

**ФЕДЕРАЛЬНАЯ ГОСУДАРСТВЕННАЯ ИНФОРМАЦИОННАЯ СИСТЕМА
ПРОСЛЕЖИВАЕМОСТИ ПЕСТИЦИДОВ И АГРОХИМИКАТОВ**

СПЕЦИАЛЬНОЕ ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

РУКОВОДСТВО ПРОГРАММИСТА

Листов 106

АННОТАЦИЯ

Настоящий документ является руководством программиста по разработке интеграционного решения, предназначенного для взаимодействия с СПО Федеральной государственной информационной системы прослеживаемости пестицидов и агрохимикатов (ФГИС «Сатурн») через интерфейсы прикладного программного обеспечения.

СОДЕРЖАНИЕ

Перечень терминов.....	4
Перечень сокращений.....	6
1. Входные и выходные данные.....	7
1.1. Взаимодействие с хозяйствующими субъектами.....	7
1.1.1. Входные данные, вводимые хозяйствующими субъектами.....	7
1.1.2. Выходные данные, получаемые хозяйствующими субъектами.....	7
1.2. Взаимодействие с приложениями ХС.....	8
1.3. Взаимодействие с компонентом «Цербер».....	8
Приложение.....	9
1. ОПИСАНИЕ.....	9
2. СТРУКТУРА ПРОТОКОЛА ОБМЕНА.....	9
3. ОБЩИЕ КОНЦЕПЦИИ.....	9
4. ОТЧЕТЫ.....	11
5. АВТОРИЗАЦИЯ.....	11
5.1. Получение ключа доступа.....	11
5.2. Генерация SID и запуск сессии.....	11
6. ОБЩЕЕ ОПИСАНИЕ SEAPI.....	13
6.1. Точки доступа.....	13
6.2. Логическая модель.....	14
6.3. Транспортный слой.....	15
6.3.1. Контейнер ответа.....	15
6.3.2. Параметры обращения.....	15
7. Описание API.....	18
1. Сущность "Warehouse".....	18
2. Сущность "Invoice".....	24
3. Сущность "PatUsagePlanRec".....	41
4. Сущность "Inventorization".....	47
5. Сущность "WarehouseStates".....	54
6. Сущность "PAImportRecord".....	58
7. Сущность "OpTransaction".....	67
8. Сущность "PATBatch".....	70
9. Сущность "PatUsage".....	73
10. Сущность "Acreage".....	80
11. Сущность "UsrFavourite".....	86
12. Сущность "Contractor".....	91
13. Сущность "PatProduct".....	100
14. Сущность "PatProductionAct".....	106

ПЕРЕЧЕНЬ ТЕРМИНОВ

Термин	Определение
Документ	Информационная запись, отражающая акт хозяйственной деятельности, совершаемый субъектом. Информационная запись может соответствовать как одному, так и нескольким бумажным документам, создаваемым при неавтоматизированной деятельности, или даже отражать "папку", в которой собираются документы, участвующие в каком-либо процессе. Задача документа - отражать факты хозяйственной деятельности, реже - отражать состояние какого-либо участка деятельности (например, накладная может быть выписана, согласована, груз по ней может быть оплачен, зарезервирован, отгружен, доставлен).
Места хранения ПАТ	Склады, места и площадки, на которых осуществляется хранение ПАТ
Россельхознадзор	Федеральная служба по ветеринарному и фитосанитарному надзору
Справочник	Набор информационных записей. Виды справочников: - классификаторы - содержат классифицирующие записи; - реестры – представляют собой перечень различных сущностей материального мира или понятий (например, реестр хозяйствующих субъектов, реестр зарегистрированных ПА). Задача справочников - содержать "перечисления признаков", которые используются при работе с другими сущностями (справочниками и документами).
ФГИС «ВетИС»	Федеральная Государственная информационная система в области ветеринарии «ВетИС»
ФГИС «Сатурн», Система	Федеральная государственная информационная система прослеживаемости пестицидов и агрохимикатов
«Цербер»	Компонент «Цербер» ФГИС «ВетИС»
JSON	JavaScript Object Notation - текстовый формат обмена данными, основанный на JavaScript.
FK	Foreign key – внешний ключ базы данных
HRJSON	HTML Representation Javascript Object Notation – формат сериализации информационных объектов, набор спецификаций, описывающих с помощью

	формата JSON-структуру информационной сущности с учётом ее связей с другими сущностями.
HTTP	HyperText Transfer Protocol - протокол прикладного уровня передачи данных
SEAPI	Stateable Entities Application Programming Interface – программный интерфейс управления сущностями с жизненным циклом
SID	Security Identifier - идентификатор безопасности

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

Сокращение	Расшифровка
ГОСТ	Нормативно-правовой документ, в соответствии требованиями которого производится стандартизация производственных процессов и оказания услуг
ЖЦ	Жизненный цикл
МХ ПАТ	Места хранения ПАТ
ОГВ	Органы государственной власти Российской Федерации - участники процесса контроля безопасного обращения с пестицидами и агрохимикатами
ПА	Пестициды и агрохимикаты
ПАТ	Пестициды, агрохимикаты и производимые на их основе тукоsmеси
СМЭВ	Система межведомственного электронного взаимодействия
СПО	Специальное программное обеспечение
ТЧ	Табличная часть
ХС	Хозяйствующий субъект
API	Программный сетевой интерфейс, обеспечивающий приложениям ХС доступ к функционалу Системы

1. ВХОДНЫЕ И ВЫХОДНЫЕ ДАННЫЕ

1.1. Взаимодействие с хозяйствующими субъектами

1.1.1. Входные данные, вводимые хозяйствующими субъектами

Хозяйственные субъекты в ручном режиме или с помощью пользовательского приложения (через API) вводят данные следующих справочников и документов:

- карточка справочника пестицидов, агрохимикатов и тукосмесей;
- карточка справочника хозяйствующих субъектов;
- универсальный передаточный документ (накладная);
- накладная импорта ПАТ;
- производственная накладная;
- план применения ПА;
- акт применения ПА.

1.1.2. Выходные данные, получаемые хозяйствующими субъектами

Хозяйственные субъекты получают информацию из следующих справочников и документов:

- справочник пестицидов, агрохимикатов и тукосмесей;
- справочник мест хранения ПАТ;
- справочник хозяйствующих субъектов;
- реестр партий ПА;
- универсальный передаточный документ (накладная);
- накладная импорта ПА;
- производственная накладная;
- акт инвентаризации;
- справочник мест применения;
- план применения ПА;
- акт применения ПА.

1.2. Взаимодействие с приложениями ХС

1.2.1. Обмен информации СПО с приложениями ХС обеспечивается через API, используя сформированные HTTP-запросы (GET/POST).

1.2.2. Перечень и структура обрабатываемых запросов описаны в Приложении.

1.3. Взаимодействие с компонентом «Цербер»

Обмен информации СПО с компонентом «Цербер» ФГИС «ВетИС» включает в себя:

- проверку авторизации по учётным данным пользователя;
- импорт списка зарегистрированных хозяйствующих субъектов;
- импорт списка зарегистрированных ПА;
- импорт списка поднадзорных объектов.

Обмен информации СПО с компонентом «Цербер» ФГИС «ВетИС» осуществляется через предоставляемый программный интерфейс или файлы форматов *.xls, *.xlsx.

Формы и форматы обмена данными с компонентом «Цербер» ФГИС «ВетИС» описаны в справочной системе ФГИС «ВетИС» (https://help.vetrif.ru/wiki/Компонент_Ветис.API).

ПЕРЕЧЕНЬ И СТРУКТУРА ОБРАБАТЫВАЕМЫХ ЗАПРОСОВ

1. ОПИСАНИЕ

Обмен данными с Системой через программный сетевой интерфейс SEAPI производится посредством JSON-сообщений, передаваемых по протоколу HTTP.

Кодировка сообщений UTF-8.

2. СТРУКТУРА ПРОТОКОЛА ОБМЕНА

Протокол обмена можно разделить на 2 слоя: транспортный (он же сервисный) и логический (бизнес-функции или функции ЖЦ сущностей).

Транспортный (сервисный) слой - соглашения о наборах атрибутов JSON-пакетов запроса и ответа, и набор команд, которые позволяют получить информацию о метаданных сущностей и их ЖЦ, фазах ЖЦ (состояниях) и командах, доступных для управления сущностями.

Более подробно транспортный слой описан в п.6.3.

Логический слой (бизнес-функции) - вызов операций над сущностями, которые зарегистрированы в SEAPI.

Набор доступных команд зависит от типа сущности, состояния объекта и прав обращающегося.

Список доступных команд для каждого состояния каждой сущности можно получить через команды сервисного слоя.

Более подробно транспортный слой описан в п.6.2.

Иные спецификации

Все сущности и массивы сущностей (а также информация о метаданных сущности) сериализуются в формате HRJSON.

3. ОБЩИЕ КОНЦЕПЦИИ

Протокол обмена предполагает вызов операций у информационных сущностей, находящихся в БД.

Получение списка сущностей и поиск также являются операциями, вызываемыми у конкретных сущностей (или их классов).

Каждый класс сущности имеет ЖЦ для объектов - набор состояний, в которых могут находиться объекты,

Каждое состояние имеет свой фиксированный набор операций, который можно вызвать только у объектов в этом состоянии.

Выделяется 2 псевдо-состояния, общих для всех сущностей: *static* и *any*, к которым "прикрепляются" операции, "доступные без указания конкретного объекта" (статические), и доступные для объектов в любом состоянии. Объекты не могут находиться в этих состояниях, потому они и называются "псевдо-состояниями", хотя они и присутствуют в описании ЖЦ.

Как правило, у сущности имеются следующие операции у псевдо-состояний *static* и *any*:

- *static/getList()*
- *static/createNew()*
- *any/getFullCard()*

Имена операций имеют имена, включающие имя состояния и имя команды.

ВАЖНО! Операции, прикрепленные к разным состояниям, - это разные операции, например:, *draft/updateCard* и *actual/updateCard()* - это разные команды: одна для состояния документа «черновик», вторая для состояния «актуально».

Действие каждой команды определяется отдельно для каждой конкретной сущности, т.е. *draft/updateCard()* у сущности *Invoice* и *draft/updateCard()* у сущности *Contractor* могут, в теории, выполнять совершенно разные действия.

Каждая функция имеет набор входных аргументов (передается на уровень транспортного слоя в аргумент *opargs*) и набор результатов (передается на уровне транспортного слоя в поле *resData*).

Описания функций и аргументов приведены в п.7.

4. ОТЧЕТЫ

Отчеты - это такие же сущности, зарегистрированные в SEAPI-механизмах, с той разницей, что у них не может быть объектов, а все функции прикреплены к фазе *static*.

5. АВТОРИЗАЦИЯ

На уровне обращения к API требуется предоставить SID - идентификатор сессии, или токен, подписанный используемым сервером авторизации.

По SID Система определяет уровень доступа, список прикрепленных к аккаунту организаций и предоставляет эту информацию функциям сущностей, которые обрабатывают с учетом выданной аккаунту роли, списка прикрепленных организаций, а также иных атрибутов.

5.1. Получение ключа доступа

Для доступа внешнего ПО к Системе требуется получить ключ доступа. Ключ генерируется на странице Системы

<доменное имя хоста системы>/probeInnerArm/innerArm/seapiAuth

на которой необходимо авторизоваться под логином ФГИС «ВетИС». Этот ключ копируется в настройки пользовательского приложения, и далее используется для расчета SID по алгоритму, описанному ниже.

5.2. Генерация SID и запуск сессии.

Для выполнения операций SEAPI внешнему приложению требуется код сессии (SID). Для его генерации требуются login, domain, ключ доступа к API.

Ключ доступа - индивидуален для каждого аккаунта. Код сессии (SID) индивидуален для каждой сессии.

Генерация SID и запуск сессии производится в 3 шага:

Шаг 1. Получить с сервера salt - случайное число, которое будет использоваться в коде этой сессии.

Отправить http-запрос с параметрами на точку доступа сервиса api-авторизации */seapiAuth (.../probeInnerArm/innerArm/seapiAuth)* с параметрами:

- op: startAuth;
- login: ваш логин;
- domain: домен вашего аккаунта. по умолчанию "local".

Все параметры передаются одним из способов:

- get-параметры;
- post-параметры формы (Content-type: form/multipart);
- атрибуты JSON-контейнера (post-запрос с телом в виде JSON и Content-type: application/json),

аналогично обращениям к основным функциям SEAPI.

Шаг 2. Генерация SID.

Генерация SID производится во внешнем приложении с использованием полученного *salt* и уже имеющегося ключа доступа к API (*accessKey*).

Для этого необходимо:

- сконкатенировать *salt* с ключем доступа через точку, как строки в кодировке UTF-8: *salt+"."+ accessKey*
- рассчитать хеш SHA256 этой строки, представить его в виде hex-строки в нижнем регистре и снабдить префиксом "SEAPISID:". Результатом этого действия и будет SID, который требуется для выполнения SEAPI-операций.

Шаг 3. Подтверждение/проверка SID и запуск сессии.

Для проверки правильности генерации SID необходимо открыть сессию. В случае удачного открытия сессии SID сгенерирован правильно.

Примечания:

- сессия валидна по умолчанию примерно 15 минут, для ее продления необходимо выполнить Шаг 2;
- продлить можно только сессию, которая не закрыта;
- сессия может быть закрыта принудительно или по таймауту.

Пример реализации механизма авторизации на языке JavaScript приведен на тестовом и демонстрационных стендах в тексте HTML-страницы */probeInnerArm/testLinksG_seapi_withKeyLinks.html*.

6. ОБЩЕЕ ОПИСАНИЕ SEAPI

6.1. Точки доступа

Существуют 3 сервиса для работы с API для внешнего приложения, они же "точки доступа":

- */seapi* - непосредственно выполнение команд и операций. Для обмена данными используется JSON. Предназначен только для программного обмена данными. Для обращения через это сервис требуется актуальный SID.
- */seapiAuth* - сервис авторизации. Осуществляет проверку сгенерированного SID, открытие сессии, продление сессии, закрытие сессии. Для обмена данными используется JSON. Данный сервис предназначен только для программного обмена данными.
- */seapiCP* - контрольная HTML-страница управления ключом доступа к API: получение ключа доступа, смена ключа доступа, отключение ключа доступа к API и запрет пользователю авторизовываться по SID-коду сессии в API. Данная страница предназначена для доступа ХС (через браузер с авторизацией по *OAuth* в «ВетИС-паспорте») к управлению ключом доступа для его дальнейшего добавления в параметры своего клиентского ПО.

Работа с этими точками доступа производится следующим образом:

- Получение ключа программного доступа к SEAPI (с помощью */seapiCP*). Выполняется пользователем вручную, с помощью первой точки доступа – страницы */probeInnerArm/innerArm/seapiCP*. Данная операция выполняется однократно или при необходимости смены ключа;
- Авторизация и запуск сессии (с помощью */seapiAuth*). Выполняется программно: полностью - при начале очередного сеанса работы, частично - при необходимости продления времени сессии (см команду подтверждение SID). Процесс генерации и подтверждения SID описан в разделе 5.2.
- Выполнение команд и операций (с помощью */seapi*). Выполняется программно, многократно внутри сессии. Структура пакета и формат передачи описаны в п. 6.3.

6.2. Логическая модель

API предназначено для работы с сущностями, реализующими интерфейс *ILifeCycledEntity*, у которых описаны жизненные циклы и операции доступные для каждой из фаз ЖЦ.

Все обращения к API можно разделить на:

- системные: базовые функции, список доступных классов сущностей, метаданные сущности, ЖЦ сущности и тп;
- общие: получение состояния сущности (по её id) и доступные для данного объекта перечни операций;
- вызов операций ЖЦ - классов или объектов.

Важно отметить, что для подавляющего числа сущностей:

- получение списка сущностей осуществляется статической операцией *"static/getList()"*;
- получение карточки сущности осуществляется доступной для любого состояния операцией *any/getFullCard()*;

- операция обновления карточки чаще всего осуществляется операцией *draft/doPartUpdate()* или ей подобной (в зависимости от ЖЦ конкретной сущности).

6.3. Транспортный слой

6.3.1. Контейнер ответа

В подавляющем числе случаев, в ответ будет отдана JSON-структура-контейнер, у которой в корне присутствуют следующие поля:

- **resCode** - код с общим результатом выполнения:
 - *меньше 0*: ошибка, нештатное поведение, метод не выполнен из-за неожиданной ошибки, ошибки в аргументах;
 - *равен 0*: метод выполнен без ошибок;
 - *больше 0*: метод выполнен, но возникли обработанные, или ожидаемые/запланированные исключения;
- **resMsg** - текстовое сообщение об ошибке;
- **resDescription** – словесное описание выходного состояния;
- **resData** – объект, представляющий собой результат выполнения операции (в виде JSONObject, перечень полей которого соответствует перечню результатов операции). Поле не обязательно при ошибке.

Список полей может расширяться.

6.3.2. Параметры обращения

Обращение предоставляет собой набор, описывающий команду и аргументы функции API или операции ЖЦ.

В простом случае, все эти параметры - аргументы GET-запроса.

Допускается вызов с через запрос POST, но с обязательным использованием параметра *Content-Type: multipart/form-data*, а также JSON-пакет-контейнером через XMLHttpRequest с обязательным использованием параметра "Content-type: application/json".

Набор параметров, описывающих вызываемые API - команды:

- **SID** - идентификатор сессии
- **com** - команда API, которую надо выполнить:
 - *getRegisteredEntityNames* – получить имена сущностей;
 - *getEntityMD* - получить метаданные сущности (для каждой табличной части сущности надо делать отдельный запрос);
 - *getEntityLCNames* - отдать имена допустимых ЖЦ;
 - *getEntityLC* - получить описание ЖЦ сущности, по имени ЖЦ (все состояния и все операции всех состояний);
 - *getEntityStates* – получить состояния сущности, по всем ЖЦ, которые допустимы у сущности;
 - *getAllowedEntityOperations* - получить перечень допустимых операций для заданной сущности;
 - *execOperation* - выполнение операции ЖЦ сущности (выполняется по умолчанию);
- **otype** - имя класса сущности (необходимо при выполнении команды, касающиеся класса или объекта сущности);
- **op** - имя операции ЖЦ, вызываемой у сущности (тип операции указан в **otype**, идентификатор – в **oid**);
- **opargs** - строка в формате JSON-объекта с аргументами операции; (аргументы описаны в описании операций);
- **oid** - идентификатор сущности в Системе (он же - идентификатор записи в БД). Указывается, если выполняется команда над объектом Системы.

Иных параметров на транспортном уровне в настоящее время не предусмотрено.

Набор аргументов операций передается в параметре **opargs** - JSON-объекта с именами атрибутов, которые требуются для выполнения конкретной операции жизненного цикла конкретной сущности. Состав аргументов перечисляется в описании функции.

Результаты работы функции отдаются в поле `resData` в контейнере ответа (см выше) в виде JSON-структуры, с именами полей, равными именам результатов, перечисленных в описании операции.

7. ОПИСАНИЕ API

1. Сущность "Warehouse"

Наименование	Warehouse
Описание	Склад

Наименование ЖЦ: mainlc

1.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования.

1.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

1.2 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

1.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

1.4 Состояние "static"

Наименование: static

Описание: статические операции

1.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

1.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

1.5 Состояние "draft"

Наименование: draft

Описание: черновик карточки, не используется, пока карточка не будет переведена в состояние «Активная»

1.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
-------------	--------------------

Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons.storages.flatObjects.AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением</p>

Имя аргумента	Тип значения	Описание
		<p>'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

1.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

1.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состоянии «Отменено»
Описание	переводит карточку в состояние «Отменено» – информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

1.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

1.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

2. Сущность "Invoice"

Наименование	Invoice
Описание	<p>Накладная (УПД): Универсальный передаточный документ, который, в зависимости от проведенной операции, может отражать и перемещение ПА между складами и смену владельца</p> <ul style="list-style-type: none"> * Создается исходным владельцем партии. * Отражается в 'Исходящих УПД' у Продавца ПАТ * Принимается покупателем (новым владельцем партии). * Отражается во 'Входящих УПД' у Покупателя ПАТ <p>Грузоотправитель и Грузополучатель присутствуют только косвенно, в виде своих складов.</p>

Наименование ЖЦ: mainlc

2.1 Состояние "static"

Наименование: static

Описание: статические операции

2.1.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100

Имя аргумента	Тип значения	Описание
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '_NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

2.1.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

2.2 Состояние "deliveredToDestroy"**Наименование:** deliveredToDestroy**Описание:** ПА отправлен на утилизацию**2.2.1 Функция "deliveredToDestroy/setArchived()"**

Имя функции	deliveredToDestroy/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.**Результаты:** отсутствуют.**2.3 Состояние "delivered"****Наименование:** delivered**Описание:** ПА доставлен получателю**2.3.1 Функция "delivered/setArchived()"**

Имя функции	delivered/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.**Результаты:** отсутствуют.

2.4 Состояние "canceled"**Наименование:** canceled**Описание:** отменен (к удалению, в учете не участвовал)**2.5 Состояние "archived"****Наименование:** archived**Описание:** перевести карточку в состояние «Архив» (к удалению, запись участвовала в учете, но сейчас уже подлежит удалению из активной БД)**2.6 Состояние "onStore"****Наименование:** onStore**Описание:** ПА на отв. хранении**2.6.1 Функция "onStore/setArchived()"**

Имя функции	onStore/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.**Результаты:** отсутствуют.

2.7 Состояние "onRetailSale"

Наименование: onRetailSale

Описание: ПА передан в розничную продажу

2.7.1 Функция "onRetailSale/setArchived()"

Имя функции	onRetailSale/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.8 Состояние "lost"

Наименование: lost

Описание: ПА не доставлен, груз утерян

2.8.1 Функция "lost/setArchived()"

Имя функции	lost/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9 Состояние "draft"

Наименование: draft

Описание:

2.9.1 Функция "draft/doSend()"

Имя функции	draft/doSend()
Действие	отгрузить получателю
Описание	уведомить о продаже с отправкой получателю

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.2 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons.storages.flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля:</p> <ul style="list-style-type: none"> * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя. <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных</p>

Имя аргумента	Тип значения	Описание
		<p>частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p>

Имя аргумента	Тип значения	Описание
		На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

2.9.3 Функция "draft/doStore()"

Имя функции	draft/doStore()
Действие	перевести на отв. хранение
Описание	уведомить о продаже с перевести на отв. хранение

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.4 Функция "draft/doSendToRetale()"

Имя функции	draft/doSendToRetale()
Действие	отгрузить в розничную продажу
Описание	отметить груз как отправленный в направлении розничной торговли

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.5 Функция "draft/doSendToDestroy()"

Имя функции	draft/doSendToDestroy()
Действие	отгрузить ПА на утилизацию
Описание	отметить партии ПА как отправленные в направлении организации, занимающейся уничтожением ПА и хим. отходов.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.6 Функция "draft/doSendToIndustrial()"

Имя функции	draft/doSendToIndustrial()
Действие	отгрузить индустриальному потребителю
Описание	отметить груз как отправленный в направлении розничной торговли

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.7 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.9.8 Функция "draft/doSendToExport()"

Имя функции	draft/doSendToExport()
Действие	отгрузить индустриальному потребителю
Описание	отметить груз как отправленный в направлении розничной торговли

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.10 Состояние "onIndustrSale"

Наименование: onIndustrSale

Описание: ПА передан в промышленное использование

2.10.1 Функция "onIndustrSale/setArchived()"

Имя функции	onIndustrSale/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.11 Состояние "any"

Наименование: any

Описание: все состояния

2.11.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.11.2 Функция "any/undoLastTransaction()"

Имя функции	any/undoLastTransaction()
Действие	отменить последнюю операцию по остаткам по документу
Описание	выбирается последняя транзакция, совершенная документом в регистре остатков, и проводится попытка отменить эту транзакцию.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.12 Состояние "returned"

Наименование: returned

Описание: полностью возвращен грузоотправителю

2.12.1 Функция "returned/setArchived()"

Имя функции	returned/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.13 Состояние "onReturning"

Наименование: onReturning

Описание: полностью возвращен грузоотправителю

2.13.1 Функция "onReturning/notifyLost()"

Имя функции	onReturning/notifyLost()
Действие	отметить утерянным
Описание	отметить возвращаемый груз утерянным

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.13.2 Функция "onReturning/updateReturnedCounts()"

Имя функции	onReturning/updateReturnedCounts()
Действие	установить дату возврата, количество полученного груза и статус/причину расхождения
Описание	обновить количество фактически возвращенного - подготовка к переводу в состояние 'принято'

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'=-1' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

2.13.3 Функция "onReturning/notifyDelivered()"

Имя функции	onReturning/notifyDelivered()
Действие	отметить доставленным
Описание	отметить возвращаемый груз доставленным

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.14 Состояние "onWay"

Наименование: onWay

Описание: ПА в доставке к получателю

2.14.1 Функция "onWay/notifyLost()"

Имя функции	onWay/notifyLost()
Действие	уведомить об утере
Описание	уведомить о факте утери - ПА не доставлен до приемного склада, груз не поступил, или утерян, или поврежден весь.

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.14.2 Функция "onWay/notifyDelivered()"

Имя функции	onWay/notifyDelivered()
Действие	подтвердить доставку
Описание	уведомить о фактической доставке ПА до приемного склада, без потерь

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.14.3 Функция "onWay/updateAcceptedCounts()"

Имя функции	onWay/updateAcceptedCounts()
Действие	установить дату приемки, количество принятого груза и статус/причину расхождения

Описание	<p>обновить количество фактически доставленного - подготовка к переводу в состояние 'принято' всё что надо отправить назад - приходится и потом отправляется назад отдельной накладной. Операция эквивалентна updateCard() но позволяет изменить только отдельные поля</p>
----------	--

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	<p>commons. storages. flatObjects. AMDObject</p>	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

2.14.4 Функция "onWay/sendBack()"

Имя функции	onWay/sendBack()
-------------	------------------

Действие	вернуть обратно
Описание	отправить груз обратно отправителю

Аргументы: отсутствуют.

Результаты: отсутствуют.

2.15 Состояние "onExportSent"

Наименование: onExportSent

Описание: ПА передан в промышленное использование

2.15.1 Функция "onExportSent/setArchived()"

Имя функции	onExportSent/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

3. Сущность "PatUsagePlanRec"

Наименование	PatUsagePlanRec
Описание	Запись о планируемом применении ПАТ

Наименование ЖЦ: mainlc

3.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования

3.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

3.2 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

3.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

3.4 Состояние "static"

Наименование: static

Описание: статические операции

3.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

3.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

3.5 Состояние "draft"

Наименование: draft

Описание: черновик карточки, не используется, пока карточка не будет переведена в состояние «Активная»

3.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1'</p>

Имя аргумента	Тип значения	Описание
		<p>- строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

3.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
-------------	-------------------

Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

3.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

3.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

3.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

4. Сущность "Inventorization"

Наименование	Inventorization
Описание	Акт инвентаризации

Наименование ЖЦ: mainlc

4.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования

4.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

4.2 Состояние "archived"

Наименование: archived

Описание: перевести карточку в состояние «Архив» (к удалению, запись участвовала в учете но сейчас уже подлежит удалению из активной БД)

4.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

4.4 Состояние "static"

Наименование: static

Описание: статические операции

4.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

4.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

4.5 Состояние "draft"

Наименование: draft

Описание: черновик

4.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

4.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
Действие	перевести карточку в состояние «Актуально»

Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.
----------	--

Аргументы: отсутствуют.

Результаты: отсутствуют.

4.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

4.6 Состояние "any"

Наименование: any

Описание: все состояния

4.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

4.6.2 Функция "any/undoLastTransaction()"

Имя функции	any/undoLastTransaction()
Действие	отменить последнюю операцию по остаткам по документу

Описание	выбирается последняя транзакция, совершенная документом в регистре остатков - и проводится попытка отменить эту транзакцию.
----------	---

Аргументы: отсутствуют.

Результаты: отсутствуют.

5. Сущность "WarehouseStates"

Наименование	WarehouseStates
Описание	Получение отчётов

Наименование ЖЦ: mainlc

5.1 Состояние "static"

Наименование: static

Описание: статические операции

5.1.1 Функция "static/getTotals()"

Имя функции	static/getTotals()
Действие	получить остатки по складам
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
timeStamp	int	момент времени, в одном из форматов ISO. если часовой пояс не указан, считается, что это GMT+3, Москва, например: «ГТТГ.ММ.ДД чч:мм:сс.щщ» (микросекунды).
warehouseId	String	идентификатор склада, по которому запрашиваем остатки
batchId	int	идентификатор партии, или пустое значение для разворачивания итогов, или JSON-массив со списком партий (как в фильтре). По умолчанию - пустое значение (разворачивать список партий)
patProductId	int	идентификатор ПАТ или пустое значение для разворачивания итогов, или JSON-массив со списком партий (как в фильтре). По умолчанию - пустое значение (разворачивать список партий)
ownerId	int	владелец партии

Имя аргумента	Тип значения	Описание
fExtractionMode	String	тип извлечения остатков, 'normal' или 'justBefore', включая текущий момент, на момент перед запрашиваемым (например, так можно получить состояние перед проведением документа). По умолчанию - normal
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100

Результаты:

Имя результата	Тип значения	Описание
totalsTable	String	JSON, 2-мерный. Первая строка - имена заголовков. . (может отдаваться без '__NAME' колонок!) ['warehouseId', 'batchId', 'quantity', 'ownerContractorId']

5.1.2 Функция "static/getBatchConnectionsReport()"

Имя функции	static/getBatchConnectionsReport()
Действие	создание отчёта 'Связи партий'
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
batchId	long	id партии, предков и потомков которой хотим найти

Результаты:

Имя результата	Тип значения	Описание
HTMLPage	String	HTML-страница с данными отчета. На текущий момент просто HTML-заглушка

5.1.3 Функция "static/getTurnoversReport()"

Имя функции	static/getTurnoversReport()
Действие	создание отчёта 'Остатки на складах'
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
dateFrom	String	момент времени, в одном из форматов ISO. если часовой пояс не указан, считаем что это GMT+3, Москва, например: 'ГГГГ.ММ.ДД чч:мм:сс.ццц' (микросекунды).
dateTo	String	момент времени, в одном из форматов ISO. если часовой пояс не указан, считаем что это GMT+3, Москва, например: 'ГГГГ.ММ.ДД чч:мм:сс.ццц' (микросекунды).
warehouseId	int	идентификатор склада, по которому делаем отчет по партии
productId	int	идентификатор партии или пустое значение для разворачивания итогов, или JSON-массив со списком партий (как в фильтре). По умолчанию - пустое значение (разворачивать список партий)
ownerId	int	владелец партии

Результаты:

Имя результата	Тип значения	Описание
HTMLPage	String	HTML-страница с данными отчета. На текущий момент - просто HTML-заглушка

5.1.4 Функция "static/getPatApplicationReport()"

Имя функции	static/getPatApplicationReport()
Действие	Создание отчёта 'Применение ПАТ на полях'
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
dateFrom	String	момент времени, в одном из форматов ISO. если часовой пояс не указан, считаем что это GMT+3, Москва, например: 'ГГГГ.ММ.ДД чч:мм:сс.ццц' (микросекунды).
dateTo	String	момент времени, в одном из форматов ISO. если часовой пояс не указан, считаем что это GMT+3, Москва, например: 'ГГГГ.ММ.ДД чч:мм:сс.ццц' (микросекунды).
acreageId	int	идентификатор поля для применения, по которому делаем отчет по ПАТ

Результаты:

Имя результата	Тип значения	Описание
HTMLPage	String	HTML-страница с данными отчета.

6. Сущность "PAImportRecord"

Наименование	PAImportRecord
Описание	Запись об импорте ПА

Наименование ЖЦ: mainlc

6.1 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

6.2 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

6.3 Состояние "static"

Наименование: static

Описание: статические операции

6.3.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100

Имя аргумента	Тип значения	Описание
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

6.3.2 Функция "static/createImported()"

Имя функции	static/createImported()
Действие	создать запись об импорте ПА в РФ
Описание	только для автоматических процедур импорта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

6.4 Состояние "imported"**Наименование:** imported**Описание:** груз растаможен, выехал в направлении склада**6.4.1 Функция "imported/setCanceled()"**

Имя функции	imported/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.**Результаты:** отсутствуют.**6.4.2 Функция "imported/setWarehouse()"**

Имя функции	imported/setWarehouse()
Действие	обновить запись об импорте
Описание	Установить целевой склад, целевого владельца, дату получения и комментарий. (targetWarehouseId, dateReceive, description)

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p>

Имя аргумента	Тип значения	Описание
		<p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию
theCard	string	Структура с изменяемыми полями json, с идентифицирующими hrjson-полями ('_id','_type'), hrjson-контейнеров табличных частей.

Результаты: отсутствуют.

6.4.3 Функция "imported/setReceived()"

Имя функции	imported/setReceived()
Действие	уведомить о доставке ПА до склада
Описание	уведомить о доставке ПА до склада

Аргументы: отсутствуют.

Результаты: отсутствуют.

6.4.4 Функция "imported/techUpdate()"

Имя функции	imported/techUpdate()
Действие	обновить карточку при импорте
Описание	техническая процедура для обновления карточки.

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля:</p> <ul style="list-style-type: none"> * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя. <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня,</p>

Имя аргумента	Тип значения	Описание
		<p>т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см. typeSpecs аргумента theCard</p>

Имя аргумента	Тип значения	Описание
comment	String	заметки оператора, о том, зачем он совершает эту операцию
theCard	string	Структура с изменяемыми полями json, с идентифицирующими hrjson-полями ('_id','_type'), hrjson-контейнеров табличных частей.

Результаты: отсутствуют.

6.5 Состояние "received"

Наименование: received

Описание: груз доставлен склада

6.5.1 Функция "received/setArchived()"

Имя функции	received/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

6.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

6.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

7. Сущность "OpTransaction"

Наименование	OpTransaction_SEW
Описание	Запись о вызове операции ЖЦ

Наименование ЖЦ: mainlc

7.1 Состояние "actual"

Наименование: actual

Описание: действующие операции, не снятые с регистрации

7.2 Состояние "canceled"

Наименование: canceled

Описание: отмененные операции - отменено, пусть и временно.

7.3 Состояние "archived"

Наименование: archived

Описание: отмененные операции, архивные

7.4 Состояние "static"

Наименование: static

Описание: статические операции

7.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100

getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '_NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

7.5 Состояние "draft"**Наименование:** draft**Описание:** техническое состояние, заполняется атрибутами.**7.6 Состояние "any"****Наименование:** any**Описание:** операции над карточками в любом состоянии**7.6.1 Функция "any/getFullCard()"**

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности

Описание	отдает 'глубокую сериализацию' карточки сущности
----------	--

Аргументы: отсутствуют.

Результаты: отсутствуют.

8. Сущность "PATBatch"

Наименование	PATBatch
Описание	Запись о партии ПА

Наименование ЖЦ: mainlc

8.1 Состояние "actual"

Наименование: actual

Описание: партии, когда либо находившиеся в обороте

8.2 Состояние "canceled"

Наименование: canceled

Описание: отмененные партии

8.3 Состояние "archived"

Наименование: archived

Описание: партии, которые подлежат удалению перевести карточку в состояние «Архив» из рабочей БД

8.4 Состояние "static"

Наименование: static

Описание: статические операции

8.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0

size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

8.5 Состояние "draft"**Наименование:** draft**Описание:** техническое состояние, заполняется атрибутами.**8.6 Состояние "any"****Наименование:** any**Описание:** операции над карточками в любом состоянии**8.6.1 Функция "any/getFullCard()"**

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

9. Сущность "PatUsage"

Наименование	PatUsage
Описание	Применение ПА

Наименование ЖЦ: mainlc

9.1 Состояние "actual"

Наименование: actual

Описание: актуально

9.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

9.2 Состояние "archived"

Наименование: archived

Описание: архив

9.3 Состояние "canceled"

Наименование: canceled

Описание: архив

9.4 Состояние "static"

Наименование: static

Описание: статические операции

9.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

9.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

9.5 Состояние "draft"

Наименование: draft

Описание: черновик

9.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDOBJECT	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='1' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

9.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

9.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

9.6 Состояние "any"

Наименование: any

Описание: все состояния

9.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

9.6.2 Функция "any/undoLastTransaction()"

Имя функции	any/undoLastTransaction()
Действие	отменить последнюю операцию по остаткам по документу
Описание	выбирается последняя транзакция, совершенная документом в регистре остатков, и проводится попытка отменить эту транзакцию.

Аргументы: отсутствуют.

Результаты: отсутствуют.

10. Сущность "Acreage"

Наименование	Acreage
Описание	Место применения ПА

Наименование ЖЦ: mainlc

10.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования

10.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

10.2 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

10.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

10.4 Состояние "static"**Наименование:** static**Описание:** статические операции**10.4.1 Функция "static/getList()"**

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

10.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

10.5 Состояние "draft"

Наименование: draft

Описание: черновик карточки, не используется, пока карточка не будет переведена в состояние «Активная»

10.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'=-</p>

Имя аргумента	Тип значения	Описание
		<p>l' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

10.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
-------------	-------------------

Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

10.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

10.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

10.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

11. Сущность "UsrFavourite"

Наименование	UsrFavourite
Описание	Запись об избранном

Наименование ЖЦ: mainlc

11.1 Состояние "actual"

Наименование: actual

Описание: активное состояние, обычная карточка

11.2 Состояние "canceled"

Наименование: canceled

Описание: отмененные записи

11.3 Состояние "static"

Наименование: static

Описание: статические операции

11.3.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1

Имя аргумента	Тип значения	Описание
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: " <code>[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]</code> ", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: " <code>['colNameA','colNameB', ... , 'colNameZ']</code> "

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами ' <code>__NAME</code> ' для указания имен ссылочных объектов. отдается, если не отдается objList

11.3.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

11.4 Состояние "any"

Наименование: any

Описание: операции над карточками в любом состоянии

11.4.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

11.4.2 Функция "any/updateCard()"

Имя функции	any/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDOObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля:</p> <ul style="list-style-type: none"> * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя. <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит</p>

		<p>строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

12. Сущность "Contractor"

Наименование	Contractor
Описание	Контрагент

Наименование ЖЦ: mainlc

12.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования.

12.1.1 Функция "actual/updateCard()"

Имя функции	actual/updateCard()
Действие	изменить свою карточку
Описание	вы можете исправить некоторые поля в своей карточке. профиль деятельности, указать публичную контактную информацию и описание.

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDOObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p>

Имя аргумента	Тип значения	Описание
		<p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='1' - строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части,</p>

Имя аргумента	Тип значения	Описание
		могут быть наложены ограничения - см. typeSpecs аргумента theCard
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

12.1.2 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

12.2 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

12.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

12.4 Состояние "static"

Наименование: static

Описание: статические операции

12.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

12.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

12.4.3 Функция "static/getOneForAuthNeeds()"

Имя функции	static/getOneForAuthNeeds()
Действие	получить краткую карточку контрагента
Описание	получить краткую карточку контрагента по ИНН или guid business entity компонента «Цербер» для заполнения атрибутов копии аккаунта в Keycloak, и, при необходимости, загрузить из сервиса ФГИС «ВетИС» через SOAP-WS-сервис.

Аргументы:

Имя аргумента	Тип значения	Описание
inn	String	ИНН, по которому ищется контрагент
beGuid	String	business entity ХС компонента «Цербер», по которому ищется контрагент

Результаты:

Имя результата	Тип значения	Описание
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '_NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

12.5 Состояние "draft"

Наименование: draft

Описание: черновик карточки, не используется, пока карточка не будет переведена в состояние «Активная»

12.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDBObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля:</p> <ul style="list-style-type: none"> * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя. <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

12.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

12.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

12.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

12.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

13. Сущность "PatProduct"

Наименование	PatProduct
Описание	Запись о Пестициде или Агрехимикате

Наименование ЖЦ: mainlc

13.1 Состояние "actual"

Наименование: actual

Описание: актуальная карточка, со сведениями, которые соответствуют действительности, или которая предназначена для использования

13.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

13.2 Состояние "archived"

Наименование: archived

Описание: архивная карточка, выведенная из использования

13.3 Состояние "canceled"

Наименование: canceled

Описание: отмененная карточка, не бывшая в активном состоянии, или создание которой признано ошибочным (не путать с состоянием «Архив»!).

13.4 Состояние "static"

Наименование: static

Описание: статические операции

13.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '_NAME' для указания имен ссылочных объектов. отдается, если не отдается objList
-----------	--------	---

13.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

13.5 Состояние "draft"

Наименование: draft

Описание: черновик карточки, не используется, пока карточка не будет переведена в состояние «Активная»

13.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDOObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='-1'</p>

Имя аргумента	Тип значения	Описание
		<p>- строка табличной части будет создана.</p> <p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

13.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
-------------	-------------------

Действие	перевести карточку в состояние «Актуально»
Описание	переводит карточку в состояние «Актуально» - актуальная информация для применения.

Аргументы: отсутствуют.

Результаты: отсутствуют.

13.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состояние «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

13.6 Состояние "any"

Наименование: any

Описание: операции над всеми карточками, в любых состояниях

13.6.1 Функция "any/getFullCard()"

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.

Результаты: отсутствуют.

14. Сущность "PatProductionAct"

Наименование	PatProductionAct
Описание	Производство ПАТ

Наименование ЖЦ: mainlc

14.1 Состояние "actual"

Наименование: actual

Описание: актуально

14.1.1 Функция "actual/setArchived()"

Имя функции	actual/setArchived()
Действие	перевести карточку в состояние «Архив»
Описание	переводит карточку в состояние «Архив» - устаревшая или уже неактуальная информация.

Аргументы: отсутствуют.

Результаты: отсутствуют.

14.2 Состояние "archived"

Наименование: archived

Описание: актуально

14.3 Состояние "canceled"

Наименование: canceled

Описание: актуально

14.4 Состояние "static"

Наименование: static

Описание: статические операции

14.4.1 Функция "static/getList()"

Имя функции	static/getList()
Действие	просмотреть список
Описание	отдает перечень заголовков объектов

Аргументы:

Имя аргумента	Тип значения	Описание
pos	int	смещение начала списка (от 0), значение по умолчанию: 0
size	int	длина списка (от 1). значение по умолчанию: 1, максимальное значение: 100
getFullCards	int	1 - возвращать полносериализованные карточки в objList, или 0 - только перечень атрибутов, в attrTable значение по умолчанию: 1
filters	String	JSON String, JSON array объектов с описанием условий фильтра, например: "[{'column':'colName','condition':'=','value':'value'}, {...}, ..., {...}]", при этом value может быть числом, строкой или массивом строк или чисел. Перечисленные условия объединяются по AND.
orderBy	String	JSON String, JSON array строк с именами колонок, например: "['colNameA','colNameB', ... , 'colNameZ']"

Результаты:

Имя результата	Тип значения	Описание
objList	String	'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable
attrTable	String	JSON массив, 2-мерный. Первая строка - имена заголовков, в конец массива добавляются колонки с суффиксами '__NAME' для указания имен ссылочных объектов. отдается, если не отдается objList

14.4.2 Функция "static/createNew()"

Имя функции	static/createNew()
Действие	создать объект
Описание	конструктор нового объекта

Аргументы: отсутствуют.

Результаты:

Имя результата	Тип значения	Описание
objList	String	Объект, который создан - 'связанный' JSON-массив HRJSON-объектов (глубокая сериализация сущностей с агрегацией всех ссылочных объектов). отдается, если не отдается attrTable

14.5 Состояние "draft"

Наименование: draft

Описание: черновик

14.5.1 Функция "draft/updateCard()"

Имя функции	draft/updateCard()
Действие	изменить карточку сущности
Описание	

Аргументы:

Имя аргумента	Тип значения	Описание
theCard	commons. storages. flatObjects. AMDObject	<p>Карточка объекта, сериализованная в HRJSON, или простой JSON-объект с именами полей, равными изменяемым атрибутам объекта,</p> <p>Во всех JSON-объектах обязательными являются 2 поля: * '_class' с полным именем java-класса (пакеты и имя класса), * '_id' с идентификатором объекта в хранилищах бизнес-слоя.</p> <p>Табличные части передаются в поле '_tparts' - это должен быть JSON-объект, с полями, равными именам табличных частей, в которых лежат массивы объектов с данными строк табличных частей.</p> <p>При разборе значений анализируются значения только первого уровня, т.е. их непосредственные значения (а для вложенных HRJSON-объектов в FK-полях - поле '_id')</p> <p>Вложенные HRJSON-объекты не создаются!</p> <p>ДОБАВНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с '_id'='1' - строка табличной части будет создана.</p>

Имя аргумента	Тип значения	Описание
		<p>УДАЛЕНИЕ СТРОК ТЧ: Если объект в ТЧ приведен с существующим '_id', но и присутствует спец атрибут '_toDelete' со значением 'true' - значит строка ТЧ будет удалена.</p> <p>ПРОПУСК СТРОК ТЧ: Если строка в ТЧ приведена (есть '_id' и '_class'), но не приводятся атрибуты - значит объект меняться не будет, и не будет удаляться.</p> <p>Если строка в ТЧ не приведена (объект пропущен), строка в ТЧ не будет обновляться, и не будет удаляться.</p> <p>Таким образом, имеет смысл приводить только изменяемые/обновляемые строки в ТЧ.</p> <p>Перечни атрибутов рассматриваются как те, которые можно изменять.</p> <p>Если атрибуты будут пропущены, они не будут изменены.</p> <p>На список допустимых для обновления атрибутов и возможность добавлять/убирать строки табличной части, могут быть наложены ограничения - см .typeSpecs аргумента theCard</p>
comment	String	заметки оператора, о том, зачем он совершает эту операцию

Результаты: отсутствуют.

14.5.2 Функция "draft/setActual()"

Имя функции	draft/setActual()
Действие	уведомить о доставке ПА до склада
Описание	уведомить о доставке ПА до склада

Аргументы: отсутствуют.

Результаты: отсутствуют.

14.5.3 Функция "draft/setCanceled()"

Имя функции	draft/setCanceled()
Действие	перевести карточку из состояния «Черновик в состоянии «Отменено»
Описание	переводит карточку в состояние «Отменено» - информация, которая не была в работе, или признана ошибочно заведенной.

Аргументы: отсутствуют.

Результаты: отсутствуют.

14.6 Состояние "any"**Наименование:** any**Описание:** все состояния**14.6.1 Функция "any/getFullCard()"**

Имя функции	any/getFullCard()
Действие	получить полную карточку сущности
Описание	отдает 'глубокую сериализацию' карточки сущности

Аргументы: отсутствуют.**Результаты:** отсутствуют.**14.6.2 Функция "any/undoLastTransaction()"**

Имя функции	any/undoLastTransaction()
Действие	отменить последнюю операцию по остаткам по документу
Описание	выбирается последняя транзакция, совершенная документом в регистре остатков - и проводится попытка отменить эту транзакцию.

Аргументы: отсутствуют.**Результаты:** отсутствуют.