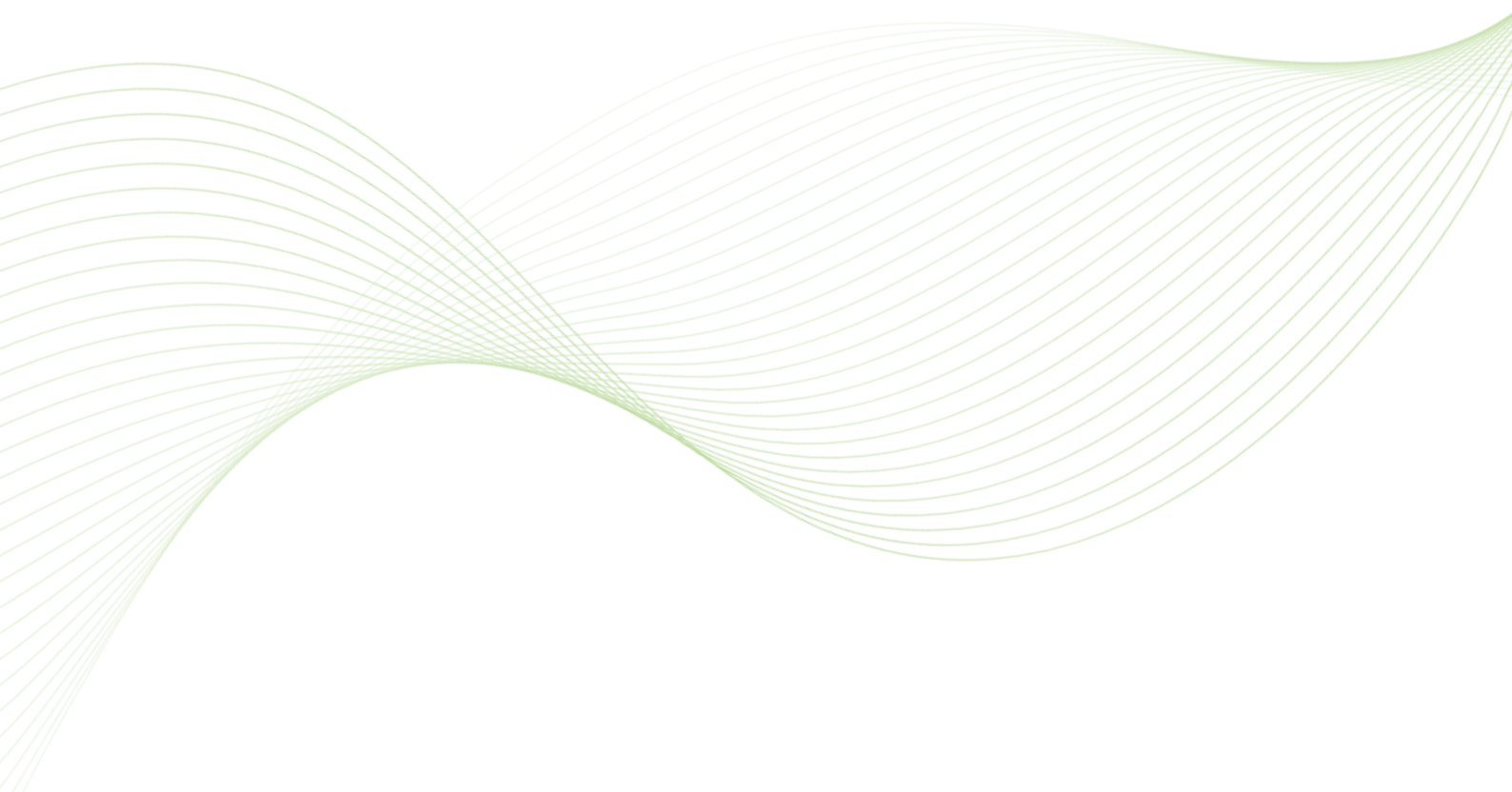




# Информация для разработчика (2.16.0)

Zyfra Industrial Internet of Things Platform  
(ZIIoT)



## Изменения в документе

Версия	Дата	Автор	Описание
1.0	01.12.2023	Пеплин Ф.Н.	Создание документа
1.1	08.02.2024	Пеплин Ф.Н.	zif-om-object: пояснение о переносе запрета на создание конфигураций свойств с типом формула

# Содержание

<b>1. Изменения системы авторазвертывания (zifctl и инфраструктуры) .....</b>	<b>7</b>
<b>2. Версионирование компонентов Платформы для релиза 2.16.0 .....</b>	<b>10</b>
2.1. Совместимость системы разворачивания Платформы с Kubernetes .....	10
2.2. Совместимость Платформы с инфраструктурными компонентами .....	11
2.3. Версии инфраструктурных и вспомогательных сервисов в составе инструмента по авторазвертыванию (zifctl) .....	11
<b>3. Изменения в работе отдельных сервисов .....</b>	<b>14</b>
3.1. Администрирование .....	14
3.1.1. zif-hierarchies .....	14
3.1.2. zif-notifications .....	14
3.1.3. zif-portal-settings-dotnet .....	15
3.1.4. zif-security .....	16
3.1.5. zui-app-nds-configurator .....	16
3.1.6. zui-app-security-settings .....	17
3.1.7. zui-app-shell .....	18
3.2. Концептуальная OM .....	19
3.2.1. zif-om-datareferences .....	19
3.2.2. zif-om-graphql .....	19
3.2.3. zif-om-object .....	20
3.2.4. zif-om-objectmodel2excel .....	22
3.2.5. zif-om-relationship .....	22
3.2.6. zif-om-uom .....	23
3.2.7. zui-app-om .....	23
3.3. Прикладная OM .....	24
3.3.1. Общие критические изменения модуля .....	24
3.3.2. zif-material-lot .....	24
3.3.3. zif-rdm-common .....	25
3.3.4. zif-sm-directories .....	26
3.3.5. zif-sm-domain-api .....	27
3.3.6. zif-sm-operationdefinition .....	28
3.3.7. zif-sm-operationperformance .....	33
3.3.8. zif-sm-operationschedule .....	35
3.3.9. zif-sm-process .....	37
3.3.10. zif-sm-testspecification .....	40
3.3.11. zif-sm-workcalendar .....	41
3.3.12. zif-sm-workdefinition .....	42
3.3.13. zif-sm-workperformance .....	44

3.3.14.	zif-sm-workschedule .....	45
3.3.15.	zui-app-rdm-common.....	46
3.3.16.	zui-sm-directories .....	47
3.3.17.	zif-om-graphql.....	47
3.4.	Приложения .....	59
3.4.1.	zui-app-dashboard-dx .....	59
3.4.2.	zui-app-datalinkeditor .....	59
3.4.3.	zui-app-documentexplorer .....	59
3.4.4.	zui-app-events-registry .....	60
3.4.5.	zui-app-mnemoeditor .....	61
3.4.6.	zui-app-reporteditor .....	61
3.4.7.	zif-mnemoschemes-storage .....	62
3.4.8.	zif-datalink-xl и плагин Excel .....	62
3.4.9.	zif-datalink .....	63
3.4.10.	zif-reporting .....	64
3.4.11.	zif-dashboards .....	64
3.4.12.	zif-propertyset .....	64
3.5.	Расчеты.....	64
3.5.1.	zif-bp-calculate-specification-worker .....	64
3.5.2.	zif-cm-context-functions.....	65
3.5.3.	zif-cm-engine-mvel .....	65
3.5.4.	zif-cm-metadata .....	65
3.5.5.	zif-om-testspecification .....	66
3.5.6.	zui-app-calculation-service .....	67
3.6.	Сбор данных .....	67
3.6.1.	zif-interface-opcda .....	67
3.6.2.	zif-interface-device .....	67
3.6.3.	zif-interface-opchda .....	68
3.6.4.	zif-interface-opcua .....	68
3.6.5.	zif-data-emulator .....	69
3.6.6.	zif-datainput.....	69
3.6.7.	zif-events-integration .....	70
3.6.8.	zif-export-nifi-collectors.....	70
3.6.9.	zif-interface-manager .....	70
3.6.10.	zif-opcua-server.....	71
3.6.11.	zif-universal-datamart .....	71
3.6.12.	zui-app-datainput .....	71
3.6.13.	zui-app-interface-manager .....	72
3.6.14.	zui-app-nifi .....	72

3.6.15.	zui-app-universal-datamart.....	72
3.7.	Хранение данных .....	72
3.7.1.	zif-bp-* .....	72
3.7.2.	zif-document-archive .....	73
3.7.3.	zif-events .....	73
3.7.4.	zif-file-storage .....	74
3.7.5.	zif-om-sqldatasource.....	75
3.7.6.	zif-quality-service.....	76
3.7.7.	zif-rtdb-data, zif-rtdb-background-services .....	76
3.7.8.	zif-rtdb-metadata.....	77
3.7.9.	zif-workflow .....	79
3.7.10.	zif-workflow-relocator .....	79
3.7.11.	zui-app-workflow .....	80
3.7.12.	zui-app-event-types .....	80
3.7.13.	zui-app-rtdb .....	80
3.8.	UDL .....	81
3.8.1.	zif-udl-services .....	81
<b>4.</b>	<b>Поддержка встроенной (native) OpenID Connect аутентификации для Apache NiFi ..</b>	<b>83</b>
4.1.	Два типа аутентификации для NiFi .....	83
4.2.	Генерация сертификатов для узлов NiFi.....	83
4.3.	Управление пользователями в NiFi .....	84
<b>5.</b>	<b>Требования по проектированию приложений на платформе.....</b>	<b>85</b>
5.1.	Модальные глаголы .....	85
5.2.	Требования .....	85
5.2.1.	Моделирование .....	85
5.2.2.	Хранение данных .....	86
5.2.3.	Доступ к данным .....	86
5.2.4.	Рабочие процессы, события и уведомления .....	87
5.2.5.	Расчеты.....	87
5.2.6.	Инфраструктура .....	88
5.2.7.	Разработка, конфигурация и запуск сервисов .....	88
5.2.8.	Мониторинг и наблюдаемость .....	89
5.2.9.	Безопасность.....	90
5.2.10.	UI.....	91
5.2.11.	Представление архитектуры приложения .....	92
5.3.	Уровни требований .....	92
5.3.1.	Приложение, разрабатываемое под платформу ZIIoT .....	93
5.3.2.	Приложение, адаптированное (смигрированное) под платформу ZIIoT .....	93
5.3.3.	Приложение, интегрированное с платформой ZIIoT .....	94

**6. Контакты технической поддержки .....95**

# 1. Изменения системы авторазвертывания (zifctl и инфраструктуры)

Добавлено:

- Обновлена версия Keycloak, система развертывания устанавливает версию 21.1.1 по умолчанию. Версия 17 поддерживается в режиме обратной совместимости и будет удалена в следующем релизе. Для выбора версии добавлено поле `keycloak.version`. Подробнее см. раздел Изменения Apache Keycloak, начиная с релиза Платформы 2.16.0 в документе Руководство для системы развертывания (zifctl) с поставкой в виде docker-образа. [#28583](#)
- Темы из Keycloak версии 17 интегрированы в Keycloak версии 21. [#28102](#)
- Профили PostgreSQL - возможность подключения к мультикластерным инсталляциям PostgreSQL. Подробнее см. раздел Профили postgres (postgres profiles) или настройки postgres для мультикластерных PG-инсталляций" в документе Руководство для системы развертывания (zifctl)\_2.16.0. [#11978](#)
- Создание топиков в Kafka с указанными параметрами (replicas, partitions) и обновление параметров топиков. Управляются только "статические" топики, состав и названия которых известны заранее на этапе конфигурации системы. Также добавлена функциональность управления топиками Debezium. Параметры топиков определяются в файле `/services/kafka-topics-list.yaml`, по формату и структуре аналогичному `/services/service-list-patch.yaml`. Подробнее см. раздел Автоматическое создание и конфигурация топиков Kafka в процессе развертывания в документе Руководство для системы развертывания (zifctl)\_2.16.0. [#8865](#)
- Осуществлено обновление Apache Kafka на версию 3.5. [#24329](#)
- В Redis, поставляемый платформой, добавлена поддержка TLS, а также возможность включения аутентификации. Подробнее см. раздел Включение TLS, аутентификации и авторизации в Redis в документе Руководство для системы развертывания (zifctl)\_2.16.0. [#24946](#)
- Возможность внешнего подключения к Redis извне кластера. [#14909](#)
- Подключение к Redis для сервиса `zif-om-data-observer` и `zif-om-uom`. [#32510](#), [#32265](#)
- Сервисы `zif-om-data-observer` и `zif-opcua-server` включены в авторазвертывание. [#25194](#), [#30944](#)
- В helm-чарт Apache Nifi добавлена возможность хранения директории распакованных NAR архивов `/opt/nifi/nifi-current/work/nar` в персистентном хранилище вместо эфемерного, с целью уменьшения времени запуска Apache Nifi после перезапуска, а также решения проблемы недостаточного размера эфемерных хранилищ. [#29018](#)
- Расширение `uuid-osp` для PostgreSQL в сервисах `zif-material-lot`, `zif-sm-operationperformance` и `zif-sm-process`. [#30224](#), [#30584](#), [#30585](#)
- Возможность установки отдельного экземпляра Redis, используемый только инфраструктурными компонентами платформы. Дополнительный экземпляр Redis `zif-`

`redis-infra` при развертывании по умолчанию выключен, при его включении необходимо перенастроить связанные с ним инфраструктурные компоненты (например `Nifi`) на его использование вручную. Подробнее см. раздел Включение отдельного `Redis` для инфраструктуры и настройка `Apache Nifi` в документе Руководство для системы развертывания (`zifctl`)\_2.16.0. [#30282](#)

- Генерация документации по допустимым параметрам и их значениям конфигурации авторазвертывания `env-values.yaml`. Добавлена команда `zifctl doc env-values`. Подробнее см. раздел Генерация документа (`zifctl doc`) `env-values` - конфигурирования переменных в документе Руководство для системы развертывания (`zifctl`)\_2.16.0. [#22292](#)
- Подкоманда `zifctl debug reboot-udl` для перезагрузки `UDL` на стенде. [#18635](#)

#### Безопасность:

- Обновлены образы инфраструктурных образов с целью устранения уязвимостей, найденных в предыдущем релизе. [#31146](#)
- В логах `init`-заданий при установке платформы с любым уровнем логирования `init`-заданий (`--init-log-verbosity`) удалены/обфусцированы пароли от административных `УЗ` инфраструктурных сервисов и пароли/секреты пользователей/клиентов `Keycloak`, создаваемых платформой. В `zifctl` добавлен новый флаг `--show-init-log-keycloak-secrets`, при установке с которым в логах `init`-заданий будет выводиться отладочная информация о пользователях/клиентах (с паролями и секретами) `Keycloak`, создаваемых платформой. [#33360](#)
- В логах `init` контейнеров `startup-guardian` удалены/обфусцированы пароли от административных `УЗ` инфраструктурных сервисов и пароли/секреты пользователей/клиентов `Keycloak`, создаваемых Платформой. [#33512](#)
- Шифрование параметра `nifi.sensitive.props.key` в `Apache Nifi`. [#26836](#)
- Образ `opensearch` пересобран с использованием `alpine:3.18.3` в качестве базового. [#28711](#)
- Образ `opensearch-dashboards` пересобран с использованием `node:10.24.1-alpine3.11` в качестве базового. [#28712](#)
- Добавляется файл `.gitignore` в `env-path` при выполнении команды `zifctl init`. [#28686](#)

#### Изменено:

- Тип развертывания сервиса `zif-om-relationship` с `java` на `netcore`. [#32740](#)
- При использовании `cert-manager` для генерации сертификатов `Ingress` добавлена передача наименований хостов в виде массива в чарт `zif-infra-ingress`. По умолчанию помимо `.Values.externalBaseHostname` передаются `hostname`'ы для `s3` хранилища: `...-s3` и `...-s3-ui`. [#27727](#)
- Значениями ресурсов для `zookeeper` можно управлять независимо от ресурсов `Apache Kafka`. [#30828](#)
- Роли и политики `ABAC` для сервиса `zif-om-object`, для записи значения в свойстве должно использоваться действие `Publish`. [#32820](#)



- Сервис авто-тестирования `ziiot-tests` устанавливается как `deployment` вместо `cronjob`. [#31397](#)

Исправлено:

- Проблема периодического перезапуска Apache Cassandra в режиме высокой доступности (`infraHA=true`), увеличено время ожидания (`timeoutSeconds`) `readiness` и `liveness` проб. [#33218](#)
- Предупреждение в логге Keycloak об отсутствующей теме аккаунта `zif.v2`. [#32387](#)
- Некорректный тег по умолчанию `DEFAULT_IMAGE_TAG` `zifctl` при сборке образа. [#32336](#)
- Опечатка в описании клиента `rabbitmq-ui` в Keycloak. [#31592](#)
- Некорректное задание ресурсов для `init`-контейнера `zookeeper` в Apache Nifi. [#30785](#)
- Неработающая интеграция с Keycloak в `pgAdmin`. [#30736](#)
- Переменная окружения `AUDIT_ENABLED`. [#30672](#)
- Скорректированы значения ресурсов для `zif-rabbitmq` при настройке `infrastructureCapacity` равной `sm` и `m`. [#30372](#)
- Скорректированы значения ресурсов для `init`-контейнера `tls-external` в Apache Nifi. [#30373](#)
- Исправлено развертывание `deployment` `zif-minio`, выставляется переменная `spec.replicas`. [#29787](#)
- Для поддержки режима `Pod Security Standart = Restricted` исправлен `securityContext` во всех инфраструктурных и платформенных сервисах [#30642](#)
- Проблема с большим размером JWT токена в сервисах `kafka-ui` и `zif-workflow`, увеличен максимально допустимый размер заголовка и тела запроса. Настройкой можно управлять используя параметр `ingress.maxBodySize` в `env-values`. [#31174](#)
- Пути для Apache Nifi объединены в один `Ingress` для решения проблемы обращения контроллера к разным узлам кластера при развертывании более одной реплики. [#30382](#)

## 2. Версионирование компонентов Платформы для релиза 2.16.0

### 2.1. Совместимость системы развертывания Платформы с Kubernetes

С помощью инструментов авторазвертывания **Платформа** может быть установлена только на системы оркестрации контейнеров, а именно на **Kubernetes** или производные от него дистрибутивы **Red Hat OpenShift Container Platform (OpenShift/OKD)**.

**Таблица 2.1 Совместимость системы развертывания ZIIoT 2.16.0 с Kubernetes**

Дистрибутив	Назначение	Версия
Kubernetes	Системы оркестрации контейнеров, совместимые со стандартами CNCF.	1.21 и выше
OCP (OpenShift/OKD)		4.8 и выше

Системы оркестрации контейнеров должны содержать некоторые компоненты, без которых **Платформа** не может быть установлена. Перечисленные компоненты устанавливаются, как правило, в рамках всего кластера и поэтому не могут быть включены в состав **Платформы**.

**Таблица 2.2. Версии инфраструктурных компонентов в ZIIoT 2.16.0**

Компоне нт	Обязател ьность	Версия
StorageClass	+	Сущность для хранения параметров подключения к системе хранения данных.
Container Network Interface (CNI)	+	Контроллер внутренней сети кластера. Входит в состав любого дистрибутива Kubernetes. Должен поддерживать работу ~150 IP сервисов, входящих в состав Платформы. Конкретная реализация (OpenShift SDN, Flannel, Canico и т.д.) не регламентируется.
external LoadBalancer	+	Общий сетевой LoadBalancer для организации сетевого взаимодействия конечных пользователей сервисов, развернутых с помощью оркестратора Kubernetes. Входит в состав любого дистрибутива Kubernetes и должен быть корректно сконфигурирован и доступен.  Использование иных вариантов балансировки входящего трафика (например, установка внешнего Nginx на ноды кластера или использование NodePort) несовместимо с системой развертывания Платформы.
Ingress Controller	+	Контроллер входящего трафика для Kubernetes LoadBalancer. Для Kubernetes - только базовый <a href="#">ingress-nginx</a> , для OCP это стандартный Route Controller, входящий в состав дистрибутива OCP.  Для корректного заполнения IP адресов в сообщениях аудита, необходимо включить use-forwarded-headers.
cert-manager	-	Необязательный компонент Kubernetes, используется для генерации TLS-сертификатов: <a href="#">cert-manager.io</a> . В типовом случае используется для генерации TLS-сертификатов для Ingress. Платформа может быть установлена и без него, в этом случае необходимо сгенерировать и указать в параметрах конфигурации (env-values.yaml) TLS-сертификаты для объектов Ingress вручную.
cert-manager.io /v1 Issuer или Cluster Issuer	-	Certificate Issuer для cert-manager - управляет TLS-сертификатами, в том числе Ingress Controller. Как и <a href="#">cert-manager</a> , не является обязательным компонентом.

## 2.2. Совместимость Платформы с инфраструктурными компонентами

Данные компоненты могут либо устанавливаться вместе с Платформой в **Kubernetes/OCP**, либо использоваться в качестве внешних зависимостей, установленных в другом месте.

В связи с этим, в таблице совместимости перечислены только требования к версиям этих компонентов, без указания конкретной реализации.

**Примечание.** Совместимость с иными реализациями **S3 (Ceph, ...)** не протестирована. Точно известно, что с отличными от **MinIO** хранилищами несовместима автоматика развертывания (инициализация **bucket, access/secret key и policy**).

**Таблица 2.3. Версии инфраструктурных компонентов в ZIIoT 2.16.0**

Компонент	Назначение	Версия
PostgreSQL	<p>рСУБД (Реляционная система управления базами данных).</p> <p><b>Примечание.</b> Для работы сервисов Платформы необходима установка следующих extensions для PostgreSQL, не входящих в базовый пакет:</p> <ul style="list-style-type: none"> <li>▪ tablefunc;</li> <li>▪ uuid-osp;</li> <li>▪ pg_trgm;</li> <li>▪ btree_gist.</li> </ul> <p>Для большинства операционных систем для установки этих extensions достаточно установить пакет postgresql-contrib, в зависимостях у этого пакета - библиотеки от Perl.</p> <p>postgresql-contrib входит в состав базового образа PostgreSQL при его установке в Kubernetes с помощью Stolon. в случае использования внешнего PostgreSQL в закрытых контурах необходимо убедиться, что там всё это установлено</p>	12 и выше
Apache Cassandra	БД временных рядов	3.11
Redis	key-value БД (кэш)	6
Apache Kafka	Брокер сообщений	2.6
RabbitMQ	Брокер сообщений (используется в сервисе UDL).	3.11
Apache NiFi	Сервис управления потоками данных	1.19
KeyCloak	Сервис авторизации	17
S3	API объектного хранилища	MiniO 2022.8.8

## 2.3. Версии инфраструктурных и вспомогательных сервисов в составе инструмента по авторазвертыванию (zifctl)

Релиз платформы версии 2.16.0 поставляется с инструментом по автоматическому развертыванию Платформы.

Версия инструмента (**zifctl**) аналогична версии платформы — 2.16.0. С помощью инструмента автоматического развертывания развертываются сервисы платформы, вошедшие в релиз 2.16.0, а также инфраструктурные и вспомогательные сервисы из таблицы ниже:

**Таблица 2.4. Версии вспомогательных сервисов (zifctl) в ZIIoT 2.16.0**

Компонент	Реализация	Назначение	Версия	Образ	Лицензия
PostgreSQL	Zyfra	PostgreSQL + утилиты для создания отказоустойчивого кластера	12.14	zif-stolon-pg12:0.17.0-v4-230802	Apache License 2.0
	Zyfra	PostgreSQL + утилиты для создания отказоустойчивого кластера	14.8	zif-stolon-pg14:1.1.2-v3-230626	Apache License 2.0
pgAdmin4	pgAdmin	Графический клиент для PostgreSQL	7.1	zif-pgadmin4:7.5-v1-230802	The PostgreSQL Licence
Apache Cassandra	Bitnami + Zyfra	БД временных рядов	3.11.13	zif-cassandra:3.11.13-v6-230802	Apache License 2.0
Redis	Bitnami + Zyfra	key-value БД (кэш)	6.2.12	zif-redis:6.2.13-v1-230802	Apache License 2.0
Redis Exporter	Bitnami + Zyfra	Мониторинг Redis	1.50.0	zif-redis-exporter:1.50.0-v2-230529	Apache License 2.0
Apache Kafka	Confluent + Zyfra	Брокер сообщений	6.0.13	zif-kafka:6.0.13-v6-230802	Apache License 2.0
Kafka Exporter	Bitnami + Zyfra	Мониторинг Apache Kafka	1.7.0	1.7.0-v3-230802	Apache License 2.0
Apache Zookeeper	Confluent + Zyfra	Распределенное хранение конфигураций Apache Kafka и Apache NiFi	6.0.13	zif-zookeeper:6.0.13-v3-230529	Apache License 2.0
Kafka Rest	Confluent + Zyfra	REST Аpi для Apache Kafka	6.0.13	zfi-kafka-rest:6.0.13-v6-230802	Apache License 2.0
Kafka Schema Registry	Confluent + Zyfra	Схемы сообщений для топиков Apache Kafka	6.0.13	zif-schema-registry:6.0.13-v6-230802	Apache License 2.0
Kafka Connect	Confluent + Zyfra	Утилиты для интеграции различных источников данных с Apache Kafka. В Zyfra добавлена утилита jq и несколько коннекторов (например, Debezium)	6.0.13	zif-kafka-connect:6.0.13-v6-230802	Apache License 2.0
Debezium Connector	Red Hat	Сбор, сериализация и отправка в Apache Kafka логов транзакций PostgreSQL	1.7.1	см. Kafka Connect	Apache License 2.0
JDBC Connector	Confluent	Сбор, сериализация и отправка данных в Apache Kafka (как Source) и РСУБД (как Sink)	10.2.5	см. Kafka Connect	Apache License 2.0

JMX Exporter	Bitnami Zyfra +	Экспорт метрик JVM для сервисов экосистемы Kafka. В Zyfra на релиз по хэш-коммиту выставлен тег Stable	0.18.0	zif-jmx-exporter:0.18.0-v2-230529	Apache License 2.0
Kafka UI	Provectus Labs Zyfra +	UI для Apache Kafka, Kafka Connect, Kafka Schema Registry	0.4.0	zif-kafka-ui:0.7.1-v1-230802	Apache License 2.0
RabbitMQ	Bitnami Zyfra +	Брокер сообщений	3.11.16	zif-rabbitmq:3.11.20-v1-230802	Apache License 2.0
Apache NiFi	Apache Foundation + Zyfra	Сервис управления потоками данных. В Zyfra добавлен ряд процессоров для интеграции со сторонними источниками данных	1.19.1	zif-nifi:1.19.1-v9-230802	Apache License 2.0
KeyCloak	RedHat Zyfra +	Сервис авторизации. stolon В Zyfra добавлен корпоративный дизайн фронтенда и несколько плагинов	17.0.1	zif-keycloak:17.0.1-v11-230802	Apache License 2.0
KeyCloak RabbitMQ Event Listener	community	Экспорт KeyCloak Events в топика RabbitMQ	3.0.2	см. KeyCloak	Apache License 2.0
KeyCloak Metrics SPI	community	Экспорт метрик KeyCloak	2.5.3	см. KeyCloak	Apache License 2.0
OAuth2 Proxy	community	Прокси для авторизации в KeyCloak для Kafka UI, Apache NiFi	7.4.0	zif-oauth2-proxy:7.4.0-v6-230802	MIT
Alpine	Docker Inc + Zyfra	Набор утилит	3.18.0	zif-alpine-util:3.18.2-v2-230802	GPLv2
MinIO	Bitnami Zyfra +	S3-совместимое объектное хранилище	2022.8.8	zif-minio:2022.8.8-v6-230802	Apache License 2.0
Opensearch	community + Zyfra	Масштабируемая утилита полнотекстового поиска и аналитики	1.3.9	zif-opensearch:1.3.10-v2-230613	Apache License 2.0
Fluentd	Bitnami Zyfra +	Система сбора, парсинга, перенаправления и агрегации логов	1.14.6	zif-fluentd-opensearch:1.14.6-v7-230802	Apache License 2.0
Opensearch Dashboards	community + Zyfra	UI для Opensearch	1.3.9	opensearch-dashboards:1.3.10-v2-230613	Apache License 2.0

## 3. Изменения в работе отдельных сервисов

### 3.1. Администрирование

#### 3.1.1. zif-hierarchies

[2.2.0] - 2023-11-10

Добавлено:

- Обновление пакетов для метрик `http_server_duration` и `http_server_resource_waiting_duration`. [#27988](#)
- Поддержка TLS соединения с PostgreSQL. [#31050](#)

Изменено:

- **BREAKING:** переход на пакет `zbc-pkg-aspnet-configuration-postgres` для формирования строки подключения к БД. Переменные PG поддерживаемые пакетом описаны в `readme`. В `POSTGRES_TLS` изменен тип `string` на `bool`. Вместо нее `POSTGRES_TLS_MODE`. Все остальные переменные сервиса для PG поддерживаются в том же виде, плюс добавлены новые.
- Обновлен пакет с метриками до 7.1.6, добавлен `fallback` хэндлер в сервисы. [#32707](#)

Исправлено:

- Запросы с ошибками в ответе отображаются в контроллере как успешные. [#30763](#)

#### 3.1.2. zif-notifications

[4.9.1] - 2023-11-28

Добавлено:

- Утилита - тестовый клиент для работы с SignalR. [#33789](#)
- Поддержка TLS соединения с PostgreSQL. [#31050](#)
- Заполнение `AssemblyInfo` атрибуты при сборке сервиса. [#28906](#)
- Добавление метрик `http_server_duration` и `http_server_resource_waiting_duration` в сервис. [#27990](#)

Изменено:

- **BREAKING:** изменения в переменных для TLS в PostgreSQL. В `POSTGRES_TLS` изменен тип `string` на `bool` только для включения/выключения `tls`. Вместо нее для указания режима TLS используется `POSTGRES_TLS_MODE`.
- Обновить пакет с метриками до 7.1.5 добавить `fallback` хэндлер в сервисы. [#32705](#)

Исправлено:

- Ошибка с кодом 414 — URI Too Long при большом токене. KestrelServerOptions для решения проблемы через настройки. [#33789](#)
- Доработан API zif-notifications - убран пароль. [#31495](#)
- 503 ошибка сервиса при параметрах TLS True и POSTGRES\_TLS\_MODE Allow. [#32710](#)
- Бесконечная отправка запросов (фикс работы SignalR и Redis с включенным TLS). [#32219](#)
- Запросы со статус-кодами 4-- и 5-- в ответе не отображаются на графике Количество запросов в секунду. [#30890](#)
- Ошибка с кодом 500 на GET /sendingsources. [#33400](#)
- Не открывается контроллер метрик сервиса уведомлений. [#33480](#)

### 3.1.3. zif-portal-settings-dotnet

[2.8.0] - 2023-11-10

#### Добавлено:

- Обновление пакетов для метрик http\_server\_duration и http\_server\_resource\_waiting\_duration. [#27932](#)
- Заполнение AssemblyInfo атрибуты при сборке сервиса. [#28906](#)
- Поддержка TLS соединения с PostgreSQL. [#31050](#)

#### Изменено:

- **BREAKING:** изменен возвращаемый тип данных для настроек приложений и компонентов на JsonElement. Для эндпоинта with-rights добавлена структура данных результата.
- **BREAKING:** переход на пакет zbk-pkg-aspnet-configuration-postgres для формирования строки подключения к БД. Переменные PG поддерживаемые пакетом описаны в readme. В POSTGRES\_TLS изменен тип string на bool. Вместо нее POSTGRES\_TLS\_MODE. Все остальные переменные сервиса для PG поддерживаются в том же виде и добавленные новые.
- Обновление пакетов Microsoft для предотвращения дефектов ИБ. [#29045](#)
- Обновление метрик до 7.1.6, добавлен fallback хэндлер. [#32704](#)

#### Исправлено:

- Некорректный ответ на запрос POST /api/v1/components/settings. [#31450](#)
- Отсутствие описания типов данных ответов в OpenAPI. [#30090](#)
- Некорректный формат ответа на запрос PATCH /api/v1/applications/{id}/settings/user. [#30717](#)
- Ошибка валидации на уникальность поля baseHref при создании или редактировании приложения. [#32973](#)
- Ошибка валидации при создании приложений с одинаковым baseHref, используя пробелы. [#32959](#)

- В контроллере не отображаются запросы со статус-кодами 404, 409, 422. [#30780](#)

### 3.1.4. zif-security

[3.2.0] - 2023-11-10

Добавлено:

- **BREAKING:** целевое решение: большое количество ролей в токене `jwt`. [#27830](#)
- **BREAKING:** интеграция с `RabbitMQ` для получения информации об изменениях в `Keycloak`. [#28754](#)
- **BREAKING:** метод поиска и выгрузки метаданных профилей пользователей. [#30200](#)
- **BREAKING:** метрики для расчета SLO/SLA по методике `APDEX`. [#25653](#).
- **BREAKING:** переменная `POSTGRES_TLS_MODE` для установки типа шифрования. Переменная `POSTGRES_TLS` остается в рамках обратной совместимости. [#30946](#)
- **BREAKING:** добавлен пакет `automaxprocs` для ограничения количества потоков путем автоматического выбора `GOMAXPROCS`. До этого изменения `zif-security` выбирал размер тред пула по размеру ноды `Kubernetes`, а не по лимитам контейнера, что приводило к плохой производительности при большом количестве ядер на нодах. [#33561](#)
- Добавлена проверка `ruleId` при создании `PolicyRule`. [#32162](#)

Изменено:

- **BREAKING:** обратно включен `TLS`, если выбран `POSTGRES_TLS_MODE allow` или `prefer`. [#32963](#)
- В докер-файлах поменены образы, которые брались с `docker.io`, на образы из нашего репозитория. [#33250](#)

Исправлено:

- Сообщение сигнала завершения процесса на корректное. [#31464](#)
- При вводе валидной комбинации символов в поле `names` в эндпоинте `POST/users/search` случайно выдает юзера, где комбинация встречается. [#30601](#)
- Отсутствует валидация на пустое поле в эндпоинте `POST /users/search`. [#30524](#)
- Отключена проверка издателя по умолчанию. [#30281](#)
- Несоответствие `api-схемы`. [#27601](#)

### 3.1.5. zui-app-nds-configurator

[4.0.1] - 2023-11-10

Изменено:

- Осуществлен переход на `Angular 15`. [#28689](#)
- Убрать поле `пароль` в параметрах источника отправки уведомлений по email и SMS. [#31492](#)

Исправлено:



- Нефиксированная ширина полей в настройках собрания шаблона рассылки. [#31658](#)
- Хинты с ошибкой валидации не содержат значений. [#32956](#)
- Снимается выделение с типа события при выборе трех условий уведомления. [#29992](#)
- Ошибка `Invalid date` при загрузке Журнала отправки уведомлений. [#33284](#)
- Не работает пагинация в окне добавления спецссылок шаблона рассылки. [#33152](#)
- Неверный текст ошибки при выполнении создании/редактировании/удалении сущностей пользователем с правами `ReadOnly`. [#28599](#)
- В состоянии по умолчанию отображения панели редактирования групп рассылок поехала верстка заголовков. [#28857](#)
- Не отображается текст хинта предупреждения поля `Часовой пояс`. [#30069](#)
- Не отображаются сохранённые изменения в группах и шаблонах рассылки. [#32374](#)
- В обязательном поле `Локализация` настроек контакта отображается опция `Не выбрано`. [#30232](#)
- Не отображаются названия полей в настройках собрания шаблона рассылки. [#31656](#)
- При превышении допустимого количества символов не отображается ошибка валидации под полем. [#30227](#)
- Кнопка проигрывания звукового файла отображается в некорректном месте. [#30121](#)
- Поле ввода текста шаблона рассылки не выделяется красным при ошибке валидации. [#30114](#)
- Некорректное отображение полей ввода в настройках повторения шаблона рассылки. [#30193](#)
- При переходе в горизонтальный вид предпросмотра автоматически отображается тултип `Перейти в вертикальный вид` и не скрывается. [#30215](#)

### 3.1.6. zui-app-security-settings

[5.0.1] - 2023-11-10

Изменено:

- Осуществлен переход на `Angular 15`. [#28699](#)

Исправлено:

- Кнопка `На весь экран` не соответствует макету `Prizm`. [#32390](#)
- Иконки действий и скроллбар не соответствует гайдлайну `Prizm`. [#33179](#)
- Кнопки в тулбаре отображаются активными, хотя недоступны для нажатия. [#30639](#)
- Некорректное отображение элементов в тулбаре. [#29714](#)
- Кнопка открытия панели предпросмотра на странице `Политики` не соответствует гайдлайну `Prizm`. [#29905](#)

- Некорректное отображение поля `Фильтр по политике` при длинном названии политики. [#30290](#)
- Некорректное отображение элемента `Добавить условие` при создании правила. [#30291](#)
- Неверный тип курсора при наведении на поля с выпадающим списком. [#30292](#)
- В полях ввода параметров субъекта и объекта не отображается чекбокс, подтверждения введённого значения и не работает сохранение при клике на эту кнопку. [#30293](#)
- Неверный тип курсора при наведении на поле выбора количества отображаемых записей на странице. [#30804](#)

### 3.1.7.zui-app-shell

[4.0.0] - 2023-11-10

#### Добавлено:

- **BREAKING:** добавлена переменная `REST_ZIF_SECURITY_URL`, создан метод `getProfileSecuritySettings` предоставляющий данные пользователя.

#### Изменено:

- Обновлен `readme zui-app-shell`. [#32536](#)
- В `zui-app-shell` обновить `keycloak` до 21 версии. [#30527](#)
- Осуществлен переход на `Angular 15`. [#28320](#)

#### Исправлено:

- Некорректное значение сортировки уведомлений по умолчанию. [#32933](#)
- Любое действие в панели сортировки уведомлений закрывает панель уведомлений. [#33529](#)
- Плавающий баг со скроллингом при раскрытии разделов в иерархическом меню портала. [#32887](#)
- Ошибка с кодом 405 запроса `PUT /applications/specifications/`. [#33354](#)
- Приложения с длинным непрерывным названием некорректно отображаются на стартовых экранах. [#33277](#)
- Ошибка `single-spa` при создании нового приложения. [#32862](#)
- Некорректное отображение количества уведомлений в счётчике, если уведомлений более или равно 100. [#33050](#)
- При изменении условий сортировки в панели уведомлений каждый раз появляется уведомление о событии. [#32916](#)
- Некорректная вёрстка контейнеров в панели уведомлений при отображении уведомления с большим количеством символов в заголовке. [#32968](#)
- Отсутствие горизонтального прокручивания при разрешении экрана меньше 1280px. [#32712](#)
- Тултип `Скрыть список возможностей` не гаснет и закрывает часть верхней панели инструментов. [#32552](#)

- После редактирования `matchRoute` приложение QA01 Настройки безопасности не открывается до обновления страницы. [#28172](#)
- Ошибка UI в поле Описание при создании или редактировании приложения. [#32548](#)
- Не настроена валидация поля `base href`. [#32484](#)
- В шапке портала не отображаются роли пользователя. [#30884](#)
- Недопустимые значения в файле конфигурации `settings.json` отображаемые при входе на портал. [#30981](#)
- Некорректное отображение >3 изображений в карточке приложения на стартовом экране. [#30374](#)

## 3.2. Концептуальная OM

### 3.2.1. zif-om-datareferences

[3.3.0] - 2023-11-10

Исправлено:

- Политики `abac` корректно применяются при работе с данными сервиса. [#28446](#)

Изменено:

- Схема для источника данных `calctag`. [#30810](#), [#29280](#)

Добавлено:

- Поддержка TLS для `postgres`. [#30951](#)
- Обновление пакета с метриками, `fallback`-хендлер. [#32256](#)
- Обновлён пакет логирования. теперь в лог пишется тело ответа при статус кодах 400–599. [#32163](#)

### 3.2.2. zif-om-graphql

[3.10.1] - 2023-11-28

Удалено:

- **BREAKING:** удалены запросы и сущности связанные с хэштегами. [#22256](#)

Исправлено:

- Запросы на спецификации ресурсов в сервис определений операций корректно обрабатывает фильтр по классам. [#28676](#)
- Ошибка при создании сегмента операции, связанная с типом операции. [#28885](#)
- Ошибка в запросе при получении `WorkMaster`. [#27452](#)
- Получение объектов, у которых не проставлен идентификатор модели. [#30729](#)
- Поле `contentStrict` для страниц с обязательностью для элементов. [#31989](#)

#### Добавлено:

- Фильтрация дочерних объектов по ОПИ. [#31653](#)
- Добавлен фильтр по кодам прототипов моделей в запрос поиска объектов. [#29379](#)
- В метод поиска возможность поиска/фильтрации партий по атрибуту `description`. [#28664](#)
- Параметр `calculateTotalCount` позволяющий не подсчитывать общее количество элементов при запросе сущностей сервиса `zif-om-object`. [#29160](#)
- Расширен поиск ОПИ. [#19223](#)
- В методы поиска для представлений свойств (`propertyView`) добавлены фильтры по источникам данных и по идентификатору модели. [#28651](#)
- Получение сегмента процесса в запрос на получение сегмента операции. [#30206](#)
- Поддержка хранения шаблонов свойств классов у свойств спецификаций в сервисе `zif-sm-process`. [#30280](#)
- Групповое удаление ОПИ. [#30329](#)
- Запрос на получение связей для класса. [#30783](#)
- Мутации для копирования сегментов операций. [#30576](#)

#### Изменено:

- **BREAKING:** методы получения данных с группировкой по ОПИ отмечены устаревшими в сервисе `zif-om-graphql`. [#32923](#)
- Метод получения определения операций не поддерживает группировку по ОПИ. [#29636](#)
- Обновлены контракты для работы с сервисом `zif-sm-directories`. [#29323](#)
- Методы группировки ОПИ помечены как устаревшие и будут удалены в версии платформы 2.17.0. [#32926](#)
- `Zyfra.AspNetCore.Configuration.Redis 7.2.2`. [#34411](#)

### 3.2.3. zif-om-object

[3.6.1] - 2023-11-28

#### Изменено:

- **BREAKING:** полностью удалена работа с хэштегами. [#22256](#)
- **BREAKING:** запланированный к введению в текущей версии запрет на создание конфигураций свойств с типом `формула`, в которых есть ссылки на свойства других моделей, операций или процессов был перенесён в версию 2.17.0 и изменен: произведено исправление валидации примитивов свойств в рамках задачи [#26914](#).
- В конфигурации свойства константа разрешено пустое значение, как и в шаблонах. [#29228](#)
- Убран роут `properties/version`. [#29522](#)
- Расширен метод создания/редактирования связи доп атрибутами связи. [#27324](#)

- Убрано создание конфигурации свойства при установке/смене класса объекта. [#30317](#)
- `Zyfra.AspNet.Configuration.Redis 7.2.2`. [#34411](#)

#### Добавлено:

- Добавлены методы тестирования данных для реализации возможности запуска проверки консистентности системы. [#24088](#), [#25173](#)
- Массовое создание или обновление `ClassesLink`. [#26639](#)
- Валидации для шаблонов свойств и прототипов моделей. [#25242](#)
- Поиск `propertyView` по `datareferenceId` и `modelId`. [#28651](#)
- Метод для поиска `ClassesLinks`. [#27323](#)
- Параметры поиска для `Links`. [#27325](#)
- Проверки полей `datatypeId` и `valuetypeId` при создании/изменении шаблона свойства. [#28546](#)
- Обязательность полей `CurrentId` и `NewId` для копирования. [#27541](#)
- Валидация при копировании - `CurrentId` и `NewId` не должны быть одинаковыми. [#27542](#)
- Смена класса у объекта. [#27880](#)
- Проверка кода `property primitive`. [#29170](#)
- Распределённая блокировка при создании и обновлении сущностей сервиса. [#30968](#)
- Создание связей классов из разных групп одновременно. [#31728](#)

#### Исправлено:

- Наследование ОПИ для свойств от примитивов свойств. [#32493](#)
- Оптимизирован запрос валидации наличия свойства при редактировании. [#30554](#)
- Проверка наличия объекта на указанной в `url` модели. [#30876](#)
- `bulk`-создание линков с несколькими типами отношений. [#30794](#)
- Код ответа для `GET /properties/{id}/properties` АВАС. [#29523](#)
- Замена идентификатора в конфигурациях свойств в рамках объекта и всех его дочерних. [#28494](#)
- Работа АВАС в методах представлений свойств (`PropertyView`). [#29222](#)
- Поиск `PropertyPrimitive` по несуществующему `classId`. [#28742](#)
- Получение данных о правах на свойство в контроллере `PropertyRights` с применением АВАС. [#28929](#)

## 3.2.4. zif-om-objectmodel2excel

[1.13.1] - 2023-11-28

Удалено:

- **BREAKING:** удалена работа с хэштегам. [#22256](#)

Добавлено:

- Валидация при смене единицы измерения у свойства в другой класс. [#27941](#), [#28109](#)
- Экспорт свойств с конфигурацией из `zif-om-object` на которое указывает свойство элемента спецификации. [#28240](#)
- Поле `EntityId` в свойствах сегментов операций. [#28240](#)
- Валидация на изменение объектов в свойствах. [#27945](#)
- Смена класса у объекта. [#27880](#)
- Поле `propertyPrimitiveClassId`. [#30285](#)
- Поддержка TLS для `postgres`. [#30951](#)

Исправлено:

- Исправление валидации `entity path` для свойства спецификации. [#29569](#)
- Поиск свойств объекты которых находятся в других сервисах. [#28003](#)
- Синхронизация `SignalR` через `MassTransit`. [#29131](#)

Изменено:

- Обновлён пакет с метриками. [#32256](#)
- Обновление пакета логирования для вывода в лог тела ответа при статусе кода 400-599. [#32163](#)
- В журнале импорта выводится полное сообщение сервиса в случае получения кода 400-500. [#33362](#)
- `Zyfra.AspNetCore.Configuration.Redis 7.2.2`. [#34411](#)

## 3.2.5. zif-om-relationship

[3.3.0] - 2023-11-10

Изменено:

- Поддержка TLS для работы с `Postgres`. [#30951](#)
- Обновлены пакеты инфраструктуры до последних версий. [#27732](#)
- Обновление пакета логирования для вывода в лог тела ответа при статусе кода 400-599. [#32163](#)

## 3.2.6. zif-om-uom

[3.6.1] - 2023-11-28

### Добавлено:

- **BREAKING:** в релизе Платформы 2.18.0 в сервисе `zif-om-uom` сущностям `Uom` и ``ClassUom` будет добавлено поле `Code`. Оно будет предварительно заполнено и будет обязательным при обновлении\создании. Также на это поле будет наложено ограничение по уникальности.
- Запрет на удаление канонической ЕИ. [#29276](#)
- Поддержка TLS, переход на CRM. [#30951](#)
- Блокировка при записи и обновлении ЕИ и классов. [#31016](#)
- Обновление пакета логирования для вывода в лог тела ответа при статусе кода 400-599. [#32163](#)

### Исправлено:

- ЕИ Неопределено и Неопределенно переименовано в Не определено. [#28774](#)
- Валидация `canonical factor` - не равен нулю. [#29189](#)

### Изменено:

- `Zyfra.AspNet.Configuration.Redis 7.2.2`. [#34411](#)

## 3.2.7. zui-app-om

[2.28.0] - 2023-11-10

### Исправлено:

- Ошибка UI в OM. [#28145](#)
- В OM у свойств поиск тэгов осуществляется без учета параметра `Строк` на странице. [#29789](#)
- Недоступен просмотр исторических данных OM. [#31555](#)
- OM - Классы объектов: при копировании класса с шаблоном свойства формула. [#32008](#)

### Добавлено:

- Функциональность группировки по ОПИ в разделах Ресурсы и Шаблоны свойств будет удален в версии платформы 2.17.0. [#32756](#)
- Связи между объектами. Атрибутивный состав. [#24336](#)
- Смена класса объекта ресурсной модели. [#25121](#)
- Множественные связи между классами. [#20862](#)

## 3.3. Прикладная ОМ

### 3.3.1. Общие критические изменения модуля

Изменено:

- **BREAKING:** начиная с версии 2.18.0 в сервисах прикладной объектной модели начнет внедряться пакет по работе с DataVault версии 8. Это приведет к удалению параметра фильтрации `isPaged`. Таким образом, все поиски автоматически станут "пейджированными". Корректная работа режима `isPaged = false` более не гарантируется.
- **BREAKING:** начиная с версии 2.18.0 будет осуществлен переход на новую модель «пейджированного» ответа в сервисах прикладной объектной модели:

```
{  "content": [],
  "pageNumber": 0, "pageSize": 0,
  "numberOfElements": 0, "sort": {
    "orders": [      {
      "direction": "Asc",      "property": "string"
    }    ]
  }, "totalPages": 0,
  "totalElements": 0,}
```

### 3.3.2. zif-material-lot

[2.5.0] - 2023-11-06

Добавлено:

- **BREAKING:** в 2.18.0 из модели `MaterialLotSearchParameter` будет удалена возможность искать по одному `MaterialDefinitionId`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`. Следует использовать параметр `MaterialDefinitionIds`.
- Миграция и включение проверки уникальности в `zif-material-lot` и `zif-sm-directories`. [#28120](#)
- Доработка метода `find` - поиск по `description`. [#25003](#)
- Реализован поиск по массиву `MaterialDefinitionId`. [#23001](#)

Изменено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT-эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#33258](#)
  - PUT `/materiallots/{id}`;
  - PUT `/materiallotlinks/{id}`;



- PUT /statuses/{id}.
- **BREAKING:** в 2.17.0 в сервисе при поискам по строковым полям в качестве WildCards нужно будет использовать \* и ?, WildCards % и \_ более не поддерживаются.
- Обновлены библиотеки по работе с Redis. [#34411](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Значение по умолчанию для переменной CASSANDRA\_ENABLED теперь равно true. [#30512](#)
- Возвращается статус 500 при вызове POST /properties. [#30696](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Не создается партия с пустым необязательным полем код. [#31688](#)
- Эндпоинты для valueProperty завершаются со статусом 500. [#32118](#)
- Обновлен пакет с метриками. [#32257](#)
- В методе /materiallots/find при задании атрибута includeStatusIds сервис возвращает 500. [#32966](#)
- Заполнить комментарии к БД. [#32353](#)

### 3.3.3. zif-rdm-common

[3.1.0] - 2023-11-06

#### Добавлено:

- Улучшена поддержка АВАС в эндпоинтах сервиса. [#28761](#)
- Добавлены новые эндпоинты с поддержкой пейджирования. [#29682](#)
  - GET /directories/{directoryId}/referencespage;
  - GET /directories/favoritespage;
  - GET /directories/{directoryId}/fieldspage;
  - GET /directories/{directoryId}/fields/{fieldId}/referencespage;
  - GET /directories/{directoryId}/fields/referencespage.

#### Исправлено:

- Пользователь с правами на чтение (rdm.viewer) не может добавить справочник в избранное. [#31782](#)
- Отсутствует разметка обязательных полей для методов POST и PUT. [#30821](#)
- Исправлена ошибка, из-за которой в панели Избранные у заблокированных справочников не отображается пиктограмма замка. [#31395](#)
- Исправлена ошибка, из-за которой настройки валидации типа поля текст справочника работают некорректно. [#32081](#)

- Обновлен пакет с метриками. [#32257](#)
- Исправлена проблема, из-за которой нельзя было создать справочник со значением `Fields` равным `null`. [#31082](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)

Удалено:

- **BREAKING:** список устаревших эндпоинтов, которые будут удалены в релизе версии 2.18:
  - GET `/directories/{directoryId}/references` используйте GET `/directories/{directoryId}/referencespage`;
  - GET `/directories/favorites` используйте GET `/directories/favoritespage`;
  - GET `/directories/{directoryId}/fields` используйте GET `/directories/{directoryId}/fieldspage`;
  - GET `/directories/{directoryId}/fields/{fieldId}/references` используйте GET `/directories/{directoryId}/fields/{fieldId}/referencespage`;
  - GET `/directories/{directoryId}/fields/references` используйте GET `/directories/{directoryId}/fields/referencespage`;
  - GET `/directories/count`.

### 3.3.4. zif-sm-directories

[1.10.0] - 2023-11-06

Добавлено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31004](#)
  - PUT `/discreteness/{id}`;
  - PUT `/hierarchyscopes/{id}`;
  - PUT `/models/{id}`;
  - PUT `/priorities/{id}`;
  - PUT `/resourceusetypes/{id}`;
  - PUT `/worktypes/{id}`.
- Миграция и включение проверки уникальности в `zif-material-lot` и `zif-sm-directories`. [#28120](#)
- Расширение поисковых возможностей (контекстный поиск, рекурсия). [#19223](#)

Исправлено:

- В запросах поиска `hierarchyscopes` если указать фильтр `"description": ""` - не будут возвращаться ОПИ у которых `description null`. [#29630](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- При `get` запросе всех `resourceusetypes` присутствует `Body`. [#30732](#)
- Нарушена иерархия ОПИ. [#30752](#)
- Обновлен пакет с метриками. [#32257](#)

#### Удалено:

- **BREAKING:** POST `/models` помечен устаревшим и будет удален в 2.18.0, используйте PUT `/models/{id}`;
- **BREAKING:** POST `/priorities` помечен устаревшим и будет удален в 2.18.0, используйте PUT `/priorities/{id}`;
- **BREAKING:** POST `/resourceusetypes` помечен устаревшим и будет удален в 2.18.0, используйте PUT `/resourceusetypes/{id}`;
- **BREAKING:** POST `/worktypes` помечен устаревшим и будет удален в 2.18.0, используйте PUT `/worktypes/{id}`.
- АПИ для работы с хештегами. [#30259](#)

### 3.3.5. zif-sm-domain-api

[1.3.0] - 2023-11-06

#### Добавлено:

- `SegmentDependency` в запросах факта Domain API. [#27944](#)

#### Изменено:

- При поиске `equipmentRequirement` поиск осуществляется по `Object`, а не по `equipmentRequirement`. [#30276](#)
- При поиске `operationSchedule` нужно не учитывать фильтр по `hierarchyScopes` если он пустой. [#30278](#)
- В поиске `/findOperationPerformanceData` фильтр `Id` у `segmentResponse` всегда не находит `segmentResponse`. [#29757](#)
- В поиске `/findOperationPerformanceData` не учитывается фильтр `segmentRequirement` у `segmentResponse`. [#29754](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- DomainApi запрос `/findOperationPerformanceData` не выводит `discretenessId` для `operationResponse`. [#31597](#)
- Обновлен пакет с метриками. [#32257](#)

## 3.3.6. zif-sm-operationdefinition

[1.17.0] - 2023-11-06

### Добавлено:

- Унификация подхода по уникальности использования поля код для сущности DependencyType. [#27857](#)
- Перемещение сегментов операции. [#15400](#)

### Изменено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31262](#)
  - PUT /equipmentspecificationproperties/{id};
  - PUT /materialspecificationproperties/{id};
  - PUT /personnelspecificationproperties/{id};
  - PUT /physicalassetspecificationproperties/{id};
  - PUT /equipmentspecifications/{id};
  - PUT /materialspecifications/{id};
  - PUT /personnelspecifications/{id};
  - PUT /physicalassetspecifications/{id};
  - PUT /dependencytypes/{id};
  - PUT /links/{id};
  - PUT /operationdefinitions/{id};
  - PUT /operationmaterialbills/{id};
  - PUT /operationmaterialbillitems/{id};
  - PUT /operationsegments/{id};
  - PUT /parameterspecifications/{id}.
- **BREAKING:** в 2.17.0 в сервисе при поискам по строковым полям в качестве WildCards нужно будет использовать \* и ?, вместо % и \_.
- Обновлены библиотеки по работе с Redis. [#34411](#)
- Атрибуты valueTypeId и dataTypeId являются обязательными для эндпоинта POST /equipmentspecificationproperties, но в swagger не помечены обязательными. [#31047](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)

- Пользователь без прав доступа получает контент при поиске. [#31278](#)
- При удалении сегмента операции не удаляются Параметры. [#31823](#)
- Обновлен пакет с метриками. [#32257](#)
- Возвращается статус 500 при невозможности создания свойства при вызове /createbyprimitive. [#32448](#)

Удалено:

- **BREAKING:** метод POST /equipmentspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /equipmentspecificationproperties/{id}.
- **BREAKING:** метод POST /materialspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /materialspecificationproperties/{id}.
- **BREAKING:** метод POST /personnelspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /personnelspecificationproperties/{id}.
- **BREAKING:** метод POST /physicalassetspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /physicalassetspecificationproperties/{id}.
- **BREAKING:** метод POST /equipmentspecifications устарел и будет удален в 2.18.0, используйте вместо него PUT /equipmentspecifications/{id}.
- **BREAKING:** метод POST /materialspecifications устарел и будет удален в 2.18.0, используйте вместо него PUT /materialspecifications/{id}.
- **BREAKING:** метод POST /personnelspecifications устарел и будет удален в 2.18.0, используйте вместо него PUT /personnelspecifications/{id}.
- **BREAKING:** метод POST /physicalassetspecifications устарел и будет удален в 2.18.0, используйте вместо него PUT /physicalassetspecifications/{id}.
- **BREAKING:** метод POST /dependencytypes устарел и будет удален в 2.18.0, используйте вместо него PUT /dependencytypes/{id}.
- **BREAKING:** метод POST /links устарел и будет удален в 2.18.0, используйте вместо него PUT /links/{id}.
- **BREAKING:** метод POST /operationdefinitions устарел и будет удален в 2.18.0, используйте вместо него PUT /operationdefinitions/{id}.
- **BREAKING:** метод POST /operationmaterialbills устарел и будет удален в 2.18.0, используйте вместо него PUT /operationmaterialbills/{id}.
- **BREAKING:** метод POST /operationmaterialbillitems устарел и будет удален в 2.18.0, используйте вместо него PUT /operationmaterialbillitems/{id}.
- **BREAKING:** метод POST /operationsegments устарел и будет удален в 2.18.0, используйте вместо него PUT /operationsegments/{id}.
- **BREAKING:** метод POST /operationsegmentlinks устарел и будет удален в 2.18.0, используйте вместо него PUT /operationsegments/{id}.
- **BREAKING:** метод POST /parameterspecifications устарел и будет удален в 2.18.0, используйте вместо него PUT /parameterspecifications/{id}.

- **BREAKING:** метод GET  
equipmentspecificationproperties/{id}/equipmentspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /equipmentspecificationproperties/equipmentspecificationproperties с указанием параметра specificationId равным null и parentId равным {id} из path.
- **BREAKING:** метод GET  
materialspecificationproperties/{id}/materialspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /materialspecificationproperties/materialspecificationproperties с указанием параметра specificationId равным null и parentId равным {id} из path.
- **BREAKING:** метод GET  
personnelspecificationproperties/{id}/personnelspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /personnelspecificationproperties/personnelspecificationproperties с указанием параметра specificationId равным null и parentId равным {id} из path.
- **BREAKING:** метод GET  
physicalassetspecificationproperties/{id}/physicalassetspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetspecificationproperties/physicalassetspecificationproperties с указанием параметра specificationId равным null и parentId равным {id} из path.
- **BREAKING:** метод GET  
equipmentspecifications/{id}/equipmentspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /equipmentspecificationproperties/equipmentspecificationproperties с указанием параметра specificationId равным {id} из path и parentId равным null.
- **BREAKING:** метод GET  
materialspecifications/{id}/materialspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST /materialspecificationproperties/materialspecificationproperties с указанием параметра specificationId равным {id} из path и parentId равным null.

```
{  "content": [],
  "pageNumber": 0, "pageSize": 0,
  "numberOfElements": 0, "sort": {
    "orders": [    {
      "direction": "Asc",      "property": "string"
    }  ]
}, "totalPages": 0,
"totalElements": 0,}
```

- **BREAKING:** метод GET  
personnelspecifications/{id}/personnelspecificationproperties устарел и будет удален в 2.18.0, используйте вместо него POST

/personnelspecificationproperties/personnelspecificationproperties C  
указанием параметра specificationId равным {id} из path и parentId равным null.

- **BREAKING:** метод GET  
physicalassetspecifications/{id}/physicalassetspecificationproperties  
устарел и будет удален в 2.18.0, используйте вместо него POST  
/physicalassetspecificationproperties/physicalassetspecificationpropert  
ies с указанием параметра specificationId равным {id} из path и parentId равным  
null.
- **BREAKING:** свойство EquipmentPropertyId модели  
EquipmentSpecificationProperty устарело и будет удалено в 2.18.0, используйте  
вместо него EntityId со значением EntityType равным Property, так же этот параметр  
будет удален из запросов в GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство EquipmentClassPropertyId модели  
EquipmentSpecificationProperty устарело и будет удалено в 2.18.0, используйте  
вместо него EntityId со значением EntityType равным ClassProperty, так же этот  
параметр будет удален из запросов в GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство PropertyId модели EquipmentSpecificationProperty устарело  
и будет удалено в 2.18.0, используйте вместо него EntityId со значением EntityType  
равным Property, так же этот параметр будет удален из запросов в GraphQL в сервисе  
zif-om-graphql.
- **BREAKING:** свойство ClassPropertyId модели EquipmentSpecificationProperty  
устарело и будет удалено в 2.18.0, используйте вместо него EntityId со значением  
EntityType равным ClassProperty, так же этот параметр будет удален из запросов в  
GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство EquipmentSpecificationId модели  
EquipmentSpecificationProperty устарело и будет удалено в 2.18.0, используйте  
вместо него ResourceSpecificationId, так же этот параметр будет удален из запросов в  
GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство MaterialDefinitionPropertyId модели  
MaterialSpecificationProperty устарело и будет удалено в 2.18.0, используйте вместо  
него EntityId со значением EntityType равным Property, так же этот параметр будет  
удален из запросов в GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство MaterialClassPropertyId модели  
MaterialSpecificationProperty устарело и будет удалено в 2.18.0, используйте вместо  
него EntityId со значением EntityType равным ClassProperty, так же этот параметр  
будет удален из запросов в GraphQL в сервисе zif-om-graphql.
- **BREAKING:** свойство PropertyId модели MaterialSpecificationProperty устарело и  
будет удалено в 2.18.0, используйте вместо него EntityId со значением EntityType  
равным Property, так же этот параметр будет удален из запросов в GraphQL в сервисе  
zif-om-graphql.
- **BREAKING:** свойство ClassPropertyId модели MaterialSpecificationProperty  
устарело и будет удалено в 2.18.0, используйте вместо него EntityId со значением  
EntityType равным ClassProperty, так же этот параметр будет удален из запросов в  
GraphQL в сервисе zif-om-graphql.

- **BREAKING:** свойство `MaterialSpecificationId` модели `MaterialSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `ResourceSpecificationId`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PersonPropertyId` модели `PersonnelSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `Property`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PersonnelClassPropertyId` модели `PersonnelSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `ClassProperty`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PropertyId` модели `PersonnelSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `Property`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `ClassPropertyId` модели `PersonnelSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `ClassProperty`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PersonnelSpecificationId` модели `PersonnelSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `ResourceSpecificationId`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PhysicalAssetPropertyId` модели `PhysicalAssetSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `Property`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PhysicalAssetClassPropertyId` модели `PhysicalAssetSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `ClassProperty`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PropertyId` модели `PhysicalAssetSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `Property`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `ClassPropertyId` модели `PhysicalAssetSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `EntityId` со значением `EntityType` равным `ClassProperty`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** свойство `PhysicalAssetSpecificationId` модели `PhysicalAssetSpecificationProperty` устарело и будет удалено в 2.18.0, используйте вместо него `ResourceSpecificationId`, так же этот параметр будет удален из запросов в GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** в 2.18.0 будет произведена унификация работы с ОПИ:



- во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет удалено, используйте существующие поле `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`.
- поле в БД `hierarchyScopeId` будет удалено и автоматически добавлено в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равно `null` переноситься не будет.

### 3.3.7. zif-sm-operationperformance

[1.11.0] - 2023-11-28

#### Добавлено:

- `SegmentDependency` в запросах факта `Operation Performance`. [#25860](#)
- Унификация подхода по уникальности использования поля код для сущности `DependencyType`. [#27857](#)
- Сделать проверку корректности времени начала и окончания операций. [#16354](#)

#### Исправлено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31262](#)
  - `PUT /operationperformancesagg;`
  - `PUT /equipmentactualproperties/{id};`
  - `PUT /propertymeasurements/{id};`
  - `PUT /equipmentactuals/{id};`
  - `PUT /links/{id};`
  - `PUT /materialactualproperties/{id};`
  - `PUT /materialactuals/{id};`
  - `PUT /operationperformances/{id};`
  - `PUT /personnelactualproperties/{id};`
  - `PUT /personnelactuals/{id};`
  - `PUT /physicalassetactualproperties/{id};`
  - `PUT /physicalassetactuals/{id};`
  - `PUT /segmentdata/{id};`
  - `PUT /segmentresponsedependencies/{id};`
  - `PUT /segmentresponses/{id}.`
- Обновлены библиотеки по работе с Redis. [#34411](#)
- При создании `OperationPerformance` вместо ожидаемого 400 статус 500. [#30175](#)

- Статус 200 при ответе на PUT /personnelactualproperties и PUT /segmentresponsedependencies вместо 204. [#30698](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Возвращается статус 500 при вызове POST /properties. [#30696](#)
- Метод POST /materialactuals/find отдаёт контент пользователю без прав доступа к сервису. [#30982](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Эндпоинты для valueProperty завершаются со статусом 500. [#32118](#)
- Обновлен пакет с метриками. [#32257](#)

Удалено:

- **BREAKING:** метод POST/dependencytypes устарел и будет удален в 2.18.0, используйте вместо него PUT /dependencytypes/{id}.
- **BREAKING:** метод POST/equipmentactualproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /equipmentactualproperties/{id}.
- **BREAKING:** метод POST/equipmentactuals устарел и будет удален в 2.18.0, используйте вместо него PUT /equipmentactuals/{id}.
- **BREAKING:** метод POST/links устарел и будет удален в 2.18.0, используйте вместо него PUT /links/{id}.
- **BREAKING:** метод GET/links устарел и будет удален в 2.18.0, используйте вместо него POST/links/find.
- **BREAKING:** метод POST/materialactualproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /materialactualproperties/{id}.
- **BREAKING:** метод POST/materialactuals устарел и будет удален в 2.18.0, используйте вместо него PUT /materialactuals/{id}.
- **BREAKING:** метод POST/operationperformances устарел и будет удален в 2.18.0, используйте вместо него PUT /operationperformances/{id}.
- **BREAKING:** метод POST/personnelactualproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /personnelactualproperties/{id}.
- **BREAKING:** метод POST/personnelactuals устарел и будет удален в 2.18.0, используйте вместо него PUT /personnelactuals/{id}.
- **BREAKING:** метод POST/physicalassetactualproperties устарел и будет удален в 2.18.0, используйте вместо него PUT /physicalassetactualproperties/{id}.
- **BREAKING:** метод POST/physicalassetactuals устарел и будет удален в 2.18.0, используйте вместо него PUT /physicalassetactuals/{id}.
- **BREAKING:** метод POST/segmentdata устарел и будет удален в 2.18.0, используйте вместо него PUT /segmentdata/{id}.

- **BREAKING:** метод `POST/segmentresponsedependencies` устарел и будет удален в 2.18.0, используйте вместо него `PUT /segmentresponsedependencies/{id}`.
- **BREAKING:** метод `POST/segmentresponses` устарел и будет удален в 2.18.0, используйте вместо него `PUT /segmentresponses/{id}`.
- **BREAKING:** параметр фильтрации `hierarchyScopesIds` будет переименован в `hierarchyScopeIds`. Так же этот параметр будет переименован в запросах GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** фильтрация по параметрам `operationSegmentId`, `operationDefinitionId`, `hierarchyScopeId` будет удалена, используйте параметры `operationSegmentIds`, `operationDefinitionIds`, `hierarchyScopeIds`. Аналогичные замены произойдут в запросах GraphQL в сервисе `zif-om-graphql`.
- **BREAKING:** в 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет удалено, используйте существующие поле `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически добавлено в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равное `null` переноситься не будет.
- **BREAKING:** в 2.18.0 будет произведено удаление возможности поиска данных по параметру `TS`:
  - во всех методах поиска будет удален параметр `ts`, как из `get` так и `post` запросов.
  - хранение истории изменений будет удалено вместе с `DataVault`. Существующая схема БД будет автоматически мигрирована в новый формат.

### 3.3.8. zif-sm-operationschedule

[1.12.0] - 2023-11-06

Добавлено:

- Сделать проверку корректности времени начала и окончания операций. [#16354](#)

Изменено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#33258](#)
  - PUT `/equipmentrequirements/{id}/equipmentrequirementproperties/{prpid}`;
  - PUT `/materialrequirements/{id}/materialrequirementproperties/{prpid}`;
  - PUT `/personnelrequirements/{id}/personnelrequirementproperties/{prpid}`;

- PUT `/physicalAssetrequirements/{id}/physicalAssetrequirementproperties/{prpid}`;
  - PUT `/links/{id}`;
  - PUT `/operationsrequests/{id}`;
  - PUT `/operationsrequests/{id}/segmentresponses/{rid}`;
  - PUT `/operationsscheduleaggs/{id}`;
  - PUT `/operationsschedules/{id}`;
  - PUT `/segmentrequirements/{id}`;
  - PUT `/segmentrequirements/{id}/segmentparameters/{pid}`.
- Обновлены библиотеки по работе с ABAC. [#29718](#)
  - Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
  - Запрос на удаление свойств физических активов всегда возвращает статус 403. [#31778](#)
  - Запрос на создание сущности `segmentResponse` всегда отвечает 500 при включенном ABAC. [#31799](#)
  - Обновлен пакет с метриками. [#32257](#)

Удалено:

- **BREAKING:** метод `POST/materialrequirements` устарел и будет удален в 2.18.0, используйте вместо него `PUT /materialrequirements/{id}`.
- **BREAKING:** метод `POST/equipmentrequirements` устарел и будет удален в 2.18.0, используйте вместо него `PUT /equipmentrequirements/{id}`.
- **BREAKING:** метод `POST/links` устарел и будет удален в 2.18.0, используйте вместо него `PUT /links/{id}`.
- **BREAKING:** метод `POST/operationsrequests` устарел и будет удален в 2.18.0, используйте вместо него `PUT /operationsrequests/{id}`.
- **BREAKING:** метод `POST/operationsschedules` устарел и будет удален в 2.18.0, используйте вместо него `PUT /operationsschedules/{id}`.
- **BREAKING:** метод `POST/personnelrequirements` устарел и будет удален в 2.18.0, используйте вместо него `PUT /personnelrequirements/{id}`.
- **BREAKING:** метод `POST/physicalAssetrequirements` устарел и будет удален в 2.18.0, используйте вместо него `PUT /physicalAssetrequirements/{id}`.
- **BREAKING:** метод `POST/segmentrequirements` устарел и будет удален в 2.18.0, используйте вместо него `PUT /segmentrequirements/{id}`.
- **BREAKING:** фильтрация по параметрам `name`, `description`, `resourceRequirementId`, `hierarchyScopeId` будет удалена, используйте параметры `names`, `descriptions`,

resourceRequirementIds, hierarchyScopeIds. Аналогичные замены произойдут в запросах GraphQL в сервисе zif-om-graphql.

- **BREAKING:** параметр фильтрации requestState будет переименован в requestStates в модели поиска OperationsRequestsSearchRequest. Аналогичные замены произойдут в запросах GraphQL в сервисе zif-om-graphql.
- **BREAKING:** в 2.18.0 будет произведена унификация работы с ОПИ:
  - во всех CRUD методах свойство и параметр фильтрации hierarchyScopeId будет удалено, используйте существующие поле hierarchyScopeIds. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе zif-om-graphql
  - поле в БД hierarchyScopeId будет удалено и автоматически добавлено в поле hierarchyScopeIds, значение hierarchyScopeId равно null переноситься не будет.

### 3.3.9. zif-sm-process

[1.10.0] - 2023-11-28

#### Добавлено:

- Унификация подхода по уникальности использования поля код для сущности DependencyType. [#27857](#)
- Добавление спецификаций ресурсов Процессов 2.0 на Классы 2.0. [#26065](#)
- Поддержка хранения в БД дробных значений свыше 100 000. [#8763](#)

#### Изменено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31262](#)
  - PUT /dependencytypes/{id};
  - PUT /links/{id};
  - PUT /processsegments/{id};
  - PATCH /processsegments/{id};
  - PUT /segmentdependencies/{id};
  - PUT /equipmentsegmentspecifications/{id};
  - PUT /equipmentsegmentspecifications/properties/{propertyId};
  - PUT /personnelsegmentspecifications/{id};
  - PUT /personnelsegmentspecifications/properties/{propertyId};
  - PUT /materialsegmentspecifications/{id};
  - PUT /materialsegmentspecifications/properties/{propertyId};
  - PUT /physicalassetsegmentspecifications/{id};

- PUT /physicalassetsegmentspecifications/properties/{propertyId}.
- Обновлены библиотеки по работе с Redis. [#34411](#)
- Статус 400 при удалении несуществующей сущности. [#30263](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Обновлен пакет с метриками. [#32257](#)

Удалено:

- **BREAKING:** метод GET /processsegments устарел и будет удален в 2.18.0, используйте вместо него POST /processsegments.
- **BREAKING:** метод GET /physicalassetsegmentspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications/properties с указанием параметра specificationId.
- **BREAKING:** метод GET /physicalassetsegmentspecifications/properties устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications/properties.
- **BREAKING:** метод POST /physicalassetsegmentspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications/properties с указанием параметра specificationId.
- **BREAKING:** метод GET /physicalassetsegmentspecifications/properties/{propertyId}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications/properties без параметра specificationId, с onlyRoot равным false и parentId равным {propertyId} из path.
- **BREAKING:** метод DELETE /physicalassetsegmentspecifications/{id}/properties/{propertyId} устарел и будет удален в 2.18.0, используйте вместо него DELETE /physicalassetsegmentspecifications/properties/{propertyId}.
- **BREAKING:** метод GET /physicalassetsegmentspecifications устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications.
- **BREAKING:** метод GET /physicalassetsegmentspecifications/{id}/physicalassetsegmentspecifications устарел и будет удален в 2.18.0, используйте вместо него POST /physicalassetsegmentspecifications без параметра processSegmentId, с onlyRoot равным false и parentId равным {id} из path.
- **BREAKING:** метод GET /personnelsegmentspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentspecifications/properties с указанием параметра specificationId.

- **BREAKING:** метод GET /personnelsegmentsspecifications/properties устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentsspecifications/properties.
- **BREAKING:** метод POST /personnelsegmentsspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentsspecifications/properties с указанием параметра specificationId.
- **BREAKING:** метод GET /personnelsegmentsspecifications/properties/{propertyId}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentsspecifications/properties без параметра specificationId, с onlyRoot равным false и parentId равным {propertyId} из path.
- **BREAKING:** метод DELETE /personnelsegmentsspecifications/{id}/properties/{propertyId} устарел и будет удален в 2.18.0, используйте вместо него DELETE /personnelsegmentsspecifications/properties/{propertyId}.
- **BREAKING:** метод GET /personnelsegmentsspecifications устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentsspecifications.
- **BREAKING:** метод GET /personnelsegmentsspecifications/{id}/personnelsegmentsspecifications устарел и будет удален в 2.18.0, используйте вместо него POST /personnelsegmentsspecifications без параметра processSegmentId, с onlyRoot равным false и parentId равным {id} из path.
- **BREAKING:** метод GET /materialsegmentsspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /materialsegmentsspecifications/properties с указанием параметра specificationId.
- **BREAKING:** метод GET /materialsegmentsspecifications/properties устарел и будет удален в 2.18.0, используйте вместо него POST /materialsegmentsspecifications/properties.
- **BREAKING:** метод POST /materialsegmentsspecifications/{id}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /materialsegmentsspecifications/properties с указанием параметра specificationId.
- **BREAKING:** метод GET /materialsegmentsspecifications/properties/{propertyId}/properties устарел и будет удален в 2.18.0, используйте вместо него POST /materialsegmentsspecifications/properties без параметра specificationId, с onlyRoot равным false и parentId равным {propertyId} из path.
- **BREAKING:** метод DELETE /materialsegmentsspecifications/{id}/properties/{propertyId} устарел и будет удален в 2.18.0, используйте вместо него DELETE /materialsegmentsspecifications/properties/{propertyId}.
- **BREAKING:** метод GET /materialsegmentsspecifications устарел и будет удален в 2.18.0, используйте вместо него POST /materialsegmentsspecifications.

- **BREAKING:** метод `GET /materialsegmentspecifications/{id}/materialsegmentspecifications` устарел и будет удален в 2.18.0, используйте вместо него `POST /materialsegmentspecifications` без параметра `processSegmentId`, с `onlyRoot` равным `false` и `parentId` равным `{id}` из `path`.
- **BREAKING:** метод `GET /equipmentsegmentspecifications/{id}/properties` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications/properties` с указанием параметра `specificationId`.
- **BREAKING:** метод `GET /equipmentsegmentspecifications/properties` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications/properties`.
- **BREAKING:** метод `POST /equipmentsegmentspecifications/{id}/properties` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications/properties` с указанием параметра `specificationId`.
- **BREAKING:** метод `GET /equipmentsegmentspecifications/properties/{propertyId}/properties` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications/properties` без параметра `specificationId`, с `onlyRoot` равным `false` и `parentId` равным `{propertyId}` из `path`.
- **BREAKING:** метод `GET /equipmentsegmentspecifications` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications`.
- **BREAKING:** метод `GET /equipmentsegmentspecifications/{id}/equipmentsegmentspecifications` устарел и будет удален в 2.18.0, используйте вместо него `POST /equipmentsegmentspecifications` без параметра `processSegmentId`, с `onlyRoot` равным `false` и `parentId` равным `{id}` из `path`.
- **BREAKING:** В 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет удалено, используйте существующие поле `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`.
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически добавлено в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равное `null` переноситься не будет.

### 3.3.10. zif-sm-testspecification

[1.10.0] - 2023-11-06

Изменено:

- **BREAKING:** В 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31262](#)
  - `PUT /evaluatedproperties/{id};`



- PUT /links/{id};
  - PUT /propertymeasurements/{id};
  - PUT /testresults/{id};
  - PUT /testspecifications/{id};
  - PUT /testspecificationcriteria/{id};
  - PUT /testspecificationproperties/{id}.
- Обновлен пакет с метриками. [#32257](#)
  - Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
  - Исправлено применение политик ABAC. [#30880](#)

Удалено:

- **BREAKING:** метод POST `evaluatedproperties` устарел и будет удален в 2.18.0, используйте вместо него PUT /evaluatedproperties/{id}.
- **BREAKING:** метод POST `links` устарел и будет удален в 2.18.0, используйте вместо него PUT /links/{id}.
- **BREAKING:** метод POST `propertymeasurements` устарел и будет удален в 2.18.0, используйте вместо него PUT /propertymeasurements/{id}.
- **BREAKING:** метод POST `testresults` устарел и будет удален в 2.18.0, используйте вместо него PUT /testresults/{id}.
- **BREAKING:** метод POST `testspecifications` устарел и будет удален в 2.18.0, используйте вместо него PUT /testspecifications/{id}.
- **BREAKING:** метод POST `testspecificationcriteria` устарел и будет удален в 2.18.0, используйте вместо него PUT /testspecificationcriteria/{id}.
- **BREAKING:** метод POST `testspecificationproperties` устарел и будет удален в 2.18.0, используйте вместо него PUT /testspecificationproperties/{id}.
- **BREAKING:** в 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет заменено на `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`.
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически мигрировано в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равное `null` переноситься не будет.

### 3.3.11. zif-sm-workcalendar

[1.6.0] - 2023-11-06

Исправлено:

- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Возвращается статус 500 при поиске сущностей от пользователя с правами только на чтение и/или с правами на запись. [#30613](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Обновлен пакет с метриками. [#32257](#)

### 3.3.12. zif-sm-workdefinition

[2.11.0] - 2023-11-06

Изменено:

- Изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31080](#)
  - PUT /equipmentspecifications/{id};
  - PUT /equipmentspecifications{id}/equipmentspecificationproperties/{prpid};
  - PUT /materialspecifications/{id};
  - PUT /materialspecifications{id}/materialspecificationproperties/{prpid};
  - PUT /personnelspecifications/{id};
  - PUT /personnelspecifications{id}/personnelspecificationproperties/{prpid};
  - PUT /physicalAssetspecifications/{id};
  - PUT /physicalAssetspecifications{id}/physicalAssetspecificationproperties/{prpid};
  - PUT /links/{id};
  - PUT /workdirectives/{id};
  - PUT /workmasters/{id};
  - PUT /workflowspecifications/{id}.
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Исправлена ошибка при попытке использовать эндпоинт PUT /physicalAssetspecifications/{id}/physicalAssetspecificationproperties/{prpid}. [#31136](#)

- Исправлена ошибка при попытке использовать эндпоинт `PUT /materialspecifications/{id}/materialspecificationproperties/{prpid}`. [#31115](#)
- Исправлена ошибка при попытке использовать эндпоинт `PUT /equipmentspecifications/{id}/equipmentspecificationproperties/{prpid}`. [#31112](#)
- Исправлена ошибка при попытке использовать эндпоинт `PUT /personnelspecifications/{id}/personnelspecificationproperties/{prpid}`. [#31122](#)
- Исправлена ошибка при попытке использовать эндпоинт `PUT /workdirectives/{id}/parameterspecifications/{psid}`. [#31057](#)
- Обновлен пакет с метриками. [#32257](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)

Удалено:

- **BREAKING:** в 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет удалено, используйте существующие поле `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`;
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически добавлено в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равно `null` переноситься не будет.
- **BREAKING:** метод `POST equipmentspecifications` устарел и будет удален в 2.18.0, используйте вместо него `PUT /equipmentspecifications/{id}`.
- **BREAKING:** метод `POST materialspecifications` устарел и будет удален в 2.18.0, используйте вместо него `PUT /materialspecifications/{id}`.
- **BREAKING:** метод `POST personnelspecifications` устарел и будет удален в 2.18.0, используйте вместо него `PUT /personnelspecifications/{id}`.
- **BREAKING:** метод `POST physicalAssetspecifications` устарел и будет удален в 2.18.0, используйте вместо него `PUT /physicalAssetspecifications/{id}`.
- **BREAKING:** метод `POST links` устарел и будет удален в 2.18.0, используйте вместо него `PUT /links/{id}`.
- **BREAKING:** метод `POST workdirectives` устарел и будет удален в 2.18.0, используйте вместо него `PUT /workdirectives/{id}`.
- **BREAKING:** метод `POST workmasters` устарел и будет удален в 2.18.0, используйте вместо него `PUT /workmasters/{id}`.
- **BREAKING:** метод `POST workflowspecifications` устарел и будет удален в 2.18.0, используйте вместо него `PUT /workflowspecifications/{id}`.

### 3.3.13. zif-sm-workperformance

[1.8.0] - 2023-11-06

#### Добавлено:

- Сделать проверку корректности времени начала и окончания операций (работ). [#16354](#)

#### Исправлено:

- **BREAKING:** в 2.17.0 в сервисе будут изменены коды ответов в PUT эндпоинтах. Код 204 будет заменен на 200 с возвращением обновленной сущности. [#31167](#)
  - PUT /equipmentactuals/{id};
  - PUT /equipmentactuals{id}/equipmentactualproperties/{prpid};
  - PUT /jobresponses/{id};
  - PUT /jobresponses/{id}/jobresponsedata/{pid};
  - PUT /jobresponselists/{id};
  - PUT /links/{id};
  - PUT /materialactuals/{id};
  - PUT /materialactuals/{id}/materialactualproperties/{prpid};
  - PUT /personnelactuals/{id};
  - PUT /personnelactuals/{id}/personnelactualproperties/{prpid};
  - PUT /physicalAssetactuals/{id};
  - PUT /physicalAssetactuals/{id}/physicalAssetactualproperties/{prpid};
  - PUT /workperformances/{id};
  - PUT /workresponse/{id}.
- Удаление некоторых сущностей возвращает статус 200 вместо 204. [#31660](#)
- Поля name и location не помечены обязательными при создании сущностей в сервисе. [#30240](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- С включенным АВАС запрос на создание и чтение jobresponsedata завершается со статусом 500. [#31662](#)
- У эндпоинта DELETE /jobresponses/{id}/jobresponsedata/{pid} id и pid перепутаны. [#31963](#)
- Обновлен пакет с метриками. [#32257](#)

#### Удалено:

- **BREAKING:** метод `POST /equipmentactuals` устарел и будет удален в 2.18.0, используйте вместо него `PUT /equipmentactuals/{id}`.
- **BREAKING:** метод `POST /jobresponses` устарел и будет удален в 2.18.0, используйте вместо него `PUT /jobresponses/{id}`.
- **BREAKING:** метод `POST /jobresponselists` устарел и будет удален в 2.18.0, используйте вместо него `PUT /jobresponselists/{id}`.
- **BREAKING:** метод `POST /links` устарел и будет удален в 2.18.0, используйте вместо него `PUT /links/{id}`.
- **BREAKING:** метод `POST /materialactuals` устарел и будет удален в 2.18.0, используйте вместо него `PUT /materialactuals/{id}`.
- **BREAKING:** метод `POST /personnelactuals` устарел и будет удален в 2.18.0, используйте вместо него `PUT /personnelactuals/{id}`.
- **BREAKING:** метод `POST /physicalAssetactuals` устарел и будет удален в 2.18.0, используйте вместо него `PUT /physicalAssetactuals/{id}`.
- **BREAKING:** метод `POST /workperformances` устарел и будет удален в 2.18.0, используйте вместо него `PUT /workperformances/{id}`.
- **BREAKING:** метод `POST /workresponse` устарел и будет удален в 2.18.0, используйте вместо него `PUT /workresponse/{id}`.
- **BREAKING:** В 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет заменено на `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`;
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически мигрировано в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равное `00000000-0000-0000-0000-00000000000000` переноситься не будет.

### 3.3.14. zif-sm-workschedule

[1.8.0] - 2023-11-06

#### Добавлено:

- Сделать проверку корректности времени начала и окончания операций (работ). [#16354](#)

#### Исправлено:

- Метод `Delete /materialrequirements/{materialRequirementId}/materialrequirementproperties/{PropertyId}` возвращает 200 вместо 204. [#31001](#)
- Исправлена ошибка, из-за которой эндпоинт `PUT /equipmentrequirements/{id}/equipmentrequirementproperties/{prpid}` возвращал ошибку валидации при корректном Id. [#30183](#)
- Обновлены библиотеки по работе с АВАС. [#29758](#)

- Исправлена ошибка, из-за которой эндпоинт `POST /physicalAssetrequirements` позволял создавать сущности от роли пользователя с правами только на чтение. [#30998](#)
- Исправлена ошибка, из-за которой эндпоинт `DELETE /joborders/{id}/joborderparameters/{pid}` возвращает 404 при передаче правильных параметров для удаления. [#31040](#)
- Обновлена библиотека по работе с PostgreSQL для включения возможности использовать TLS. [#31350](#)
- Обновлен пакет с метриками. [#32257](#)

#### Удалено:

- **BREAKING:** В 2.18.0 будет произведена унификация работы с ОПИ:
  - Во всех CRUD методах свойство и параметр фильтрации `hierarchyScopeId` будет заменено на `hierarchyScopeIds`. Аналогичные изменения будут выполнены в запросах к GraphQL в сервисе `zif-om-graphql`;
  - Поле в БД `hierarchyScopeId` будет удалено и автоматически мигрировано в поле `hierarchyScopeIds`, значение `hierarchyScopeId` равное `00000000-0000-0000-0000-000000000000` переноситься не будет.

### 3.3.15. zui-app-rdm-common

[1.17.0] - 2023-11-06

#### Исправлено:

- НСИ: возможно создание элемента справочника с пустыми значениями полей. [#28822](#)
- Справочники: при отправке данных на сервер произошла ошибка. HTTP-статус - 400: "Unknown". [#30157](#)
- НСИ. FE.: нельзя сохранить элемент со значениями 0 и 0, 0. [#30428](#)
- Справочники: скриптовая ошибка `Error: Required parameter id was null or undefined when calling getStatisticsByDirectoryIdUsingGET`. [#30893](#)
- НСИ: при повторном редактировании элемента справочника, поля значений в расширенных настройках не соответствуют действующим значениям настройки актуальности. [#29908](#)
- НСИ: при выборе строки справочника появляется активная рамка. [#29914](#)
- НСИ: в общем журнале изменений список изменений отображается сквозь шапку грида. [#31531](#)
- Справочники - Общий журнал изменений: невозможно прочитать длинное название в выпадающем списке. [#31541](#)
- Справочники: длинное наименование справочника не умещается на UI при открытом окне области ОПИ. [#31621](#)
- НСИ: Приложение Справочники вызывает ошибку стилей в других приложениях. [#28229](#)

### 3.3.16. zui-sm-directories

[2.16.0] - 2023-11-06

#### Добавлено:

- Перенос ОПИ из zui-app-om в zui-sm-directories. [#24648](#)

#### Исправлено:

- Перестают отображаться параметры поиска при возврате из окна редактирования ОПИ. [#26596](#)
- В гриде ОПИ отображаются не все ОПИ за раз. [#26383](#)
- Сбрасывается сортировка и параметры поиска при возврате из окна редактирования ОПИ. [#26378](#)
- Поиск по ОПИ. Непонятный состав фильтров. [#22334](#)
- При одновременном удалении корневого элемента иерархии и его вложенного элемента иерархии, эти элементы не пропадают сами, а только при обновлении страницы. [#28758](#)

### 3.3.17. zif-om-graphql

[3.10.0] - 2023-11-10

#### Удалено:

- **BREAKING:** В 2.18.0 будут удалены следующие устаревшие мутации:
  - `createOrUpdateSegmentDependencySmProcess(segmentDependency: SegmentDependencyInputSmProcess): SegmentDependencySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteSegmentDependencySmProcess(id: Uuid!): SegmentDependencySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdateProcessSegmentSmProcess(processSegment: ProcessSegmentInputSmProcess): ProcessSegmentSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteProcessSegmentSmProcess(id: Uuid!): ProcessSegmentSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdatePhysicalAssetSegmentSpecificationSmProcess(specification: PhysicalAssetSegmentSpecificationInputSmProcess): PhysicalAssetSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deletePhysicalAssetSegmentSpecificationSmProcess(id: Uuid!): PhysicalAssetSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdatePhysicalAssetSegmentSpecificationPropertySmProcess(property: PhysicalAssetSegmentSpecificationPropertyInputSmProcess):`

- ```
PhysicalAssetSegmentSpecificationPropertySmProcess
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
```
- `deletePhysicalAssetSegmentSpecificationPropertySmProcess (specificationId: Uuid! propertyId: Uuid!): PhysicalAssetSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdatePersonnelSegmentSpecificationSmProcess (specification: PersonnelSegmentSpecificationInputSmProcess): PersonnelSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deletePersonnelSegmentSpecificationSmProcess (id: Uuid!): PersonnelSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdatePersonnelSegmentSpecificationPropertySmProcess (property: PersonnelSegmentSpecificationPropertyInputSmProcess): PersonnelSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deletePersonnelSegmentSpecificationPropertySmProcess (specificationId: Uuid! propertyId: Uuid!): PersonnelSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdateMaterialSegmentSpecificationSmProcess (specification: MaterialSegmentSpecificationInputSmProcess): MaterialSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteMaterialSegmentSpecificationSmProcess (id: Uuid!): MaterialSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdateMaterialSegmentSpecificationPropertySmProcess (property: MaterialSegmentSpecificationPropertyInputSmProcess): MaterialSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteMaterialSegmentSpecificationPropertySmProcess (specificationId: Uuid! propertyId: Uuid!): MaterialSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdateLinkSmProcess (link: LinkInputSmProcess): LinkSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteLinkSmProcess (id: Uuid!): LinkSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `createOrUpdateEquipmentSegmentSpecificationSmProcess (specification: EquipmentSegmentSpecificationInputSmProcess): EquipmentSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
  - `deleteEquipmentSegmentSpecificationSmProcess (id: Uuid!): EquipmentSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`



- `createOrUpdateEquipmentSegmentSpecificationPropertySmProcess` (property: EquipmentSegmentSpecificationPropertyInputSmProcess): EquipmentSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- `deleteEquipmentSegmentSpecificationPropertySmProcess` (specificationId: Uuid! propertyId: Uuid!): EquipmentSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- `createOrUpdateDependencyTypeSmProcess` (dependencyType: DependencyTypeInputSmProcess): DependencyTypeSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- `deleteDependencyTypeSmProcess` (id: Uuid!): DependencyTypeSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- `operationDefinitionCreateOrUpdatePhysicalAssetSpecification` (request: PhysicalAssetSpecificationCreateRequestInput): OdPhysicalAssetSpecification @deprecated(reason: "Устарело. Будет удалено в версии 2.18.0. Используйте новый API.");
- `OperationDefinitionCreateOrUpdatePersonnelSpecification` (request: PersonnelSpecificationCreateRequestInput): OdPersonnelSpecification @deprecated(reason: "Устарело. Будет удалено в версии 2.18.0. Используйте новый API.");
- `createOrUpdateOperationSegmentFromProcessSegment` ("Сегмент операции" operationSegment: OperationSegmentInput): OperationSegment;
- `createOrUpdateOperationSegment` ("Сегмент операции" operationSegment: OperationSegmentInput): OperationSegment;
- `operationDefinitionCreateOrUpdateMaterialSpecification` (request: MaterialSpecificationCreateRequestInput): OdMaterialSpecification @deprecated(reason: "Устарело. Будет удалено в версии 2.18.0. Используйте новый API.");
- `operationDefinitionCreateOrUpdateEquipmentSpecification` (request: EquipmentSpecificationCreateRequestInput): OdEquipmentSpecification @deprecated(reason: "Устарело. Будет удалено в версии 2.18.0. Используйте новый API.");
- **BREAKING: В 2.18.0 будут удалены следующие устаревшие запросы:**
  - `segmentDependencySmProcess` (id: Uuid! ts: String): SegmentDependencySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - `segmentDependenciesSmProcess` (page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid ts: String description: String): [SegmentDependencySmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - `segmentDependenciesSmProcessPage` (page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid ts: String description: String): SegmentDependencyPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");

- processSegmentSmProcess(id: Uuid! ts: String): ProcessSegmentSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- processSegmentsSmProcess(page: Int size: Int sort: [String] isPaged: Boolean = false search: String name: String description: String ts: String hierarchyScopeIds: [Uuid!] onlyRoot: Boolean! = true filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [ProcessSegmentSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- processSegmentsSmProcessPage(page: Int size: Int sort: [String] isPaged: Boolean = false search: String name: String description: String ts: String hierarchyScopeIds: [Uuid!] onlyRoot: Boolean! = true filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): ProcessSegmentPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- childrenProcessSegmentsSmProcess(id: Uuid! search: String name: String description: String ts: String hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [ProcessSegmentSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- childrenProcessSegmentsSmProcessPage(id: Uuid! search: String name: String description: String ts: String hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): ProcessSegmentPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- groupedProcessSegmentsSmProcess(groupBy: [GroupParameterInput] page: Int size: Int sort: [String] isPaged: Boolean = false search: String name: String description: String ts: String hierarchyScopeIds: [Uuid!] onlyRoot: Boolean! = true filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [HierarchyGroup] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- physicalAssetSegmentSpecificationSmProcess(id: Uuid! ts: String): PhysicalAssetSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- physicalAssetSegmentSpecificationsSmProcess(hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [PhysicalAssetSegmentSpecificationSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- physicalAssetSegmentSpecificationsSmProcessPage(hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): PhysicalAssetSegmentSpecificationPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- childrenPhysicalAssetSegmentSpecificationsSmProcess(id: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String]

- ```
isPaged: Boolean = false ts: String name: String description: String
search: String filterType: HierarchyScopeIdsFilterType filterMode:
FilterMode): [PhysicalAssetSegmentSpecificationSmProcess]
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
```
- childrenPhysicalAssetSegmentSpecificationsSmProcessPage(id: Uuid!  
hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String]  
isPaged: Boolean = false ts: String name: String description: String  
search: String filterType: HierarchyScopeIdsFilterType filterMode:  
FilterMode): PhysicalAssetSegmentSpecificationPage  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - physicalAssetSegmentSpecificationPropertySmProcess(specificationI  
d: Uuid! propertyId: Uuid! ts: String):  
PhysicalAssetSegmentSpecificationPropertySmProcess  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - physicalAssetSegmentSpecificationPropertiesSmProcess(specificatio  
nId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort:  
[String] isPaged: Boolean = false onlyRoot: Boolean ts: String name:  
String filterType: HierarchyScopeIdsFilterType filterMode:  
FilterMode): [PhysicalAssetSegmentSpecificationPropertySmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - physicalAssetSegmentSpecificationPropertiesSmProcessPage(specific  
ationId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort:  
[String] isPaged: Boolean = false onlyRoot: Boolean ts: String name:  
String filterType: HierarchyScopeIdsFilterType filterMode:  
FilterMode): PhysicalAssetSegmentSpecificationPropertyPage  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - childrenPhysicalAssetSegmentSpecificationPropertiesSmProcess(spec  
ificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!]  
page: Int size: Int sort: [String] isPaged: Boolean = false ts:  
String name: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode):  
[PhysicalAssetSegmentSpecificationPropertySmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - childrenPhysicalAssetSegmentSpecificationPropertiesSmProcessPage(  
specificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!]  
page: Int size: Int sort: [String] isPaged: Boolean = false ts:  
String name: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode):  
PhysicalAssetSegmentSpecificationPropertyPage @deprecated(reason:  
"2.18.0 Устарело. Используйте новый API.");
  - personnelSegmentSpecificationSmProcess(id: Uuid! ts: String):  
PersonnelSegmentSpecificationSmProcess @deprecated(reason: "2.18.0  
Устарело. Используйте новый API.");
  - personnelSegmentSpecificationsSmProcess(hierarchyScopeIds: [Uuid!]  
page: Int size: Int sort: [String] isPaged: Boolean = false  
processSegmentId: Uuid ts: String search: String name: String  
description: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode): [PersonnelSegmentSpecificationSmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - personnelSegmentSpecificationsSmProcessPage(hierarchyScopeIds:  
[Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false  
processSegmentId: Uuid ts: String search: String name: String

- ```
description: String filterType: HierarchyScopeIdsFilterType
filterMode: FilterMode): PersonnelSegmentSpecificationPage
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
```
- childrenPersonnelSegmentSpecificationsSmProcess(id: Uuid!  
hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String]  
isPaged: Boolean = false ts: String search: String name: String  
description: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode): [PersonnelSegmentSpecificationSmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - childrenPersonnelSegmentSpecificationsSmProcessPage(id: Uuid!  
hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String]  
isPaged: Boolean = false ts: String search: String name: String  
description: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode): PersonnelSegmentSpecificationPage  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - personnelSegmentSpecificationPropertySmProcess(specificationId:  
Uuid! propertyId: Uuid! ts: String):  
PersonnelSegmentSpecificationPropertySmProcess @deprecated(reason:  
"2.18.0 Устарело. Используйте новый API.");
  - personnelSegmentSpecificationPropertiesSmProcess(specificationId:  
Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String]  
isPaged: Boolean = false onlyRoot: Boolean ts: String name: String  
filterType: HierarchyScopeIdsFilterType filterMode: FilterMode):  
[PersonnelSegmentSpecificationPropertySmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - personnelSegmentSpecificationPropertiesSmProcessPage(specificatio  
nId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort:  
[String] isPaged: Boolean = false onlyRoot: Boolean ts: String name:  
String filterType: HierarchyScopeIdsFilterType filterMode:  
FilterMode): PersonnelSegmentSpecificationPropertyPage  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - childrenPersonnelSegmentSpecificationPropertiesSmProcess(specific  
ationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!] page:  
Int size: Int sort: [String] isPaged: Boolean = false ts: String  
name: String filterType: HierarchyScopeIdsFilterType filterMode:  
FilterMode): [PersonnelSegmentSpecificationPropertySmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - childrenPersonnelSegmentSpecificationPropertiesSmProcessPage(spec  
ificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!]  
page: Int size: Int sort: [String] isPaged: Boolean = false ts:  
String name: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode): PersonnelSegmentSpecificationPropertyPage  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
  - materialSegmentSpecificationSmProcess(id: Uuid! ts: String):  
MaterialSegmentSpecificationSmProcess @deprecated(reason: "2.18.0  
Устарело. Используйте новый API.");
  - materialSegmentSpecificationsSmProcess(hierarchyScopeIds: [Uuid!]  
page: Int size: Int sort: [String] isPaged: Boolean = false  
processSegmentId: Uuid ts: String search: String name: String  
description: String filterType: HierarchyScopeIdsFilterType  
filterMode: FilterMode): [MaterialSegmentSpecificationSmProcess]  
@deprecated(reason: "2.18.0 Устарело. Используйте новый API.");

- o materialSegmentSpecificationsSmProcessPage (hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): MaterialSegmentSpecificationPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o childrenMaterialSegmentSpecificationsSmProcess (id: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [MaterialSegmentSpecificationSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o childrenMaterialSegmentSpecificationsSmProcessPage (id: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): MaterialSegmentSpecificationPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o materialSegmentSpecificationPropertySmProcess (specificationId: Uuid! propertyId: Uuid! ts: String): MaterialSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o materialSegmentSpecificationPropertiesSmProcess (specificationId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false onlyRoot: Boolean ts: String name: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [MaterialSegmentSpecificationPropertySmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o materialSegmentSpecificationPropertiesSmProcessPage (specificationId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false onlyRoot: Boolean ts: String name: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): MaterialSegmentSpecificationPropertyPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o childrenMaterialSegmentSpecificationPropertiesSmProcess (specificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String name: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [MaterialSegmentSpecificationPropertySmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o childrenMaterialSegmentSpecificationPropertiesSmProcessPage (specificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String name: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): MaterialSegmentSpecificationPropertyPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o linkSmProcess (id: Uuid! ts: String): LinkSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");
- o linksSmProcess (page: Int size: Int sort: [String] isPaged: Boolean = false sourceId: Uuid destinationId: Uuid linkType: LinkTypeSmProcess ts: String): [LinkSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");

- `equipmentSegmentSpecificationSmProcess(id: Uuid! ts: String): EquipmentSegmentSpecificationSmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `equipmentSegmentSpecificationsSmProcess(hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode ts: String): [EquipmentSegmentSpecificationSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `equipmentSegmentSpecificationsSmProcessPage(hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false processSegmentId: Uuid search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode ts: String): EquipmentSegmentSpecificationPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `childrenEquipmentSegmentSpecificationsSmProcess(id: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): [EquipmentSegmentSpecificationSmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `childrenEquipmentSegmentSpecificationsSmProcessPage(id: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String search: String name: String description: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode): EquipmentSegmentSpecificationPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `equipmentSegmentSpecificationPropertySmProcess(specificationId: Uuid! propertyId: Uuid! ts: String): EquipmentSegmentSpecificationPropertySmProcess @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `equipmentSegmentSpecificationPropertiesSmProcess(specificationId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false onlyRoot: Boolean ts: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode name: String): [EquipmentSegmentSpecificationPropertySmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `equipmentSegmentSpecificationPropertiesSmProcessPage(specificationId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false onlyRoot: Boolean ts: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode name: String): EquipmentSegmentSpecificationPropertyPage @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `childrenEquipmentSegmentSpecificationPropertiesSmProcess(specificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts: String filterType: HierarchyScopeIdsFilterType filterMode: FilterMode name: String): [EquipmentSegmentSpecificationPropertySmProcess] @deprecated(reason: "2.18.0 Устарело. Используйте новый API.");`
- `childrenEquipmentSegmentSpecificationPropertiesSmProcessPage(specificationId: Uuid! propertyId: Uuid! hierarchyScopeIds: [Uuid!] page: Int size: Int sort: [String] isPaged: Boolean = false ts:`

- ```
String filterType: HierarchyScopeIdsFilterType filterMode:
FilterMode name: String):
EquipmentSegmentSpecificationPropertyPage @deprecated(reason:
"2.18.0 Устарело. Используйте новый API.");
```
- dependencyTypeSmProcess(id: Uuid! ts: String):  
DependencyTypeSmProcess @deprecated(reason: "2.18.0 Устарело.  
Используйте новый API.");
  - dependencyTypesSmProcess(page: Int size: Int sort: [String]  
isPaged: Boolean = false search: String name: String description:  
String ts: String): [DependencyTypeSmProcess] @deprecated(reason:  
"2.18.0 Устарело. Используйте новый API.");
  - dependencyTypesSmProcessPage(page: Int size: Int sort: [String]  
isPaged: Boolean = false search: String name: String description:  
String ts: String): DependencyTypePage @deprecated(reason: "2.18.0  
Устарело. Используйте новый API.");
  - operationDefinitionPhysicalAssetSpecifications(operationSegmentId  
: Uuid! page: Int size: Int ispaged: Boolean ts: String):  
PageOfOdPhysicalAssetSpecification @deprecated(reason: "Устарело.  
Будет удалено в версии 2.18.0. Используйте новый API.");
  - operationDefinitionSegmentSpecificationPhysicalAssetSpecification  
(request: OperationDefinitionSpecificationsRequestInput):  
PageOfPhysicalAssetSpecification @deprecated(reason: "Устарело.  
Будет удалено в версии 2.18.0. Используйте новый API.");
  - operationDefinitionPhysicalAssetSpecification(request:  
OdGetByIdRequestInput!): OdPhysicalAssetSpecification  
@deprecated(reason: "Устарело. Будет удалено в версии 2.18.0.  
Используйте новый API.");
  - operationDefinitionPhysicalAssetSpecificationProperties(specifica  
tionId: Uuid! page: Int size: Int ispaged: Boolean ts: String):  
PageOfOdPhysicalAssetSpecificationProperty;
  - operationDefinitionPhysicalAssetSpecificationProperty(request:  
OdGetByIdRequestInput!): OdPhysicalAssetSpecificationProperty;
  - operationDefinitionPersonnelSpecifications(operationSegmentId:  
Uuid! page: Int size: Int ispaged: Boolean ts: String):  
PageOfOdPersonnelSpecification @deprecated(reason: "Устарело.  
Будет удалено в версии 2.18.0. Используйте новый API.");
  - operationDefinitionSegmentSpecificationPersonnelSpecification(req  
uest: OperationDefinitionSpecificationsRequestInput):  
PageOfPersonnelSpecification @deprecated(reason: "Устарело. Будет  
удалено в версии 2.18.0. Используйте новый API.");
  - operationDefinitionPersonnelSpecification(request:  
OdGetByIdRequestInput!): OdPersonnelSpecification  
@deprecated(reason: "Устарело. Будет удалено в версии 2.18.0.  
Используйте новый API.");
- **BREAKING:** В 2.18.0 будут удалены следующие устаревшие запросы в типе `type ZifSmProcess`:
    - `equipmentSegmentSpecificationProperties(request: ZifSmProcess_Clients_GetEquipmentSegmentSpecificationPropertiesRe`

- ```
questInput) :  
  [ZifSmProcess_Clients_EquipmentSegmentSpecificationProperty!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationPropertiesPage");
```
- ```
equipmentSegmentSpecificationPropertiesChildren (request:  
  ZifSmProcess_Clients_GetChildrenSegmentEquipmentSegmentSpecif  
  icationPropertiesRequestInput!) :  
  [ZifSmProcess_Clients_EquipmentSegmentSpecificationProperty!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationPropertiesPage");
```
  - ```
equipmentSegmentSpecificationPropertiesChildrenPage (request:  
  ZifSmProcess_Clients_GetChildrenSegmentEquipmentSegmentSpecif  
  icationPropertiesRequestInput!) :  
  ZifSmProcess_Clients_EquipmentSegmentSpecificationPropertyPage!  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationPropertiesPage");
```
  - ```
equipmentSegmentSpecifications (request:  
  ZifSmProcess_Clients_GetEquipmentSegmentSpecificationsRequestInpu  
  t) : [ZifSmProcess_Clients_EquipmentSegmentSpecification!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationsPage");
```
  - ```
equipmentSegmentSpecificationsChildren (request:  
  ZifSmProcess_Clients_GetChildrenSegmentEquipmentSegmentSpecif  
  icationsRequestInput!) :  
  [ZifSmProcess_Clients_EquipmentSegmentSpecification!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationsPage");
```
  - ```
equipmentSegmentSpecificationsChildrenPage (request:  
  ZifSmProcess_Clients_GetChildrenSegmentEquipmentSegmentSpecif  
  icationsRequestInput!) :  
  ZifSmProcess_Clients_EquipmentSegmentSpecificationPage!  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  EquipmentSegmentSpecificationsPage");
```
  - ```
materialSegmentSpecificationProperties (request:  
  ZifSmProcess_Clients_GetMaterialSegmentSpecificationPropertiesReq  
  uestInput) :  
  [ZifSmProcess_Clients_MaterialSegmentSpecificationProperty!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  MaterialSegmentSpecificationPropertiesPage");
```
  - ```
materialSegmentSpecificationPropertiesChildren (request:  
  ZifSmProcess_Clients_GetChildrenSegmentMaterialSegmentSpecif  
  icationPropertiesRequestInput!) :  
  [ZifSmProcess_Clients_MaterialSegmentSpecificationProperty!!]  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  MaterialSegmentSpecificationPropertiesPage");
```
  - ```
materialSegmentSpecificationPropertiesChildrenPage (request:  
  ZifSmProcess_Clients_GetChildrenSegmentMaterialSegmentSpecif  
  icationPropertiesRequestInput!) :  
  ZifSmProcess_Clients_MaterialSegmentSpecificationPropertyPage!  
  @authorize(apply: BEFORE_RESOLVER) @deprecated(reason: "2.18.0 use  
  MaterialSegmentSpecificationPropertiesPage");
```



- materialSegmentSpecifications (request: ZifSmProcess\_Clients\_GetMaterialSegmentSpecificationsRequestInput): [ZifSmProcess\_Clients\_MaterialSegmentSpecification!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use MaterialSegmentSpecificationsPage");
- materialSegmentSpecificationsChildren (request: ZifSmProcess\_Clients\_GetChildrenSegmentMaterialSegmentSpecificationsRequestInput!): [ZifSmProcess\_Clients\_MaterialSegmentSpecification!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use MaterialSegmentSpecificationsPage");
- materialSegmentSpecificationsChildrenPage (request: ZifSmProcess\_Clients\_GetChildrenSegmentMaterialSegmentSpecificationsRequestInput!): ZifSmProcess\_Clients\_MaterialSegmentSpecificationPage! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use MaterialSegmentSpecificationsPage");
- personnelSegmentSpecificationProperties (request: ZifSmProcess\_Clients\_GetPersonnelSegmentSpecificationPropertiesRequestInput): [ZifSmProcess\_Clients\_PersonnelSegmentSpecificationProperty!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationPropertiesPage");
- personnelSegmentSpecificationPropertiesChildren (request: ZifSmProcess\_Clients\_GetChildrenSegmentPersonnelSegmentSpecificationPropertiesRequestInput!): [ZifSmProcess\_Clients\_PersonnelSegmentSpecificationProperty!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationPropertiesPage");
- personnelSegmentSpecificationPropertiesChildrenPage (request: ZifSmProcess\_Clients\_GetChildrenSegmentPersonnelSegmentSpecificationPropertiesRequestInput!): ZifSmProcess\_Clients\_PersonnelSegmentSpecificationPropertyPage! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationPropertiesPage");
- personnelSegmentSpecifications (request: ZifSmProcess\_Clients\_GetPersonnelSegmentSpecificationsRequestInput): [ZifSmProcess\_Clients\_PersonnelSegmentSpecification!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationsPage");
- personnelSegmentSpecificationsChildren (request: ZifSmProcess\_Clients\_GetChildrenSegmentPersonnelSegmentSpecificationsRequestInput!): [ZifSmProcess\_Clients\_PersonnelSegmentSpecification!]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationsPage");
- personnelSegmentSpecificationsChildrenPage (request: ZifSmProcess\_Clients\_GetChildrenSegmentPersonnelSegmentSpecificationsRequestInput!): ZifSmProcess\_Clients\_PersonnelSegmentSpecificationPage! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use PersonnelSegmentSpecificationsPage");

- `physicalAssetSegmentSpecificationProperties` (request: `ZifSmProcess_Clients_GetPhysicalAssetSegmentSpecificationPropertiesRequestInput`):  
[`ZifSmProcess_Clients_PhysicalAssetSegmentSpecificationProperty!`]  
! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationPropertiesPage`");
- `physicalAssetSegmentSpecificationPropertiesChildren` (request: `ZifSmProcess_Clients_GetChildrenSegmentPhysicalAssetSegmentSpecificationPropertiesRequestInput!`):  
[`ZifSmProcess_Clients_PhysicalAssetSegmentSpecificationProperty!`]  
! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationPropertiesPage`");
- `physicalAssetSegmentSpecificationPropertiesChildrenPage` (request: `ZifSmProcess_Clients_GetChildrenSegmentPhysicalAssetSegmentSpecificationPropertiesRequestInput!`):  
`ZifSmProcess_Clients_PhysicalAssetSegmentSpecificationPropertyPage!`  
! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationPropertiesPage`");
- `physicalAssetSegmentSpecifications` (request: `ZifSmProcess_Clients_GetPhysicalAssetSegmentSpecificationsRequestInput`): [`ZifSmProcess_Clients_PhysicalAssetSegmentSpecification!!`]  
@authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationsPage`");
- `physicalAssetSegmentSpecificationsChildren` (request: `ZifSmProcess_Clients_GetChildrenSegmentPhysicalAssetSegmentSpecificationsRequestInput!`):  
[`ZifSmProcess_Clients_PhysicalAssetSegmentSpecification!!`]  
@authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationsPage`");
- `physicalAssetSegmentSpecificationsChildrenPage` (request: `ZifSmProcess_Clients_GetChildrenSegmentPhysicalAssetSegmentSpecificationsRequestInput!`):  
`ZifSmProcess_Clients_PhysicalAssetSegmentSpecificationPage!`  
@authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `PhysicalAssetSegmentSpecificationsPage`");
- `processSegmentsByFilter` (request: `ZifSmProcess_Clients_GetProcessSegmentsRequestInput`):  
[`ZifSmProcess_Clients_ProcessSegment!!`]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `ProcessSegmentsPage`");
- `processSegmentsChildren` (request: `ZifSmProcess_Clients_GetChildrenSegmentProcessSegmentsRequestInput!`): [`ZifSmProcess_Clients_ProcessSegment!!`]! @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `ProcessSegmentsPage`");
- `processSegmentsChildrenPage` (request: `ZifSmProcess_Clients_GetChildrenSegmentProcessSegmentsRequestInput!`): `ZifSmProcess_Clients_ProcessSegmentPage!` @authorize(apply: BEFORE\_RESOLVER) @deprecated(reason: "2.18.0 use `ProcessSegmentsPage`");

- **BREAKING:** В 2.18.0 помеченные атрибутом @deprecated на момент выхода 2.16.0 свойства сущностей и типов, а также запросы и мутации могут быть удалены.

## 3.4. Приложения

### 3.4.1. zui-app-dashboard-dx

[11.0.0] - 2023-11-13

Изменено:

- Обновлены версии компонентов `DevExpress` до 22.1.11. [#30873](#)

Исправлено:

- Устранены уязвимости.
- Логика обработки запроса на получение нового дашборда при смене выбранного дашборда в дереве. [#32649](#)
- Установлен приоритет для `z-index` из файла со стилями. [#32558](#)
- Испралена логика при смене выбранного дашборда в дереве. [#32646](#)

### 3.4.2. zui-app-datalinkeditor

[2.0.0] - 2023-11-13

Добавлено:

- Возможность настройки правил автоматической генерации. [#27822](#)

Исправлено:

- Устранены уязвимости.

### 3.4.3. zui-app-documentexplorer

[15.0.0] - 2023-11-13

Добавлено:

- Реализована навигация по ссылке для элемента группы. [#30500](#)
- Реализовано всплывающее меню элемента для группы. [#30501](#)
- Возможность выбирать формат загружаемого файла. [#30865](#)

Изменено:

- Обновлены версии компонентов `DevExpress` до 22.1.11. [#30873](#)

Исправлено:

- Невозможно выбрать время для параметра отчета формата `дата и время`. [#31062](#)
- Ошибка при отображении наименований параметров отчета. [#30268](#)
- Ошибка при передаче параметров отчета. [#30697](#)
- Нет возможности заполнять значения параметров в синтаксисе `ZifTime`. [#30791](#)
- Параметры отчета теперь отображаются корректно. [#31125](#)

- Исправлено правило выбора версии. [#31671](#)
- Используется локальный часовой пояс вместо выбранного в поле. [#32773](#)
- Отчет формата `XLSM` не меняется на `XLS` при скачивании с фронта. [#32881](#)
- Интерфейс `Параметры шаблона` выходит за пределы страницы. [#32402](#)
- Пропадают границы поля в шаблоне повторений. [#32403](#)
- Смещение по времени активно при выключенной опции автогенерации. [#31549](#)
- Некорректные расширения файлов отчета при отправке отчета по email. [#32178](#)
- Исправлена логика при смене выбранного дашборда в дереве. [#32646](#)
- Исправлена логика обработки запроса на получение нового дашборда при смене выбранного дашборда в дереве. [#32649](#)
- Не работает контроль устаревания данных на мнемосхемах. [#30586](#)
- Не работает групповое правило мигания или скрытия для части элементов. [#31659](#)
- После появления скрытого элемента на мнемосхеме, изменение масштаба приводит к его скрытию. [#31803](#)
- Отключается ручной ввод при перемещении элемента на мнемосхеме. [#31026](#)
- Запросы на подписку продолжают проходить после прекращения режима просмотра. [#32231](#)
- Пропадает `Smart Trend` после добавления дискретного набора. [#32930](#)
- Отображение количества найденных элементов при поиске привязки. [#32554](#)
- Множественный выбор элементов при поиске привязки. [#32547](#)
- Изменение размера шрифта при раскрытии списков. [#33582](#)
- Отображение курсора для группы или элемента, содержащих ссылки. [#30759](#)
- Ошибка при сохранении введенного значения в режиме просмотра мнемосхемы через ручной ввод. [#29939](#)
- Устранены уязвимости.

### 3.4.4. zui-app-events-registry

[4.4.15] - 2023-11-13

Исправлено:

- Проверка на наличие циклических ссылок событий задач. [#30027](#)
- Добавлено поле `acknowledgedUser`. [#30801](#)
- Отображение кнопок `Связать` и `Отвязать` при наличии действия `bindEvent` в настройках. [#30802](#)
- Корректная валидация `GUID` / `JSON` полей. [#30892](#)
- Не меняет цвет кнопка при переходе в Режим работы в реальном времени [#32477](#)

- Некорректное сообщение валидации и ошибка 400 при попытке сохранить событие с некорректным `uid`. [#33623](#)
- Устранены уязвимости.

### 3.4.5. zui-app-mnemoeditor

[15.0.0] - 2023-11-13

Добавлено:

- Реализована навигация по ссылке для элемента группы. [#30500](#)
- Реализовано всплывающее меню элемента для группы. [#30501](#)

Исправлено:

- Не работает контроль устаревания данных на мнемосхемах. [#30586](#)
- Не работает групповое правило мигания или скрытия для части элементов. [#31659](#)
- После появления скрытого элемента на мнемосхеме, изменение масштаба приводит к его скрытию. [#31803](#)
- Отключается ручной ввод при перемещении элемента на мнемосхеме. [#31026](#)
- Запросы на подписку продолжают проходить после прекращения режима просмотра. [#32231](#)
- Пропадает `Smart Trend` после добавления дискретного набора. [#32930](#)
- Отображение кол-ва найденных элементов при поиске привязки. [#32554](#)
- Множественный выбор элементов при поиске привязки. [#32547](#)
- Изменение размера шрифта при раскрытии списков. [#33582](#)
- Отображение курсора для группы или элемента, содержащих ссылки. [#30759](#)
- Ошибка при сохранении введенного значения в режиме просмотра мнемосхемы через ручной ввод. [#29939](#)
- Устранены уязвимости.

### 3.4.6. zui-app-reporteditor

[11.0.0] - 2023-11-13

Изменено:

- Обновлено версии компонентов `DevExpress` до 22.1.11. [#30873](#)

Исправлено:

- Пропадают границы поля в шаблоне повторений. [#32403](#)
- Интерфейс `Параметры шаблона` выходит за пределы страницы. [#32402](#)
- Некорректные расширения файлов отчета при отправке отчета по email. [#32178](#)
- Смещение по времени активно при выключенной опции автогенерации. [#31549](#)

- Устранены уязвимости.

### 3.4.7. zif-mnemoschemes-storage

[6.1.4] - 2023-11-13

Изменено:

- Обновлён пакет `zbc-pkg-aspnet-configuration-metrics` до версии 7.1.2. [#27735](#)
- Обновлён пакет `zbc-pkg-aspnet-configuration-postgres` до версии 7.4.2. [#31197](#)

Исправлено:

- При удалении мнемосхемы не удаляются связанные атрибуты. [#33389](#)

### 3.4.8. zif-datalink-xl и плагин Excel

[1.7.0] - 2023-11-13

Изменено:

- Обновлён пакет `zbc-pkg-aspnet-configuration-metrics` до версии 7.1.2. [#27735](#)
- Перейти на актуальные методы UDL (v2). [#25153](#)
- Реализовать настройку и вывод атрибутов качества и аннотации. [#29982](#)
- Разрешить запросы агрегирования `count` для нечисловых типов данных. [#31188](#)
- Расчёт и рендер функций перенесён в сервис.
- Добавлен пакет `Zyfra.AspNet.Configuration.Serilog` для реализации структурированного лога.
- Не возвращаем открытое событие с флагом `opened`, если у него есть дата завершения в будущем. [#31389](#)
- Обновить пакет с метриками до 7.1.3+ (или 6.2.5 для шестой версии), добавить `fallback` хэндлер в сервисы. [#32259](#)
- Использовать режим извлечения `Auto` для функции `ZfSampDat()`. [#32650](#)

Исправлено:

- Привести `swagger` при сериализации `TimeSpan` к нужному виду. [#31805](#)
- `ZfSqlQuery` возвращает ошибку при изменении массива вывода. [#32405](#)
- Панель функции событий в `Excel` обрезана. [#32433](#)
- Модальное окно поиска свойств обрезано. [#32434](#)
- Не рассчитывается значение функции, если ее аргументом выступает значение другой функции `Datalink`. [#32327](#)
- `ZfSqlQuery` перезаписывает стили и форматы ячеек. [#32445](#)
- Сбивается конфигурация `ZfSqlQuery` при повторном нажатии на ячейку с результатом. [#32501](#)

- Поиск свойств по моделям и операциям выводит пустые объекты без свойств. [#32641](#)

## 3.4.9. zif-datalink

[1.7.0] - 2023-11-13

### Добавлено:

- Внесены изменения контрактов для добавления `usePreview` и `modeOfProvision`. [#27894](#)
- Добавлен метод возврата событий на основе `CRON` выражения. [#27776](#)
- Добавлен атрибут `name` в параметры в метаданных версии. [#28503](#)
- Добавлен атрибут `referenceType` в определение параметров. [#28144](#)
- Создать методы работы с корзиной. [#27571](#)
- Скрипты мигрированы в файловое хранилище. [#29800](#)
- Добавлен список прав в возвращаемые данные при включенном `ABAC`. [#26858](#)
- Выполнение генерации по расписанию в указанном часовом поясе. [#30135](#)
- Параметры типа `timestamp` могут принимать значение в виде выражения `ZifTime`. [#30137](#)
- Обновлён пакет `zbk-pkg-aspnet-configuration-metrics` до версии 7.1.2. [#27735](#)
- Обновлён пакет `zbk-pkg-aspnet-configuration-postgres` до версии 7.4.2. [#31197](#)
- Перейти на актуальные методы `UDL`. [#25153](#)
- Реализовать настройку и вывод атрибутов качества и аннотации. [#29982](#)
- Изменить механизм удаления пустых строк. [#31128](#)
- Разрешить запросы агрегирования `count` для нечисловых типов данных. [#31188](#)
- Расчёт и рендер функций перенесён в сервис.
- Добавлен `Fallback handler` в сервис. [#32259](#)

### Изменено:

- Изменен механизм работы с расписаниями для версий. [#29471](#)
- Не возвращаем открытое событие с флагом `opened`, если у него есть дата завершения в будущем. [#31389](#)
- Использовать режим извлечения `Auto` для функции `ZfSampDat()`. [#32650](#)

### Исправлено:

- Ошибка из-за синтаксиса формулы. [#31735](#)
- Редактор шаблонов `Datalink`. Исправить расширения файлов. [#31673](#)
- Ошибка при удалении версии. [#32502](#)
- При добавлении шаблона через `swagger` теперь создаются и метаданные. [#32796](#)

### 3.4.10. zif-reporting

[5.7.0] - 2023-11-13

Изменено:

- DevExpress обновлен до 22.1.11. [#30871](#)
- Обновлён пакет `zbc-pkg-aspnet-configuration-postgres` до версии 7.4.2. [#31197](#)
- Привести `swagger` при сериализации `TimeSpan` к нужному виду. [#31805](#)

### 3.4.11. zif-dashboards

[2.4.0] - 2023-11-13

Изменено:

- DevExpress обновлен до 22.1.11. [#30871](#)
- Обновлён пакет `zbc-pkg-aspnet-configuration-postgres` до версии 7.4.2. [#31197](#)
- Обновить пакет с метриками до 7.1.3+ (или 6.2.5 для шестой версии), добавить `fallback` хэндлер в сервисы. [#32259](#)

### 3.4.12. zif-propertyset

[1.4.0] - 2023-11-13

Изменено:

- Обновлён пакет `zbc-pkg-aspnet-configuration-metrics` до версии 7.1.2. [#27735](#)
- Обновлён пакет `zbc-pkg-aspnet-configuration-postgres` до версии 7.4.2. [#31197](#)
- Обновить пакет с метриками до 7.1.3+ (или 6.2.5 для шестой версии), добавить `fallback` хэндлер в сервисы. [#32259](#)

Добавлено:

- В метод `api/categories` добавлен параметр `categoryNameMask`. [#28138](#)

## 3.5. Расчёты

### 3.5.1. zif-bp-calculate-specification-worker

[2.0.2] - 2023-10-23

Исправлено:

- **BREAKING:** при отсутствии в результате дополнительных атрибутов не работает закрытие события. [#31558](#)



## 3.5.2. zif-cm-context-functions

[1.2.1] - 2023-11-20

Добавлено:

- **BREAKING:** функция `val` некорректное значение по умолчанию при `arcMode: After` и `AtOrAfter`. [#32312](#)
- **BREAKING:** поддержка типа `float`. [#30546](#)

Изменено:

- Обновлена версия базового образа. [#31370](#)
- Библиотека `Zyfra.AspNetCore.Configuration.Metrics` обновлена до версии 7.1.4, добавлен `fallback` хэндлер. [#32255](#)

## 3.5.3. zif-cm-engine-mvel

[2.2.0] - 2023-11-16

Добавлено:

- **BREAKING:** поддержка типа `float`. [#30546](#)
- **BREAKING:** увеличено значение по умолчанию для максимального размер заголовка HTTP и GRPC запросов. [#33499](#)
- Переменные:
  - `SERVER_MAX_HTTP_HEADER_SIZE` максимальный размер заголовка HTTP запроса.
  - `GRPC_NETTY_SERVER_MAX_INBOUND_METADATA_SIZE` максимальный размер заголовка GRPC запроса.

Изменено:

- Рефакторинг проекта. [#24627](#)
- Обновлен базовый образ `docker.idp.yc.ziiot.ru/devops/jre19-jammy`. [#31457](#)

Исправлено:

- Функция `Fn.val`, некорректная работа с `DATETIME` свойствами. [#28842](#)
- Встроенное логирование `Tomcat`, неучитывающее настройки `zbc-pkg-spring-structured-logging`. [#24627](#)
- Преобразование строковых параметров к подходящему типу значения. [#30169](#)

## 3.5.4. zif-cm-metadata

[5.2.0] - 2023-11-20

Добавлено:

- **BREAKING:** функциональность массового импорта расчетов и их перезапуска. [#26214](#)

- **BREAKING:** уникальный индекс в базе данных, предотвращающие появление нескольких активных конфигураций расчетов. [#25662](#)
- **BREAKING:** поддержка типа float. [#30546](#)
- **BREAKING:** добавлена поддержка TLS соединения с PostgreSQL. [#30950](#)
- Преобразование нетипизированных констант к конкретному типу параметра. [#30169](#)
- **BREAKING:** добавлена поддержка коэффициента преобразования для агрегатов за период. [#32361](#)
- Заполнение необходимых AssemblyInfo атрибутов при сборке. [#28903](#)
- Обновлены пакеты, необходимые для полей аудита. [#28916](#)
- В некоторые GET запросы добавлены новые поля для фильтрации задач расчета/перерасчета по датам во временных промежутках. [#31191](#)
- Преобразование нетипизированных констант к конкретному типу параметра. [#30169](#)

Изменено:

- **BREAKING:** Некоторые настройки соединения с RabbitMQ сделаны конфигурируемыми через переменные окружения. [#33393](#)
- Библиотека Zyfra.AspNet.Configuration.Metrics обновлена до версии 7.1.4, добавлен fallback хэндлер. [#32255](#)
- Для расчета Агрегации за период id свойства агрегации берётся из коллекции переменных, а не из отдельного поля конфигурации свойства. [#30864](#)
- В логах сервиса скрыты пароли. [#33543](#)

Исправлено:

- **BREAKING:** при вычислении агрегата за период исключается левая граница периода. [#32587](#)

## 3.5.5. zif-om-testspecification

[3.3.0] - 2023-11-20

Добавлено:

- **BREAKING:** добавлена поддержка TLS соединения с PostgreSQL. [#30950](#)
- Заполнение необходимых AssemblyInfo атрибутов при сборке. [#28903](#)
- Обновлены пакеты, необходимые для полей аудита. [#28916](#)

Изменено:

- Библиотека Zyfra.AspNet.Configuration.Metrics обновлена до версии 7.1.4, добавлен fallback хэндлер. [#32255](#)

## 3.5.6. zui-app-calculation-service

[1.4.0] - 2023-11-15

Исправлено:

- **BREAKING:** Ошибка загрузки конфигурации расчета. [#31997](#)
- Исправления дизайна. [#29020](#)
- Поведение таблицы после действий. [#32416](#)
- Показ времени периодической конфигурации. [#32161](#)
- Высота таблицы версий расчета. [#32422](#)
- Путь агрегации за период. [#32795](#)

Добавлено:

- Логгер платформы. [#27744](#)

Изменено:

- **BREAKING:** обновлен `Angular` до версии 15. [#26106](#)

## 3.6. Сбор данных

### 3.6.1. zif-interface-opcda

[4.16.1] - 2023-11-17

Исправлено:

- Некорректное преобразование `Float` в `Double`. [#32068](#)

Добавлено:

- Настройка подключения `OPC DA` интерфейса к источнику без авторизации. [#32884](#)

### 3.6.2. zif-interface-device

[2.4.0] - 2023-11-17

Изменено:

- Переход на `.net 7`. [#28592](#)
- Добавлены метаданные для сборок. [#30346](#)
- Убран `swagger` для служебных `endpoint`. [#30363](#)
- Исправлена генерация пары ключей при установке. [#32461](#)

### 3.6.3. zif-interface-opchda

[1.12.0] - 2023-11-17

#### Добавлено:

- Параметры конфигурации:
  - `SnapshotGetter:SnapshotUrl` — Url для получения текущих значений из БДВР (по умолчанию ``).
  - `SnapshotGetter:login` — Логин для базовой аутентификации при получении текущих значений (по умолчанию `admin`).
  - `SnapshotGetter:Timeout` — Timeout запросов для получения текущих значений (по умолчанию `00:00:05`).
  - `SnapshotGetter:password` — Пароль для базовой аутентификации при получении текущих значений (по умолчанию ``).
  - Настройка подключения интерфейса к источнику без авторизации. [#32884](#)

#### Изменено:

- Определение глубины чтения предыстории с учетом временных меток текущих значений из БДВР. [#27124](#)
- Значение по умолчанию для параметра конфигурации `recoveryMaxTime` на `2.00:00:00`. [#27124](#)
- Версия пакета `Zif.Interface.Core`.

#### Удалено:

- Поддержка режима «горячего» резервирования. [#27124](#)

#### Исправлено:

- Утечка памяти. [#30057](#)

### 3.6.4. zif-interface-opcua

[2.14.0] - 2023-11-17

#### Добавлено:

- Добавлены метаданные для сборок. [#30347](#)

#### Исправлено:

- В конце установки `Explorer` появляется ошибка. [#28806](#)
- Установка интерфейса вместе с `Explorer` приводит к ошибке при запуске интерфейса. [#30351](#)
- Функция записи доступна в `docker`-образе без поддержки записи. [#30750](#)
- Исправлена генерация пары ключей при установке. [#32461](#)

#### Изменено:

- Целевая платформа .NET Core 7.0. [#28594](#)
- Обновлена версия Workstation.UaClient до 3.2.1.

### 3.6.5. zif-data-emulator

[2.1.0] - 2023-11-17

Исправлено:

- Ошибка при выполнении валидации. [#30314](#)

Изменено:

- Добавлена пагинация для методов получения сессий импорта и воспроизведения. [#27087](#), [#29770](#)
- Добавлены метаданные для сборок. [#30349](#)

### 3.6.6. zif-datainput

[3.1.0] - 2023-11-17

Добавлено:

- Пакет zbk-pkg-aspnet-configuration-swagger 6.5.1 для стандартизации поддержки swagger. [#28944](#)
- Пакеты zbk-pkg-aspnet-common 7.0.0 и zbk-pkg-aspnet-configuration-routes 7.0.0 для отображения метаданных сервиса в полях аудита. [#28913](#)
- Переменная окружения MAX\_PAGE\_SIZE - максимальный допустимый размер страницы в запросах с пагинацией. [#30659](#)
- Переменная окружения KAFKA\_DEBEZIUM\_CONSUMER\_GROUP - группа консьюмеров для сообщений об изменениях в Объектной модели из Debezium (значение по умолчанию zif-datainput\_om\_changes). [#33490](#)
- Атрибуты сборки полях аудита. [#30343](#)

Изменено:

- Целевая платформа .NET Core 7.0. [#28565](#)
- В API сервиса коллекции объектов возвращаются с пагинацией. [#27082](#)
- Обновлены версии зависимых пакетов zbk-pkg-aspnet-abac-core 7.2.2, zbk-pkg-aspnet-configuration-health 7.1.2, zbk-pkg-aspnet-configuration-metrics 7.1.2, zif-bp-httpclient-pkg 2.3.1, zif-rtddb-common 5.0.23, zbk-pkg-aspnet-configuration-serilog 7.0.0, zbk-pkg-aspnet-configuration-tracing 7.0.0, zbk-pkg-aspnet-configuration-auth 7.0.1, zbk-pkg-aspnet-configuration-redis 7.2.1.
- Версия пакета zbk-pkg-aspnet-configuration-redis 7.2.2. [#34407](#)

Исправлено:

- В журнал валидации не выводятся ошибки по свойствам, не входящим в класс. [#27814](#)

- Множество созданных консьюмеров сервисом `zif-datainput`. [#33490](#)

### 3.6.7. zif-events-integration

[4.2.0] - 2023-11-17

Исправлено:

- Ошибка при обработке события, если тип события создан после старта сервиса. [#28997](#)
- Не работает фильтрация входящих событий по переменной `EVENT_SOURCE_ID_FILTER`. [#29326](#)

Изменено:

- Значение по умолчанию переменной `DEFAULT_EVENT_TYPE` на `Opс_Event` - новый тип событий. [#24580](#) Если переменная `DEFAULT_EVENT_TYPE` ранее не переопределялась, то все события с типом по умолчанию относились к типу события `тип4`. Если необходимо сохранить данное поведение, нужно явно указать для данной переменной значение `тип4`, в противном случае, начиная с данной версии сервиса, все события с типом по умолчанию будут относиться к типу события `Opс_Event`.
- Значение по умолчанию переменной `EVENT_SOURCE_ID_FILTER` на ````. [#29326](#)

Добавлено:

- Автоматическое создание типов событий. [#24580](#)

### 3.6.8. zif-export-nifi-collectors

[1.4.0] - 2023-11-17

Изменено:

- Добавлена пагинация для методов получения сессий импорта/экспорта. [#27091](#), [#29768](#)
- Формирование сообщений аудита в `CEF` формате. [#30356](#)

Исправлено:

- Проверка связи с одним и тем же `url nifi`, но разными типами авторизации. [#30289](#)
- Поддержка типа проблем при импорте `EXTERNAL_VARIABLES`. [#29838](#)

### 3.6.9. zif-interface-manager

[1.9.0] - 2023-11-17

Добавлено:

- Зависимость от пакета `Zyfra.AspNetCore.Configuration.Routes`. [#30354](#)

Изменено:

- Обновлена версия `Zyfra.AspNetCore.Configuration.Serilog` до 7.0.0. [#28589](#)
- Обновлена версия `Zyfra.AspNetCore.Configuration.Tracing` до 7.0.0. [#28589](#)
- Обновлена версия `Zyfra.Abac.Core` до 7.2.6. [#28677](#)

- Обновлена версия `Zyfra.AspNet.Configuration.Metrics` до 7.1.2. [#28589](#)
- Обновлена версия `Zyfra.Abac.Abstractions` до 7.1.1. [#28589](#)
- Обновлена версия `Zyfra.AspNet.Configuration.Postgres` до 7.4.2. [#30988](#)
- Обновлена версия `Zyfra.AspNet.Configuration.Auth` до 7.0.1. [#28589](#)

Исправлено:

- Некорректно заполняются поля аудита. [#29261](#)
- В методе `POST /api/v1.0/Agents/{id}/instance` при попытке создать экземпляр недоступного интерфейса возвращается код ответа 404, а должен 403. [#28677](#)

### 3.6.10. zif-opcua-server

[1.0.0] - 2023-11-17

Добавлено:

- Реализация OPC UA сервера `ziiot`.
- Построение дерева тегов, получение атрибутов тегов.
- Получение текущих значений тегов.
- Получение исторических значений тегов.
- Получение агрегированных значений тегов.
- Поддержка записи значений тегов.
- Подписка на изменение значений тегов.
- Упрощенная ABAC авторизация на выполнение операций чтения/записи.

### 3.6.11. zif-universal-datamart

[4.2.0] - 2023-11-17

Изменено:

- Добавлена пагинация для методов. [#27090](#)

### 3.6.12. zui-app-datainput

[3.3.0] - 2023-11-17

Изменено:

- Актуализированы методы API с учетом пагинации. [#29449](#)

Добавлено:

- Поддержка ограничения ручного ввода в листах по требованию. [#22945](#)

### 3.6.13. zui-app-interface-manager

[3.0.1] - 2023-11-17

Исправлено:

- Устранены уязвимости. [#33558](#)

### 3.6.14. zui-app-nifi

[5.0.0] - 2023-11-17

Изменено:

- Переход на родные группы NiFi в Управлении коллекторами. [#23716](#)
- Переход на Angular 14. [#21886](#)
- Обновление приложения согласно новым макетам. [#24258](#)

### 3.6.15. zui-app-universal-datamart

[1.6.0] - 2023-11-17

Изменено:

- Актуализированы методы API с учетом пагинации. [#28972](#)

Исправлено:

- Обозначения обязательных полнй формы свойства. [#28955](#)

## 3.7. Хранение данных

### 3.7.1. zif-bp-\*

[2.10.1] - 2023-10-03

Добавлено:

- Заполнение AssemblyInfo атрибутов при сборке образа. [#28901](#)

Изменено:

- Изменен базовый образ. [#26535](#)

[2.10.0] - 2023-08-08

Изменено:

- Пакет Zyfra.Bp.WorkerHelperPkg поднят до 1.3.0.



## 3.7.2. zif-document-archive

[1.3.0] - 2023-10-23

### Добавлено:

- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Заполнения полей аудита и формирования `CEF` сообщений. [#28914](#)

### Изменено:

- Актуализированы `api`-клиенты `OM`. [#17841](#)
- Изменен базовый образ. [#26535](#)
- Пакет `Zyfra.AspNetCore.Configuration.Serilog` поднят до версии 7.2.0.
- Пакет `Zyfra.AspNetCore.Configuration.Auth` поднят до версии 7.0.1.
- Пакет `Zyfra.AspNetCore.Configuration.Health` поднят до версии 6.2.2.
- Пакет `Zyfra.AspNetCore.Configuration.Metrics` поднят до версии 7.1.4. [#32279](#)
- Пакет `Zyfra.AspNetCore.Configuration.Postgres` поднят до версии 7.4.2.
- Пакет `Zyfra.AspNetCore.Configuration.Redis` поднят до версии 7.2.1.
- Пакет `Zyfra.AspNetCore.Configuration.Routes` поднят до версии 7.0.0.
- Пакет `Zyfra.AspNetCore.Configuration.Swagger` поднят до версии 6.5.3.
- Пакет `Microsoft.AspNetCore.Authorization` поднят до версии 7.0.11.
- Пакет `Zyfra.Abac.Abstractions` убран из зависимостей.
- Пакет `Microsoft.AspNetCore.HeaderPropagation` убран из зависимостей.

### Исправлено:

- Ошибка 400 при редактировании документа. [#28734](#)
- Сущности в `OM` не создавались при миграции из-за ответа `HttpCode 410`. [#31992](#)

## 3.7.3. zif-events

[5.9.1] - 2023-11-20

### Добавлено:

- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Поддержку `TLS` при подключении к `PostgreSQL`. [#31504](#)

### Изменено:

- Пакет `Microsoft.AspNetCore.SignalR.StackExchangeRedis` обновлен до версии 6.0.21.
- Изменен базовый образ. [#26535](#)

- Обновлены пакеты для заполнения полей аудита и формирования CEF сообщений. [#28914](#)
- Пакет `Zyfra.AspNet.Configuration.Metrics` обновлен версии 7.1.4. [#32275](#)

Исправлено:

- Ошибка при выгрузке таблицы. [#28971](#)
- Ошибка при выгрузке таблицы. [#30553](#)
- Сортировка данных в методе `api GET /partitions`.

[5.9.0] - 2023-08-15

Изменено:

- **BREAKING:** методы `POST /restapi/partitions` и `DELETE /restapi/partitions` помечены как `obsolete` и будут удалены в версии платформы 2.17.0.
- Nuget `Microsoft.Extensions.DependencyInjection.Abstractions` - версия повышена до 7.0.0.
- Nuget `Zyfra.AspNet.Configuration.Tracing` - версия повышена до 7.0.0.
- Базовый образ изменен с `mcr.microsoft.com/dotnet/aspnet:6.0-alpine3.16` на `docker.idp.yc.ziiot.ru/devops/dotnet-aspnet6.0-alpine:latest`.

Добавлено:

- Инъекцию `IMetricsService` в сервис для считывания ожидания времени запросов к `postgres, redis, kafka, abac`.
- Поддержка TLS и аутентификации Redis. [#24943](#)
- Автоматическое создание партиций. [#23439](#)
- Метод API `GET /restapi/partitions/createNextPartitionsGroup`. [#23439](#)
- Добавлены переменные окружения `AUTO_PARTITIONS_ENABLED`, `AUTO_PARTITIONS_PERIOD`, `AUTO_PARTITIONS_INDENT_DAY`, `AUTO_PARTITIONS_CHECK_PERIOD_MIN`, `AUTO_PARTITIONS_PERIOD_AHEAD_DAY`, `AUTO_PARTITIONS_LOCK_EXPIRY_SEC`. [#23439](#)

Исправлено:

- Отсутствие валидации дат в некоторых методах API. [#25992](#)

### 3.7.4. zif-file-storage

[1.2.7] - 2023-11-20

Изменено:

- Базовый образ.

[1.2.6] - 2023-11-17

Добавлено:

- Заполнение полей аудита. [#28914](#)

## [1.2.5] - 2023-09-29

### Добавлено:

- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Параметр `AUTH_ANONYMOUS_ACCESS_ENABLED` для включения/отключения анонимного доступа. [#27805](#)

### Изменено:

- Изменен базовый образ. [#26535](#)
- Версия пакета `Zyfra.AspNetCore.Configuration.Auth` поднята до 7.0.1.
- Версия пакета `Zyfra.AspNetCore.Configuration.Health` поднята до 6.2.2.
- Версия пакета `Zyfra.AspNetCore.Configuration.Routes` поднята до 7.0.0.
- Версия пакета `Zyfra.AspNetCore.Configuration.Kafka` поднята до 6.0.1.
- Версия пакета `Zyfra.AspNetCore.Configuration.Serilog` поднята до 7.2.0.
- Версия пакета `Zyfra.AspNetCore.Configuration.Metrics` поднята до 7.1.4.
- Версия пакета `Zyfra.Abac.Core` поднята до 7.2.6.
- Версия пакета `Zyfra.Abac.Abstractions` поднята до 7.1.1.

## 3.7.5. zif-om-sqldatasource

### [1.9.0] - 2023-11-20

### Добавлено:

- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Обработчик ошибок БД.

### Изменено:

- Изменен базовый образ. [#26535](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-routes` до 7.0.0. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-auth` до 7.0.1. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-serilog` до 7.2.0. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-abac-core` до 7.2.6. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-abac-metrics` до 7.1.4. [#32277](#)
- Обновлен пакет `Npgsql` до 7.0.6. [#31507](#)

## 3.7.6. zif-quality-service

[1.4.2] - 2023-11-20

Добавлено:

- Описания таблиц и столбцов базы данных. [#33013](#)

[1.4.1] - 2023-10-23

Добавлено:

- Добавление `DefaultPolicyId`. [#22404](#)
- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Добавлен пакет `zbc-pkg-aspnet-configuration-postgres` 7.4.2 [#31503](#)

Изменено:

- Обновлен пакет `zbc-pkg-aspnet-configuration-metrics` до 7.1.1. [#27729](#)
- Изменен базовый образ. [#26535](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-routes` до 7.0.0. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-auth` до 7.0.0. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-configuration-serilog` до 7.1.0. [#28914](#)
- Обновлен пакет `zbc-pkg-aspnet-abac-core` до 7.2.0. [#28914](#)
- Переменная `POSTGRES_TLS` переименована в `POSTGRES_TLS_MODE`. [#31503](#)
- Пакет `Zyfra.AspNet.Configuration.Metrics` обновлен версии 7.1.4. [#32282](#)

[1.4.0] - 2023-02-27

Добавлено:

- Реализовать ABAC-авторизацию. [#20920](#)
- Аудирование. [#22854](#)

Изменено:

- Приведение базовых образов сервисов к единым. [#19301](#)

## 3.7.7. zif-rtdb-data, zif-rtdb-background-services

[2.11.0] - 2023-11-20

Добавлено:

- Поддержка TLS при подключении к Cassandra. [#31973](#)
- Библиотека `Zyf.AspNet.Configuration.Cassandra` версии 7.0.0. [#31973](#)
- Создание задач на расчет предварительных агрегаций для текущих значений. [#27278](#)

- Создание задач на перерасчет предварительных агрегаций. [#27279](#)
- Расчет предварительных агрегаций для тегов, у которых уже был включен атрибут Рассчитывать предагрегации. [#29436](#)
- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Параметр `CONVERSION_FACTOR_MODE` для смены поведения `ConversionFactor` в `GET /api/v3/point-values/summary` [#31402](#)

Изменено:

- **BREAKING:** новый формат сообщения для топиков `metadata-service-updates`, `metadata-service-updates-dict`. [#27284](#)
- **BREAKING:** изменена модель ответа для запросов, завершившихся ошибкой. Для обратной совместимости нужно установить настройку `ERROR_RESPONSE_TYPE=Text`. [#26843](#)
- **BREAKING:** `ConversionFactor` в `GET /api/v3/point-values/summary` по умолчанию работает в режиме `All` и применяется к значениям `Minimum,Maximum,Average`. [#31402](#)
- **BREAKING:** в `api v2` при выводе значения тега типа `datetime` добавлен указатель на универсальную временную зону. [#32509](#)
- Обновлена библиотека `Zyfra.Abac.Core` до версии 7.2.6. [#28914](#)
- Изменен базовый образ. [#26535](#)
- Обновлены пакеты для заполнения полей аудита. [#28914](#)
- В `api v2` добавлен параметр `DateTimeValueFormat` для определения формата времени: `OldStandard` и `Iso8601`.

Исправлено:

- Расчет исторических предварительных агрегаций левее самого первого значения тега. [#29967](#)
- `Health check` проба во время выполнения миграций для `Cassandra`. [#30468](#)
- Инвалидация кеша после переименования тега. [#30237](#)
- Наименование ключей в распределенном кеше. [#31369](#)
- Интерполяция событий со значением `null` и качеством `Bad`. [#31092](#)
- Учёт правой границы при построении графиков. [#32298](#)
- Ответ для групповых операций в случаи отсутствия тега, переданного в запросе. [#31829](#), [#32408](#)
- Интерполяция на временные метки будущего времени. [#32943](#)

## 3.7.8. zif-rtdb-metadata

[3.7.1] - 2023-11-22

Исправлено:

- Ошибка с кодом 500 при выгрузке тегов с разным набором атрибутов. [#34027](#)

- Смещение столбцов при выгрузке тегов с разным набором атрибутов. [#34027](#)

[3.7.0] - 2023-11-20

#### Добавлено:

- Переменная окружения `KAFKA_CONNECT_TIMEOUT_SEC`. [#27430](#)
- Экспорт и импорт описаний пользовательских атрибутов. [#25156](#)
- Отображение имен пользователей. [#29976](#)
- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Поддержка TLS для подключения к Postgres. [#31506](#)
- Использование библиотеки `Zyfra.AspNet.Configuration.Tracing` версии 7.0.0. [#31506](#)

#### Изменено:

- Рефакторинг импорта дискретных наборов. [#27430](#)
- Формат сообщения для топиков `metadata-service-updates`, `metadata-service-updates-dict`. [#27283](#)
- Код ответа метода API `POST api/v3/points/bulk/delete` при удачном удалении с 204 на 200 с массивом `id` удаленных тегов в теле ответа. [#30190](#)
- Изменен базовый образ [#26535](#)
- Обновлена библиотека `Zyfra.AspNet.Configuration.Serilog` до версии 7.2.0. [#31506](#)
- Обновлена библиотека `Zyfra.AspNet.Configuration.Auth` до версии 7.0.1. [#31506](#)
- Обновлена библиотека `Zyfra.AspNet.Configuration.Postgres` до версии 7.4.2. [#31506](#)
- Обновлена библиотека `Zyfra.Abac.Core` до версии 7.2.6. [#31506](#)
- Обновлена библиотека `Zyfra.AspNet.Configuration.Metrics` до версии 7.1.3. [#32276](#)
- Обновлены пакеты для заполнения полей аудита. [#28914](#)

#### Исправлено:

- Исправлено зависание при потере соединения с Kafka при импорте дискретных наборов. [#27430](#)
- Некорректный код ошибки возвращаемый методом `POST api/v3/points/export/csv` при запрете доступа к тегу. [#29268](#)
- Модель ошибки возвращаемая в `api v3`.
- Некорректный код ошибки при запрете тега на удаление в методе API `POST api/v3/points/bulk/delete`. [#30190](#)
- Поддержка АВАС в методах API `POST api/v1.0/points` и `DELETE api/v1.0/points`.

- Ошибка параметра `http_route` в метриках приложения. [#32276](#)
- Описание действий `Abac` для методов массового удаления тегов. [#32364](#)

### 3.7.9. zif-workflow

[2.11.0] - 2023-11-20

Добавлено:

- Возможность вести аудит действий пользователя (переменная `AUDIT_ENABLED`). [#28914](#)

Изменено:

- Начиная с версии `ZIIoT 2.18` прекращается поддержка `Camunda JavaDelegate` находящихся в `zif-workflow` (и библиотеке `zif-workflow-common`). Существующие `JavaDelegate` не будут дорабатываться под новую функциональность. Начиная с версии `2.16` и `2.17` необходимо адаптировать свои `BPMN` под `Camunda ExternalTasks` (сервисы `zif-bp*`), которые являются рекомендованным способом взаимодействия `BPMN` с сервисами `ZIIoT`.
- Использует структурированное логирование [#28914](#)

[2.10.9] - 2023-11-10

Добавлено:

- Добавлена возможность задавать режим `TLS` при работе с `POSTGRES` (переменная `POSTGRES_TLS_MODE`). [#31497](#)

Изменено:

- Изменен базовый образ. [#33190](#)

Изменено:

- Ошибка запуска сервиса в режимах `TLS: verify-ca, verify-full`. [#33482](#)

### 3.7.10. zif-workflow-relocator

[1.1.1] - 2023-10-24

Добавлено:

- Заполнение `AssemblyInfo` атрибутов при сборке образа. [#28901](#)
- Добавлен пакет `Zyfra.AspNet.Configuration.Auth 7.0.1`. [#28914](#)
- Добавлен пакет `Zyfra.AspNet.Configuration.Routes 7.0.1`. [#28914](#)

Изменено:

- Пакет `Zyfra.Bp.HttpClient` поднят до `2.4.0`.
- Пакет `Zyfra.AspNet.Configuration.Metrics` поднят до `7.1.6`.
- Пакет `Zyfra.AspNet.Configuration.Tracing` поднят до `7.0.0`.
- Изменен базовый образ. [#26535](#)

- Пакет `Zyfra.Abac.Core` поднят до 7.2.6. [#28914](#)
- Пакет `Zyfra.AspNetCore.Configuration.Serilog` поднят до 7.2.0. [#28914](#)

### 3.7.11. `zui-app-workflow`

[3.2.0] - 2023-10-31

Изменено:

- Обновлено приложение до Angular 16. [#29671](#)

Исправлено:

- Удаление развертываний. Нет множественного выбора. [#25232](#)
- Смена локализации (RU-EN). Создание развертывания не выполнена смена локализации. [#25642](#)

[3.1.1] - 2023-10-30

Исправлено:

- Не видно подсказку при превышении длины в поле `TenantId`. [#21916](#)
- Редактор диаграмм: длинное название `bpmn` файла ломает UI. [#30847](#)
- Не отображаются свойства активити при редактировании и создании рабочего процесса. [#31186](#)

### 3.7.12. `zui-app-event-types`

[2.0.2] - 2023-10-30

Изменено:

- Обновлен базовый образ. [#33197](#)

Добавлено:

- Добавлено логирование. [#27743](#)

[2.0.1] - 2023-10-09

Изменено:

- Обновлен базовый образ.

### 3.7.13. `zui-app-rtdb`

[3.1.0] - 2023-11-20

Изменено:

- Таблица переведены на Prizm. [#27848](#)

Добавлено:

- Добавлено отображение количества отфильтрованных тегов. [#24921](#)



- Добавлено отображение имен пользователей. [#29975](#)
- Добавлен вывод информации о создании и изменении тега. [#27065](#)
- Добавлено логирование. [#27743](#)

Исправлено:

- Исправлено отображение ошибки при создании атрибута. [#28531](#)
- Исправлено обновление дискретных наборов. [#28558](#)
- Исправлена фильтрация по Типу тега. [#28605](#)

## 3.8. UDL

### 3.8.1. zif-udl-services

[2.9.1] - 2023-11-23

Исправлено:

- **BREAKING:** `zif-udl-dfawebapi` зависает при поступлении одновременного большого количества запросов сразу после старта. [#34354](#)

[2.9.0] - 2023-11-20

Добавлено:

- Заполнения полей аудита и формирования CEF сообщений. [#28915](#)
- Процент «хороших» событий в агрегационных методах. [#29262](#)
- `AssemblyInfo` атрибуты сервисов. [#28902](#)
- Метрики `http_server_resource_waiting_duration_ms` в сервисах `zif-udl-dsentitywebapi`, `zif-udl-omconfigloader`, `zif-udl-rtdwebapi`. [#27730](#)
- Автоматическое получение значений из справочников. [#28060](#)
- Метод с пагинацией для получения тегов в `zif-udl-mdswebapi`. [#30905](#)
- Параметр `conversionFactor` для методов агрегаций в сервисе `zif-udl-dfawebapi`. [#30822](#)
- Групповой метод запроса данных для невременных рядов в `zif-udl-dfawebapi v2`. [#30104](#)
- `Fallback handler` в сервисы UDL. [#32254](#)
- Метод очистки кеша для заданного источника данных в сервисе `zif-udl-dsentitywebapi`. [#32683](#)

Изменено:

- **BREAKING:** В Платформе 2.17.0 настройка конфигурации источников данных будет перенесена из PostgreSQL в переменные окружения. [#25481](#)
- Сервис `zif-udl-dfawebapi` запрашивает права у `zif-om-object`, только если переменная `ABAC_ENABLED` равна `true`. [#24879](#)

- Названия очередей RabbitMQ для обработки команд вынесены в настройки. [#32538](#)
- Используются базовые Docker-образы Ubuntu Jammy вместо Alpine. [#33121](#)
- Все очереди RabbitMQ используемые UDL, включая очереди команд, учитывают параметр RABBITMQ\_QUEUE\_EXPIRES\_TIMEOUT\_SEC. [#32772](#)

Исправлено:

- **BREAKING:** в сервисе `zif-udl-dsentitywebapi` удален метод `/restapi/MirrorManager/{uid}/rebuild`. При очистке Redis конфигурация системных тегов не восстанавливается. [#33944](#)
- Запрос агрегаций с параметром `All` для свойства невременного ряда возвращает `null`. [#30099](#)
- Перестают работать подписки при большом количестве клиентов. [#28142](#)
- Не возвращаются метки времени для `Min/Max` агрегаций. [#30405](#)
- При массовых подписках достигается лимит подключений к RabbitMQ. [#28142](#)
- Не работает запрос с множеством агрегационных типов. [#30930](#)
- Не возвращается аннотация при запросе `Min/Max` агрегаций. [#30914](#)
- При запросе значения из ссылочного справочника, возвращается `id` значения. [#31102](#)
- Не возвращается `statusCode` при получении `plot` значений. [#31516](#)
- Параметр `plotType` требует обязательного заполнения. [#31766](#)
- Сервис `zif-udl-mdswebapi` не возвращает дискретный набор для справочников. [#31679](#)
- Сервис `zif-udl-mdswebapi` в ответе не выводит нулевые миллисекунды. [#30933](#)
- Кластерный узел UDL остаётся запущенным при потере связи с `Seed` и деактивации. [#31346](#)
- Кластер UDL не всегда восстанавливается после переразвертывания/обновления. [#33270](#)
- Неверный ответ запроса сервиса `zif-udl-dfawebapi` при записи нескольких массивов значений на одно свойство. [#33196](#)
- Ошибка `pipelining of requests forbidden` в сервисе `zif-udl-rtdwebapi`. [#33567](#)
- При большом количестве запросов на подписку часть завершается с ошибкой `An item with the same key has already been added`. [#33829](#)

## 4. Поддержка встроенной (native) OpenID Connect аутентификации для Apache NiFi

### 4.1. Два типа аутентификации для NiFi

Касательно Apache NiFi, поставляемом в составе Платформы, начиная с релиза 2.13.0 произошли следующие существенные изменения:

- Повышение поставляемой версии Apache NiFi до 1.19.1.
- Добавлена поддержка native-аутентификации пользователей через KeyCloak, с использованием протокола OpenId Connect.
- Добавлена возможность генерации TLS-сертификатов для узлов NiFi при развертывании, что необходимо для включения native-аутентификации.

В результате этих доработок, Платформы теперь поддерживает два варианта аутентификации и авторизации пользователей для NiFi:

1. Аутентификация с использованием OAuth Proxy (oauth-proxy), этот режим существовал и в предыдущих версиях.
  - Плюсы: проще в настройке: не требует сертификатов на узлах NiFi, не требует TLS на узлах NiFi, не требует создания учетных записей в NiFi.
  - Минусы: нет полноценной авторизации (разделения прав доступа) - любой пользователь, который сможет аутентифицироваться через OAuth Proxy в NiFi будет являться полным администратором и фактически на уровне NiFi нет разделения учетных записей пользователей (а сама возможность аутентификации на proxy ограничивается одной общей группой в KeyCloak).
2. Аутентификация NiFi напрямую в OpenID Connect Provider (в KeyCloak), она же native-аутентификация
  - Плюсы: с точки зрения NiFi пользователи обладают отдельными полноценными учетными записями, которым можно предоставлять и ограничивать права средствами самого NiFi, все действия в NiFi реализуются от имени конкретной (а не общей) учетной записи.
  - Минусы: требуется включение TLS на узлах NiFi, что в свою очередь требует выдачи и обновления TLS-сертификатов узлов. Так же требуется создание и управление учетными записями в NiFi (фактически надо поддерживать два согласованных набора учетных записей для пользователей - в KeyCloak и отдельно в NiFi).

### 4.2. Генерация сертификатов для узлов NiFi

Для поддержки аутентификации пользователей средствами самого NiFi (а не внешними вроде oauth-proxy), узлы Apache NiFi должны быть сконфигурированы в т.н. защищенный кластер и на них должен быть включен TLS.

Для работы TLS необходимо обеспечить каждый узел кластера NiFi собственным уникальным закрытым ключом и TLS-сертификатом.

Для решения этой задачи в систему установки Платформы начиная с версии 2.13.0 добавлен инструментарий PKI, позволяющий генерировать сертификаты на этапе развертывания для узлов некоторых сервисов инфраструктуры, включая NiFi. Инструментарий поддерживает два режима работы:

- Генерация сертификатов силами решения `cert-manager`, который должен быть предварительно установлен в кластере. В этом случае `zifctl` создает Namespace-level Issuer (если нужно) и создает объекты `certificate` с необходимыми параметрами. Это рекомендованный метод создания и управления сертификатами.
- Генерация сертификатов силами самого `zifctl`: в этом случае на этапе создания конфигурации создается RootCA-сертификат, уникальный для данной инсталляции, и на этапе развертывания `zifctl` выпускает необходимые TLS-сертификаты и самостоятельно загружает их в объекты `secret` кластера `Kubernetes`. Обновление сертификатов выполняется при очередных обновлениях платформы (`zifctl sync`).

### 4.3. Управление пользователями в NiFi

Если используется `oauth-proxy` аутентификация, то управление пользователями в NiFi невозможно: все пользователи с точки зрения Apache NiFi являются анонимными с полными административными правами (фактически в самом NiFi аутентификация и авторизация отключены). Администратор платформы может только ограничить список тех, кто может зайти в NiFi с помощью специальной роли в KeyCloak: `apache-nifi-client:nifi-admin` (`client role`).

Если используется `native` аутентификация, то пользователи будут представлены в NiFi полноценными индивидуальными учетными записями, которым можно назначить права и ограничения средствами самого NiFi. Но при это Apache NiFi не предоставляет штатных средств синхронизации этих учетных записей с провайдером аутентификации. Для логина пользователя с использованием KeyCloak необходимо:

- Создать учетную запись пользователя в KeyCloak (или синхронизировать ее из ActiveDirectory).
- Создать точно такую же учетную запись в NiFi (она должна совпадать с полем `email` УЗ в KeyCloak).
- Назначить права доступа учетной записи в NiFi (и/или включить ее в соответствующие группы в NiFi).
- При удалении учетной записи из KeyCloak, необходимо удалить и соответствующую учетную запись из NiFi.

После начального развертывания в NiFi будет создана только одна учетная запись (`initial admin identity`), соответствующая создаваемой `zifctl` учетной записи в KeyCloak `nifi-admin`.

## 5. Требования по проектированию приложений на платформе

### 5.1. Модальные глаголы

1. Необходимо (MUST), а также требуется (REQUIRED) и должно (SHALL) — абсолютное требование.
2. Недопустимо (MUST NOT), а также не должно, не допускается (SHALL NOT) — абсолютный запрет.
3. Следует (SHOULD), а также рекомендуется (RECOMMENDED) — требования, от которых можно отказаться при наличии разумных причин и обоснований.
4. Не следует (SHOULD NOT), а также не рекомендуется (NOT RECOMMENDED) — требования, нарушение которых возможно при наличии обоснований, но может вызвать проблемы.
5. Допускается, могут, возможно (MAY), а также необязательный (OPTIONAL) — возможности, предоставляемые платформой, использование которых допустимо, но не обязательно.

### 5.2. Требования

При редактировании документа новые требования должны добавляться в конец списков, чтобы не нарушать нумерацию требований. Требования могут удаляться.

#### 5.2.1. Моделирование

1. Все данные приложений, к которым необходим доступ другим приложениям, должны быть включены в Объектную Модель и доступны другим приложениям через сервисы OM/UDL.
2. Приложения должны использовать Единую Объектную Модель (ЕОМ), описывающую ресурсы и деятельность всего предприятия.
3. Следует проектировать разделяемые между приложениями модели ЕОМ. Допускается разделение данных ЕОМ на несколько областей применения, приложения могут создавать собственные области применения, если существующие не позволяют описать предметную область.
4. Приложениям следует использовать семантические сервисы OM, реализующие модели ISA-95, при автоматизации соответствующих процессов.
5. Взаимодействие с внешними относительно платформы системами (например, 1С, SAP) следует проектировать путем создания в платформе процессов переноса данных в/из внешних систем на базе Apache NiFi или других коннекторов к системе сбора данных платформы.
6. Следует рассмотреть перенос максимум данных из внешних источников в платформенные хранилища.
7. Возможно подключение внешних источников данных (SQL, Hive, GE Historian etc) через средства DataReference UDL/OM.
8. Не допускается хранить логику или расчеты приложения во внешних источниках данных.
9. Приложениям следует выполнять миграции данных в сервисах OM в соответствии с документом ADR Миграция данных приложений в объектную модель.

10. Для моделирования документов, их структуры, набора полей, связи с S3 хранилищем рекомендуется использовать сервис документов (`zif-document-archive`).
11. При моделировании объектов, выбор источника данных для свойств следует производить на основе сравнительного анализа гибкости конфигурации и производительности выборки данных.
12. При использовании источника данных Константа, следует принимать во внимание, что ее значение хранится в конфигурации свойства OM, что усложняет процесс переноса конфигураций OM между стендами. Константы рекомендуется использовать для свойств, значения которых не меняются с течением времени.
13. Приложения могут создавать адаптационные модели для упрощения выборки/агрегации данных из общих моделей EOM.

## 5.2.2. Хранение данных

1. Данные, для хранения которых в платформе есть специализированные сервисы (OM, БДВР, семантические модели ISA-95) следует хранить в соответствующих сервисах.
2. Временные ряды должны храниться в сервисах БДВР платформы.
3. Кэширование следует реализовывать с использованием Redis.
4. Приложениям не следует разворачивать самостоятельно сервисы, аналогичные предоставляемым платформой:
  - Реляционная БД — приложения могут использовать платформенную PostgreSQL.
  - БДВР — приложения могут использовать платформенную БДВР для хранения временных рядов связанных со свойствами OM.
  - Брокер сообщений — приложения могут использовать платформенный Apache Kafka\Schema registry, RabbitMq.
  - Неструктурированные данные/файлы — приложения могут использовать платформенное S3-совместимое хранилище доступное через s3-proxy сервис `zif-file-storage`.
5. Приложениям следует использовать сервисы UDL для получения WebSocket подписки на изменения значений свойств OM.
6. Приложения могут использовать шаблон издатель-подписчик для обмена данными с другими приложениями или сервисами платформы через брокеров сообщений Apache Kafka\RabbitMq:
  - Apache Kafka будет предпочтительней в случаях, когда требуется большая пропускная способность/поточная обработка передаваемых данных.
  - RabbitMq будет предпочтительней в сценариях, требующих маршрутизации/точной доставки сообщений контрактным подписчикам.
7. Для формирования витрин данных по моделям EOM/значениям свойств рекомендуется использовать сервис универсальной витрины данных — `zif-universal-datamart`.

## 5.2.3. Доступ к данным

1. Приложения должны обращаться к данным через сервисы OM/UDL/Events, либо через их публичные шины сообщений.
2. Сервисы не должны напрямую обращаться к платформенной СУБД Cassandra.

3. Сервисы не должны обращаться к тегам БДВР по имени, все обращения к данным должны быть через свойства ОМ и сервисы UDL.
4. Приложения не должны напрямую подключаться к внешним источникам данных.
5. Приложению должно получать доступ к справочникам НСИ через `zif-rdm-common` (ранее `zif-om-directories`) не зависимо от того, является ли мастер-системой для конкретного справочника платформа ZIIoT или внешний источник, данные из которого переносятся в `zif-rdm-common`.
6. Не допускается использование Apache NiFi для формирования бизнес логики приложений, выходящей за рамки ETL процессов или в случаях наличия специализированных сервисов.
7. В случае необходимости в взаимодействиях с внешними приложениями, которые по обоснованным причинам не могут быть реализованы в качестве приложений на платформе, рекомендуется разрабатывать отдельные сервисы-адаптеры.
8. В случае необходимости в взаимодействиях с внешними источниками данных, которые по обоснованным причинам не могут быть перенесены в платформу, рекомендуется разрабатывать отдельные сервисы-адаптеры.

### 5.2.4. Рабочие процессы, события и уведомления

1. При разработке бизнес-логики приложений, в том числе с участием пользователей, следует использовать нотацию BPMN и сервис рабочих процессов, обеспечивающих ее выполнение.
2. Производственные события, если они требуют регистрации, используются для интеграции с рабочими процессами/уведомлениями или другими решениями, должны храниться в сервисе событий (`zif-events`).
3. Следует использовать сервис уведомлений для рассылки уведомления о зарегистрированных событиях через выбранные каналы.
4. Следует использовать сервис уведомлений для рассылки уведомления о событиях внешних систем не зарегистрированных в сервисе событий.
5. Следует минимизировать создание скриптовых обработок внутри BPMN процессов и по возможности всегда использовать паттерн внешних задач для обработки и взаимодействия с данными (сравнение подходов).
6. Создание скриптовых обработок внутри BPMN допустимо в случаях:
  - отсутствия необходимых внешних обработчиков и невозможности их реализовать;
  - необходимости выполнить запрос с минимально возможной задержкой и небольшой длительностью выполнения.
7. Для взаимодействия с данными ZIIoT из BPMN процессов следует использовать внешние задачи (`external tasks`) исполняемые в сервисах `zif-bp-*`.

### 5.2.5. Расчеты

1. При необходимости расчетов следует использовать свойства объектной модели с источником данных Формула (`calctag`).
2. Для расчета формул из ОМ не следует использовать сервис потоковых расчетов `zif-sm-xxx` напрямую, в обход ОМ.
3. Допускается для генерации событий по спецификациям проверок использовать решения `EventGenUdl` на базе Apache Spark.

4. Допускается использовать кастомные расчеты на базе Apache Spark в тех случаях, когда функционал невозможно реализовать на сервисе потоковых расчетов.

## 5.2.6. Инфраструктура

1. Приложение должно быть рассчитано на работу внутри кластера платформы (Kubernetes 1.19+/OpenShift 4.0+) и не требовать для размещения отдельных виртуальных машин.
2. Приложение не должно требовать создания кастомных CRD или требовать предоставления прав уровня кластера для своего развертывания или функционирования.
3. Сервисы, составляющие приложение, должны поставляться как docker-образы на базе ОС Linux,
4. Докер образы сервисов должны быть собраны с учетом запуска от имени непривелигированного пользователя со случайным UID (см. требования по созданию docker-образов для [OpenShift](#)).
5. У сервисов должна быть версия в формате [SemVer2](#).
6. Сервисы приложений, доступные снаружи кластера (например, доступные из фронтенд) должны быть опубликованы только стандартным для Платформы образом с использованием Ingress-контроллера (Route-контроллера для OpenShift) кластера.
7. Не допускается делать доступным снаружи кластера сервисы приложений через незащищенный HTTP.
8. Приложение не должно зависеть от каких-либо сервисов, кроме платформенных или сервисов приложения, развернутых внутри кластера. Не допускается развертывать зависимости приложения в отдельных виртуальных машинах.

## 5.2.7. Разработка, конфигурация и запуск сервисов

1. В большинстве случаев рекомендуется создавать stateless сервисы, допустимо использование stateful-сервисов, в сценариях, когда наличие состояние значительно повышает эффективность решения, сервисы должны обеспечивать сохранение состояния в соответствующих Платформенных хранилищах (OM, PostgreSQL, Kafka, S3 storage и т. д.) и не терять данные во время перезапуска.
2. Сервисы должны получать настройки через переменные окружения контейнеров.
3. Если приложению необходимы для старта какие-то данные или конфигурацию в любых постоянных хранилищах, то приложение должно самостоятельно создать всю начальную конфигурацию в размере, достаточном для запуска и дальнейшей конфигурации средствами приложения.
4. Возможно требовать от пользователя/администратора после первого старта приложения дополнения конфигурации средствами приложения.
5. Для всех настроек, для которых это осмысленно, следует иметь значения, по умолчанию зашитые внутрь контейнера. Этим значениям рекомендуется подходить для большинства применений.
6. Настройки, зависящие от окружения, не должны зашиваться (в т.ч. IP-адреса, ссылки на домены, включая localhost, имена серверов СУБД, и тому подобное) и не должны иметь значения по умолчанию.
7. Сервисы должны поддерживать горизонтальное масштабирование, с балансировкой средствами сервисов Kubernetes в случае использования простых алгоритмов балансировки таких как round-robin, допускается использования собственных сервисов балансировки, в случаях, когда требуются специализированные алгоритмы, не поддерживаемые в Kubernetes.



8. Сервисы должны обрабатывать ситуацию временной недоступности зависимостей, ожидать и возобновлять работу при восстановлении их доступности (состояние работы сервиса должно отражаться в `readiness`-пробе, см. ниже).
9. Сервис, которому не заданы обязательные параметры при старте, должен сам завершить свою работу. В этом случае сервису следует выдать в лог детальное описание ошибки.
10. Если на старте сервиса он не смог подключиться к необходимым зависимостям (например, к БД), он должен либо самостоятельно завершить работу с выдачей на консоль детального описания ошибки, либо перейти в цикл попыток подключения к инфраструктуре и продолжить начальную инициализацию после появления доступа.
11. Сервис должен запускаться обладая только минимально необходимыми правами на инфраструктурные зависимости. Например, сервис не должен ожидать наличия прав на создание базы данных у той учетной записи, которая ему передана.
12. Переменные окружения, связанные с сервисами платформы (например, указывающий на инфраструктурные сервисы или сервисы платформы) следует иметь стандартные имена согласно документу Именованние переменных. `POSTGRES_XXX`, `AUTH_XXX`, `REST_XXX` и т.д. Это упростит развертывание приложения, т.к. можно использовать автоматически создаваемые `ConfigMap` Платформы для задания значений этих переменных, и облегчит в дальнейшем переход на платформенные средства развертывания через маркетплейс.
13. Приложения должны обеспечить работоспособность при откате версии сервисов на предыдущую, без отката схемы данных. При разработке приложения необходимо соблюдать обратную совместимость схемы хранения данных в рамках минорных версий сервисов.
14. Сервисам следует использовать `s3` хранилище для хранения файлов с структурированными и неструктурированными данными.
15. Сервисы не должны использовать файловую систему узла для хранения разделяемых между экземплярами сервиса данных.
16. Сервисы не должны предъявлять требования на использование `persistent volumes` с режимом `ReadWriteMany` при развёртывании в среде, так как используемый `class storage` может не поддерживать данный режим.
17. Корневая файловая система контейнера должна монтироваться в режиме только чтение. Для контейнера должны быть явно выделены ресурсы для записи через `Persistent Volumes Claims`.
18. Состав пробрасываемых в контейнер портов должен быть минимально достаточен для работы. Неиспользуемые в ходе работы контейнеризированного приложения порты не должны пробрасываться в контейнер.
19. Каждый процесс приложения должен быть запущен в отдельном контейнере, связанные контейнеры должны быть объединены в один `Pod`.

## 5.2.8. Мониторинг и наблюдаемость

1. Сервис должен предоставлять стандартные `health-check` пробы типов `liveness` и `readiness` по URL-путям `/health/liveness` и `/health/readiness` соответственно. Подробнее о семантике проб см. [документацию kubernetes](#) и требованиях по реализации проб.
2. Сервис может предоставлять пробу типа `startup` по URL `/health/startup`.
3. Сервис должен иметь `endpoint` для отдачи метрик в [формате Prometheus](#) по адресу `/metrics`.
4. Сервис должен отдавать базовые метрики ([для ASP.NET Core](#) и `runtime` или общий пакет) о состоянии рантайма (например, объем потребляемой памяти и т.д.).

5. Сервису следует отдавать специфические для сервиса метрики (например, количество вычитанных и обработанных событий).
6. Сервис должен выдавать свой лог на консоль, рекомендации по логированию описаны в документе [Технические рекомендации по реализации логирования в сервисах](#).
7. Сервису следует логировать все ключевые события (в том числе обработку запроса, разрешение доступа, отказ в доступе)
8. Сервису следует иметь возможность включения/отключения детализированного (отладочного) логирования через переменную окружения.
9. Сервисы должны поддерживать трассировку в формате `OpenTracing` (`Jaeger`).
10. Сервисы, содержащие HTTP API, должны предоставлять описание своего API в формате `OpenAPI/Swagger`.

### 5.2.9. Безопасность

1. Сервисы должны обеспечивать аутентификацию при любом обращении к API или фронтенду и только с помощью протокола `OpenID Connect`, включая аутентификацию при взаимодействиях между бекенд-сервисами (справочно: [Технические рекомендации по безопасности при микросервисной архитектуре](#)).
2. Frontend-сервисы, встроенные в портал, должны получать токен авторизации от портала.
3. Frontend-сервисы, не встроенные в портал или встроенные через `IFrame`, должны использовать процедуру `OpenID Connect Authorization Code Flow` для аутентификации пользователей.
4. Фронтенд-сервисы должны передавать полученный токен пользователя в любых запросах к бэкенд-сервисам.
5. Бэкенд-сервисы должны соблюдать принцип `identity propagation` (вызов всех http сервисов в цепочке обработки запроса должен осуществляться от имени исходного пользователя) – передавая токен входящего http-запроса во все исходящие запросы, совершенные в рамках конвейера обработки запроса. Токен входящего запроса также должен быть использован и для случаев, когда часть обработки идет асинхронно вне конвейера http-запроса, в этом случае токен рекомендуется передавать обработчику в виде части сообщения.
6. В случае, если сервис выполняет функциональность общего назначения, не связанный с действиями конкретного пользователя, то вызовы к другим сервисам следует производить от имени клиента сервиса (`oauth client`) используя поток получения токена доступа `*OpenID Connect Client Credentials Flow` (например, это могут быть фоновые расчеты или прогрев данных кэша и т.д.).
7. При реализации авторизации по любой модели, сервисы не должны использовать дополнительных Claim внутри JWT-токена, помимо стандартных для [JSON Web Token](#), `roles` и `user attributes`.
8. Приложения не должны ожидать, что `KeyCloak` будет подключен к конкретному провайдеру аутентификации (например, именно к `Active Directory`).
9. Приложениям рекомендуется использовать авторизация по модели ABAC (`Attribute-Based Access Control`) с использованием платформенного сервиса авторизации (`zif-security`).
10. Приложениям не рекомендуется (но возможно) использовать авторизацию по модели RBAC (`Role-Base Access Control`) на базе собственной ролевой модели приложения, основываясь на данных из JWT-токена.
11. При выборе модели авторизации следует руководствоваться ADR [Требования к аутентификации, авторизации и аудиту сервисов](#).

12. Сервисы должны обеспечивать возможность использования сертификата TLS для включения HTTPS соединения.
13. Docker образы приложений не должны содержать зарегистрированных CVE уязвимостей (Common Vulnerabilities and Exposures) уровня High и выше и строиться на актуальных версиях операционных систем и пакетов приложений.
14. Не допускается пробрасывать привилегированные порты (до 1024) в контейнер.
15. Не допускается наличие в образе контейнера файлов с полномочиями `setuid`, `setgid`.
16. Не допускается устанавливать и запускать ssh-сервер и прочие средства удаленного управления в контейнере.
17. Контейнер приложения должен быть максимально ограничен в части использования Linux Capabilities (например, обычно не требуются возможности `NET_ADMIN`, `SYS_ADMIN`, `SYS_MODULE`), а также не должен выходить за рамки Seccomp-профиля. Исключения прорабатываются отдельно для каждого приложения.

## 5.2.10. UI

1. Приложения, разработанные или адаптированные к платформе, должны быть встроены в портал платформы.
2. Приложения, разработанные на платформе, должны подключаться в портал платформы с использованием библиотеки [single-spa.js](#), а адаптированным — следует это сделать.
3. Приложения, адаптированные к платформе и интегрированные с ней, могут работать через `iframe`-встраивание.
4. Приложения рекомендуется реализовывать с использованием фреймворка Angular или библиотек React/Vue, т.к. подключение их в настоящий момент протестировано и поддерживается.
5. Приложениям рекомендуется следовать требованиям Guideline ZIIoT — рекомендации по проектированию пользовательских интерфейсов.
6. Использование дизайн-системы для ZIIoT:
  - Проектируемым приложениям рекомендуется использовать дизайн-систему Prizm (версии 4.0 - Prizm это развитие дизайн-системы ZIIoT версии 3.0).
  - Разрабатываемым приложениям рекомендуется запланировать переход на дизайн-систему Prizm.
  - Если приложение не использовало компонентную базу, то рекомендуется использовать дизайн-систему Prizm.
7. Если приложение использовало (начало миграцию) компонентную базу версию 3.0, то рекомендуется продолжить ее использование (продолжить миграцию), до разработки мигратора до Prizm.
  - Проектам, разработанным без компонентной базы, рекомендуется запланировать переход сразу на дизайн-систему Prizm, если планируется продолжать разработку приложения год или более.
  - Рекомендуется использовать графики из Prizm для отображения визуализации в приложениях.
8. Приложениям рекомендуется использовать платформенные средства визуализации (отчеты, дашборды, мнемосхемы) для отображения данных из ОМ, что позволит максимально гибко

- отображать данные (в т.ч. показывать на экранах приложений данные из других приложений).
9. Приложения должны работать с оболочкой портала только с использованием средств предоставленных SDK.
  10. Приложения должны инкапсулировать собственные стили. Не допускается влиять или создавать стили на уровне глобальной области видимости. Необходимо очищать свои стили при разрушении приложения.
  11. Приложениям следует работать в своем окружении. Не следует определять в `globalThis`, `window`, `document`.
  12. Приложения, разработанные на платформе, для взаимодействия с функционалом портала должны использовать сервис оболочку портала (`zui-app-shell`).
  13. Приложениям следует использовать сервис Сервис настроек приложения (`zif-portal-settings-dotnet`) для хранения настроек/состояния приложения, компонентов, сетки.
  14. Приложения не должны содержать UI в виде нативных десктоп приложений.
  15. Приложения должны использовать собственные ресурсы, не зависеть от ресурсов `AppShell` (`assets`, `css`).
  16. При разработке приложений рекомендуется поддерживать все браузеры, попадающие под следующие условия (согласно `browserslist`) и, в случае использования, `Prizm`:
    - распространенность более 0.5% на глобальном `www`;
    - последние 2 версии каждого браузера;
    - Firefox ESR;
    - есть поддержка;
    - не IE 9-11.

### 5.2.11. Представление архитектуры приложения

Рекомендуется представлять архитектуру приложений в соответствии с документом [Соглашение по моделированию архитектуры](#).

## 5.3. Уровни требований

При проектировании приложения уровень его интеграции с платформой ZIIoT устанавливается при утверждении плана проекта. Уровни интеграции:

1. Приложение, разрабатываемое под платформу ZIIoT, то есть приложение, при проектировании которого изначально закладывалась необходимость работы на платформе ZIIoT.
2. Приложение, адаптированное (смигрированное) под платформу ZIIoT, то есть приложение, изначально при проектировании которого не закладывалась необходимость работы на платформой ZIIoT или закладывалась не в полном объеме, но потом было принято решение об необходимости работы на платформе ZIIoT.
3. Приложение, интегрированное с платформой ZIIoT, то есть приложение, которое разрабатывается для работы без платформы ZIIoT, но имеется необходимость заложить опциональную возможность работы с платформой ZIIoT.

### 5.3.1. Приложение, разрабатываемое под платформу ZIIoT

Приложение, разрабатываемое под платформу ZIIoT, проектируется и реализуется, максимально используя все возможности платформы.

Отступление от обязательных требований (должно) не допускается. В случае, если решением архитектурного комитета ZIIoT установлена такая необходимость, то инициируется процесс пересмотра или дополнения обязательных требований.

Отступление от рекомендуемых требований (следует, рекомендуется) допускается, только если необходимость такого отступления обоснована и доказана архитектурным проектированием приложения, и выполнение рекомендаций приведет к невыполнению (не полному выполнению) приложением функциональных или нефункциональных требований. Ссылка на то, что выполнение требования приведет к необходимости существенной переработки архитектуры приложений или уже реализованных программных модулей не допускается.

Необходимость подтверждается на защите на архитектурном комитете ZIIoT.

### 5.3.2. Приложение, адаптированное (смигрированное) под платформу ZIIoT

При адаптации существующего приложения должен быть разработан поэтапный план миграции на целевую архитектуру. Целевая и промежуточная архитектура согласуется с архитектурным комитетом ZIIoT.

Промежуточная архитектура:

В промежуточной архитектуре представляется и защищается на архитектурном комитете ZIIoT список требований к приложениям, которые временно не соблюдаются. Как правило, приложениям не следует в промежуточной архитектуре отступать от обязательных требований (должно). В частности, не допускается нарушения требований:

- Интеграция с другими приложениями через Объектной Модели/UDL (1.1).
- Работа приложения в порталного решения платформы (10.1).
- Разворачивание приложения (6.1–6.8).
- Аутентификация (9.1–9.8).
- Не исполнение других требований, если нарушение их приведет или может привести к нестабильности работы платформы ZIIoT или других приложений.

Целевая архитектура:

Отступление от обязательных требований (должно) в целевой архитектуре не допускается.

Отступление от рекомендуемых требований (следует, рекомендуется) в целевой архитектуре допускается если выполнение рекомендуемых требований нецелесообразно (в том числе по причине необходимости существенной переработки приложения без получения существенной выгоды для функциональных или нефункциональных требований).

В частности, разрешается сохранять и не перерабатывать уже реализованное:

- Хранение и доступ к данным в выделенной под приложение БД PostgreSQL.
- Бизнес-логику, логику расчетов, классификации, обработки событий и др., даже при наличии в платформе аналогичных механизмов.
- Средства визуализации и ввода данных.
- Ролевая модель приложения, авторизация пользователей (но не аутентификация).

### 5.3.3. Приложение, интегрированное с платформой ZIIoT

При интеграции с платформой со стороны приложения предъявляются требования:

- Интеграция с системой аутентификации платформы.
- Все данные, которые потенциально требуются другим системам, должны быть доступны через Объектную Модель /UDL (1.1).
- Внешние взаимодействия могут быть реализованы через сервисы адаптеры.

Все остальные сервисы платформы используются по желанию. Приложение хостится вне кластера платформы и пользуется только сервисами платформы, доступными через `gateway`.

## 6. Контакты технической поддержки

Таблица 6.1. Контакты технической поддержки

| Вид поддержки | Значение                                                                            |
|---------------|-------------------------------------------------------------------------------------|
| Портал        | <a href="https://jira.zyfra.com/servicedesk">https://jira.zyfra.com/servicedesk</a> |
| Email         | <a href="mailto:dp-support@zyfra.com">dp-support@zyfra.com</a>                      |

Регистрация запросов производится круглосуточно. Рабочее время — с 8.00 до 17.00 (время московское), перерыв с 12.00 до 13.00, в рабочие дни. Запросы, поступившие в нерабочее время, обрабатываются на следующий рабочий день.

Таблица 6.2. Контакты технической поддержки ЦИП

| Вид поддержки | Значение                                                                                                                  |
|---------------|---------------------------------------------------------------------------------------------------------------------------|
| Портал        | <a href="https://jira.zyfra.com/servicedesk/customer/portal/42">https://jira.zyfra.com/servicedesk/customer/portal/42</a> |
| Email         | <a href="mailto:support@idpllc.ru">support@idpllc.ru</a>                                                                  |

Регистрация запросов производится круглосуточно. Рабочее время — с 9.00 до 18.00 (время московское), перерыв с 13.00 до 14.00, в рабочие дни. Запросы, поступившие в нерабочее время, обрабатываются на следующий рабочий день.