА LEGALENTITY

3.6.1 Логика обмена «Контрагенты»

- 3.6.1.1.1 Контрагенты – это внешние поставщики и получатели (юридические и физические лица). К Контрагентам также относятся внутренние подразделения, МОЛы и склады Заказчика, по которым не нужно вести учет остатков товаров по адресам хранения на складах WMS, т.к. в этом случае поступление от них и отгрузка им товаров в WMS будет производится документами WMS «Поступление» и «Отгрузка», как и для внешних контрагентов. Контрагенты передаются в WMS из КИС. WMS не является точкой ввода новых Контрагентов – все Контрагенты требуемые в системе, должны быть переданы из КИС.
- 3.6.1.1.2 Элементы справочника КИС «Контрагенты» должны передаваться из КИС при создании/изменении элемента справочника «Контрагенты», как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта».
- 3.6.1.1.3 Элементы справочника КИС «Контрагенты» должны передаваться из КИС при передаче объектов обмена, содержащих ссылки на эти элементы, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.6.1.1.4 При передаче из КИС в WMS элемента справочника КИС «Контрагенты» тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `legalEntity`. Описание всех свойств класса WMS `legalEntity` в связке с атрибутами элемента справочника КИС «Контрагенты» приведено в табл. 3.

3.6.2 POST – метод для передачи данных по Контрагентам из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.6.3 POST – тело JSON для передачи данных по Контрагенту из КИС в WMS

```
{
  "legalEntity": [
    {
      "externalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
      "id": "3212",
      "name": "Компания Zima\\Leto",
      "fullName": "000 \\"Компания Zima\\Leto\\\"",
      "address": "Москва",
      "archived": false,
      "phone": "123-45-45, 123-45-46",
      "email": "example1@email.com",
      "INN": "1564213212",
      "juridicalAddress": "МОСКВА"
    }
  ]
}
```

3.6.4 GET – метод для получения в КИС данных по конкретному Контрагенту из WMS

```
http://localhost:8080/exec?action=getObject&p=fc0e33a2-7db3-4b74-b135-9fb1c23ad56e
```

Где fc0e33a2-7db3-4b74-b135-9fb1c23ad56e – это externalId конкретного Контрагента.

3.6.5 GET – метод для получения в КИС данных по всем Контрагентам из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsLegalEntity
```

Где LegalEntity – это название класса WMS с заглавной буквы.

3.6.6 GET – тело ответа JSON с данными по Контрагенту из WMS

```
{
  "externalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
  "barcodes": [],
  "id": "3212",
  "name": "Компания Zima\\Leto",
  "fullName": "000 \"Компания Zima\\Leto\"",
  "address": "Москва",
  "phone": "123-45-45, 123-45-46",
  "email": "example1@email.com",
  "INN": "1564213212",
  "juridicalAddress": "МОСКВА"
}
```

3.6.7 JSON – описание свойств класса legalEntity

Таблица 3. Описание свойств класса legalEntity

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	id	Короткий код контрагента в КИС	STRING[50]			
Да	name	Краткое название контрагента	STRING[92]			
	fullName	Полное название контрагента	STRING[200]			
	address	Адрес	STRING[150]			
	phone	Телефон	STRING[30]			
	email	Электронная почта	STRING[30]			
	INN	ИНН	STRING[12]			
	juridicalAddress	Юридический адрес	STRING[150]			
	archived	Неактивный	BOOLEAN			

3.7 СПРАВОЧНИК WMS «ЕДИНИЦЫ ИЗМЕРЕНИЯ». ОПИСАНИЕ СВОЙСТВ КЛАССА UOM

3.7.1 Логика обмена «Единицы измерения»

- 3.7.1.1.1 Единица измерения (ЕИ) – это базовая (или основная) единица измерения товара, например «штука». ЕИ передаются в WMS из КИС. WMS не является точкой ввода новых ЕИ – все ЕИ требуемые в системе, должны быть переданы из КИС.
- 3.7.1.1.2 Весь обмен данными по количеству товаров в документах, например, плановое и фактически принятое количество товара в документе «Поступление», производится в ЕИ.
- 3.7.1.1.3 Элементы справочника КИС «Единицы измерения» должны передаваться из КИС при создании/изменении этого элемента, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта»
- 3.7.1.1.4 Элементы справочника КИС «Единицы измерения» должны передаваться из КИС при передаче объектов обмена, содержащих ссылки на эти элементы, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.7.1.1.5 При передаче из КИС в WMS элемента справочника КИС «Единицы измерения» тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS uom. Описание всех свойств класса WMS uom в связке с атрибутами элемента справочника КИС «Единицы измерения» приведено в табл. 4.

3.7.2 POST – метод для передачи данных по ЕИ из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.7.3 POST – тело JSON для передачи данных по ЕИ из КИС в WMS

```
{
  "uom": [
    {
      "externalId": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
      "id": "PCE",
      "name": "Штука",
      "shortName": "шт.",
      "okei": "796",
      "denySplit": true
    }
  ]
}
```

3.7.4 GET – метод для получения в КИС данных по конкретной ЕИ из WMS

```
http://localhost:8080/exec?action=getObject&p=ec6fa0e6-d196-4f7c-adfa-913c54bb47a8
```

Где ec6fa0e6-d196-4f7c-adfa-913c54bb47a8 – это externalId конкретной ЕИ.

3.7.5 GET – метод для получения в КИС данных по всем ЕИ из WMS

```
http:// localhost:8080/exec?action=IncomeApi.getObjectsUom
```

Где Uom – это название класса WMS с заглавной буквы.

3.7.6 GET – тело ответа JSON с данными по ЕИ из WMS

```
{
  "externalId": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
  "barcodes": [],
  "id": "PCE",
  "name": "Штука",
  "shortName": "шт.",
  "okei": "796",
  "denySplit": true
}
```

3.7.7 JSON – описание свойств класса uom

Таблица 4. Описание свойств класса uom

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	id	Код ЕИ в КИС или, например, международное кодовое обозначение ЕИ	STRING[3]			
Да	name	Полное наименование ЕИ	STRING[50]			
Да	shortName	Краткое наименование ЕИ	STRING[10]			
	okei	Код ОКЕИ	STRING[4]			
	denySplit	Запрет дробного. Если установлен, то при сканировании на ТСД ШК товара с этой ЕИ отключается физическая кнопка десятичной точки	BOOLEAN			

3.8 СПРАВОЧНИК WMS «КАТЕГОРИИ». ОПИСАНИЕ СВОЙСТВ КЛАССА CATEGORY

3.8.1 Логика обмена «Категории»

- 3.8.1.1.1 Категории в WMS – это группы товаров, обычно в КИС представляют собой иерархию групп элементов справочника «Номенклатура». Категории передаются в WMS из КИС. WMS не является точкой ввода новых Категорий – все Категории требуемые в системе, должны быть переданы из КИС.
- 3.8.1.1.2 Иерархия групп элементов справочника КИС «Номенклатура» должны передаваться из КИС при создании/изменении группы в справочнике «Номенклатура», как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта»
- 3.8.1.1.3 Иерархия групп элементов справочника КИС «Номенклатура» должна передаваться из КИС при передаче объектов обмена, содержащих ссылки на эти группы, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.8.1.1.4 При передаче из КИС в WMS иерархии групп элементов справочника КИС «Номенклатура» тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `category`. Описание всех свойств класса WMS `category` в связке с атрибутами групп справочника КИС «Номенклатура» приведено в табл. 5.

3.8.2 POST – метод для передачи данных по Категории из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.8.3 POST – тело JSON для передачи данных по Категориям из КИС в WMS

```
{
  "category": [
    {
      "externalId": "11740b13-48ba-4781-85e2-6fd17f1ffbe3",
      "name": "Категория 1"
    },
    {
      "externalId": "d187220c-9578-434f-ad8a-dc3a650577e2",
      "name": "Категория 2.1",
      "parent": "11740b13-48ba-4781-85e2-6fd17f1ffbe3",
      "notGetQuantity": true,
      "isSeriesNumber": true,
      "isExpirable": true,
      "fromProductionDate": true,
      "yearShelfLife": 1,
      "monthShelfLife": 2,
      "dayShelfLife": 3,
    }
  ]
}
```

```
    "askDimensionsOnScan": true
  },
  {
    "externalId": "de40577d-7923-4550-9e01-8ea6539b0c62",
    "name": "Категория 2.2",
    "parent": "11740b13-48ba-4781-85e2-6fd17f1ffbe3"
  }
]
```

3.8.4 GET – метод для получения в КИС данных по конкретной Категории из WMS

```
http://localhost:8080/exec?action=getObject&p=d187220c-9578-434f-ad8a-dc3a650577e2
```

Где d187220c-9578-434f-ad8a-dc3a650577e2 – это externalId конкретной Категории.

3.8.5 GET – метод для получения в КИС данных по всем Категориям из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsCategory
```

Где Category – это название класса WMS с заглавной буквы.

3.8.6 GET – тело ответа JSON с данными по Категории из WMS

```
{
  "externalId": "d187220c-9578-434f-ad8a-dc3a650577e2",
  "barcodes": [],
  "name": "Категория 2.1",
  "parentExternalId": "11740b13-48ba-4781-85e2-6fd17f1ffbe3",
  "notGetQuantity": true,
  "seriesNumberReceipt": true,
  "askDimensionsOnScan": true,
  "isExpirable": true,
  "fromProductionDate": true,
  "yearShelfLife": 1,
  "monthShelfLife": 2,
  "dayShelfLife": 3
}
```

3.8.7 JSON – описание свойств класса `category`

Таблица 5. Описание свойств класса `category`

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	<code>externalId</code>	Уникальный код в рамках WMS (GUID)	STRING[50]			
Да	<code>name</code>	Название категории	STRING[50]			
Да	<code>parent</code>	POST <code>externalId</code> группы-родителя. Не передается для группы 1 уровня в иерархии	STRING[50]			
	<code>parentExternalId</code>	GET <code>externalId</code> группы-родителя	STRING[50]			
	<code>notGetQuantity</code>	Не запрашивать количество – признак товара, если установлен, то в операции на ТСД требуется ввод ШК каждой упаковки вместо ввода количества. При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	BOOLEAN			
	<code>seriesNumberReceipt</code>	Учитывать по серийным номерам – признак товара, если установлен, то при приемке на ТСД выводится запрос на ввод серийных номеров. При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	BOOLEAN			
	<code>askDimensionsOnScan</code>	Ввод ВГХ при сканировании – признак товара, если установлен, то при сканировании ШК товара на ТСД выводится запрос на ввод ВГХ товара. При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	BOOLEAN			
	<code>isExpirable</code>	Со сроком годности – признак товара, по которому требуется учитывать сроки годности. При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	BOOLEAN			
	<code>fromProductionDate</code>	Указывать дату производства – признак товара, если установлен, то срок годности партии товара определяется по дате производства и количеству дней, месяцев, лет годности. При установке на категорию, все	BOOLEAN			

		подчиненные элементы в этой категории наследуют этот признак.				
	yearShelfLife	Срок годности, лет При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	INTEGER			
	monthShelfLife	Срок годности, месяцев При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	INTEGER			
	dayShelfLife	Срок годности, дней При установке на категорию, все подчиненные элементы в этой категории наследуют этот признак.	INTEGER			

3.9 СПРАВОЧНИК WMS «КАТЕГОРИИ 3PL». ОПИСАНИЕ СВОЙСТВ КЛАССА BILLINGCATEGORY

3.9.1 Логика обмена «Категории 3PL»

- 3.9.1.1.1 «Категории 3PL» в WMS – это справочник из модуля «Биллинг» и второй после Категорий классификатор товаров, предназначен для возможности определения для товаров альтернативной группировки в целях тарификации услуг ответ. хранения по каким-либо объединяющим признакам. Например, в бытовой технике может быть несколько разных групп (категорий) для однотипного товара, которые объединяют товары по виду, типу или производителю. Но трудозатраты на грузопереработку этих товаров будут зависеть только от их габаритов, поэтому для тарификации услуг для этих товаров назначают «Категории 3PL»: Крупногабарит, Среднегабарит, Мелкогабарит.
- 3.9.1.1.2 Категории 3PL передаются в WMS из КИС или создаются в WMS.
- 3.9.1.1.3 Если «Категории 3PL» передаются из КИС, то они должны передаваться при создании/изменении этого атрибута товара или элемента отдельного справочника КИС, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта»
- 3.9.1.1.4 Если «Категории 3PL» передаются из КИС, то объекты обмена, содержащие ссылки на эти категории, должны передаваться в WMS, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.9.1.1.5 При передаче из КИС в WMS категорий 3PL тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `billingCategory`. Описание всех свойств класса WMS `billingCategory` приведено в табл. 6.

3.9.2 POST – метод для передачи данных по Категории 3PL из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.9.3 POST – тело JSON для передачи данных по Категориям 3PL из КИС в WMS

```
{
  "billingCategory": [
    {
      "externalId": "50effedf-7e13-40cd-a526-8e885126313a",
      "name": "Категории 3PL"
    },
    {
      "externalId": "92842ab0-2f6c-4a72-bfd6-fd78483a4261",
      "name": "МБТ",
      "id": "mbt",
      "parent": "50effedf-7e13-40cd-a526-8e885126313a"
    }
  ]
}
```

```
}

```

3.9.4 GET – метод для получения в КИС данных по конкретной Категории 3PL из WMS

```
http://localhost:8080/exec?action=getObject&p=02c15091-54ef-4026-9e10-aed1d7400189
```

Где 02c15091-54ef-4026-9e10-aed1d7400189 – это externalId конкретной Категории 3PL.

3.9.5 GET – метод для получения в КИС данных по всем Категориям 3PL из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsBillingCategory
```

Где BillingCategory – это название класса WMS с заглавной буквы.

3.9.6 GET – тело ответа JSON с данными по Категории 3PL из WMS

```
{
  "externalId": "02c15091-54ef-4026-9e10-aed1d7400189",
  "barcodes": [],
  "name": "КБТ",
  "id": "kbt",
  "parentExternalId": "50effedf-7e13-40cd-a526-8e885126313a"
}
```

3.9.7 JSON – описание свойств класса billingCategory

Таблица 6. Описание свойств класса billingCategory

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
Да	name	Название категории 3PL	STRING[50]			
	id	Код категории 3PL	STRING[50]			
Да	parent	POST externalId группы-родителя. Не передается для группы 1 уровня в иерархии	STRING[50]			
	parentExternalId	GET externalId группы-родителя	STRING[50]			

3.10 СПРАВОЧНИК WMS «ТОВАРЫ». ОПИСАНИЕ СВОЙСТВ КЛАССА ITEM

3.10.1 Логика обмена «Товары»

- 3.10.1.1.1 Товары в WMS – это ТМЦ, с которыми производятся складские операции, обычно в КИС представляют собой элементы справочника «Номенклатура». Товары передаются в WMS из КИС. WMS не является точкой ввода новых Товаров – все товары требуемые в системе, должны быть переданы из КИС.
- 3.10.1.1.2 Элементы справочника КИС «Номенклатура» должны передаваться из КИС при создании/изменении элемента справочника «Номенклатура», как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта»
- 3.10.1.1.3 Элементы справочника КИС «Номенклатура» должны передаваться из КИС при передаче объектов обмена, содержащих ссылки на эти элементы, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.10.1.1.4 При передаче из КИС в WMS элементов справочника КИС «Номенклатура» тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `item`. Описание всех свойств класса WMS `item` в связке с атрибутами элемента справочника КИС «Номенклатура» приведено в табл. 7.

3.10.2 POST – метод для передачи данных по Товару из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.10.3 POST – тело JSON для передачи данных по Товару из КИС в WMS

```
{
  "item": [
    {
      "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
      "id": "K12345",
      "name": "Товар_123",
      "fullName": "Товар_123 полное наименование",
      "article": "A12345",
      "nameCountry": "Россия",
      "uom": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
      "category": "d187220c-9578-434f-ad8a-dc3a650577e2",
      "billingCategory": "92842ab0-2f6c-4a72-bfd6-fd78483a4261",
      "description": "Текстовое описание Товара 123",
      "length": 200,
      "width": 300,
      "height": 400,
      "netWeight": 2,
```

```

    "grossWeight": 3,
    "fillFactor": 0.9,
    "askDimensionsOnScan": false,
    "isExpirable": false,
    "fromProductionDate": false,
    "yearShelfLife": 1,
    "monthShelfLife": 2,
    "dayShelfLife": 3,
    "notGetQuantity": false,
    "seriesNumberReceipt": false
  }
]
}

```

3.10.4 GET – метод для получения в КИС данных по конкретному Товару из WMS

```
http://localhost:8080/exec?action=getObject&p=85840e5c-aefb-4cc1-b33f-c9343cf4370e
```

Где 85840e5c-aefb-4cc1-b33f-c9343cf4370e – это externalId конкретного Товара.

3.10.5 GET – метод для получения в КИС данных по всем Товарам из WMS

```
http://localhost:8080/exec?action=IncomeApi.getItemsItem
```

Где Item – это название класса WMS с заглавной буквы.

3.10.6 GET – тело ответа JSON с данными по Товару из WMS

```

{
  "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
  "barcodes": [
    "9999999999999999"
  ],
  "id": "K12345",
  "name": "Товар_123",
  "fullName": "Товар_123 полное наименование",
  "article": "A12345",
  "nameCountry": "Россия",
  "uomExternalId": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
  "categoryExternalId": "d187220c-9578-434f-ad8a-dc3a650577e2",

```

```

"notGetQuantity": true,
"description": "Текстовое описание Товара 123",
"length": 200,
"width": 300,
"height": 400,
"netWeight": 2,
"grossWeight": 3,
"fillFactor": 0.9,
"seriesNumberReceipt": true,
"askDimensionsOnScan": true,
"isExpirable": true,
"fromProductionDate": true,
"yearShelfLife": 1,
"monthShelfLife": 2,
"dayShelfLife": 3,
"checkReceipt": true,
"billingCategoryExternalId": "92842ab0-2f6c-4a72-bfd6-fd78483a4261"
}

```

3.10.7 JSON – описание свойств класса item

Таблица 7. Описание свойств класса item

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	barcodes	GET Массив штрихкодов товара (из всех партий для ЕИ товара) в WMS				
Да	id	Код товара в справочнике КИС	STRING[50]			
Да	name	Наименование товара	STRING			
Да	fullName	Полное наименование товара	STRING			
	article	Артикул	STRING[50]			
	nameCountry	Страна происхождения	STRING			
Да	uom	POST externalId единицы измерения	STRING[50]			
	uomExternalId	GET externalId единицы измерения	STRING[50]			
Да	category	POST externalId категории	STRING[50]			
	categoryExternalId	GET externalId категории	STRING[50]			
	billingCategory	POST externalId категории 3PL	STRING[50]			
	billingCategoryExternalId	GET externalId категории 3PL	STRING[50]			
	description	Описание Товара	TEXT			
	length	Длина, мм	INTEGER			
	width	Ширина, мм	INTEGER			
	height	Высота, мм	INTEGER			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
	netWeight	Масса нетто, кг	NUMERIC[11,3]			
	grossWeight	Масса брутто, кг	NUMERIC[11,3]			
	fillFactor	Коэффициент заполнения ячейки	NUMERIC[4,1]			
	notGetQuantity	Не запрашивать количество – признак товара, если установлен, то в операции на ТСД требуется ввод ШК каждой упаковки вместо ввода количества	BOOLEAN			
	seriesNumberReceipt	Учитывать по серийным номерам – признак товара, если установлен, то при приемке на ТСД выводится запрос на ввод серийных номеров	BOOLEAN			
	askDimensionsOnScan	Ввод ВГХ при сканировании – признак товара, если установлен, то при сканировании ШК товара на ТСД выводится запрос на ввод ВГХ товара	BOOLEAN			
	isExpirable	Со сроком годности – признак товара, по которому требуется учитывать сроки годности	BOOLEAN			
	fromProductionDate	Указывать дату производства – признак товара, если установлен, то срок годности партии товара определяется по дате производства и количеству дней, месяцев, лет годности	BOOLEAN			
	yearShelfLife	Срок годности, лет	INTEGER			
	monthShelfLife	Срок годности, месяцев	INTEGER			
	dayShelfLife	Срок годности, дней	INTEGER			
	checkReceipt	Проверка товара при приемке – признак товара, если установлен, то при приемке товара на ТСД выводится сообщение о необходимости вскрыть упаковку и проверить товар	BOOLEAN			

3.11 ОБЪЕКТ WMS «УПАКОВКА ТОВАРА». ОПИСАНИЕ СВОЙСТВ КЛАССА USERPACK

3.11.1 Логика обмена «Упаковка товара»

3.11.1.1.1 Упаковки товаров в WMS – это дополнительные к ЕИ упаковки или несколько упаковок в иерархии, в которых обозначено количество вложенных ЕИ или упаковок предыдущего уровня. В WMS поддерживается многоуровневая иерархическая структура вложенности упаковок товара неограниченной глубины (см. рис 1 в разделе 3.12). Например:

```
Товар (ЕИ: шт., м, кг и т.д.)
├─ Упаковка 2 уровня (упак., коробка, пакет, блистер)
│   └─ Упаковка 3 уровня (короб)
│       └─ Упаковка N уровня (паллет, фрейм)
```

3.11.1.1.2 Упаковки товаров передаются в WMS из КИС. WMS не является точкой ввода новых упаковок – все упаковки требуемые в системе, должны быть переданы из КИС при создании/изменении аналогичного объекта в КИС, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта».

3.11.1.1.3 Упаковки товара должны передаваться из КИС при передаче объектов обмена, содержащих ссылки на них, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».

3.11.1.1.4 При передаче из КИС в WMS упаковок товара тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `userPack`. Описание всех свойств класса WMS `userPack` в связке с атрибутами объекта из КИС приведено в табл. 8.

3.11.2 POST – метод для передачи данных по Упаковке товара из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.11.3 POST – тело JSON для передачи данных по Упаковкам товара из КИС в WMS

```
{
  "userPack": [{
    "externalId": "04f5f3bd-3565-4055-a5cb-b2da5223cf9a",
    "parent": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
    "name": "упак.",
    "capacity": 10,
    "length": 500,
    "width": 600,
    "height": 600,
    "netWeight": 50,
```



```

    "grossWeight": 52,
    "fillFactor": 0.8
  }, {
    "externalId": "38b99bdc-65ec-4886-b7bc-a05ee05e6b15",
    "parent": "04f5f3bd-3565-4055-a5cb-b2da5223cf9a",
    "name": "коробка",
    "capacity": 50,
    "length": 1000,
    "width": 800,
    "height": 1500,
    "netWeight": 100,
    "grossWeight": 102,
    "fillFactor": 0.9
  }
]
}

```

3.11.4 GET – метод для получения в КИС данных по конкретной Упаковке товара из WMS

```
http://localhost:8080/exec?action=getObject&p=04f5f3bd-3565-4055-a5cb-b2da5223cf9a
```

Где 04f5f3bd-3565-4055-a5cb-b2da5223cf9a – это externalId конкретной Упаковки.

3.11.5 GET – метод для получения в КИС данных по всем Упаковкам товара из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsUserPack
```

Где UserPack – это название класса WMS с заглавной буквы.

3.11.6 GET – тело ответа JSON с данными по Упаковке товара из WMS

```

{
  "externalId": "04f5f3bd-3565-4055-a5cb-b2da5223cf9a",
  "barcodes": [
    "9999999999999999",
    "7777777777777777",
    "000000000012345"
  ],
  "parentExternalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
  "name": "упак.",

```

```

"capacity": 10,
"length": 500,
"width": 600,
"height": 600,
"netWeight": 50,
"grossWeight": 52,
"fillFactor": 0.8,
"volume": 0.18
}

```

3.11.7 JSON – описание свойств класса userPack

Таблица 8. Описание свойств класса userPack

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
Да	parent	POST externalId родителя. Для упаковки второго уровня – это externalId объекта item Для упаковок третьего и последующих уровней – это externalId объекта userPack предыдущего уровня См. описание в разделе 3.12	STRING[50]			
	parentExternalId	GET externalId родителя				
Да	name	Наименование упаковки	STRING			
Да	capacity	Вместимость упаковки – количество упаковок предыдущего уровня	NUMERIC[15,3]			
	length	Длина, мм	INTEGER			
	width	Ширина, мм	INTEGER			
	height	Высота, мм	INTEGER			
	netWeight	Масса нетто, кг	NUMERIC[11,3]			
	grossWeight	Масса брутто, кг	NUMERIC[11,3]			
	fillFactor	Коэффициент заполнения ячейки	NUMERIC[4,1]			
	volume	GET Объем, м3	NUMERIC[11,6]			

3.12.1.1.1 Визуальное представление, что из себя представляет структура зависимостей классов uom, item, userPack, lot, lotPack, lotPackBarCode, обозначено на рисунке 1.

Рисунок 1. Структура зависимостей классов uom, item, userPack, lot, lotPack, lotPackBarCode

Где:

- 1) Единица измерения товара – `uom`
- 2) Товар – `item`. В матрице упаковок отображается его ЕИ, для визуализации матрицы упаковок относительно названий единицы измерения и упаковок.
- 3) Упаковки товара – `userPack`
 - Для всех упаковок, для которых верхним узлом в иерархии будет `item` – это упаковки 2-го уровня. Родителем упаковок 2-го уровня является `item` (на рис. 1 у упаковки с названием «упак.» родителем является товар).
 - Для всех упаковок, для которых верхним узлом в иерархии будет другая упаковка предыдущего уровня в иерархии, родителем является `userPack` предыдущего уровня (на рис. 1 у упаковки с названием «коробка» родителем является «упак.»).
- 4) Партия товара – `lot`. (см. описание ниже в разделе «3.13 СПРАВОЧНИК WMS «ПАРТИИ». ОПИСАНИЕ СВОЙСТВ КЛАССА LOT»)
- 5) Внутренний объект WMS – `lotPack`. Представляет собой объект системы из связки объектов `item+lot` или `userPack+lot` в зависимости от уровня в иерархии упаковок:
 - Для первого уровня в иерархии упаковок – `item+lot`
 - Для второго и следующих уровней – `userPack+lot`
- 6) Штрихкоды конкретного `lotPack` – `lotPackBarCode` (см. описание ниже в разделе «3.14 СПРАВОЧНИК WMS «ШТРИХКОДЫ». ОПИСАНИЕ СВОЙСТВ КЛАССОВ LOTPACK, LOTPACKBARCODE»).

3.13 СПРАВОЧНИК WMS «ПАРТИИ». ОПИСАНИЕ СВОЙСТВ КЛАССА LOT

3.13.1 Логика обмена «Партии»

- 3.13.1.1.1 В WMS реализован учет по партиям для всех товаров. При создании объектов `item` и `userPack` эти объекты (товары и их упаковки) присваиваются к единой партии по умолчанию с названием «без партии». Для товаров, по которым требуется учитывать определенные атрибуты партии (например, срок годности), партии могут быть получены из КИС или созданы в WMS автоматически или вручную.
- 3.13.1.1.2 Партии в WMS – это партии товаров, с которыми производятся складские операции, обычно в КИС представляют собой элементы справочника «Серии». Партии передаются в WMS из КИС. WMS не является точкой ввода новых партий – все партии требуемые в системе, должны быть переданы из КИС.
- 3.13.1.1.3 Элементы справочника КИС «Серии» должны передаваться из КИС при создании/изменении элемента справочника «Серии», как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта»
- 3.13.1.1.4 Элементы справочника КИС «Серии» должны передаваться из КИС при передаче объектов обмена, содержащих ссылки на эти элементы, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных».
- 3.13.1.1.5 При передаче из КИС в WMS элементов справочника КИС «Серии» тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `lot`. Описание всех свойств класса WMS `lot` в связке с атрибутами элемента справочника КИС «Серии» приведено в табл. 9.

3.13.2 POST – метод для передачи данных по Партии из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.13.3 POST – тело JSON для передачи данных по Партии из КИС в WMS

```
{
  "lot": [
    {
      "externalId": "31a7f146-6dbe-496a-a715-dea41d60d497",
      "description": "Название Партии",
      "expireDate": "01.01.2025",
      "isExpirationed": false,
      "note": "123",
      "attribute1": "123",
      "items": [
        {
          "itemExternalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e"
        }
      ]
    }
  ]
}
```

```

    }
  ]
}

```

3.13.4 GET – метод для получения в КИС данных по конкретной Партии из WMS

```
http://localhost:8080/exec?action=getObject&p=31a7f146-6dbe-496a-a715-dea41d60d497
```

Где 31a7f146-6dbe-496a-a715-dea41d60d497 – это externalId конкретной Партии.

3.13.5 GET – метод для получения в КИС данных по всем Партиям из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsLot
```

Где Lot – это название класса WMS с заглавной буквы.

3.13.6 GET – тело ответа JSON с данными по Партии из WMS

```

{
  "externalId": "31a7f146-6dbe-496a-a715-dea41d60d497",
  "barcodes": [],
  "description": "Название Партии",
  "expireDate": "01.01.2025",
  "isExpirationed": true,
  "note": "123",
  "attribute1": "123",
  "items": [
    {
      "itemExternalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e"
    }
  ]
}

```

3.13.7 JSON – описание свойств класса lot

Таблица 9. Описание свойств класса lot

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
Да	description	Наименование партии	STRING			
	expireDate	Дата «Годен до»	DATE			
	isExpirationed	Статус партии «Не годен»	BOOLEAN			
	denySelection	Запрещен подбор – признак партии, если установлен, то все остатки товаров с этой партией игнорируются при расчете задач на подбор, если эта партия явно не указана в расходном документе	BOOLEAN			
	allItems	Все товары – признак партии, если установлен, то все вновь создаваемые товары автоматически будут добавляться в эту партию	BOOLEAN			
	note	Примечание к партии	STRING[100]			
	attributel	Атрибут 1	STRING[100]			
Да	items	Массив. Товары входящие в партию				
Да	itemExternalId	Уникальный код в рамках WMS (GUID), externalId товара, входящего в партию	STRING[50]			

3.14 СПРАВОЧНИК WMS «ШТРИХКОДЫ». ОПИСАНИЕ СВОЙСТВ КЛАССОВ LOTPACK, LOTPACKBARCODE

3.14.1 Логика обмена «Штрихкоды»

- 3.14.1.1.1 В WMS реализован учет по партиям для всех товаров. При создании объектов `item` и `userPack` эти объекты (товары и их упаковки) присваиваются к единой партии по умолчанию с названием «без партии» или, если партия передается из КИС, к конкретной партии КИС. Связка объектов `item` и `userPack` с `lot` представляет собой объект системы `lotPack` (см. описание в разделе 3.12). Все штрихкоды товаров и упаковок конкретных партий в системе принадлежат объекту `lotPack`.
- 3.14.1.1.2 То есть, в WMS штрихкод товара или его упаковки обязательно принадлежат к какой-либо партии:
- к партии по умолчанию «без партии» – обязательно. Именно в эту партию должны передаваться ШК производителя товара (EAN);
 - к партии, полученной из КИС – при наличии, уникальный ШК партии товара или упаковки;
 - к партии, созданной в WMS автоматически или вручную – при наличии, уникальный ШК партии товара или упаковки.
- 3.14.1.1.3 Для корректной передачи в WMS штрихкодов товаров и упаковок в КИС перед такой передачей должен быть получен `externalId` партии по умолчанию с названием «без партии» из WMS. Для этого используется метод представленный в разделе «3.13.5 GET – метод для получения в КИС данных по всем Партиям из WMS». В полученном массиве партий будет партия с названием «без партии», `externalId` которой обязателен для передачи ШК (EAN) товаров и их упаковок из КИС в WMS.

Пример тела ответа JSON на запрос GET `getObjectsLot`

```
{
  "lot": [
    {
      "externalId": "203",
      "barcodes": [],
      "description": "без партии",
      "items": [
        {
          "itemExternalId": "1709221"
        },
        {
          "itemExternalId": "1978880f-3b8d-4fc2-ab9b-9104544396af"
        }
      ]
    }
  ]
}
```

- 3.14.1.1.4 Штрихкоды должны передаваться из КИС при создании/изменении штрихкода, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта». Описание всех свойств классов WMS `lotPackBarCode` и `lotPack` приведено в табл. 10, 11.

3.14.2 POST – метод для передачи данных по Штрихкодам из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.14.3 POST – тело JSON для передачи данных по Штрихкодам из КИС в WMS

```
{
  "lotPackBarCode": [
    {
      "barCodeId": "000000000012345",
      "lotExternalId": "203",
      "packExternalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e"
    },
    {
      "barCodeId": "000000000099999",
      "lotExternalId": "203",
      "packExternalId": "38b99bdc-65ec-4886-b7bc-a05ee05e6b15"
    }
  ]
}
```

3.14.4 GET – метод для получения в КИС данных по всем штрихкодам из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsLotPack
```

Где `LotPack` – это название класса WMS с заглавной буквы.

3.14.5 GET – метод для получения в КИС данных `lotPack` и Штрихкодам из WMS по `externalId` товара или упаковки и партии

```
http://localhost:8080/exec?action=IncomeApi.getObjectLotPack&p=<externalId_1>&p=<externalId_2>
```

Где `externalId_1` – это `externalId` упаковки (`item` или `userPack`), а `externalId_2` – это `externalId` партии (`lot`), или наоборот (порядок следования этих параметров не имеет значения).

3.14.6 GET – метод для получения в КИС данных по штрихкоду конкретного товара или конкретной упаковки товара с конкретной партией (`lotPack`) из WMS

```
http://localhost:8080/exec?action=getObject&p=39768
```

Где `39768` – это `externalId` конкретного `lotPack`.

3.14.7 GET – тело ответа JSON с данными по lotPack из WMS

```
{
  "externalId": "39768",
  "barcodes": [
    "LP39768",
    "000000000088888"
  ],
  "lotExternalId": "382",
  "packExternalId": "39762"
}
```

3.14.8 JSON – описание свойств класса lotPackBarCode, LotPack.

Таблица 10. Описание свойств класса lotPackBarCode

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	barCodeId	POST Штрихкод товара или упаковки товара	STRING			
Да	lotExternalId	POST Уникальный код в рамках WMS (GUID) externalId партии по умолчанию или конкретной партии товара, для которой передается ШК	STRING [50]			
Да	packExternalId	POST Уникальный код в рамках WMS (GUID) externalId товара или упаковки товара, для которых передается ШК	STRING [50]			

Таблица 11. Описание свойств класса LotPack

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
	externalId	GET Уникальный код в рамках WMS (GUID) объекта WMS LotPack	STRING [50]			
	barcodes	Массив штрихкодов для LotPack				
	lotExternalId	GET Уникальный код в рамках WMS (GUID) externalId партии по умолчанию или конкретной партии товара, для которой передается ШК	STRING [50]			
	packExternalId	GET Уникальный код в рамках WMS (GUID) externalId товара или упаковки товара, для которых передается ШК	STRING [50]			

3.15 ДОКУМЕНТ WMS «ПОСТУПЛЕНИЕ». ОПИСАНИЕ СВОЙСТВ КЛАССА USERRECEIPT

3.15.1 Логика обмена «Поступление»

- 3.15.1.1.1 Документ WMS «Поступление» – это приходный документ, в котором отражается перечень товаров и их плановое количество, которое нужно принять на склад, а также отражается фактически принятое количество после выполнения операции «Приемка» на ТСД. Аналогом этого документа в КИС может быть любой приходный документ, например, «Поступление товаров и услуг», «Приходный ордер».
- 3.15.1.1.2 Поступления передаются в WMS из КИС или создаются в WMS вручную Оператором WMS.
- 3.15.1.1.3 Весь обмен данными по количеству товаров в документе «Поступление», а именно плановое и фактически принятое количество товара, производится в ЕИ.
- 3.15.1.1.4 Приходные документы могут передаваться из КИС при создании/изменении документа, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта» только в случае наличия в WMS всех объектов связанных с «Поступлением». То есть в WMS должны быть переданы данные по всем связанным объектам (см. раздел «2.2.4 POST – порядок и состав запросов») до передачи в WMS данных по приходному документу.
- 3.15.1.1.5 Из практики, для избежания коллизий, Приходные документы должны передаваться из КИС вместе с классами объектов обмена, ссылки на элементы которых есть в приходном документе, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных» (также см. пример в п. 3.14.4).
- 3.15.1.1.6 В КИС должны быть созданы атрибуты приходного документа и атрибуты строки ТЧ приходного документа для возможности записи значений получаемых из WMS как описано в разделе «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО». Статусы документа WMS «Поступление» и их описание представлены в табл. 13.
- 3.15.1.1.7 В КИС должен быть создан или использован стандартный идентификатор строки ТЧ приходного документа для возможности передачи и получения данных из WMS по каждой строке документа (см. раздел «3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID»). Особенности обмена данными по каждой строке документа определены в разделе «3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ».
- 3.15.1.1.8 При передаче Приходного документа из КИС в WMS тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `userReceipt`. Описание всех свойств класса WMS `userReceipt` в связке с атрибутами приходного документа из КИС приведено в табл. 12.

3.15.2 POST – метод для передачи данных по Поступлению из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.15.3 POST – тело JSON для передачи данных по Поступлению из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»

```
{
  "userReceipt": [
    {
      "externalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
      "vendor": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
      "number": "001234678",
      "dateTime": "2024-02-01 09:17:34",
      "nameLocation": "ТЕСТ 1",
      "nameStockERP": "Основной склад",
      "typeDocument": "Приходный ордер",
      "note": "Примечание",
      "inStock": true,
      "notTakeBySerialNumber": true,
      "userReceiptLine": [
        {
          "externalId": "8d569fc1-d0eb-4e99-9968-5806811edb41",
          "userReceipt": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
          "item": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
          "initialDemand": 10,
          "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
        }
      ]
    }
  ]
}
```

3.15.4 POST – тело JSON для передачи данных по Поступлению и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»

```
{
  "legalEntity": [
    {
      "externalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
      "name": "Компания Zima\\Leto"
    }
  ],
  "uom": [
```

```
{
  "externalId": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
  "shortName": "шт."
},
],
"category": [
  {
    "externalId": "11740b13-48ba-4781-85e2-6fd17f1ffbe3",
    "name": "Категория 1"
  },
  {
    "externalId": "d187220c-9578-434f-ad8a-dc3a650577e2",
    "name": "Категория 2.1",
    "parent": "11740b13-48ba-4781-85e2-6fd17f1ffbe3"
  }
],
"item": [
  {
    "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
    "id": "K12345",
    "name": "Товар_123",
    "fullName": "Товар_123 полное наименование",
    "article": "A12345",
    "uom": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
    "category": "d187220c-9578-434f-ad8a-dc3a650577e2"
  }
],
"lot": [
  {
    "externalId": "31a7f146-6dbe-496a-a715-dea41d60d497",
    "description": "Название Партии",
    "items": [
      {
        "itemExternalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e"
      }
    ]
  }
],
"userReceipt": [
  {
    "externalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
    "vendor": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
    "number": "001234678",
    "dateTime": "2024-02-01 09:17:34",
    "nameStockERP": "Основной склад",
    "typeDocument": "Приходный ордер",
    "note": "Примечание",
    "userReceiptLine": [
```

```
{
  "externalId": "8d569fc1-d0eb-4e99-9968-5806811edb41",
  "item": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
  "initialDemand": 10,
  "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
  "numbers": [
    "9999999991",
    "9999999992",
    "9999999993"
  ]
}
```

3.15.5 GET – метод для получения в КИС статуса по конкретному Поступлению из WMS

Метод `getObjectStatus` является сокращенной вариацией метода `getObject` – сервер возвращает данные только по статусу объекта. Метод `getObjectStatus` используется для получения статуса конкретного документа в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные. То есть КИС делает запросы по статусу документа до момента, когда будет получен финальный статус, и после получения финального статуса делает запрос методом `getObject` для получения полных данных по документу (см. п. 3.15.7 GET – метод для получения в КИС данных по конкретному Поступлению из WMS).

```
http://localhost:8080/exec?action=IncomeApi.getObjectStatus&p=7f0b3913-69c9-487a-83e4-31db27afc0e1
```

Где 7f0b3913-69c9-487a-83e4-31db27afc0e1 – это `externalId` конкретного Поступления.

Метод `getObjectStatus` требует обязательной базовой авторизации.

3.15.6 GET – тело ответа JSON с данными по статусу конкретного Поступления из WMS

```
{
  "userReceipt": [
    {
      "externalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
      "nameStatus": "Загружена",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Поступление» и их описание представлены в табл. 13.

3.15.7 GET – метод для получения в КИС статусов по Поступлениям за указанный период из WMS

Метод `getUserReceiptStatusesPeriod` является вариацией метода `getObjectStatus` – сервер возвращает данные по статусам всех документов Поступление за указанный период. Метод `getUserReceiptStatusesPeriod` используется для получения статуса всех Поступлений за нужный период в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные.

```
http://localhost:8080/exec?action=IncomeApi.getUserReceiptStatusesPeriod&p=20.01.2024 7:00&p=28.02.2024 7:00
```

Где `&p=20.01.2024 7:00&p=28.02.2024 7:00` – это период, за который требуется получить данные.

Метод `getUserReceiptStatusesPeriod` требует обязательной базовой авторизации.

3.15.8 GET – тело ответа JSON с данными по статусу Поступлений за указанный период из WMS

```
{
  "userReceipt": [
    {
      "externalId": "47570569",
      "nameStatus": "Принята с расхождениями",
      "idStatus": "doneDiff"
    },
    {
      "externalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
      "nameStatus": "Загружена",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Поступление» и их описание представлены в табл. 13.

3.15.9 GET – метод для получения в КИС полных данных по конкретному Поступлению из WMS

```
http://localhost:8080/exec?action=getObject&p=7f0b3913-69c9-487a-83e4-31db27afc0e1
```

Где `7f0b3913-69c9-487a-83e4-31db27afc0e1` – это `externalId` конкретного Поступления.

3.15.10 GET – метод для получения в КИС полных данных по всем Поступлениям из WMS

```
http:// localhost:8080/exec?action=IncomeApi.getObjectsUserReceipt
```

Где UserReceipt– это название класса WMS с заглавной буквы.

3.15.11 GET – тело ответа JSON с полными данными по Поступлению из WMS

```
{
  "externalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
  "barcodes": [
    "RECEIPT_48172413"
  ],
  "dateTime": "01.02.2024 9:17",
  "number": "001234678",
  "nameLocation": "ТЕСТ 1",
  "note": "Примечание",
  "vendorExternalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
  "nameStatus": "В работе",
  "idStatus": "work",
  "nameStockERP": "Основной склад",
  "typeDocument": "Приходный ордер",
  "inStock": true,
  "receiptLine": [
    {
      "externalId": "8d569fc1-d0eb-4e99-9968-5806811edb41",
      "item": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
      "initialDemand": 10,
      "done": 3,
      "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
    }
  ]
}
```

3.15.12 JSON – описание свойств класса userReceipt

Таблица 12. Описание свойств класса userReceipt

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	barcodes	GET Массив, ШК Поступления				

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	vendor	POST externalId контрагента-поставщика	STRING[50]			
	vendorExternalId	GET externalId контрагента-поставщика	STRING[50]			
Да	number	Номер документа	STRING[31]			
Да	dateTime	Дата/время документа	DATETIME			
Да	nameStockERP	Название склада в КИС	STRING			
	typeDocument	Тип документа в КИС	STRING[100]			
	nameLocation	Не используется				
	note	Примечание в документе	STRING[50]			
	inStock	Признак документа «На складе» означает, что ТС с грузом прибыло на склад и приемка может быть начата. Если установлен, то на ТСД в меню операции «Приемка» доступен выбор этого документа для начала приемки.	BOOLEAN			
	notTakeBySerialNumber	Признак документа “Не принимать по серийным номерам”, если установлен, то при приемке на ТСД игнорируется свойство товара «Приемка по серийным номерам»	BOOLEAN			
	nameStatus	GET Название статуса документа (табл. 13)	STRING[100]			
	idStatus	GET Код статуса документа (табл. 13)				
	isArchive	GET Признак документа «В архиве». Если true, то это означает, что работа с документом в WMS работа завершена и он переведен в архив.	BOOLEAN			
Да	userReceiptLine	POST Массив строк Поступления				
	receiptLine	GET Массив строк Поступления				
Да	externalId	externalId строки документа (GUID)	STRING[50]			
Да	item	externalId товара (GUID)	STRING[50]			
Да	initialDemand	Плановое количество в ЕИ	NUMERIC[16,3]			
	done	GET количество «Принято» – фактически принятое количество товара в ЕИ	NUMERIC[16,3]			
Да	lot	externalId партии. При получении в массиве userReceiptLine значения свойства lot, которого нет в WMS, будет создана новая партия. Для созданной партии свойства externalId и description	STRING[50]			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
		будут заполнены полученным значением свойства <code>lot</code> Не передается для строк с товарами, по которым не требуется учет по партиям				
	<code>numbers</code>	POST Массив серийных номеров товара из КИС. Если передан, то при приемке на ТСД запрос на ввод серийных номеров не выводится. Для получения s/n в WMS у товара должен быть включен признак «Приемка по серийным номерам»				

Таблица 13. Описание статусов документа WMS «Поступление»

Код статуса	Название статуса	Описание статуса	Тип статуса		
<code>loaded</code>	Загружена	Документ загружен в WMS	Начальный		
<code>new</code>	Новая	Не используется	Служебный		
<code>draft</code>	Черновик	Документ создан на основании документа WMS «Перемещение между складами»	Начальный		
<code>work</code>	В работе	Документ в работе, есть незавершенные задачи на приемку	Операционный		
<code>done</code>	Принята	Все задачи на приемку по документу завершены без расхождений	Операционный		
<code>doneDiff</code>	Принята с расхождениями	Все задачи на приемку по документу завершены, есть расхождения планового и фактического количества	Операционный		
<code>close</code>	Завершена	Документ не имеет расхождений и переведен Оператором WMS в статус «Завершена»	Финальный		
<code>closeDiff</code>	Завершена с расхождениями	Документ имеет расхождения и переведен Оператором WMS в статус «Завершена»	Финальный		
<code>archived</code>	В архиве	Не используется. Вместо этого статуса используется признак документа «В архиве» (свойство <code>isArchive</code> в табл. 12)	Служебный		

3.16 ДОКУМЕНТ WMS «ЗАКАЗ (ОТГРУЗКА)». ОПИСАНИЕ СВОЙСТВ КЛАССА ORDER

3.16.1 Логика обмена «Заказ (отгрузка)»

- 3.16.1.1.1 Документ WMS «Заказ (отгрузка)» – не является расходным документом в WMS, а предназначен для обозначения плана по отгрузке товара контрагенту-получателю. На основании одного или нескольких «Заказов» может быть создан расходный документ WMS «Отгрузка» (см. раздел «3.16 ДОКУМЕНТ WMS «ОТГРУЗКА»). Документ «Заказ (отгрузка)» применяется в случае, когда от одного и того же контрагента-получателя поступает множество заказов, а получение/доставка осуществляется в одной точке выдачи или по одному адресу доставки. Аналогом этого документа в КИС может быть, например, «Заказ клиента» или «Заявка на выдачу ТМЦ» или любой расходный документ.
- 3.16.1.1.2 Заказы передаются в WMS из КИС или создаются в WMS вручную Оператором WMS.
- 3.16.1.1.3 Весь обмен данными по количеству товаров в документе «Заказ (отгрузка)», а именно плановое и фактически отгруженное количество товара, производится в ЕИ.
- 3.16.1.1.4 Заказы могут передаваться из КИС при создании/изменении документа, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта» только в случае наличия в WMS всех объектов связанных с «Заказом». То есть в WMS должны быть переданы данные по всем связанным объектам (см. раздел «2.2.4 POST – порядок и состав запросов) до передачи в WMS данных по заказу на отгрузку.
- 3.16.1.1.5 Из практики, для избежания коллизий, заказы должны передаваться из КИС вместе с классами объектов обмена, ссылки на элементы которых есть в этом документе, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных» (также см. пример в п. 3.14.4 для документа WMS «Поступление»).
- 3.16.1.1.6 В КИС должны быть созданы атрибуты документа-аналога и атрибуты строки ТЧ документа-аналога для возможности записи значений получаемых из WMS как описано в разделе «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО». Статусы документа WMS «Заказ (отгрузка)» и их описание представлены в табл. 15.
- 3.16.1.1.7 В КИС должен быть создан или использован стандартный идентификатор строки ТЧ документа-аналога для возможности передачи и получения данных из WMS по каждой строке документа (см. раздел «3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID»). Особенности обмена данными по каждой строке документа определены в разделе «3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ».
- 3.16.1.1.8 При передаче заказа из КИС в WMS тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `order`. Описание всех свойств класса WMS `order` в связке с атрибутами документа из КИС приведено в табл. 14.

3.16.2 POST – метод для передачи данных по заказу из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.16.3 POST – тело JSON для передачи данных по Заказу из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»

```
{
  "order": [
    {
      "externalId": "a366506d-e556-4d42-8688-010ed003aaec",
      "legalEntity": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
      "number": "001234567",
      "dateTime": "2024.02.02 13:58:00",
      "deliveryAddress": "123456, г. Москва, ул. Свободы, д. 2",
      "scheduledDateTime": "2024.02.03 14:40:00",
      "note": "Примечание",
      "nameStockERP": "Основной",
      "nameRoute": "C0",
      "nameTruck": "567",
      "nameDriver": "Водитель1",
      "nameVault": "567",
      "typeDocument": "Отпуск ПС Доставка",
      "nameForwarder": "Экспедитор 1",
      "nameDirection": "ЗАПАДНЫЙ",
      "orderLine": [
        {
          "externalId": "35dc0e33-30d6-4ca0-b106-571b7bfc6c98",
          "item": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
          "quantity": 3,
          "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
        }
      ]
    }
  ]
}
```

3.16.4 POST – тело JSON для передачи данных по Заказу и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»

См. пример в п. 3.14.4 для документа WMS «Поступление»

3.16.5 GET – метод для получения в КИС статуса по конкретному Заказу из WMS

Метод `getObjectStatus` является сокращенной вариацией метода `getObject` – сервер возвращает данные только по статусу объекта. Метод `getObjectStatus` используется для получения статуса конкретного документа в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные. То есть КИС делает запросы по статусу документа до момента, когда будет получен финальный статус, и после получения финального статуса делает запрос методом `getObject` для получения полных данных по документу (см. п. 3.15.9 GET – метод для получения в КИС данных по конкретному Заказу из WMS).

```
http://localhost:8080/exec?action=IncomeApi.getObjectStatus&p=a366506d-e556-4d42-8688-010ed003aaec
```

Где `a366506d-e556-4d42-8688-010ed003aaec` – это `externalId` конкретного Заказа.

Метод `getObjectStatus` требует обязательной базовой авторизации.

3.16.6 GET – тело ответа JSON с данными по статусу конкретного Заказа из WMS

```
{
  "order": [
    {
      "externalId": "a366506d-e556-4d42-8688-010ed003aaec",
      "nameStatus": "Загружен",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Заказ (отгрузка)» и их описание представлены в табл. 15.

3.16.7 GET – метод для получения в КИС статусов по Заказам за указанный период из WMS

Метод `getOrderStatusesPeriod` является вариацией метода `getObjectStatus` – сервер возвращает данные по статусам всех документов «Заказ (отгрузка)» за указанный период. Метод `getOrderStatusesPeriod` используется для получения статуса всех заказов за нужный период в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные.

```
http:// localhost:8080/exec?action=IncomeApi.getOrderStatusesPeriod&p=21.01.2024 7:10&p=21.05.2024 7:11
```

Где `&p=21.01.2024 7:10&p=21.05.2024 7:11` – это период, за который требуется получить данные.

Метод `getOrderStatusesPeriod` требует обязательной базовой авторизации.

3.16.8 GET – тело ответа JSON с данными по статусу Заказов за указанный период из WMS

```
{
  "order": [
    {
      "externalId": "af460f69-aa08-4c49-b8a1-1a5b9680f1ce",
      "nameStatus": "В отгрузке",
      "idStatus": "ship"
    },
    {
      "externalId": "a366506d-e556-4d42-8688-010ed003aaec",
      "nameStatus": "Загружен",
      "idStatus": "loaded"
    },
    {
      "externalId": "59de8cb6-4191-4496-b5a5-ab4441dd9b0f",
      "nameStatus": "Загружен",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Заказ (отгрузка)» и их описание представлены в табл. 15.

3.16.9 GET – метод для получения в КИС полных данных по конкретному Заказу из WMS

```
http://localhost:8080/exec?action=getObject&p=a366506d-e556-4d42-8688-010ed003aaec
```

Где `a366506d-e556-4d42-8688-010ed003aaec` – это `externalId` конкретного Заказа.

3.16.10 GET – метод для получения в КИС полных данных по всем Заказам из WMS

```
http:// localhost:8080/exec?action=IncomeApi.getObjectsOrder
```

Где `Order` – это название класса WMS с заглавной буквы.

3.16.11 GET – тело ответа JSON с полными данными по Заказу из WMS

```
{
  "externalId": "a366506d-e556-4d42-8688-010ed003aaec",
  "barcodes": [],
  "legalEntityExternalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
  "number": "001234567",
  "dateTime": "02.02.2024 13:58",
  "deliveryAddress": "123456, г. Москва, ул. Свободы, д. 2",
  "scheduledDateTime": "03.02.2024 14:40",
  "note": "Примечание",
  "nameStatus": "Загружен",
  "idStatus": "loaded",
  "nameStockERP": "Основной",
  "nameRoute": "C0",
  "nameTruck": "567",
  "nameDriver": "Водитель1",
  "nameVault": "567",
  "typeDocument": "Отпуск ПС Доставка",
  "nameForwarder": "Экспедитор 1",
  "nameDirection": "ЗАПАДНЫЙ",
  "orderLine": [
    {
      "externalId": "35dc0e33-30d6-4ca0-b106-571b7bfc6c98",
      "item": "85840e5c-ae5b-4cc1-b33f-c9343cf4370e",
      "quantity": 3,
      "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
    }
  ]
}
```

3.16.12 JSON – описание свойств класса order

Таблица 14. Описание свойств класса order

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING [50]			
	barcodes	GET Массив, ШК Заказа				
Да	legalEntity	POST externalId контрагента-получателя	STRING [50]			
	legalEntityExternalId	GET externalId контрагента-получателя	STRING [50]			
Да	number	Номер документа	STRING [31]			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	dateTime	Дата/время документа	DATETIME			
Да	nameStockERP	Название склада в КИС	STRING			
	typeDocument	Тип документа в КИС	STRING[100]			
	deliveryAddress	Адрес доставки	STRING			
	scheduledDateTime	Планируемая дата/время доставки	DATETIME			
	note	Примечание к документу	STRING			
	nameStatus	GET Название статуса документа (табл. 15)	STRING[100]			
	idStatus	GET Код статуса документа (табл. 15)				
	isArchive	GET Признак документа «В архиве». Если true, то это означает, что работа с документом в WMS работа завершена и он переведен в архив.	BOOLEAN			
	nameRoute	Маршрут	STRING[100]			
	nameTruck	Автомобиль доставки	STRING[100]			
	nameDriver	Водитель	STRING[100]			
	nameVault	Номер сводного документа по планированию доставки из КИС	STRING[100]			
	nameForwarder	Экспедитор	STRING[100]			
	nameDirection	Направление (Регион доставки). Каждое уникальное значение формирует запись в справочнике WMS «Направление»	STRING[100]			
Да	orderLine	Массив строк Заказа (отгрузки)				
Да	externalId	externalId строки документа (GUID)	STRING[50]			
Да	item	externalId товара (GUID)	STRING[50]			
Да	quantity	Плановое количество в ЕИ	NUMERIC[16,3]			
	quantityShipped	GET количество «Отгружено» – фактически отгруженное количество товара в ЕИ	NUMERIC[16,3]			
Да	lot	externalId партии. Не передается для строк с товарами, по которым не требуется отгрузка конкретной партии	STRING[50]			

Таблица 15. Описание статусов документа WMS «Заказ (отгрузка)»

Код статуса	Название статуса	Описание статуса	Тип статуса		
loaded	Загружен	Документ загружен в WMS	Начальный		
ship	В отгрузке	На основании заказа создан документ «Отгрузка» и этот документ «Отгрузка» не завершен	Операционный		
done	Отгружено	Созданный на основании заказа	Финальный		

Код статуса	Название статуса	Описание статуса	Тип статуса		
		документ «Отгрузка» завершен			
archived	В архиве	Не используется. Вместо этого статуса используется признак документа «В архиве» (свойство <code>isArchive</code> в табл. 14)	Служебный		

3.17 ДОКУМЕНТ WMS «ОТГРУЗКА». ОПИСАНИЕ СВОЙСТВ КЛАССА SHIPMENT

3.17.1 Логика обмена «Отгрузка»

- 3.17.1.1.1 Документ WMS «Отгрузка» – это расходный документ, в котором отражается перечень товаров и их плановое количество, которое нужно подобрать и отгрузить со склада, а также отражается фактически подобранное и отгруженное количество после выполнения операций «Подбор» и «Отгрузка» на ТСД. Аналогом этого документа в КИС может быть любой расходный документ, например, «Реализация товаров и услуг», «Расходный ордер».
- 3.17.1.1.2 Отгрузки передаются в WMS из КИС или создаются в WMS вручную Оператором WMS.
- 3.17.1.1.3 Весь обмен данными по количеству товаров в документе «Отгрузка», а именно плановое и фактически подобранное и отгруженное количество товара, производится в ЕИ.
- 3.17.1.1.4 Отгрузки могут передаваться из КИС при создании/изменении документа, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта» только в случае наличия в WMS всех объектов связанных с «Отгрузкой». То есть в WMS должны быть переданы данные по всем связанным объектам (см. раздел «2.2.4 POST – порядок и состав запросов») до передачи в WMS данных по Отгрузке.
- 3.17.1.1.5 Из практики, для избежания коллизий, Отгрузки должны передаваться из КИС вместе с классами объектов обмена, ссылки на элементы которых есть в этом документе, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных» (также см. пример в п. 3.14.4 для документа WMS «Поступление»).
- 3.17.1.1.6 В КИС должны быть созданы атрибуты расходного документа и атрибуты строки ТЧ расходного документа для возможности записи значений получаемых из WMS как описано в разделе «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО». Статусы документа WMS «Отгрузка» и их описание представлены в табл. 17.
- 3.17.1.1.7 В КИС должен быть создан или использован стандартный идентификатор строки ТЧ расходного документа для возможности передачи и получения данных из WMS по каждой строке документа (см. раздел «3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID»). Особенности обмена данными по каждой строке документа определены в разделе «3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ».
- 3.17.1.1.8 При передаче Отгрузки из КИС в WMS тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `shipment`. Описание всех свойств класса WMS `shipment` в связке с атрибутами документа из КИС приведено в табл. 16.

3.17.2 POST – метод для передачи данных по Отгрузке из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.17.3 POST – тело JSON для передачи данных по Отгрузке из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»

```
{
  "shipment": [
    {
      "externalId": "cd3f8f1c-7edf-4afc-b041-e80540c5dca2",
      "legalEntity": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
      "number": "001234590",
      "dateTime": "2024.02.02 17:58:00",
      "deliveryAddress": "123456, г. Москва, ул. Свободы, д. 2",
      "scheduledDateTime": "2024.02.03 14:40:00",
      "nameStockERP": "Основной",
      "typeDocument": "Отпуск ПС Доставка",
      "inStock": true,
      "note": "Примечание",
      "nameRoute": "CO",
      "nameTruck": "567",
      "nameDriver": "Водитель1",
      "nameVault": "567",
      "nameForwarder": "Экспедитор 1",
      "nameDirection": "ЗАПАДНЫЙ",
      "shipmentLine": [
        {
          "externalId": "a29b34b9-4985-42bb-b4f2-c905a930618b",
          "item": "85840e5c-ae5b-4cc1-b33f-c9343cf4370e",
          "quantity": 5,
          "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
        }
      ]
    }
  ]
}
```

3.17.4 POST – тело JSON для передачи данных по Отгрузке и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»

См. пример в п. 3.14.4 для документа WMS «Поступление»

3.17.5 GET – метод для получения в КИС статуса по конкретной Отгрузке из WMS

Метод `getObjectStatus` является сокращенной вариацией метода `getObject` – сервер возвращает данные только по статусу объекта.

Метод `getObjectStatus` используется для получения статуса конкретного документа в случаях, когда количество документов и их состав

не позволяют достаточно быстро обработать полученные из WMS полные данные. То есть КИС делает запросы по статусу документа до момента, когда будет получен финальный статус, и после получения финального статуса делает запрос методом `getObject` для получения полных данных по документу (см. п. 3.16.9 GET – метод для получения в КИС данных по конкретной Отгрузке из WMS).

```
http://localhost:8080/exec?action=IncomeApi.getObjectStatus&p=cd3f8f1c-7edf-4afc-b041-e80540c5dca2
```

Где `cd3f8f1c-7edf-4afc-b041-e80540c5dca2` – это `externalId` конкретной Отгрузки.

Метод `getObjectStatus` требует обязательной базовой авторизации.

3.17.6 GET – тело ответа JSON с данными по статусу конкретной Отгрузки из WMS

```
{
  "shipment": [
    {
      "externalId": "cd3f8f1c-7edf-4afc-b041-e80540c5dca2",
      "nameStatus": "Загружен",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Отгрузка» и их описание представлены в табл. 17.

3.17.7 GET – метод для получения в КИС статусов по Отгрузкам за указанный период из WMS

Метод `getOrderStatusesPeriod` является вариацией метода `getObjectStatus` – сервер возвращает данные по статусам всех документов «Отгрузка» за указанный период. Метод `getOrderStatusesPeriod` используется для получения статуса всех Отгрузок за нужный период в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные.

```
http:// localhost:8080/exec?action=IncomeApi.getShipmentStatusesPeriod&p=26.01.2024 7:10&p=21.05.2024 7:11
```

Где `&p=26.01.2024 7:10&p=21.05.2024 7:11` – это период, за который требуется получить данные.

Метод `getOrderStatusesPeriod` требует обязательной базовой авторизации.

3.17.8 GET – тело ответа JSON с данными по статусу Отгрузка за указанный период из WMS

```
{
  "shipment": [
    {
      "externalId": "48173421",
      "nameStatus": "В работе",
      "idStatus": "selectionWork"
    },
    {
      "externalId": "cd3f8f1c-7edf-4afc-b041-e80540c5dca2",
      "nameStatus": "Загружен",
      "idStatus": "loaded"
    }
  ]
}
```

Статусы документа WMS «Отгрузки» и их описание представлены в табл. 17.

3.17.9 GET – метод для получения в КИС полных данных по конкретной Отгрузке из WMS

```
http://localhost:8080/exec?action=getObject&p=cd3f8f1c-7edf-4afc-b041-e80540c5dca2
```

Где cd3f8f1c-7edf-4afc-b041-e80540c5dca2 – это externalId конкретной Отгрузки.

3.17.10 GET – метод для получения в КИС полных данных по всем Отгрузкам из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsShipment
```

Где Shipment – это название класса WMS с заглавной буквы.

3.17.11 GET – тело ответа JSON с полными данными по Отгрузке из WMS

```
{
  "externalId": "cd3f8f1c-7edf-4afc-b041-e80540c5dca2",
  "barcodes": [
    "SH48173459"
  ],
}
```

```

"legalEntityExternalId": "fc0e33a2-7db3-4b74-b135-9fb1c23ad56e",
"number": "001234590",
"dateTime": "02.02.2024 17:58",
"deliveryAddress": "123456, г. Москва, ул. Свободы, д. 2",
"note": "Примечание",
"nameStatus": "Завершена",
"idStatus": "done",
"nameStockERP": "Основной",
"inStock": true,
"nameRoute": "C0",
"nameTruck": "567",
"nameDriver": "Водитель1",
"nameVault": "567",
"typeDocument": "Отпуск ПС Доставка",
"nameForwarder": "Экспедитор 1",
"nameDirection": "ЗАПАДНЫЙ",
"shipmentLine": [
  {
    "externalId": "a29b34b9-4985-42bb-b4f2-c905a930618b",
    "item": "85840e5c-ae5b-4cc1-b33f-c9343cf4370e",
    "quantity": 3,
    "quantitySelection": 3,
    "quantityShipped": 3,
    "lot": "31a7f146-6dbe-496a-a715-dea41d60d497"
  }
]
}

```

3.17.12 JSON – описание свойств класса shipment

Таблица 14. Описание свойств класса shipment

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	barcodes	GET Массив, ШК Отгрузки				
Да	legalEntity	POST externalId контрагента-получателя	STRING[50]			
	legalEntityExternalId	GET externalId контрагента-получателя	STRING[50]			
Да	number	Номер документа	STRING[31]			
Да	dateTime	Дата/время документа	DATETIME			
Да	nameStockERP	Название склада в КИС	STRING			
	typeDocument	Тип документа в КИС	STRING[100]			
	deliveryAddress	Адрес доставки	STRING			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
	scheduledDateTime	Планируемая дата/время доставки	DATETIME			
	note	Примечание к документу	STRING			
	nameStatus	GET Название статуса документа (табл. 17)	STRING[100]			
	idStatus	GET Код статуса документа (табл. 17)				
	isArchive	GET Признак документа «В архиве». Если true, то это означает, что работа с документом в WMS работа завершена и он переведен в архив.	BOOLEAN			
	nameRoute	Маршрут	STRING[100]			
	nameTruck	Автомобиль доставки	STRING[100]			
	nameDriver	Водитель	STRING[100]			
	nameVault	Номер сводного документа по планированию доставки из КИС	STRING[100]			
	nameForwarder	Экспедитор	STRING[100]			
	nameDirection	Направление (Регион доставки). Каждое уникальное значение формирует запись в справочнике WMS «Направление»	STRING[100]			
	tripExternalId	GET externalId документа WMS «Рейс», в который входит Отгрузка (модуль «Транспорт»)	STRING[50]			
Да	shipmentLine	Массив строк Отгрузки				
Да	externalId	externalId строки документа (GUID)	STRING[50]			
Да	item	externalId товара (GUID)	STRING[50]			
Да	quantity	Плановое количество в ЕИ	NUMERIC[16,3]			
	quantitySelection	GET количество «Подобрано» – фактически подобранное количество товара в ЕИ	NUMERIC[16,3]			
	quantityShipped	GET количество «Отгружено» – фактически отгруженное количество товара в ЕИ	NUMERIC[16,3]			
Да	lot	externalId партии. Не передается для строк с товарами, по которым не требуется отгрузка конкретной партии	STRING[50]			

Таблица 17. Описание статусов документа WMS «Отгрузка»

Код статуса	Название статуса	Описание статуса	Тип статуса		
loaded	Загружен	Документ загружен в WMS	Начальный		
selectionCreated	Создан подбор	На основании Отгрузки создан документ «Подбор» и этот	Операционный		

Код статуса	Название статуса	Описание статуса	Тип статуса		
		документ «Подбор» не подобран			
selectionWait	Ожидает пополнения	На основании Подбора созданы задачи на пополнение и эти задачи не выполнены	Операционный		
selectionInSelection	В подбор	На основании Подбора созданы задачи на подбор и эти задачи не выполнены	Операционный		
selectionWork	В работе	Одна из задач на подбор взята в работу	Операционный		
selectionDone	Подобран	Все задачи на подбор выполнены	Операционный		
done	Завершена	Документ «Отгрузка» завершен	Финальный		
archived	В архиве	Не используется. Вместо этого статуса используется признак документа «В архиве» (свойство isArchive в табл. 16)	Служебный		

3.18 ДОКУМЕНТ WMS «ПЕРЕМЕЩЕНИЕ МЕЖДУ СКЛАДАМИ». ОПИСАНИЕ СВОЙСТВ КЛАССА TRANSFER

3.18.1 Логика обмена «Перемещение между складами»

- 3.18.1.1.1 Документ WMS «Перемещение между складами» – это документ, в котором отражается перечень товаров и их плановое количество, которое нужно переместить между складами WMS, то есть подобрать и отгрузить с одного склада WMS и принять на другом складе WMS, а также отражается фактически подобранное, отгруженное и принятое количество после выполнения операций «Подбор», «Отгрузка» и «Приемка» на ТСД. Аналогом этого документа в КИС может быть документ, например, «Перемещение товаров».
- 3.18.1.1.2 Документ WMS «Перемещение между складами» предназначен для перемещения товаров только (!) между складами WMS. Соответственно до передачи из КИС такого перемещения требуется настройка соотношения складов в WMS. Соотношение складов КИС-WMS производится в справочнике WMS «Соотношение складов WMS-ERP» по названию склада в КИС: каждому складу из КИС, по которому определен конкретный физический склад, присваивается значение из справочника WMS «Склады». Настройка соотношения складов КИС – WMS представлена в таблице 19. В случае если соотношение для какого-то склада КИС не настроено, но этот склад передается в документе перемещения, то сервер возвращает ошибку 409 (см. п. 3.17.2).
- 3.18.1.1.3 Перемещения передаются в WMS из КИС или создаются в WMS вручную Оператором WMS.
- 3.18.1.1.4 Весь обмен данными по количеству товаров в документе «Перемещение между складами», а именно плановое и фактические количества товара по складским операциям, производится в ЕИ.
- 3.18.1.1.5 Перемещения между складами могут передаваться из КИС при создании/изменении документа, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта» только в случае наличия в WMS всех объектов связанных с этим документом. То есть в WMS должны быть переданы данные по всем связанным объектам (см. раздел «2.2.4 POST – порядок и состав запросов») до передачи в WMS данных по перемещению.
- 3.18.1.1.6 Из практики, для избежания коллизий, Перемещения должны передаваться из КИС вместе с классами объектов обмена, ссылки на элементы которых есть в этом документе, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных» (также см. пример в п. 3.14.4 для документа WMS «Поступление»).
- 3.18.1.1.7 В КИС должны быть созданы атрибуты документа перемещения и атрибуты строки ТЧ этого документа для возможности записи значений получаемых из WMS как описано в разделе «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО». Для документа WMS «Перемещение между складами» используется двух-статусная модель:
- Статус «Отгрузки» – описание статусов представлено в табл. 13
 - Статус «Поступления» – описание статусов представлено в табл. 17
- 3.18.1.1.8 В КИС должен быть создан или использован стандартный идентификатор строки ТЧ документа перемещения для возможности передачи и получения данных из WMS по каждой строке документа (см. раздел «3.2 ИДЕНТИФИКАТОР ОБЪЕКТА EXTERNALID»).

Особенности обмена данными по каждой строке документа определены в разделе «3.3 СОЗДАНИЕ, ИЗМЕНЕНИЕ И УДАЛЕНИЕ ОБЪЕКТОВ И ЗНАЧЕНИЙ ИХ СВОЙСТВ».

3.18.1.1.9 При передаче перемещения из КИС в WMS тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `transfer`. Описание всех свойств класса WMS `transfer` в связке с атрибутами документа из КИС приведено в табл. 18.

3.18.2 POST – ответ сервера 409 на запрос, в котором указан склад КИС без настроенного соотношения со складом WMS

Свойство Получатель [Transfer.receipient[Transfer.Transfer]] не задано

Перемещение между складами (ИД) Описание	Дата/Время	Наименование	Дата/время	Номер
48173688 Перемещение между складами от 2024-02-01 16:48:00 №000123	01.02.2024 16:48	000123	000123 от 2024-02-01 16:48:00	01.02.2024 16:48

3.18.3 POST – метод для передачи данных по Перемещению из КИС в WMS

`http://localhost:8080/exec?action=IncomeApi.insertUpdate`

3.18.4 POST – тело JSON для передачи данных по Перемещению из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»

```
{
  "transfer": [
    {
      "externalId": "1f1ef602-cbf0-4381-8730-0c26ec4e6c0b",
      "dateTime": "2024.02.01 16:48",
      "number": "000123",
      "note": "Примечание к документу",
      "scheduledShipmentDateTime": "2024.02.01 08:00",
      "scheduledReceiptDateTime": "2024.02.02 09:00",
      "senderStockERP": "OC",
      "receipientStockERP": "ГС",
      "transferLine": [
        {

```

```

        "externalId": "97845754-82db-4fc1-a6bd-a834fdc83b1b",
        "item": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
        "lot": "31a7f146-6dbe-496a-a715-dea41d60d497",
        "quantity": 1
      }
    ]
  }
}

```

3.18.5 POST – тело JSON для передачи данных по Перемещению и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»

См. пример в п. 3.14.4 для документа WMS «Поступление»

3.18.6 GET – метод для получения в КИС полных данных по конкретному Перемещению из WMS

```
http://localhost:8080/exec?action=getObject&p=1f1ef602-cbf0-4381-8730-0c26ec4e6c0b
```

Где 1f1ef602-cbf0-4381-8730-0c26ec4e6c0b – это externalId конкретного Перемещения.

3.18.7 GET – метод для получения в КИС полных данных по всем Перемещениям из WMS

```
http://localhost:8080/exec?action=IncomeApi.getObjectsTransfer
```

Где Transfer – это название класса WMS с заглавной буквы.

3.18.8 GET – тело ответа JSON с полными данными по Перемещению из WMS

```

{
  "externalId": "1f1ef602-cbf0-4381-8730-0c26ec4e6c0b",
  "barcodes": [
    "TR42165846"
  ],
  "dateTime": "01.02.2024 16:48",
  "number": "000123",
  "note": "Примечание к документу",
  "scheduledShipmentDateTime": "01.02.2024 08:00",

```

```

"scheduledReceiptDateTime": "02.02.2024 09:00",
"nameReceiptStatus": "Завершена",
"nameShipmentStatus": "Завершена",
"idReceiptStatus": "close",
"idShipmentStatus": "done",
"senderStockERP": "Тест",
"recepientStockERP": "ГС",
"transferLine": [
  {
    "externalId": "97845754-82db-4fc1-a6bd-a834fdc83b1b",
    "itemExternalId": "21505187",
    "lotExternalId": "7509630",
    "quantity": 1,
    "quantitySelect": 1,
    "quantityShipped": 1,
    "done": 1
  }
]
}

```

3.18.9 JSON – описание свойств класса transfer

Таблица 18. Описание свойств класса transfer

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING [50]			
	barcodes	GET Массив, ШК Перемещения				
Да	number	Номер документа	STRING [31]			
Да	dateTime	Дата и время документа	DATETIME			
	scheduledShipment DateTime	Планируемые дата и время отгрузки	DATETIME			
	scheduledReceiptD ateTime	Планируемые дата и время поступления	DATETIME			
	nameReceiptStatus	GET Название статуса документа Поступление (табл. 13)	STRING			
	nameShipmentStatu s	GET Название статуса документа Отгрузка (табл. 17)	STRING			
	idReceiptStatus	GET Код статуса документа Поступление (табл. 13)	STRING			
	idShipmentStatus	GET Код статуса документа Поступление (табл. 17)	STRING			
Да	senderStockERP	Название склада-отправителя из КИС	STRING			
Да	recepientStockERP	Название склада-получателя из КИС	STRING			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	transferLine					
Да	externalId	externalId строки документа (GUID)	STRING[50]			
Да	itemExternalId	externalId товара (GUID)	STRING[50]			
Да	lotExternalId	externalId партии. Не передается для строк с товарами, по которым не требуется перемещение конкретной партии	STRING[50]			
Да	quantity	Плановое количество в ЕИ	NUMERIC[16,3]			
	quantitySelect	GET количество «Подобрано» – фактически подобранное количество товара в ЕИ	NUMERIC[16,3]			
	quantityShipped	GET количество «Отгружено» – фактически отгруженное количество товара в ЕИ	NUMERIC[16,3]			
	done	GET количество «Принято» – фактически принятое количество товара в ЕИ	NUMERIC[16,3]			

3.18.10 Соотношение складов КИС – WMS

Таблица 19. Соотношение складов КИС – WMS

Код склада в КИС	Наименование склада в КИС	Склад в WMS	Комментарий

3.19 ДОКУМЕНТ WMS «ИНВЕНТАРИЗАЦИЯ (ПО ТОВАРУ)». ОПИСАНИЕ СВОЙСТВ КЛАССА INVENTORYITEM

3.19.1 Логика обмена «Инвентаризация (по товару)»

- 3.19.1.1.1 Документ WMS «Инвентаризация (по товару)» – это документ, в котором отражается перечень товаров их полное плановое количество, которое числится на остатках, а также отражается полное фактическое количество после выполнения операции «Инвентаризация» на ТСД. Аналогом этого документа в КИС может быть документ «Инвентаризация» или «Пересчет товаров».
- 3.19.1.1.2 Инвентаризации передаются в WMS из КИС или создаются в WMS вручную Оператором WMS.
- 3.19.1.1.3 Весь обмен данными по количеству товаров в документе «Инвентаризация», а именно плановое и фактическое количество товара, производится в ЕИ.
- 3.19.1.1.4 Инвентаризации могут передаваться из КИС при создании/изменении документа, как обозначено в разделе «2.2.2 POST – JSON для одиночного объекта» только в случае наличия в WMS всех объектов связанных с «Инвентаризацией». То есть в WMS должны быть переданы данные по всем связанным объектам (см. раздел «2.2.4 POST – порядок и состав запросов») до передачи в WMS данных по документу.
- 3.19.1.1.5 Из практики, для избежания коллизий, Инвентаризации должны передаваться из КИС вместе с классами объектов обмена, ссылки на элементы которых есть в этом документе, как обозначено в разделе «2.2.3 POST – JSON для пакетной передачи данных» (также см. пример в п. 3.14.4 для документа «Поступление»).
- 3.19.1.1.6 В КИС должны быть созданы атрибуты документа и атрибуты строки ТЧ документа для возможности записи значений получаемых из WMS как описано в разделе «3.4 СТАТУСЫ ДОКУМЕНТОВ WMS И ФАКТИЧЕСКОЕ КОЛИЧЕСТВО». Статусы документа WMS «Инвентаризация (по товару)» и их описание представлены в табл. 22.
- 3.19.1.1.7 При передаче Инвентаризации из КИС в WMS тело запроса POST в формате JSON должно содержать значения обязательных свойств класса WMS `inventoryItem`. Описание всех свойств класса WMS `inventoryItem` в связке с атрибутами инвентаризации из КИС приведено в табл. 21.

3.19.2 POST – метод для передачи данных по Инвентаризации из КИС в WMS

```
http://localhost:8080/exec?action=IncomeApi.insertUpdate
```

3.19.3 POST – тело JSON для передачи данных по Инвентаризации из КИС в WMS, как описано в разделе «2.2.2 POST – JSON для одиночного объекта»

```
{
  "inventoryItem": [
    {
      "externalId": "e3c532fd-e74c-4dcb-a8b8-508b3e2a1e2d",
      "number": "INV-000005",
      "dateTime": "2024-02-02 10:31:34",
      "nameStockERP": "ГС",
      "items": [
        {
          "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e"
        }
      ]
    }
  ]
}
```

3.19.4 POST – тело JSON для передачи данных по Инвентаризации и данных по всем связанным объектам из КИС в WMS, как описано в разделе «2.2.3 POST – JSON для пакетной передачи данных»

См. пример в п. 3.14.4 для документа WMS «Поступление»

3.19.5 GET – метод для получения в КИС полных данных по конкретной Инвентаризации из WMS по Коду документа (**vector**)

```
http://localhost:8080/exec?action=getInventoryItem&p=II41571520
```

Где II41571520 – это Код (код документа в WMS) конкретной Инвентаризации.

3.19.6 GET – метод для получения в КИС полных данных по конкретной Инвентаризации из WMS по externalId

```
http://localhost:8080/exec?action=getObject&p=e3c532fd-e74c-4dcb-a8b8-508b3e2a1e2d
```

Где e3c532fd-e74c-4dcb-a8b8-508b3e2a1e2d – это externalId конкретной Инвентаризации.

3.19.7 GET – метод для получения в КИС полных данных по Инвентаризациям из WMS за период (Вектор)

Метод `getInventoryItemStatusesPeriod` используется для получения статуса всех «Инвентаризаций (по товару)» за нужный период в случаях, когда количество документов и их состав не позволяют достаточно быстро обработать полученные из WMS полные данные.

```
http://localhost:8080/exec?action=IncomeApi.getInventoryItemStatusesPeriod&p=21.03.2023 7:10&p=21.05.2024 7:11
```

Где `&p=21.03.2023 7:10&p=21.05.2024 7:11` – это период, за который требуется получить данные.

3.19.8 GET – метод для получения в КИС полных данных по всем Инвентаризациям (по товару) из WMS

```
http:// localhost:8080/exec?action=IncomeApi.getObjectsInventoryItem
```

Где `InventoryItem` – это название класса WMS с заглавной буквы.

3.19.9 GET – тело ответа JSON с полными данными по Инвентаризации из WMS

```
{
  "externalId": "e3c532fd-e74c-4dcb-a8b8-508b3e2a1e2d",
  "barcodes": [],
  "number": "INV-000005",
  "dateTime": "02.02.2024 10:31",
  "nameStatus": "Завершена",
  "idStatus": "close",
  "idStock": "S1",
  "nameStock": "Главный",
  "nameStockERP": "ГС",
  "items": [
    {
      "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
      "externalIdInventoryItem": "e3c532fd-e74c-4dcb-a8b8-508b3e2a1e2d",
      "initialDemand": 3,
      "done": 3
    }
  ]
}
```

3.19.10 JSON – описание свойств класса `inventoryItem`

Таблица 21. Описание свойств класса `inventoryItem`

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID)	STRING[50]			
	barcodes	GET Массив, ШК Инвентаризации (по товару)				
Да	number	Номер документа	STRING[31]			
Да	dateTime	Дата/время документа	DATETIME			
Да	nameStockERP	Название склада в КИС	STRING			
	nameStatus	GET Название статуса документа (табл. 22)	STRING[100]			
	idStatus	GET Код статуса документа (табл. 22)	STRING			
	idStock	GET Код склада WMS	STRING[50]			
	nameStock	GET Название склада WMS	STRING[50]			
	isArchive	GET Признак документа «В архиве». Если true, то это означает, что работа с документом в WMS работа завершена и он переведен в архив.	BOOLEAN			
Да	items	Массив строк Инвентаризации – Товары				
Да	externalId	externalId товара (GUID)	STRING[50]			
	externalIdInventoryItem	GET externalId документа Инвентаризация	STRING[50]			
	initialDemand	Плановое количество в ЕИ	NUMERIC[16,3]			
	done	GET фактическое количество по остатку товара в ЕИ в WMS	NUMERIC[16,3]			

Таблица 22. Описание статусов документа WMS «Инвентаризация (по товару)»

Код статуса	Название статуса	Описание статуса	Тип статуса		
loaded	Загружена	Документ загружен в WMS	Начальный		
work	В работе	По документу созданы задачи на инвентаризацию или на вкладке «Детально» заполнено плановое кол-во	Операционный		
factQuantity	Введено количество факт.	По одной из строке с товаром на вкладке «Детально» внесено фактическое количество	Операционный		
canceled	Отменена	Для документа в статусе «Загружена» нажата кнопка «Отменить»	Финальный		
close	Завершена	Все действия по инвентаризации завершены	Финальный		
archived	В архиве	Не используется. Вместо этого статуса используется признак документа «В архиве» (свойство isArchive в табл. 21)	Служебный		

4 ЛОГИКА ОБМЕНА. ОПИСАНИЕ СПЕЦИАЛЬНЫХ МЕТОДОВ POST И GET

4.1 ОСТАТКИ ИЗ WMS. ОПИСАНИЕ СВОЙСТВ МЕТОДА REMAININGITEMS

4.1.1 GET – метод для получения в КИС данных по остаткам товаров из WMS

```
http:// localhost:8080/exec?action=IncomeApi.remainingItems&p=85840e5c-aefb-4cc1-b33f-c9343cf4370e
```

Где 85840e5c-aefb-4cc1-b33f-c9343cf4370e – это externalId категории товаров или externalId конкретного товара, по которым нужно получить остатки из WMS.

4.1.2 GET – тело ответа JSON с данными по остаткам из WMS

```
{
  "externalId": "85840e5c-aefb-4cc1-b33f-c9343cf4370e",
  "article": "A12345",
  "id": "K12345",
  "name": "Товар_123",
  "fullName": "Товар_123 полное наименование",
  "nameUom": "Штука",
  "externalIdUom": "ec6fa0e6-d196-4f7c-adfa-913c54bb47a8",
  "stock": [
    {
      "id": "S1",
      "name": "Главный",
      "remaining": 3
    }
  ]
}
```

4.1.3 JSON – описание свойств класса remainingItems

Таблица 23. Описание свойств класса remainingItems

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalId	Уникальный код в рамках WMS (GUID) - товар	STRING[50]			
	article	Артикул товара	STRING[50]			
	id	Код товара	STRING[50]			
	name	Наименование товара	STRING			

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
	fullName	Полное наименование товара	STRING			
	nameUom	Наименование ЕИ товара	STRING [50]			
	externalIdUom	externalId ЕИ	STRING [50]			
	stock	Массив остатков товара по складам				
	id	Код склада в WMS	STRING [30]			
	name	Название склада в WMS	STRING [100]			
	remaining	Кол-во товара всего (остаток товара в ЕИ)	NUMERIC [16,3]			
vector	quantitySelect	Кол-во товара в Упаковочных листах	NUMERIC [16,3]			
vector	availableQuantity	Кол-во товара в наличии (Кол-во всего за минусом кол-ва в УЛ)	NUMERIC [16,3]			

Обязателен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
Да	externalIdItem	externalId товара	STRING[50]			
Да	image	Эквивалентное строковое представление изображения для JSON				

4.3 СЕРИЙНЫЕ НОМЕРА ИЗ ДОКУМЕНТА WMS. ОПИСАНИЕ СВОЙСТВ КЛАССА SERIESNUMBERS

4.3.1 Логика обмена «Серийные номера из документа WMS»

4.3.1.1.1 Серийные номера из документов WMS должны быть получены в КИС после получения финального статуса документа из WMS.

4.3.2 GET – метод для получения в КИС данных по серийным номерам из документов WMS

```
http://localhost:8080/exec?action=getSeriesNumbers&p=7f0b3913-69c9-487a-83e4-31db27afc0e1
```

Где 7f0b3913-69c9-487a-83e4-31db27afc0e1 – это externalId документа, из которого нужно получить серийные номера.

4.3.3 GET – тело ответа JSON с данными по серийным номерам из документов WMS

```
{
  "documentExternalId": "7f0b3913-69c9-487a-83e4-31db27afc0e1",
  "items": [
    {
      "itemExternalId": "1709221",
      "numbers": [
        "1231231",
        "1232132",
        "1231233",
        "1231234",
        "1231235",
        "1231236"
      ]
    }
  ]
}
```

4.3.4 JSON – описание свойств класса SeriesNumbers

Таблица 25. Описание свойств класса SeriesNumbers

Обязат елен	Название свойства в JSON	Описание	Тип в WMS	Название атрибута в КИС	Описание	Тип в КИС
	documentExternalId	externalId документа	STRING[50]			
	items	Массив товаров				
	itemExternalId	externalId товара	STRING[50]			
	numbers	Массив серийных номеров				

4.4 КОДЫ МАРКИРОВКИ (ЧЕСТНЫЙ ЗНАК). ОПИСАНИЕ СВОЙСТВ КЛАССА MARKS

4.4.1 Логика обмена «Коды маркировки»

1) передача в WMS марок

POST `ехес?action=IncomeApi.marks`

Пример передаваемого JSON

```
[
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMj10dU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFCM3BD",
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSw==",
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMj10dU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFCM3BD",
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMj10dU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFCM3BD"
]
```

2) Передача в параметре EAN

GET `ехес?action=IncomeApi.mark&p=4811215024721`

Возвращает в JSON список марок в Base64, которые соответствует переданному параметру

Пример:

```
[
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMj10dU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFCM3BD",
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMj10dU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFCM3BD",
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNu2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSw=="
]
```

3) Передача в параметре марки закодированной в Base64

GET

exec?action=IncomeApi.mark&p=MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzZlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNU2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMjlOdU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFfCM3BD

Возвращает в JSON тот же код, при наличии его в справочнике марок WMS

```
[  
  "MDEwNDgxMTIxNTAyNDcyMTIxMjN2SzZlCZkhCcWhOTR05MTAwMDUdOTJrRWFEUVRNU2dmWjdUMjF2N1UxTFZEVFhMMWJhRVB5V3FBN3RqbkhOdXplSzNmd3NhdEY4WFhtMjlOdU5Lemk2ZVN3NGZRUjJUWFJlVmhlelhQeHFfCM3BD"  
]
```

