



Сведения о релизе

Zyfra Quality Lab

Версия ZQL – 5.3.0

Содержание

1. Общие сведения	4
2. Информационные данные	4
3. Что нового	4
4. Выполненные изменения	7
5. Исправленные дефекты	9
6. Известные дефекты	11
7. Особенности развертывания	11
8. Список сервисов и компонентов	17
8.1. Поставки и совместимость платформы	17
8.2. Измененные сервисы и компоненты	17
8.3. Добавленные сервисы и компоненты	20
8.4. Неизмененные сервисы	20
8.5. Устаревшие сервисы	26
9. Контакты технической поддержки	27
Приложение 1 Обновление платформы ZIIoT с 2.13.x до 2.16.2 и Helm-чарта ZQL с 4.8.4-mod-a до 5.0.0-mod-a	28
Обновление платформы с 2.13.x на 2.16.2	28
Обновление с 2.13.x до 2.14.3	28
Обновление с 2.14.3 до 2.15.1	28
Шаги после обновления до 2.15.1	28
Обновление с 2.15.1 до 2.16.2	29
Шаги после обновления до 2.16.2	29
Обновление Helm-чарта ZQL с 4.8.4-mod-a до 5.0.0-mod-a	29
Приложение 2 Инструкция по переводу сервиса zqa-equipment на версию без хештегов	31
Предварительные операции	31
Основные изменения в новой версии сервиса zqa-equipment	31
Шаги по обновлению конфигурационного файла для существующего развертывания	31
Приложение 3 Инструкция по переводу сервиса zqa-materials на версию без хештегов	37
Предварительные операции	37
Основные изменения в новой версии сервиса zqa-materials	37
Шаги по обновлению конфигурационного файла для существующего развертывания	37
Приложение 4 Инструкция по переводу сервиса zqa-personnel на версию без хештегов	41
Предварительные операции	41
Основные изменения в новой версии сервиса zqa-personnel	41
Шаги по обновлению конфигурационного файла для существующего развертывания	41

Приложение 5 Инструкция по переводу сервиса zqa-samples на версию без хештегов .45

Предварительные операции	45
Основные изменения в новой версии сервиса zqa-samples.....	45
Шаги по обновлению конфигурационного файла для существующего развертывания	45

1. Общие сведения

18 июля 2024 года вышел очередной релиз Zyfra Quality Lab (ZQL) 5.3.0. (предыдущий релиз 5.2.0). В рамках релиза в ZQL была обновлена функциональность модулей **Контроль качества, Оборудование, Нормативные документы** и др., а также исправлен ряд дефектов предыдущей версии.

2. Информационные данные

Таблица 1 Информационные данные для релиза ZQL 5.3.0

Параметр	Значение
Дистрибутив	http://docker.idp.yc.ziit.ru/zql/
Инсталлятор	5.3.0-mod-a
Отчет по уязвимостям	Отчет по измененным сервисам релиза ZQL 5.3.0
Состав дистрибутива	Версии ПО
Отчет по тестированию	Отчет по тестированию версии ZQL 5.3.0

3. Что нового

Среди наиболее значимых доработок можно отметить следующие:

В02 Управление образцами

В модуле **Контроль качества** реализована возможность загружать, просматривать и удалять вложения:

- на странице **Заявки** в таблице **Образцы заявки**;
- на форме ввода в таблице **Образцы заявки**;
- на странице **Образцы** в таблице **Образцы**.

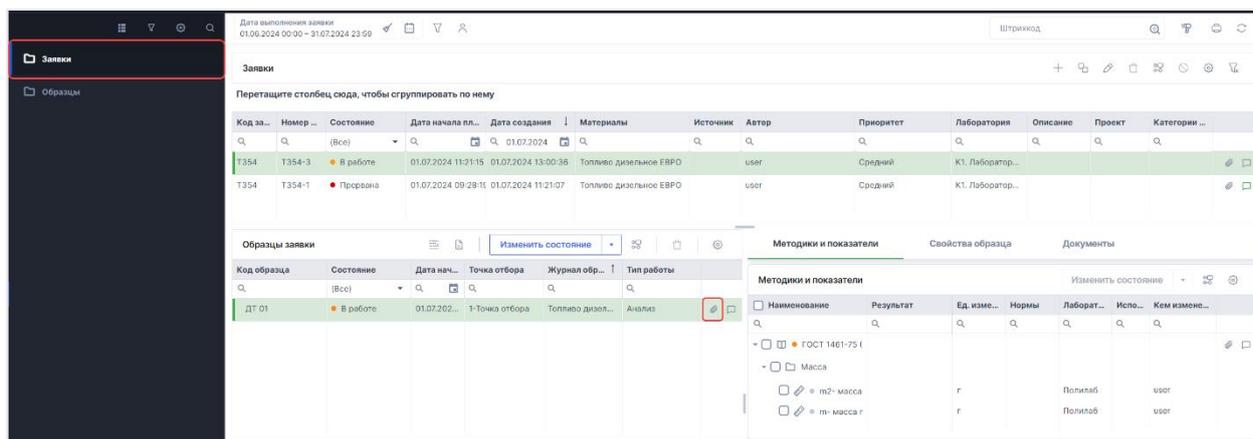


Рисунок 3.1 Вложения для таблицы Образцы заявки

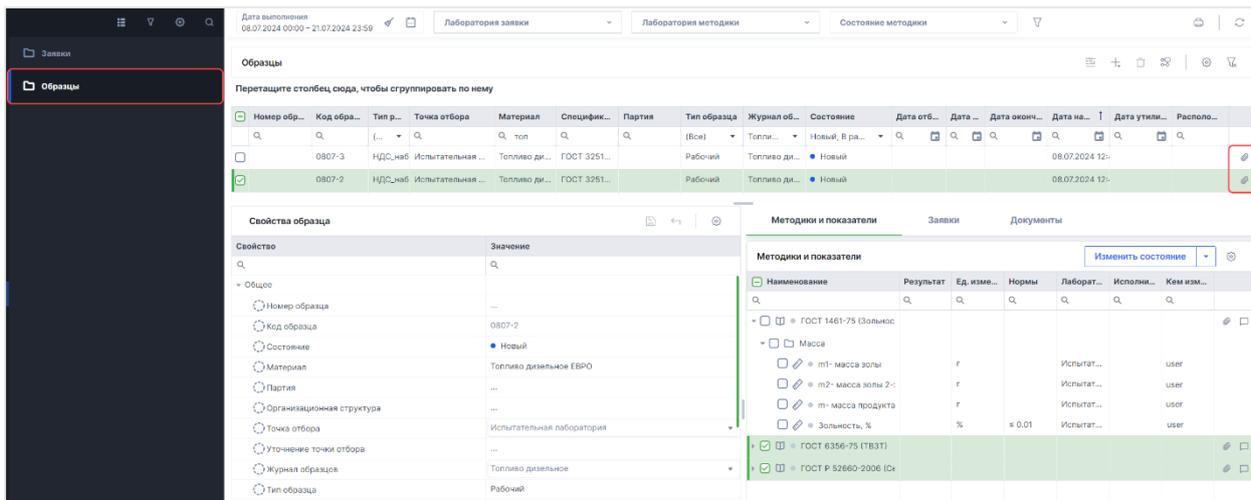


Рисунок 3.2 Вложения для таблицы Образцы

Для образца добавлены новые состояния:

- В процессе отбора;
- Принят в лабораторию;
- Зарегистрирован;
- Передан в стороннюю лабораторию;
- Возвращен заказчику;
- Переведен в образец ВЛК.

В03 Управление испытаниями

В модуле **Контроль качества** реализована возможность добавления дополнительных свойств для партии в диалоге реактивы и материалы.

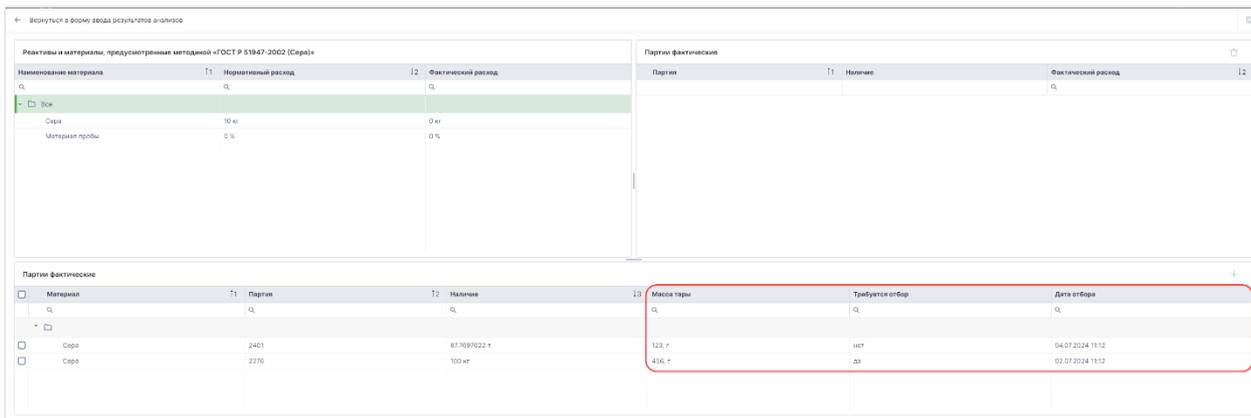


Рисунок 3.3 Дополнительные свойства для партии

Е01 Управление нормативными документами

В модуле **Нормативные документы** разделы диалога создания/редактирования нормативного документа перемещены на главную страницу в отдельные вкладки. Также добавлена возможность использовать дополнительные поля.

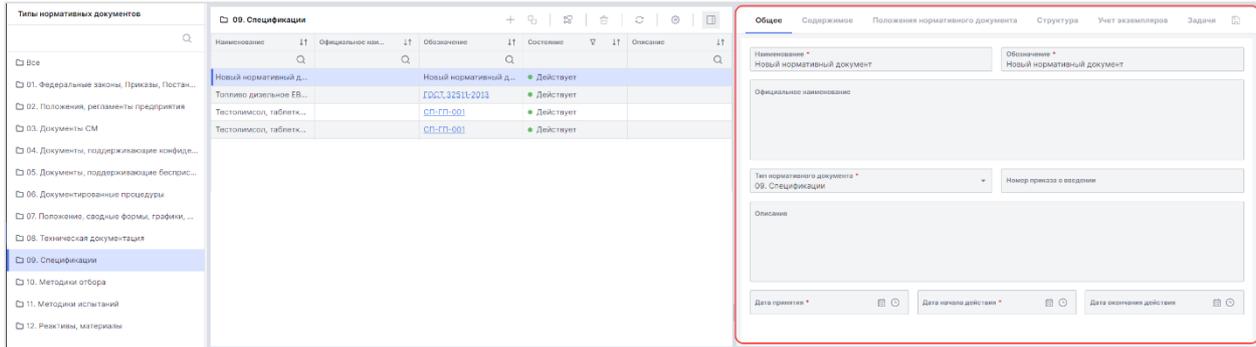


Рисунок 3.4 Вкладки нормативного документа

E04 Управление оборудованием

В модуле **Оборудование** добавлена возможность копирования мероприятия для оборудования и измерительного канала.

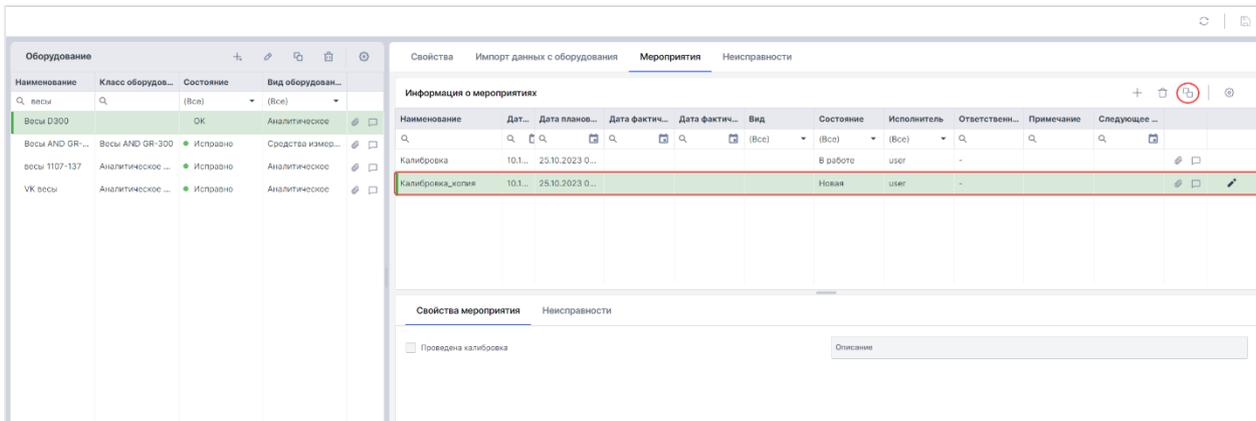


Рисунок 3.5 Копирование мероприятия для оборудования

Также теперь можно добавлять свойства к мероприятиям оборудования и измерительного канала.

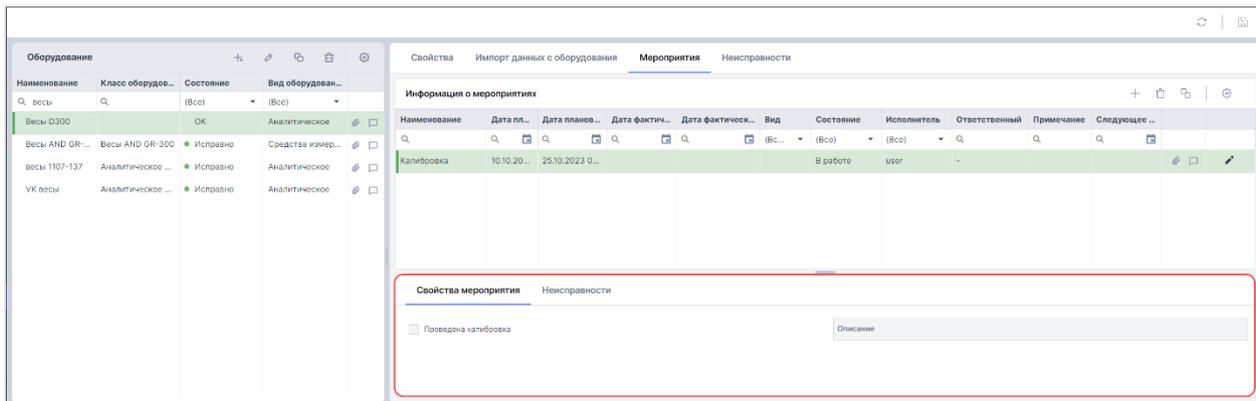


Рисунок 3.6 Дополнительные свойства мероприятия оборудования

E12 Контроль качества товарной продукции

Добавлен процесс **Контроль качества. Создание партии** и адаптирован процесс паспортизации для запуска от образца в модуле **Контроля качества**.

4.Выполненные изменения

Полный список выполненных изменений см. ниже (Таблица 2).

Таблица 2 Выполненные изменения версии 5.3.0

Ключ	Тема	Компонент
#14906	ВЛК. Оценка показателей качества. Выбор образцов для расчёта	ВЛК. Серии контроля
#17886	Заявки. Назначение исполнителя методики при смене состояния методики образца	Заявки, Рабочие процессы
#18453	Сегменты задачи. Отображение группы материала	Методики, Приложение Контроль качества, Фактические операции
#19652	v.5.3.0_BPMN процесс создания партии из КК	Мониторинг запасов, Контроль качества
#17719	E01-6. НД. Набор дополнительных свойств НД	Нормативные документы
#8053	E01-4. НД. Общие. Рефакторинг диалога создания/редактирования НД	Нормативные документы
#8054	E01-3. НД. Содержимое. Перемещение из диалога создания/редактирования раздела во вкладку	Нормативные документы
#15006	E01-2. НД. Структура. Перемещение из диалога создания/редактирования раздела во вкладку	Нормативные документы
#8055	E01-1. НД. Положение НД. Перемещение из диалога создания/редактирования раздела во вкладку	Нормативные документы
#19650	v. 5.3.0_Копирование мероприятия Оборудования	Оборудование
#19460	v. 5.3.0_Мероприятия оборудования - дополнительные атрибуты	Оборудование
#19648	v. 5.3.0_Новые статусы для образца	Образцы
#18608	Контроль качества/Заявки. Прикрепление файлов и просмотр прикрепленных файлов в таблице "Образцы заявки"	Образцы
#18602	Контроль качества/Образцы. Прикрепление файлов и просмотр прикрепленных файлов в таблице "Образцы"	Образцы
#18954	Перевод zqa-materials-workflow-workers на Net 8	Рабочие процессы
#19647	v.5.3.0_Материалы в КК. Допсвойства партии при поиске партий	Списание реактивов
#19646	v.5.3.0_Материалы в ФО. Допсвойства партии при поиске партий	Фактические операции
#19813	Актуализировать инструмент развертывания релиза ZQL v.5.3.0	-
#19795	Перевод zqa-directories-workflow-workers на Net 8	-
#19651	v. 5.3.0_Доработки ПО для процесса паспортизации	-
#18393	Перевод zqa-properties на Net 8	-
#18383	Перевод zqa-directories на Net 8	-
#18347	Отказаться от DevExtreme в таблице документов на странице заявок	-

Ключ	Тема	Компонент
#18916	Добавить поддержку древовидных таблиц в фасад гридов	-
#19875	Использовать в монорепозитории обновлённый пакет @zyfra/camunda-task (5.3.x)	-

5.Исправленные дефекты

Исправленные дефекты, отсортированные по приоритету см. ниже (Таблица 3):

Таблица 3 Исправленные дефекты версии 5.3.0

Ключ	Связанные запросы в техническую поддержку	Тема	Приоритет	Компонент
#19209	IDPFLVS-6009	Некорректная работа с файлами в модуле Проекты	Критичный	Проекты
#19486	IDPFLVS-6132	Контроль качества: не работает округление показателя в БОльшую сторону	Высокий	API
#18869	-	Из лога невозможно понять какая именно сущность мешает старту сервиса	Высокий	Install
#19682	-	ВЛК. Неверное округление параметра "Контроль повторяемости"	Высокий	ВЛК. Операции контроля
#19868	IDPFLVS-6411	Производительность фронта модуля "Контроль качества" при создании заявки и заполнении показателей образца	Высокий	Контроль качества
#19862	IDPFLVS-6396	Значение норматива повторяемости, стало отображаться с большим количеством знаков после запятой	Высокий	Контроль качества
#19786	IDPFLVS-6346	Не сохраняются изменения в ФМ Спецификации	Высокий	Спецификации
#19607	IDPFLVS-6230	УАТ. График контроль. Не корректно отображается календарь.	Средний	ГАК
#19430	-	Заявки. Отсутствует иконка удаления примененного фильтра	Средний	Заявки
#12861	-	Контроль качества. Не корректная сортировка по алфавиту в колонках "Источник" и "Проект"	Средний	Заявки
#20126	IDPFLVS-6595	zqa-normdocs	Средний	Нормативные документы
#18999	IDPFLVS-5855	Контроль качества. Отсутствует пиктограмма закрытия (крестик) у "чипсов" в компоненте отображения активных фильтров	Средний	Контроль качества
#20169	IDPFLVS-6617	Модуль "Проекты". Не корректное отображение значения свойства с типом dateTime при просмотре свойства в "Дополнительная информация"	Средний	Проекты
#20137	IDPFLVS-6604	Пустые поля при редактировании проекта в psplanning	Средний	Проекты
#16625	-	Ошибка при выполнении процесса "Изменение состояния мероприятия"	Средний	Рабочие процессы
#19788	IDPFLVS-6347	Ошибка образца при удалении спецификации и запуске процесса "Повторные испытания"	Средний	Контроль качества

Ключ	Связанные запросы в техническую поддержку	Тема	Приоритет	Компонент
#19603	-	Форма ввода. Для методики, созданной через процесс "Повторные испытания", присваивается метрология, рассчитанная для исходной методики	Средний	Форма ввода
#18113	-	В методе /Materials.MaterialGroupApi/ListMaterialGroups searchString работает только при полном совпадении с name\code группы материалов	Низкий	API
#17819	-	При выполнении метода /lms/Events.EventScopeApi/ListEventScopes в списке отображается ОПИ с кодом "QL.EventTypes"	Низкий	API
#19321	-	Мероприятия. Некорректно отображается компонент с чипсами в полях "Тип" и "Подразделение"	Низкий	Мероприятия

6. Известные дефекты

Известные дефекты критичного и высокого приоритета в релизе отсутствуют.

7. Особенности развертывания

При развертывании релиза необходимо учитывать информацию из Changelog сервисов и соблюдать инструкции, см. ниже (Таблица 4).

Таблица 4 Инструкции при развертывании релиза

№	Инструкция	Источник ограничений	Начиная с версии
1	В EntitiesConfiguration.json сервиса zqa-events изменилось наименование раздела "operationPerformances" на "operationsPerformances", необходимо актуализировать конфигурационный файл	Z10000064-16625 - Ошибка при выполнении процесса "Изменение состояния мероприятия"	5.3.0
2	В EntitiesConfiguration.json сервиса zqa-normdocs добавлен класс QL.NormDocProperties в раздел "classes" <pre> { "alias": "NormDocProperties", "create": "true", "modelPrototype": "DMP", "name": { "en": "Normative document properties", "ru": "Свойства нормативных документов" }, "code": "QL.NormDocProperties", "description": { "en": "Class containing extra properties of normative document", "ru": "Класс, содержащий дополнительные свойства нормативных документов" }, "properties": [] } </pre>	Z10000064-17719 - Ошибка при выполнении процесса "Изменение состояния мероприятия"	5.3.0
3	Необходимо обновить DMN для бизнес-процессов: <ul style="list-style-type: none"> Процесс "QL.Sample.Workflow.SampleStatusManagement" - изменилась DMN QL.Sample.DMN.ChangeState. Процесс "QL.Auto.AnalysisRequest.ChangeStateBySampleState" - изменилась dmn QL.AnalysisRequest.DMN.ChangeStateBySampleState. Процесс "QL.Auto.Sample.ChangeStateBySampleAnalysisRequestState" - изменилась dmn. 	Z10000064-19648 - v. 5.3.0_Новые статусы для образца	5.3.0

№	Инструкция	Источник ограничений	Начиная с версии
	<p>QL.Sample.DMN.ChangeStateBySampleAnalysisRequestState.</p> <ul style="list-style-type: none"> Процесс "QL.Auto.SampleTech.ChangeStateBySampleState" - изменилась dmн QL.SampleTech.DMN.ChangeStateBySampleState. 		
4	<p>1. В EntitiesConfiguration сервиса zqa-equipment необходимо добавить класс дополнительных свойств для мероприятий оборудования:</p> <pre> { ..., "classes": [..., { "alias": "MaintenanceResponseProperties", "create": "true", "modelPrototype": "EMP", "name": { "en": "Maintenance response properties", "ru": "Свойства мероприятий обслуживания оборудования" }, "code": "QL.MaintenanceResponseProperties", "description": { "en": "Maintenance response properties", "ru": "Класс содержит набор дополнительных свойств мероприятий обслуживания оборудования" } }] } </pre> <p>2. В данный класс необходимо добавить примитивы свойств, являющиеся дополнительными свойствами для мероприятий оборудования.</p>	<p>Z10000064-19460 - v. 5.3.0 5.3.0_Мероприятия оборудования дополнительные атрибуты</p>	<p>5.3.0</p>
5	<p>В EntitiesConfiguration сервиса zqa-techs необходимо добавить 2 новые области применения иерархии:</p> <pre> { ..., "hierarchyScopes": [..., { "alias": "QL", "create": "true", "code": "QL", "name": { "en": "Quality Lab", </pre>	<p>Z10000064-19647 - v.5.3.0_Материалы в КК. Допсвойства партии при поиске партий</p>	<p>5.3.0</p>

№	Инструкция	Источник ограничений	Начиная с версии
	<pre> "ru": "Лаборатория качества" }, "description": { "en": "Scope for Quality Lab", "ru": "Область применения для Quality Lab" } }, { "alias": "MaterialLots", "create": "true", "code": "QL.MaterialLots", "parent": "QL", "name": { "en": "Material Lots", "ru": "Партии материала" }, "description": { "en": "Groups the scopes related to material lots", "ru": "Группирует области применения, связанные с партиями материала" } }] } </pre>		
6	<p>Реализована миграция, создающая спецификации материалов с типом MaterialGroup для всех контекстов использования. В сервисе интеграции методик (zqa-i9n-techs-ziiot) доработана синхронизация спецификаций материалов и оборудования в zif-sm-workdefinition.</p> <p>В зависимости от порядка развертывания сервисов предусмотрены 2 сценария:</p> <ol style="list-style-type: none"> zqa-i9n-techs-ziiot → zqa-techs - выполнится синхронизация через Kafra для версий методик, в которых созданы новые MaterialSpecification; zqa-techs → zqa-i9n-techs-ziiot - выполнится синхронизация в старой реализации. В этом случае необходим запуск принудительной синхронизации через REST метод "POST /techs" интеграционного сервиса zqa-i9n-techs-ziiot. 	<p>Z10000064-17757 - Методики. Настройка материала для контекста методики</p>	5.2.0
7	<p>В рамках релиза 5.2.0 произошло объединение сервисов zps-planning и zqm-perform-operations в единый сервис zqa-projects. Переменные и API были также объединены, их структура и поведение остались без изменений.</p>	<p>Z10000064-17828 - E-13-8. ФО. Проверка доступа к сегменту через подразделение</p>	5.2.0
8	<p>Совместимость сервиса zui-app-ps с платформой 2.17.x заблокирована инцидентом в платформу. Решение ожидается в релизе 2.17.2.</p>	<p>IDPFLVS-4984 - zif-om-grapfql. Ответ 405 при запросе getMaterialActualByLotIds</p>	5.1.0

№	Инструкция	Источник ограничений	Начиная с версии
9	<p>Добавлены политики для доступа к проектам, этапам, операциям в модуле "Проекты" (см. Changelog zps-planning): zqa-projects-projects, zqa-projects-operations</p> <p>В качестве атрибута объекта политики может использоваться только атрибут ResponsibleSubdivisionCode.</p>	<p>Z10000064-15861 - E-13-5. Проекты. Безопасность команд в приложении для проектов/этапов</p> <p>Z10000064-17113 - E-13-6. Проекты. Безопасность команд в приложении для операций</p>	5.1.0
10	<p>В сервис zps-planning добавлен конфигурационный файл EntitiesConfiguraiton.json. В нем указаны существующие ОПИ "Лаборатория качества" и "Оборудование" (которые используются, например в сервисе zqa-equipment). А также добавлена новая ОПИ "Проектный центр". Если данные ОПИ имеют кастомные поля (наименование/код), необходимо актуализировать новый конфигурационный файл под конкретный полигон.</p>	<p>Z10000064-16550 - Проекты. Фильтрация подразделений в навигационном меню</p>	5.1.0
11	<p>Реализован новый механизм синхронизации статусов в модулях "Проекты" и "Фактические операции". Необходимо ознакомиться с CHANGELOG проекта zqa-workflow-definitions, добавлены новые бизнес-процессы.</p>	<p>Z10000064-15290 - Проекты. Каскадное изменение статусов (Новый, Приостановлено, Отмена)</p>	5.1.0
12	<p>Реализована совместимость бизнес-процессов ZQL с версией платформы 2.17.x. Необходимо ознакомиться с CHANGELOG проекта zqa-workflow-definitions.</p>	<p>Z10000064-17641 - Переход на платформу 2.17.0</p>	5.1.0
13	<p>В связи с наличием уязвимостей безопасности в ПО envoy, совершен переход на nginx. Оставлена опциональная возможность использовать envoy. Режим выбирается переменной хельма "grpcServer" с 2 режимами "envoy / nginx" (по умолчанию nginx).</p>	<p>Z10000064-8533 - Отказ от envoy при работе с grpc-web</p>	5.1.0
14	<p>В связи с переходом на версию платформы 2.16.x, в данный момент присутствуют проблемы с отображением свойств ресурсов операции в компоненте "Проекты". Для решения проблемы вместе с релизом необходимо поставить версию сервиса ZIIOT: zif-sm-operationdefinition:1.17.3-rc.1</p> <p>Проблема будет устранена в версии ZIIOT 2.16.3.</p>	<p>PLATFORM01-38060 - ФК-1. Фильтрация свойств оборудования по области применения QL</p>	5.0.0
15	<p>Обновление платформы с 2.13.x до 2.16.2 и Helm-чарта ZQL с 4.8.4-mod-a до 5.0.0-mod-a, см. Приложение 1. Переход на платформу версии 2.17.0 осуществляется через версию 2.16.2</p>	<p>Z10000064-16003 - Переход на платформу 2.16.0</p>	5.0.0
16	<p>Часть процессов мониторинга запасов перенесена из zps-store-bpmn в zqa-workflow-definitions (см. Changelog проекта zqa-workflow-definitions)</p>	<p>Z10000064-16601 - Перевод сервисов ZIAK на ZIIOT</p>	5.0.0
17	<p>Результаты проверки параметров средств контроля теперь сохраняются в привязке каждому показателю с использованием кода CheckProcResult. Все результаты операций контроля, полученные до выхода версии zqa-</p>	<p>Z10000064-12864 - ВЛК. Сохранение результатов проверки средства контроля для каждого показателя</p>	5.0.0

№	Инструкция	Источник ограничений	Начиная с версии
	iqс:6.0.0 теперь несовместимы и не будут отображаться.		
18	<p>(2024) Инструкция по переводу сервиса оборудования на версию без хештегов, см. Приложение 2</p> <p>(2024) Инструкция по переводу сервиса материалов на версию без хештегов, см. Приложение 3</p> <p>Инструкция по переводу сервиса персонала на версию без хештегов, см. Приложение 4</p> <p>(2024) Инструкция по переводу сервиса образцов на версию без хештегов, см. Приложение 5</p>	<p>Z10000064-16324 - Оборудование. Отказ от хештегов</p> <p>Z10000064-16325 - Материалы. Отказ от хештегов</p> <p>Z10000064-16326 - Персонал. Отказ от хештегов</p> <p>Z10000064-16327 - НДС образцов. Отказ от хештегов</p>	5.0.0
19	<p>В EntitiesConfiguraton.json сервиса оборудования (zqa-equipment) необходимо добавить описание ОПИ в раздел "hierarchyScopes":</p> <pre data-bbox="288 981 979 1433"> { "alias": "QL", "create": "true", "code": "QL", "name": { "en": "Quality Lab", "ru": "Лаборатория качества" }, "description": { "en": "Scope for Quality Lab.", "ru": "Область применения для Quality Lab." } } </pre>	<p>Z10000064-15626 - ФК-1. Фильтрация свойств оборудования по области применения QL</p>	4.8.0
20	<p>Для сервиса оборудования (zqa-equipment) в раздел "Entities" добавлена конфигурация "ExtraPropertiesClass" - класс дополнительных свойств оборудования.</p> <p>Для сервиса оборудования (zqa-equipment) в раздел "tags" добавлена конфигурация "ExtraProperties" - хэштег класса дополнительных свойств оборудования.</p>	<p>Z10000064-15622 - ФК-4. Дополнительные классы оборудования</p>	4.8.0
21	<p>Реализовано кэширование оборудования и измерительного канала в Redis для получения актуального состояния и отображения на UI. В связи с этим в состав продукта добавлен Redis.</p>	<p>Z10000064-14008 - Оборудование. Подписка на изменение состояния оборудования в таблице</p>	4.8.0
22	<p>Формирование отчёта с использованием json-источников сервиса zqa-documents.</p>	<p>Z10000064-12532 - ВЛК. Формирование отчёта для контекста контроля стабильности</p>	4.8.0

№	Инструкция	Источник ограничений	Начиная с версии
23	Реализована ролевая безопасность для методик. Добавлена новая политика zqa-techs-techs	Z10000064-14968 - Методики. Атрибутная безопасность в сервисе (АВАС) Z10000064-14969 - Методики. Расширение правил для атрибутной безопасности в сервисе (АВАС)	4.8.0
24	Реализована ролевая безопасность для заявок. Добавлена новая политика zqa-analysisrequests-analysisrequests	Z10000064-14016 - Заявки. Доступ на чтение к заявкам с учетом атрибутов (АВАС)	4.7.0
25	Исправлена ошибка, из-за которой к сущностям ZQL относились объекты из моделей, не описанных в конфигурационных файлах EntitiesConfiguration.json.	Z10000064-14219 - Сервисы ZQL должны работать с единой моделью материалов	4.7.0
26	Произошел отказ от модифицированной версии zif-sm-testspecification:1.8.5-mod-zql. Начиная с релиза ZQL 4.6.5, используется релизная версия zif-sm-testspecification:1.9.1	Z10000064-14248 - В релизе необходимо использовать zif-sm-testspecification с релизным тегом	4.7.0

8. Список сервисов и компонентов

Ниже перечислены измененные, добавленные, устаревшие и неизменные сервисы ZQL по сравнению с предыдущей версией.

8.1. Поставки ZQL, совместимость версий ZQL и Платформы

Доступны 2 варианта поставки ZQL:

- **ZQL for ZIIoT** – для установки ZQL поверх ZIIoT. Необходимые сервисы отмечены в таблицах ниже признаком **ZQL for ZIIoT**.
- **ZQL with ZIIoT SDK** – для установки ZQL без предварительной установки платформы ZIIoT. В данный вариант поставки дополнительно к **ZQL for ZIIoT** входит набор сервисов из состава ZIIoT, обеспечивающих работоспособность продукта – пакет **ZIIoT SDK**. Сервисы, которые не отмечены в таблицах ниже признаком **ZQL for ZIIoT**, входят в пакет **ZIIoT SDK**.

ZQL версии 5.3.0 совместим с платформой ZIIoT версии **2.16.3, 2.17.2**.

Рекомендованная версия платформы ZIIoT для установки – **2.17.2**.

8.2. Измененные сервисы и компоненты

В таблице ниже приведен перечень версий измененных сервисов, которые входят в стабильную версию релиза 5.3.0.

Таблица 5 Измененные сервисы в релизе ZQL 5.3.0

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
Установщик	lims-helm	5.3.0-mod-a	ДА
Сервисы фронтенд	zui-app-acs	1.14.0	ДА
	zui-app-calibrations	1.4.0	ДА
	zui-app-equipment	1.17.0	ДА
	zui-app-events	0.11.0	ДА
	zui-app-hr	1.14.0	ДА
	zui-app-importers	1.13.0	ДА
	zui-app-iqc	1.13.0	ДА
	zui-app-iqc-cs	1.7.0	ДА
	zui-app-normdocs	1.18.0	ДА
	zui-app-operation-performance	1.8.0	ДА
	zui-app-ps-planning	4.19.0	ДА
	zui-app-qa	1.34.0	ДА
	zui-app-qm-quotas	1.4.0	ДА
	zui-app-qms	1.8.0	ДА
	zui-app-rooms	1.6.0	ДА
	zui-app-sampletemplates	1.13.0	ДА
	zui-app-samplelogs	1.15.0	ДА

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	zui-app-specs	1.13.0	ДА
	zui-app-techs	1.16.0	ДА
Сервисы бэкенд	zqa-admin	1.4.9	ДА
	zqa-admin-workflow-workers	1.4.9	ДА
	zqa-analysisrequests	7.18.1	ДА
	zqa-analysisrequests-simple	3.12.2	ДА
	zqa-analysisrequests-simple-workflow-workers	3.1.9	ДА
	zqa-analysisrequests-workflow-workers	7.18.1	ДА
	zqa-calibrations	0.7.1	ДА
	zqa-calibrations-workflow-workers	0.5.4	ДА
	zqa-camunda-sampling	2.5.8	ДА
	zqa-changelog	2.2.1	ДА
	zqa-comments	2.1.1	ДА
	zqa-comments-workflow-workers	2.0.8	ДА
	zqa-contentstorage	1.5.9	ДА
	zqa-directories	0.3.0	ДА
	zqa-documents	2.7.1	ДА
	zqa-documents-workflow-workers	3.2.4	ДА
	zqa-eqp-import	1.5.1	ДА
	zqa-equipment	3.20.0	ДА
	zqa-equipment-workflow-workers	3.20.0	ДА
	zqa-events	0.6.1	ДА
	zqa-events-generator	1.6.1	ДА
	zqa-events-workflow-workers	0.4.11	ДА
	zqa-generator	3.1.1	ДА
	zqa-i9n-analysisrequests-iqc	0.9.1	ДА
	zqa-i9n-analysisrequests-ziiot	0.8.1	ДА
	zqa-i9n-techs-ziiot	0.5.0	ДА
	zqa-iqc	6.3.0	ДА
	zqa-iqc-workflow-workers	6.2.1	ДА
	zqa-masks	1.5.1	ДА
	zqa-materials	2.11.0	ДА
	zqa-materials-workflow-workers	2.11.0	ДА
	zqa-normdocs	4.0.0	ДА
zqa-normdocs-workflow-workers	3.4.2	ДА	
zqa-notifications	0.5.9	ДА	
zqa-personnel	1.12.1	ДА	

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	zqa-personnel-workflow-workers	1.6.9	ДА
	zqa-projects-workflow-workers	1.17.1	ДА
	zqa-properties	3.4.0	ДА
	zqa-properties-observer	1.7.1	ДА
	zqa-qms	1.5.1	ДА
	zqm-quotas	1.5.1	ДА
	zqa-quotas-workflow-workers	1.0.1	ДА
	zqa-relations	1.4.1	ДА
	zqa-relations-workflow-workers	1.2.4	ДА
	zqa-rooms	0.2.2	ДА
	zqa-samplelogs	2.7.1	ДА
	zqa-samplemasks	1.3.9	ДА
	zqa-samples	11.5.0	ДА
	zqa-samples-workflow-workers	11.5.0	ДА
	zqa-sampletemplates	3.8.1	ДА
	zqa-samplingpoints	2.5.1	ДА
	zqa-samplingpoints-workflow-workers	1.6.9	ДА
	zqa-schedulers	3.9.1	ДА
	zqa-schedulers-workflow-workers	2.1.9	ДА
	zqa-scripts	3.7.2	ДА
	zqa-sequences	1.6.1	ДА
	zqa-sequences-workflow-workers	1.4.10	ДА
	zqa-sm-workschedule-workflow-workers	2.3.9	ДА
	zqa-specs	3.12.1	ДА
	zqa-techs	11.5.0	ДА
	zqa-techs-workflow-workers	11.5.0	ДА
	zqa-workflow	0.7.2	ДА
	zqa-projects	2.1.0	ДА
Демо-процессы	zqa-workflow-definitions	2.16.0	ДА
BPMN-процессы	QL.SampleJobOrder.Reanalysis	2.0.1	ДА
	QL.Auto.AnalysisRequest.ChangeStateBySampleState	1.0.0	ДА
	QL.Auto.Sample.ChangeStateBySampleAnalysisRequestState	1.0.0	ДА
	QL.Auto.SampleTech.ChangeStateBySampleState	1.0.0	ДА
	QL.MaterialLot.Transfer.ApproveInvoice	1.1.8	ДА
DMN-схемы	QL.AnalysisRequest.DMN.ChangeStateBySampleState	1.0.1	ДА
	QL.SampleTech.DMN.ChangeStateBySampleState	1.0.1	ДА

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	QL.Sample.DMN.ChangeStateBySampleAnalysisRequestState	1.0.1	ДА
	QL.Sample.DMN.ChangeState	1.0.1	ДА

8.3. Добавленные сервисы и компоненты

В таблице ниже приведен перечень версий добавленных сервисов, которые входят в стабильную версию релиза 5.3.0.

Таблица 6 Добавленные сервисы в релизе ZQL 5.3.0

Тип сервиса/компонента	Сервис/компонент	Версия	ДА
Сервисы бэкенд	zqa-directories-workflow-workers	0.3.0	ДА
BPMN-процессы	QL.MaterialLot.Inventory.Update	1.1.1	ДА
	QL.Certification.CreateCertificate	1.0.0	ДА
	QL.Auto.Sample.CreateMaterialLot	1.0.0	ДА

8.4. Неизмененные сервисы

В таблице ниже приведен перечень версий неизмененных сервисов, которые входят в стабильную версию релиза 5.3.0.

Таблица 7 Неизмененные сервисы в релизе ZQL 5.3.0

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
Сервисы фронтенд	zui-app-certification	1.6.0	ДА
	zui-app-certification-configuration	2.1.1	ДА
	zui-app-guaranteed-properties	1.4.0	ДА
	zui-app-masks	1.3.3	ДА
	zui-app-ps-store	3.20.0	ДА
	zui-app-testspecification	1.6.0	ДА
	zui-bff-graphql	0.10.1	ДА
	zui-app-shell	5.0.1	-
	zui-app-dashboard-dx	12.0.2	-
	zui-app-documentexplorer	16.0.3	-
	zui-app-event-types	2.1.0	-
	zui-app-events-registry	4.5.2	-
	zui-app-nds-configurator	5.0.0	-
	zui-app-om	2.29.2	-
	zui-app-rdm-common	1.18.1	-

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	zui-app-reporteditor	12.0.1	-
	zui-app-security-settings	6.0.0	-
	zui-app-sm-directories	2.17.0	-
	zui-app-sm-operations	2.17.2	-
	zui-app-workflow	3.2.1	-
Сервисы бэкенд	zps-store-bpmn	1.2.0	ДА
	ziak-configuration-manager	1.2.0	ДА
	zif-om-extensions-core	2.0.1	ДА
	zqa-camunda-materiallots	2.8.0	ДА
	zqa-certification-configuration	1.14.0	ДА
	zqa-inspection	2.10.0	ДА
	zqa-jsonmasks	1.5.0	ДА
	zqm-certification	1.11.0	ДА
	zqm-guaranteed-properties	1.7.0	ДА
	zqm-masks	1.3.0	ДА
	zqm-research-module	1.3.0	ДА
	zqm-testspecification-validator	1.5.0	ДА
	zif-bp-dataworker	2.10.1	-
	zif-bp-events-worker	2.10.1	-
	zif-bp-material-worker	2.10.1	-
	zif-bp-message-bus-worker	2.10.1	-
	zif-bp-reference-worker	2.10.1	-
	zif-bp-sm-directories-worker	2.10.1	-
	zif-bp-sm-operation-definition-worker	2.10.1	-
	zif-bp-sm-operation-performance-worker	2.10.1	-
	zif-bp-sm-operations-schedule-worker	2.10.2	-
	zif-bp-sm-work-definition-worker	2.10.1	-
	zif-bp-sm-work-performance-worker	2.10.1	-
	zif-bp-sm-work-schedule-worker	2.10.1	-
	zif-bp-notifications-worker	2.10.1	-
	zif-bp-objects-worker	2.10.1	-
	zif-bp-properties-worker	2.10.1	-
	zif-dashboard	2.4.2	-
	zif-document-archive	1.3.3	-
	zif-events	5.10.1	-
	zif-file-storage	1.2.10	-
	zif-material-lot	2.6.1	-

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	zif-notifications	4.10.1	-
	zif-om-datareferences	3.4.1	-
	zif-om-graphql	3.11.3	-
	zif-om-object	3.7.2	-
	zif-om-testspecification	3.4.1	-
	zif-om-uom	3.7.1	-
	zif-portal-settings-dotnet	2.9.2	-
	zif-rdm-common	3.2.2	-
	zif-reporting	5.8.1	-
	zif-security	3.3.1	-
	zif-sm-directories	1.11.2	-
	zif-sm-operationdefinition	1.18.2	-
	zif-sm-operationperformance	1.12.1	-
	zif-sm-operationsschedule	1.13.1	-
	zif-sm-testspecification	1.11.1	-
	zif-sm-workdefinition	2.12.2	-
	zif-sm-workperformance	1.9.1	-
	zif-sm-workschedule	1.9.1	-
	zif-udl-rtdwebapi	2.10.2	-
	zif-udl-dfawebapi	2.10.2	-
	zif-udl-dsentitywebapi	2.10.2	-
	zif-udl-node	2.10.2	-
	zif-udl-seed	2.10.2	-
	zif-udl-mdswebapi	2.10.2	-
	zif-udl-omconfigloader	2.10.2	-
	zif-udl-omconfigupdater	2.10.2	-
	zif-workflow	2.11.1	-
BPMN-процессы	QL.Equipment.MeasureChannel.Value.Changed	1.0.0	ДА
	QL.SampleAnalysisRequest.Workflow.Reselection	2.0.0	ДА
	QL.SampleAnalysisRequest.Workflow.Quarantine	1.0.0	ДА
	QL.AnalysisRequest.Workflow.PrintingDocuments	2.0.0	ДА
	QL.AnalysisRequest.Workflow.SamplePreparation	2.0.0	ДА
	QL.Protocol.Workflow.Create	5.1.0	ДА
	QL.CheckSeries.ChangeState	1.0.0	ДА
	QL.Auto.IQC.CheckOperationsRequest.ChangeState	1.0.0	ДА
	QL.Schedule.Workflow.ChangeState	2.0.0	ДА
	QL.Sample.Workflow.SampleStatusManagement	1.0.0	ДА

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	QL.NormDocs.Workflow.NormDocIntro	2.0.0	ДА
	QL.SmDirectories.GetOrCreateResourceUseType	1.0.0	ДА
	QL.SmDirectories.GetOrCreateWorkType	1.0.0	ДА
	QL.WorkSchedule.CreateWorkRequest	1.0.0	ДА
	QL.WorkPerformance.CreateWorkResponse	1.0.0	ДА
	QL.NormDocs.Workflow.ExecuteNormDocIntro	2.0.0	ДА
	QL.NormDocRequests.Workflow.EditNormDocIntro	1.0.0	ДА
	QL.Events.GetOrCreateEventType	1.0.0	ДА
	QL.Events.Workflow.ChangeState	1.0.0	ДА
	QL.SampleTech.ChangeState	2.0.0	ДА
	QL.SampleTech.Completing	3.1.0	ДА
	QL.DocumentArchive.CreateDocument	1.0.0	ДА
	QL.AnalysisRequest.ChangeState	1.1.0	ДА
	QL.SegmentRequirement.PrintingDocuments	1.0.0	ДА
	QL.Auto.SampleAnalysisRequest.ChangeStateByAnalysisRequestState	1.0.0	ДА
	QL.Auto.Sample.ChangeStateBySampleTechState	1.0.0	ДА
	QL.Auto.Equipment.SyncStateAndQuestionable	1.0.0	ДА
	QL.Auto.IQC.AnalysisRequestStateSync	1.0.0	ДА
	QL.IQC.CheckOperationsRequest.ChangeState	1.0.0	ДА
	QL.Auto.SampleAnalysisRequest.StartTimeExpiration	1.0.0	ДА
	QL.Auto.Equipment.Questionable	3.0.1	ДА
	QL.Auto.Equipment.ChangeState	0.1.0	ДА
	QL.CheckProcSeries.CreateReport	1.0.0	ДА
	QL.Auto.Equipment.ChangeStateAtCriticalFailure	1.0.0	ДА
	QL.Calibration.ChangeState	1.0.0	ДА
	QL.NormDocCopy.ChangeState	1.0.0	ДА
	QL.MaterialLot.Transfer.CreateInvoice	1.1.14	ДА
	QL.MaterialLot.ReceiptCreation	1.0.14	ДА
	QL.Store.MaterialLotReceiptCreation	1.2.1	ДА
	QL.MaterialLot.Transfer	1.1.3	ДА
	QL.MaterialLot.Transfer.CancelInvoice	1.0.4	ДА
	QL.MaterialLot.Transfer.Defect	1.1.0	ДА
	QL.MaterialLot.Transfer.DeleteInvoice	1.1.4	ДА
	QL.MaterialLot.Transfer.ExecuteInvoice	1.1.4	ДА
	QL.MaterialLot.Transfer.RejectInvoice	1.0.8	ДА
	QL.MaterialLot.Transfer.UpdateInvoice	1.1.10	ДА

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	QL.Auto.Project.Quota.ChangeStateByProjectState	1.0.0	ДА
	QL.CheckProcSeries.CreateTestQualityEvaluationReport	1.0.0	ДА
	QL.Auto.Project.ChangeStateByProjectStageState	1.0.1	ДА
	QL.Auto.Project.ProjectStage.ChangeStateByProjectState	1.0.0	ДА
	QL.Auto.Project.Operation.ChangeStateByProjectState	1.0.0	ДА
	QL.Auto.Project.ChangeStateByOperationState	1.0.1	ДА
	QL.Auto.Project.Operation.ChangeStateBySegmentState	1.0.1	ДА
	QL.Auto.Project.Segment.ChangeStateByOperationState	1.0.0	ДА
	QL.Auto.Project.Segment.ChangeStateByPerformanceState	1.0.1	ДА
	QL.Auto.Project.Segment.ChangePerformancesStatesByState	1.0.0	ДА
	QA.InconsistencyProcess	1.0.3	ДА
	QA.ClaimProcess	1.0.2	ДА
	SegmentStatusChanged	1.0.6	ДА
	QL.HR.SendFiles	1.0.0	ДА
	QL.Internal.Sample.ChangeState	1.0.0	ДА
	QL.Internal.SampleTech.ChangeState	1.0.0	ДА
NiFi-процессы	QL.AnalysisRequest.Code.Changing	1.0.0	ДА
	TestHromatekImporter	1.0.0	ДА
Dmn-схемы	QL.SampleAnalysisRequest.DMN.StateTransition	1.0.0	ДА
	QL.SampleAnalysisRequest.DMN.ChangeStateByAnalysisRequestState	1.0.0	ДА
	QL.AnalysisRequest.DMN.StateTransition	1.0.0	ДА
	QL.CheckSeries.DMN.AvailableStates	1.0.0	ДА
	QL.IQC.AnalysisRequest.DMN.StateTransition	1.0.0	ДА
	QL.IQC.CheckOperationsRequest.DMN.StateTransition	1.0.0	ДА
	QL.IQC.CheckOperationsRequest.DMN.AvailableStates	1.0.0	ДА
	QL.Schedule.DMN.StateTransition	1.0.0	ДА
	QL.Event.DMN.ChangeState	1.0.0	ДА
	QL.Sample.DMN.ChangeStateBySampleTechState	1.0.0	ДА
	QL.SampleJobOrder.DMN.ChangeStateBySampleAnalysisRequestState	1.0.0	ДА
	QL.SampleTech.DMN.ChangeState	2.0.0	ДА
	QL.SampleTech.DMN.ChangeTestResultStatus	1.0.0	ДА
	QL.SampleTechs.DMN.CommentRequired	1.0.0	ДА
	QL.SampleTechs.DMN.SampleTechCompleting	1.0.0	ДА
	QL.SampleTechs.DMN.TestResultStatusExecution	1.0.0	ДА
	QL.Equipment.DMN.SyncStateAndQuestionableCheckAllow	1.0.0	ДА

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	QL.Equipment.DMN.SyncState	1.0.0	ДА
	QL.Equipment.DMN.SyncQuestionable	1.0.0	ДА
	QL.Equipment.DMN.ChangeEquipmentStateByFailureSeverity	1.0.0	ДА
	QL.Equipment.DMN.NeedToChangeEquipmentStatusByFailure	1.0.0	ДА
	QL.Calibration.DMN.AvailableStates	1.0.0	ДА
	QL.NormDocCopy.DMN.AvailableStates	1.0.0	ДА
	QL.Project.DMN.ActionsByState	1.0.0	ДА
	QL.Project.DMN.ActionsByParentObjectState	1.0.0	ДА
	QL.Project.Operation.DMN.ComponentsAvailability	1.0.0	ДА
	QL.Project.Segment.DMN.ComponentsAvailability	1.0.0	ДА
	QL.Project.Quota.DMN.StateByProjectState	1.0.0	ДА
	QL.Project.DMN.StateByProjectStageState	1.0.0	ДА
	QL.Project.ProjectStage.DMN.StateByProjectState	1.0.0	ДА
	QL.Project.Operation.DMN.StateByProjectState	1.0.0	ДА
	QL.Project.DMN.StateByOperationState	1.0.0	ДА
	QL.Project.Operation.DMN.StateBySegmentState	1.0.0	ДА
	QL.Project.Segment.DMN.StateByOperationState	1.0.0	ДА
	QL.Project.Segment.DMN.StateByPerformanceState	1.0.0	ДА
	QL.Project.Segment.DMN.PerformanceStateByState	1.0.0	ДА
Внешние сервисы	envoy1.25.3	2.1.3-mode-writeless	ДА
	nginx1.23-alpine	2.3.17-mode-writeless	ДА
	zif-stolon-pg14	1.2.0-v4-240422	-
	zif-pgadmin4	8.5.0-v2-240422	-
	zif-cassandra	3.11.13-v20-240422	-
	zif-kafka	7.5.3-v5-240422	-
	zif-zookeeper	7.5.3-v5-240422	-
	zif-kafka-connect	7.5.3-v5-240422	-
	zif-kafka-exporter	1.7.0-v16-240422	-
	zif-kafka-ui	0.7.2-v1-240422	-

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
	zif-nifi	1.19.1-v29-240422	-
	zif-keycloak	21.1.2-v3-240422	-
	zif-oauth2-proxy	7.5.1-v11-240422	-
	zif-minio	2024.4.18-v1-240422	-
	zif-redis	6.2.14-v5-240219	-
	zif-redis-exporter	1.58.0-v1-240219	-
	zif-kafka-rest	7.5.3-v2-240219	-
	zif-schema-registry	7.5.3-v2-240219	-
	zif-jmx-exporter	0.20.0-v5-240219	-

8.5. Устаревшие сервисы

ВНИМАНИЕ! В данном разделе перечислены сервисы, **поддержка** которых с данного релиза будет **остановлена**, что может повлиять на работу взаимодействующих сервисов.

Таблица 8 Устаревшие сервисы в релизе ZQL 5.3.0

Тип сервиса/компонента	Сервис/компонент	Версия	ZQL for ZIIoT
Сервисы фронтенд	zui-app-ps	2.7.2	ДА
Сервисы бекэнд	zif-mnemoschemes-storage	6.1.7	-
	zif-sm-process	1.11.1	-

9. Контакты технической поддержки

Таблица 9 Контакты технической поддержки

Вид поддержки	Значение
Портал	https://jira.zyfra.com/servicedesk/customer/portal/42
Email	support@idpllc.ru

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.

Приложение 1 Обновление платформы ZIIoT с 2.13.x до 2.16.2 и Helm-чарта ZQL с 4.8.4-mod-a до 5.0.0-mod-a

Обновление платформы с 2.13.x на 2.16.2

Обновление с 2.13.x до 2.14.3

Подразумевается, что в файле `env-config/services/service-list-patch.yaml` присутствуют следующие сервисы:

- `bp-workers`
- `zif-sm-testspecification 1.9.2`
- `zif-file-storage 1.2.7`

При переходе на 2.14.3 сервис `zif-file-storage` будет развернут в составе инфраструктурных сервисов платформы, поэтому его необходимо удалить из файла `env-config/services/service-list-patch.yaml`.

Пример содержимого файла `env-config/services/service-list-patch.yaml` для обновления платформы до 2.14.3:

```
modules:  
  - name: bp-workers  
    enabled: true  
  - name: sm  
    services:  
      - name: zif-sm-testspecification  
        tag: 1.9.2
```

Обновление с 2.14.3 до 2.15.1

При переходе на 2.15.1 сервис `zif-sm-testspecification` версии 1.9.2 включен в состав платформенного модуля SM, поэтому его необходимо удалить из файла `env-config/services/service-list-patch.yaml`.

Шаги после обновления до 2.15.1

Необходимо провести миграций хештегов с помощью сервиса `zif-sm-directories` версии 1.9.2.

Для этого следует добавить 2 переменные развертыванию `zif-sm-directories` вручную:

```
- name: POSTGRES_HASHTAG_HOST  
  value: 'указать хост PostgreSQL для подключения'  
- name: POSTGRES_HASHTAG_PORT  
  value: 'указать порт PostgreSQL для подключения'
```

После запуска сервиса `zif-sm-directories` с вышеуказанными переменными будет запущен процесс миграции хештегов из базы `<tenantId>__zif-om-hashtags` в базу `<tenantId>__zif-sm-directories` в таблицу `hierarchyscopes_s`.

После успешного проведения миграции, т.е. при отсутствии ошибок в логах сервиса `zif-sm-directories`, следует:

- проверить успешность миграции, сравнив хештеги в базе `<tenantId>__zif-sm-directories` и `<tenantId>__zif-om-hashtags`, следующими запросами:

```
--- База <tenantId>__zif-om-hashtags
```

```
--- Получить список записей с ID тегов, которые должны мигрировать в базу
<tenantId>__zif-sm-directories
SELECT id
  FROM public.hashtag_s
 WHERE tt is null AND isdeleted = false

--- База <tenantId>__zif-sm-directories
--- Получить список записей с code 'hashtag%'
SELECT *
  FROM public.hierarchyscopes_s
 WHERE code like 'hashtag%' AND id in ('вставить', 'список', 'id', 'тегов',
'из', 'бд', '<tenantId>__zif-om-hashtags');

--- !!!
--- Количество строк (тегов) в результатах запросов должно совпадать
```

- отключить миграцию хештегов у сервиса zif-sm-directories, удалив из файла env-config/values/om/mod-secrets.yaml строку с секретом zif-om-hashtags-psql-pass вручную:

```
zifctl secrets dec --module om --env-path ./env-config
sed -i '/zif-om-hashtags-psql-pass/d' ./env-config/om/mod-secrets.dec.yaml
zifctl secrets enc --module om --env-path ./env-config
rm ./env-config/om/mod-secrets.dec.yaml
```

Обновление с 2.15.1 до 2.16.2

Перед обновлением добавить следующие переменные для сервиса zif-sm-directories в файл env-config/values/sm/zif-sm-directories.yaml.gotmpl, если они не были добавлены в релизе zifctl 2.16.2:

```
environmentVars:
  RESOURCE_USE_TYPE_CODE_UNIQUENESS_ENABLED: 'false'
  OPI_CODE_FROM_HASHTAG_NAME: 'true'
```

Также следует закомментировать параметр keycloak.baseUri в файле env-config/env-values.yaml.

Шаги после обновления до 2.16.2

Необходимо включение расширения uuid-ossf в базе <tenantId>__zif-sm-directories, т.к. пользователь <tenantId>__zif-sm-directories не обладает правами суперпользователя в PostgreSQL.

Включение расширения производится выполнением следующего SQL-запроса в базе <tenantId>__zif-sm-directories:

```
create extension if not exists "uuid-ossf"
```

Понизить версию zif-file-storage-default с 1.2.8 до 1.2.7, т.к. версия 1.2.8 имеет проблемы с загрузкой файлов со специальными символами, пробелом, русскими буквами в именах.

Обновление Helm-чарта ZQL с 4.8.4-mod-a до 5.0.0-mod-a

Перед обновлением следует изменить конфигурации файлов, содержащих EntitiesConfiguration.json для сервисов, из-за отказа от хештегов и вывода сервиса zif-om-hashtags из состава платформы. Подробная информация:

- Changelog.md Helm-чарта для релиза 5.0.0-mod-a;
- Приложение 2 Инструкция по переводу сервиса zqa-equipment на версию без хештегов;
- Приложение 3 Инструкция по переводу сервиса zqa-materials на версию без хештегов;
- Приложение 4 Инструкция по переводу сервиса zqa-personnel на версию без хештегов;
- Приложение 5 Инструкция по переводу сервиса zqa-samples на версию без хештегов.

После обновления релиза ZQL до версии 5.0.0-mod-a необходимо провести миграцию хештегов с помощью сервисов:

- zqa-equipment
- zqa-materials
- zqa-personnel
- zqa-samples

Для включения миграции вышеуказанных сервисов ZQL необходимо изменить или добавить ручную переменную в сервисах:

```
HASHTAGS_MIGRATION_ENABLED=true
```

После проведения миграции хештегов вышеуказанными сервисами ZQL значение переменной в сервисах следует изменить на значение по умолчанию:

```
HASHTAGS_MIGRATION_ENABLED=false
```

Приложение 2 Инструкция по переводу сервиса zqa-equipment на версию без хештегов

Предварительные операции

1. Убедиться, что развернута версия платформы 2.16.x и выполнена миграция хештегов в ОПИ в корневой ОПИ с наименованием HashTagRoot.

Внимание. Коды новых ОПИ должны совпадать с наименованием ОПИ.

2. Создать резервные копии баз данных сервисов:
 - a. zif-om-object;
 - b. zif-sm-directories;
 - c. zqa-equipment.

Основные изменения в новой версии сервиса zqa-equipment

1. Изменен конфигурационный файл EntitiesConfiguration.json сервиса zqa-equipment.
2. Удалены следующие переменные:
 - a. ``REST_ZIF_OM_HASHTAGS_URL``;
 - b. ``CACHE_ABSOLUTE_EXPIRATION_TIME_HASHTAG``.
3. Добавлены следующие переменные:
 - a. ``HASHTAGS_MIGRATION_ENABLED`` - разрешена миграция хештегов, по умолчанию имеет значение ``false``;
 - b. ``HASHTAGS_MIGRATION_DELETE_DEPRECATED`` - разрешено удаление устаревших хештегов, по умолчанию имеет значение ``false``.

Шаги по обновлению конфигурационного файла для существующего развертывания

Важно. В конфигурационном файле связь между различными сущностями устанавливается с использованием свойства **alias**. Если одна сущность ссылается на другую, определенную в данной конфигурации, то в поле ссылки указывается значение свойства **alias** целевой сущности. Исключительно свойство **alias** является ключевым для установления связей между сущностями в конфигурации и, поэтому значения в нем не могут быть изменены. Остальные свойства могут быть изменены при необходимости.

1. В файле EntitiesConfiguration.json, который используется сервисом zqa-equipment, выполнить следующие изменения:
 - a. В разделе **models** удалить раздел **tags**:

```
"models": [  
  {  
    "alias": "EM",  
    "create": "true",  
    "modelPrototype": "EMP",
```

```
    "name": {
      "en": "Equipment model",
      "ru": "Модель оборудования"
    },
    "code": "EquipmentModel",
    "description": {
      "en": "Equipment model.",
      "ru": "Модель оборудования."
    },
    "tags": [          // удалить раздел
      "QL"
    ]
  }
],
```

b. В разделе **entities** для сущностей:

- i. с алиасом **EquipmentClass** раздел **tags** переименовывать в **hierarchyScopes**;

```
{
  "alias": "EquipmentClass",
  "implementedBy": "Class",
  "modelPrototype": "EMP",
  "tags": [          // переименовать в hierarchyScopes
    "Equipment"
  ],
}
```

- ii. с алиасом **MeasureChannelClass** раздел **tags** переименовать в **hierarchyScopes**, алиас ОПИ переименовать с MeasureChannel на MeasureChannelValue;

```
{
  "alias": "MeasureChannelClass",
  "implementedBy": "Class",
  "modelPrototype": "EMP",
  "tags": [          // переименовать в hierarchyScopes
    "MeasureChannel"          // переименовать в
MeasureChannelValue
  ],
}
```

- iii. с алиасом **ExtraPropertiesClass** раздел **tags** переименовать в **hierarchyScopes**, алиас ОПИ переименовать с ExtraProperties на EquipmentExtraProperties;

```
{
  "alias": "ExtraPropertiesClass",
  "implementedBy": "Class",
  "modelPrototype": "EMP",
  "tags": [          // переименовать в hierarchyScopes
    "ExtraProperties"          // переименовать в
EquipmentExtraProperties
  ],
}
```

c. В разделе **classes** для всех сущностей удалить раздел **tags**:

```
"classes": [
  {
    "alias": "Equipment",
    "create": "true",
    "modelPrototype": "EMP",
    "name": {
      "en": "Laboratory equipment",
      "ru": "Лабораторное оборудование"
    },
  },
]
```

```
    "code": "QL.Equipment",
    "description": {
      "en": "Laboratory equipment.",
      "ru": "Лабораторное оборудование."
    },
    "properties": [
      "Kind",
      "Status",
      "SerialNumber",
      "InventoryNumber",
      "Questionable",
      "QuestionableReason"
    ],
    "tags": [
      // удалить раздел
      "QL"
    ]
  },
},
```

d. В разделе **directories** для всех сущностей удалить раздел **tags**:

```
"directories": [
  {
    "alias": "EquipmentKinds",
    "create": "true",
    "name": {
      "en": "Laboratory equipment kinds",
      "ru": "Виды лабораторного оборудования"
    },
    "code": "QL.EquipmentKinds",
    "description": {
      "en": "Laboratory equipment kinds.",
      "ru": "Виды лабораторного оборудования."
    },
    "tags": [],
    // удалить раздел
  },
],
```

e. В раздел **hierarchyScopes** после области с алиасом QL добавить следующие настройки на ОПИ:

```
1313 "hierarchyScopes": [
1314   {
1315     "alias": "QL",
1316     "create": "true",
1317     "code": "QL",
1318     "name": {
1319       "en": "Quality Lab",
1320       "ru": "Лаборатория качества"
1321     },
1322     "description": {
1323       "en": "Scope for Quality Lab.",
1324       "ru": "Область применения для Quality Lab."
1325     }
1326   },
1327   {
1328     "alias": "MeasureChannelParameterGroup",
1329     "create": "true",
1330     "code": "QL.MeasureChannel.ParameterGroup",
1331     "name": {
1332       "en": "Measure channel parameter group",
1333       "ru": "Группа параметров измерительного канала"
1334     },
1335     "description": {
1336       "en": "Used to group primitives of measure channel parameters.",
1337       "ru": "Используется для группировки примитивов параметров измерительного канала."
1338     }
1339   },
1340 ],
```

← Добавить настройки на ОПИ

```
"alias": "Equipment",
"create": "true",
"code": "QL.Equipment",
"parent": "QL",
"name": {
  "en": "Equipment",
  "ru": "Оборудование"
},
"description": {
  "en": "Used as root hierarchy scope for equipment.",
  "ru": "Корневая область применения для оборудования."
}
},
{
  "alias": "EquipmentExtraProperties",
  "create": "true",
  "code": "QL.Equipment.ExtraProperties",
  "parent": "Equipment",
  "name": {
    "en": "Equipment extra properties",
    "ru": "Дополнительные свойства оборудования"
  },
  "description": {
    "en": "Used for equipment extra properties class.",
    "ru": "Используется для класса дополнительных свойств оборудования."
  }
},
{
  "alias": "MeasureChannel",
  "create": "true",
  "code": "QL.MeasureChannel",
  "parent": "Equipment",
  "name": {
    "en": "MeasureChannel",
    "ru": "Измерительный канал"
  },
  "description": {
    "en": "Used as root hierarchy scope for measure channel.",
    "ru": "Корневая область применения для измерительного канала."
  }
},
{
  "alias": "MeasureChannelValue",
  "create": "true",
  "code": "QL.MeasureChannel.Value",
  "parent": "MeasureChannel",
  "name": {
    "en": "Measure channel value",
    "ru": "Значение измеряемой величины"
  },
  "description": {
    "en": "Used for class and property of measured value.",
    "ru": "Используется для класса и свойства измеряемой величины."
  }
},
},
```

- f. В разделе **hierarchyScopes** для сущностей с алиасами `MeasureChannelParameterGroup`, `MeasureChannelParameterTypes` добавить алиас родительской ОПИ `"parent": "MeasureChannel"`:

```
{
  "alias": "MeasureChannelParameterGroup",
```

```
ОПИ
    "create": "true",
    "code": "QL.MeasureChannel.ParameterGroup",
    "parent": "MeasureChannel", // добавленный алиас родительской
    "name": {
        "en": "Measure channel parameter group",
        "ru": "Группа параметров измерительного канала"
    },
    "description": {
        "en": "Used to group primitives of measure channel
parameters.",
        "ru": "Используется для группировки примитивов параметров
измерительного канала."
    }
},
```

2. Выполнить миграцию хештегов из корневой ОПИ с наименованием HashTagRoot в ОПИ с иерархией для ZQL. Перед запуском сервиса zqa-equipment необходимо проверить соответствие:
 - a. идентификаторы ОПИ = идентификаторы ранее использовавшихся хештегов, описаны в разделе tags (обязательно).
 - b. коды ОПИ = коды ОПИ из конфигураций из раздела hierarchyScopes (обязательно).
 - c. наименования ОПИ = наименования ОПИ из конфигураций из раздела hierarchyScopes (не обязательно, рекомендуется).
 - d. коды родительских ОПИ = коды ОПИ, описанные в родительской конфигурации (не обязательно, рекомендуется)
 - e. коды ОПИ из конфигурации должны быть уникальными (обязательно)

Пары для соответствия алиасов <алиас хештега, алиас ОПИ> из конфигурации:

- <Equipment, Equipment>
- <MeasureChannel, MeasureChannelValue>
- <ExtraProperties, EquipmentExtraProperties>

Примечание. Если не выполнять миграцию, то перед запуском необходимо проверить соответствие (см. выше) для корректного отображения в пользовательском интерфейсе свойств, которые ранее были отмечены хештегами. В этом случае ОПИ будут отображены как дочерние для корневой ОПИ HashTagRoot.

Миграция может быть выполнена:

- **Автоматически.** В сервисе zqa-equipment переменная HASHTAGS_MIGRATION_ENABLED имеет значение true. При запуске сервиса ОПИ из родительского узла HashTagRoot перемещаются в иерархию с корневым ОПИ QL (Лаборатория качества), получает код, наименование, описание из соответствующей конфигурации ОПИ. Созданные механизмом автосоздания сущностей ОПИ (для пар соответствия, см. выше) по конфигурации будут помечены суффиксом deprecated. При включенной переменной HASHTAGS_MIGRATION_DELETE_DEPRECATED = true ОПИ с суффиксом deprecated будут удалены автоматически.
- **Вручную.** В сервисе zqa-equipment переменная HASHTAGS_MIGRATION_ENABLED имеет значение false. Необходимо выполнить ручное изменение кода, наименования и перемещение в соответствии с конфигурацией (например, через Swagger или пользовательский интерфейс OM)

3. Запустить сервис zqa-equipment. При запуске сервиса выполнится создание сущностей по умолчанию, и при HASHTAGS_MIGRATION_ENABLED = true выполнится описанная выше миграция.

4. Проверить, что функциональность получения классов оборудования, классов свойств значений измерительного канала, классов дополнительных свойств и дополнительных свойств работает как раньше. После этого удалить раздел хештегов из конфигурационного файла EntitiesConfiguration.json.

```
"tags": [ // удалить весь раздел
  {
    "alias": "QL",
    "create": "true",
    "name": "QL"
  },
  {
    "alias": "Equipment",
    "create": "true",
    "name": "QL.Equipment"
  },
  {
    "alias": "MeasureChannel",
    "create": "true",
    "name": "QL.MeasureChannel.Value"
  },
  {
    "alias": "ExtraProperties",
    "create": "true",
    "name": "QL.Equipment.ExtraProperties"
  }
],
```

В результате выполненных действий в модели ОПИ будет отображена иерархия ОПИ для ZQL.

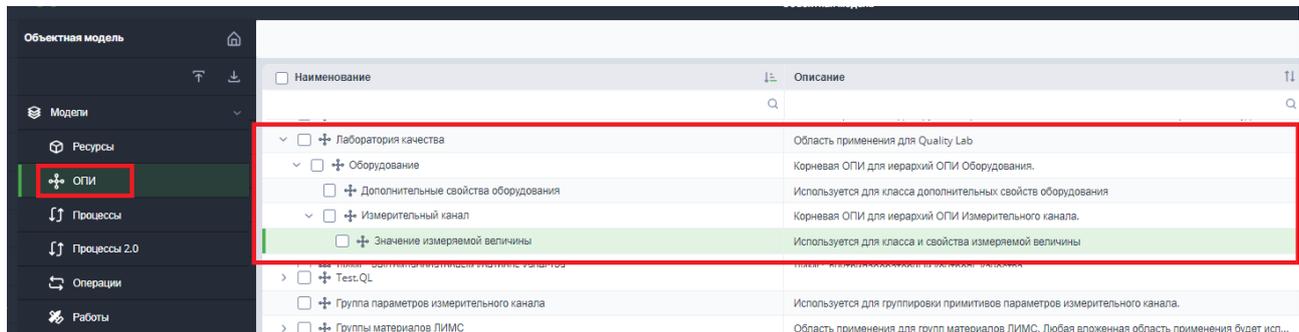


Рисунок 0.1 Иерархия ОПИ для ZQL

Приложение 3 Инструкция по переводу сервиса zqa-materials на версию без хештегов

Предварительные операции

1. Убедиться, что развернута версия платформы 2.16.x и выполнена миграция хештегов в ОПИ в корневой ОПИ с наименованием HashTagRoot.

Внимание. Коды новых ОПИ должны совпадать с наименованием ОПИ.

2. Создать резервные копии баз данных сервисов:
 - a. zif-om-object;
 - b. zif-sm-directories;
 - c. zqa-materials.

Основные изменения в новой версии сервиса zqa-materials

1. Изменен конфигурационный файл EntitiesConfiguration.json сервиса zqa-materials.
2. Удалены следующие переменные:
 - a. `REST_ZIF_OM_HASHTAGS_URL`;
 - b. `CACHE_ABSOLUTE_EXPIRATION_TIME_HASHTAG`.
3. Добавлены следующие переменные:
 - a. `HASHTAGS_MIGRATION_ENABLED` - разрешена миграция хештегов, по умолчанию имеет значение `false`;
 - b. `HASHTAGS_MIGRATION_DELETE_DEPRECATED` - разрешено удаление устаревших хештегов, по умолчанию имеет значение `false`.
4. Изменена переменная:
 - a. `REST_ZIF_OM_VALUETYPES_URL` теперь указывает на адрес сервиса `zif-om-uom` (значение по умолчанию `http://zif-om-uom:8080`)

Шаги по обновлению конфигурационного файла для существующего развертывания

Важно. В конфигурационном файле связь между различными сущностями устанавливается с использованием свойства **alias**. Если одна сущность ссылается на другую, определенную в данной конфигурации, то в поле ссылки указывается значение свойства **alias** целевой сущности. Исключительно свойство **alias** является ключевым для установления связей между сущностями в конфигурации и, поэтому значения в нем не могут быть изменены. Остальные свойства могут быть изменены при необходимости.

1. В файле EntitiesConfiguration.json, который используется сервисом zqa-materials, выполнить следующие изменения:
 - a. В разделе **models** удалить раздел **tags**:

```
"models": [  
  {  
    "alias": "MP",  
    "create": "true",  
    "modelPrototype": "MMP",
```

```
"name": {
  "en": "Material model",
  "ru": "Модель материалов"
},
"code": "MaterialModel",
"description": {
  "en": "Material model",
  "ru": "Модель материалов"
},
"tags": [
  "QL"
]
}
],
```

b. В разделе **classes** для всех сущностей удалить раздел **tags**:

```
"classes": [
  {
    "alias": "Material",
    "create": "true",
    "modelPrototype": "MMP",
    "name": {
      "en": "Laboratory material",
      "ru": "Материал для исследования в лаборатории"
    },
    "code": "QL.Material",
    "description": {
      "en": "Laboratory material",
      "ru": "Материал для исследования в лаборатории"
    },
    "properties": [
      "IsActive",
      "Code"
    ],
    "tags": [ // удалить раздел
      "QL"
    ]
  },
],
```

c. В разделе **hierarchyScopes** добавить корневую ОПИ с алиасом QL:

```
36 "hierarchyScopes":
37 [
38   { ← Добавить ОПИ
39     "alias": "MaterialGroups",
40     "create": "true",
41     "name": {
42       "en": "Groups of LIMS material",
43       "ru": "Группы материалов ЛИМС"
44     },
45     "code": "QL.MaterialGroups",
46     "description": {
47       "en": "Scope for LIMS material groups",
48       "ru": "Область применения для групп материалов ЛИМС"
49     }
50   }
51 ],
```

```
{
  "alias": "QL",
  "create": "true",
  "code": "QL",
  "name": {
    "en": "Quality Lab",
```

```
    "ru": "Лаборатория качества"  
  },  
  "description": {  
    "en": "Scope for Quality Lab",  
    "ru": "Область применения для Quality Lab"  
  }  
},
```

- d. В разделе **hierarchyScopes** для сущностей с алиасом MaterialGroups добавить алиас родительской ОПИ "parent": "QL":

```
"hierarchyscopes":  
  [  
    {  
      "alias": "MaterialGroups",  
      "create": "true",  
      "name": {  
        "en": "Groups of LIMS material",  
        "ru": "Группы материалов ЛИМС"  
      },  
      "code": "QL.MaterialGroups",  
      "parent": "QL", // добавленный алиас родительской ОПИ  
      "description": {  
        "en": "Scope for LIMS material groups",  
        "ru": "Область применения для групп материалов ЛИМС"  
      }  
    }  
  ],
```

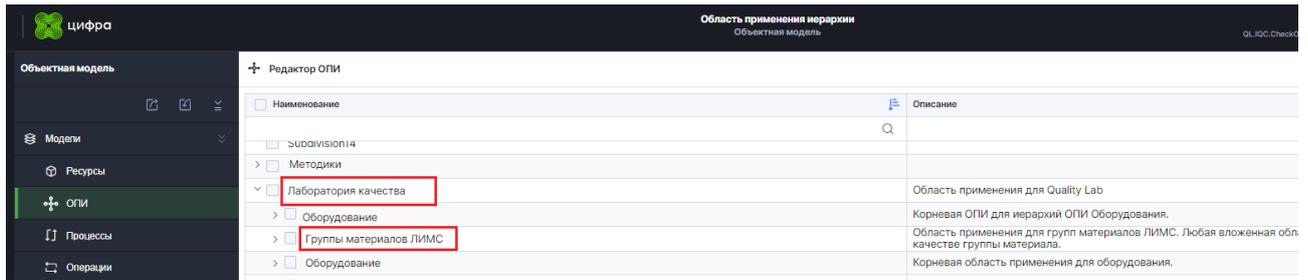
2. Выполнить миграцию хештегов из корневой ОПИ с наименованием HashTagRoot в ОПИ с иерархией для ZQL. Перед запуском сервиса zqa-materials необходимо проверить соответствие:
- коды родительских ОПИ = коды ОПИ, описанные в родительской конфигурации (не обязательно, рекомендуется)
 - коды ОПИ из конфигурации должны быть уникальными (обязательно)

Примечание. Если не выполнять миграцию, то перед запуском необходимо проверить соответствие (см. выше) для корректного отображения в пользовательском интерфейсе свойств, которые ранее были отмечены хештегами. В этом случае ОПИ будут отображены как дочерние для корневой ОПИ HashTagRoot.

Миграция может быть выполнена:

- **Автоматически.** В сервисе zqa-materials переменная HASHTAGS_MIGRATION_ENABLED имеет значение true. При запуске сервиса ОПИ из родительского узла HashTagRoot перемещаются в иерархию с корневым ОПИ QL (Лаборатория качества).
 - **Вручную.** В сервисе zqa-materials переменная HASHTAGS_MIGRATION_ENABLED имеет значение false. Необходимо выполнить ручное изменение идентификатора родительской ОПИ для ОПИ групп материалов в соответствии с конфигурацией (например, через Swagger).
3. Запустить сервис zqa-materials. При запуске сервиса выполнится создание сущностей по умолчанию, и при HASHTAGS_MIGRATION_ENABLED = true выполнится описанная выше миграция.

В результате выполненных действий в модели ОПИ будет отображена иерархия ОПИ для ZQL.



Наименование	Описание
Suborganization	
Методики	
Лаборатория качества	Область применения для Quality Lab
Оборудование	Корневая ОПИ для иерархий ОПИ Оборудования.
Группы материалов ЛИМС	Область применения для групп материалов ЛИМС. Любая вложенная область качества группы материала.
Оборудование	Корневая область применения для оборудования.

Рисунок 0.1 Иерархия ОПИ для ZQL

Приложение 4 Инструкция по переводу сервиса zqa-personnel на версию без хештегов

Предварительные операции

1. Убедиться, что развернута версия платформы 2.16.x и выполнена миграция хештегов в ОПИ в корневой ОПИ с наименованием HashTagRoot.

Внимание. Коды новых ОПИ должны совпадать с наименованием ОПИ.

2. Создать резервные копии баз данных сервисов:
 - a. zif-om-object;
 - b. zif-sm-directories;
 - c. zqa-personnel.

Основные изменения в новой версии сервиса zqa-personnel

1. Изменен конфигурационный файл EntitiesConfiguration.json сервиса zqa-personnel.
2. Удалены следующие переменные:
 - a. ``REST_ZIF_OM_HASHTAGS_URL``;
 - b. ``CACHE_ABSOLUTE_EXPIRATION_TIME_HASHTAG``.
3. Добавлены следующие переменные:
 - a. ``REST_ZIF_SM_DIRECTORIES_URL`` - адрес сервиса справочников для семантических моделей, по умолчанию имеет значение ``http://zif-sm-directories:8080``;
 - b. ``HASHTAGS_MIGRATION_ENABLED`` - разрешена миграция хештегов, по умолчанию имеет значение ``false``;
 - c. ``HASHTAGS_MIGRATION_DELETE_DEPRECATED`` - разрешено удаление устаревших хештегов, по умолчанию имеет значение ``false``.
4. Изменены следующие переменные:
 - a. ``REST_ZIF_OM_PROPERTIES_VIEW_URL`` теперь должна указывать на адрес сервиса ``zif-om-object``, по умолчанию имеет значение ``http://zif-om-object:8080/propertyview``;
 - b. ``REST_ZIF_OM_VALUETYPES_URL`` теперь должна указывать на адрес сервиса ``zif-om-uom``, по умолчанию имеет значение ``http://zif-om-uom:8080``.

Шаги по обновлению конфигурационного файла для существующего развертывания

Важно. В конфигурационном файле связь между различными сущностями устанавливается с использованием свойства **alias**. Если одна сущность ссылается на другую, определенную в данной конфигурации, то в поле ссылки указывается значение свойства **alias** целевой сущности. Исключительно свойство **alias** является ключевым для установления связей между сущностями в конфигурации и, поэтому значения в нем не могут быть изменены. Остальные свойства могут быть изменены при необходимости.

1. В файле EntitiesConfiguration.json, который используется сервисом zqa-personnel, выполнить следующие изменения:

a. В разделе **models** удалить раздел **tags**:

```
"models": [  
  {  
    "alias": "PM",  
    "create": "true",  
    "modelPrototype": "PMP",  
    "name": {  
      "en": "Personnel model",  
      "ru": "Модель персонала"  
    },  
    "code": "PersonnelModel",  
    "description": {  
      "en": "Personnel model.",  
      "ru": "Модель персонала."  
    },  
    "tags": [ // удалить раздел  
      "QL"  
    ]  
  },  
],
```

b. В разделе **entities** для сущности с алиасом **PersonnelClass** раздел **tags** переименовывать в **hierarchyScopes**:

```
{  
  "alias": "PersonnelClass",  
  "implementedBy": "Class",  
  "modelPrototype": "PMP",  
  "tags": [ // переименовать в hierarchyScopes  
    "Personnel"  
  ],  
}
```

c. В разделе **classes** для всех сущностей удалить раздел **tags**:

```
"classes": [  
  {  
    "alias": "PersonnelSystem",  
    "create": "true",  
    "modelPrototype": "PMP",  
    "name": {  
      "en": "Personnel, system properties",  
      "ru": "Персонал, системные свойства"  
    },  
    "code": "Personnel.System",  
    "description": {  
      "en": "Personnel, system properties. Changes to the property  
set are not allowed.",  
      "ru": "Персонал, системные свойства. Изменения в наборе  
свойств не допускаются."  
    },  
    "properties": [  
      "User"  
    ],  
    "tags": [ // удалить раздел  
      "QL"  
    ]  
  },  
],
```

d. Удалить раздел **directories**, если он пустой.

```
"directories": [], // удалить раздел
```

e. Добавить раздел **hierarchyScopes**:

```
"hierarchyScopes":  
[  
  {  
    "alias": "QL",  
    "create": "true",  
    "code": "QL",  
    "name": {  
      "en": "Quality Lab",  
      "ru": "Лаборатория качества"  
    },  
    "description": {  
      "en": "Scope for Quality Lab",  
      "ru": "Область применения для Quality Lab"  
    }  
  },  
  {  
    "alias": "Personnel",  
    "create": "true",  
    "code": "QL.Personnel",  
    "name": {  
      "en": "QL.Personnel",  
      "ru": "Персонал"  
    },  
    "parent": "QL",  
    "description": {  
      "en": "Used as root hierarchy scope for personnel",  
      "ru": "Корневая область применения для персонала"  
    }  
  }  
]
```

2. Выполнить миграцию хештегов из корневой ОПИ с наименованием HashTagRoot в ОПИ с иерархией для ZQL. Перед запуском сервиса zqa-personnel необходимо проверить соответствие:

- идентификаторы ОПИ = идентификаторы ранее использовавшихся хештегов, описаны в разделе tags (обязательно).
- коды ОПИ = коды ОПИ из конфигураций из раздела hierarchyScopes (обязательно).
- наименования ОПИ = наименования ОПИ из конфигураций из раздела hierarchyScopes (не обязательно, рекомендуется).
- коды родительских ОПИ = коды ОПИ, описанные в родительской конфигурации (не обязательно, рекомендуется)
- коды ОПИ из конфигурации должны быть уникальными (обязательно)

Пары для соответствия алиасов <алиас хештега, алиас ОПИ> из конфигурации:

- <Personnel, Personnel>

Примечание. Если не выполнять миграцию, то перед запуском необходимо проверить соответствие (см. выше) для корректного отображения в пользовательском интерфейсе свойств, которые ранее были отмечены хештегами. В этом случае ОПИ будут отображены как дочерние для корневой ОПИ HashTagRoot.

Миграция может быть выполнена:

- Автоматически.** В сервисе zqa-personnel переменная HASHTAGS_MIGRATION_ENABLED имеет значение true. При запуске сервиса ОПИ из родительского узла HashTagRoot перемещаются в иерархию с корневым ОПИ QL (Лаборатория качества), получает код, наименование, описание из соответствующей конфигурации ОПИ. Созданные механизмом автосоздания сущностей ОПИ (для пар соответствия, см. выше) по конфигурации будут помечены суффиксом deprecated.

При включенной переменной `HASHTAGS_MIGRATION_DELETE_DEPRECATED = true` ОПИ с суффиксом `deprecated` будут удалены автоматически.

- **Вручную.** В сервисе `zqa-personnel` переменная `HASHTAGS_MIGRATION_ENABLED` имеет значение `false`. Необходимо выполнить ручное изменение кода, наименования и перемещение в соответствии с конфигурацией (например, через Swagger или пользовательский интерфейс OM)
3. Запустить сервис `zqa-personnel`. При запуске сервиса выполнится создание сущностей по умолчанию, и при `HASHTAGS_MIGRATION_ENABLED = true` выполнится описанная выше миграция.
 4. Проверить, что функциональность получения классов персонала работает как раньше. После этого удалить раздел хештегов из конфигурационного файла `EntitiesConfiguration.json`.

```
"tags": [ // удалить весь раздел
  {
    "alias": "Personnel",
    "create": "true",
    "name": "Personnel"
  },
  {
    "alias": "QL",
    "create": "true",
    "name": "QL"
  }
]
```

В результате выполненных действий в модели ОПИ будет отображена иерархия ОПИ для ZQL.

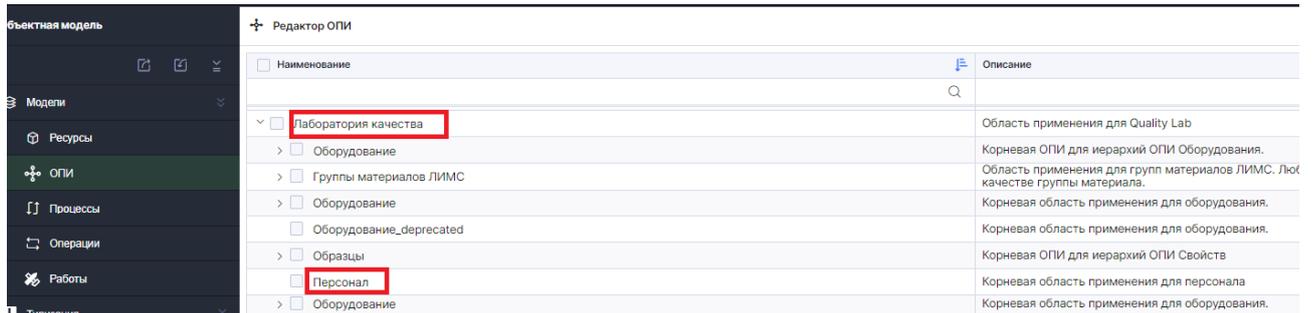


Рисунок 0.1 Иерархия ОПИ для ZQL

Приложение 5 Инструкция по переводу сервиса zqa-samples на версию без хештегов

Предварительные операции

1. Убедиться, что развернута версия платформы 2.16.x и выполнена миграция хештегов в ОПИ в корневой ОПИ с наименованием HashTagRoot. Среди дочерних для HashTagRoot должен быть ОПИ с именем QL.Sample.ExtraFields (имя хештега раньше настраивалось на фронтенд-сервисе zui-app-qa в переменной OM_SAMPLE_EXTRAPROPERTYSET_CLASS_HASHTAG_NAME, может отличаться от указанного здесь).

Внимание. Коды новых ОПИ должны совпадать с наименованием ОПИ.

2. Создать резервные копии баз данных сервисов:
 - a. zif-om-object;
 - b. zif-sm-directories;
 - c. zqa-samples.

Основные изменения в новой версии сервиса zqa-samples

1. Изменен конфигурационный файл EntitiesConfiguration.json сервиса zqa-samples.
2. Удалены следующие переменные:
 - a. ``REST_ZIF_OM_HASHTAGS_URL``;
 - b. ``CACHE_ABSOLUTE_EXPIRATION_TIME_HASHTAG``.
3. Добавлены следующие переменные:
 - a. ``REST_ZIF_SM_DIRECTORIES_URL`` - адрес сервиса справочников для семантических моделей, по умолчанию имеет значение ``http://zif-sm-directories:8080``.
 - b. ``HASHTAGS_MIGRATION_ENABLED`` - разрешена миграция хештегов, по умолчанию имеет значение ``false``;
 - c. ``HASHTAGS_MIGRATION_DELETE_DEPRECATED`` - разрешено удаление устаревших хештегов, по умолчанию имеет значение ``false``.
4. Изменена переменная:
 - a. ``REST_ZIF_OM_PROPERTIES_VIEW_URL`` теперь должна указывать на адрес сервиса ``zif-om-object``, по умолчанию имеет значение ``http://zif-om-object:8080/propertyview``.

Шаги по обновлению конфигурационного файла для существующего развертывания

Важно. В конфигурационном файле связь между различными сущностями устанавливается с использованием свойства **alias**. Если одна сущность ссылается на другую, определенную в данной конфигурации, то в поле ссылки указывается значение свойства **alias** целевой сущности. Исключительно свойство **alias** является ключевым для установления связей между сущностями в конфигурации и, поэтому значения в нем не могут быть изменены. Остальные свойства могут быть изменены при необходимости.

1. В файле EntitiesConfiguration.json, который используется сервисом zqa-samples, выполнить следующие изменения:

- a. В разделе **modelPrototypes** добавить настройку на прототип модели, к которому привязывается класс дополнительных свойств образца. Убедиться, что код прототипа модели в конфигурации совпадает с кодом прототипа модели в ОМ (обязательно). Наименование, описание привести в соответствие (не обязательно, рекомендуется).

```
{
  "alias": "MMP",
  "create": "true",
  "name": {
    "en": "Material model prototype",
    "ru": "Прототип модели материалов"
  },
  "code": "MaterialModelPrototype",
  "description": {
    "en": "Material model prototype",
    "ru": "Прототип модели материалов"
  }
}
```

- b. В раздел **entities** для сущности класса дополнительных свойств добавить настройку с алиасом `SampleExtraPropertySet`:

```
{
  "alias": "SampleExtraPropertySet",
  "implementedBy": "class",
  "modelPrototype": "MMP",
  "hierarchyScopes": [
    "SampleExtraFields"
  ],
  "properties": [
    {
      "name": "Id",
      "source": "Id"
    },
    {
      "name": "Name",
      "source": "Name"
    }
  ]
}
```

- c. В разделе **directories** для всех сущностей удалить раздел **tags**:

```
"directories": [
  {
    "alias": "SampleTypes",
    "create": "true",
    "name": {
      "en": "Sample types",
      "ru": "Типы образцов"
    },
    "code": "QL.SampleType",
    "description": {
      "en": "Sample types.",
      "ru": "Типы образцов."
    },
    "tags": [], // удалить раздел
  }
]
```

- d. Удалить раздел **tags**:

```
"tags": [ // удалить раздел
  {
```

```
        "alias": "QL",  
        "create": "true",  
        "name": "QL"  
    }  
]
```

е. Добавить раздел **hierarchyScopes**.

```
"hierarchyScopes": [  
  {  
    "alias": "QL",  
    "create": "true",  
    "code": "QL",  
    "name": {  
      "en": "Quality Lab",  
      "ru": "Лаборатория качества"  
    },  
    "description": {  
      "en": "Scope for Quality Lab",  
      "ru": "Область применения для Quality Lab"  
    }  
  },  
  {  
    "alias": "Samples",  
    "create": "true",  
    "code": "QL.Samples",  
    "parent": "QL",  
    "name": {  
      "en": "QL.Samples",  
      "ru": "Образцы"  
    },  
    "description": {  
      "en": "Used as the root folder for Samples hierarchy scope",  
      "ru": "Корневая область применения для образцов"  
    }  
  },  
  {  
    "alias": "SampleExtraFields",  
    "create": "true",  
    "code": "QL.Sample.ExtraFields",  
    "parent": "Samples",  
    "name": {  
      "en": "QL.Samples.ExtraFields",  
      "ru": "Дополнительные свойства образца"  
    },  
    "description": {  
      "en": "Scope for sample extra fields",  
      "ru": "Область применения для идентификации дополнительных  
свойств образца"  
    }  
  },  
],  
]
```

ф. Если необходима функциональность управления дополнительными свойствами образца, то в раздел **hierarchyScopes** следует добавить настройку ОПИ с алиасом **SampleExtraField**:

```
{  
  "alias": "SampleExtraField",  
  "create": "true",  
  "code": "QL.Sample.ExtraFields",  
  "parent": "Samples",  
  "name": {
```

```
        "en": "QL.Samples.ExtraFields",  
        "ru": "Дополнительные свойства образца"  
    },  
    "description": {  
        "en": "Scope for sample extra fields",  
        "ru": "Область применения для идентификации дополнительных  
свойств образца"  
    }  
}
```

2. Выполнить миграцию хештегов из корневой ОПИ с наименованием HashTagRoot в ОПИ с иерархией для ZQL. Перед запуском сервиса zqa-equipment необходимо проверить соответствие:
 - a. идентификатор ОПИ для классов дополнительных свойств = идентификатор ранее использовавшегося хештега, описанного в переменной OM_SAMPLE_EXTRAPROPERTYSET_CLASS_HASHTAG_NAME (обязательно).
 - b. коды ОПИ = коды ОПИ из конфигураций из раздела hierarchyScopes (обязательно).
 - c. наименования ОПИ = наименования ОПИ из конфигураций из раздела hierarchyScopes (не обязательно, рекомендуется).
 - d. коды родительских ОПИ = коды ОПИ, описанные в родительской конфигурации (не обязательно, рекомендуется)
 - e. коды ОПИ из конфигурации должны быть уникальными (обязательно)

Пары для соответствия алиасов <ИМЯ хештега, алиас ОПИ> из конфигурации:

- <"QL.Sample.ExtraFields", SampleExtraFields>

Т.к. настройки на хештег классов дополнительных свойств не было в разделе tags конфигурации, чтобы миграция для мигрированного из хештега ОПИ выполнялась, имя этой ОПИ должно быть QL.Sample.ExtraFields. Если имя ОПИ отличается, то предварительно перед миграцией его можно переименовать. Родительской ОПИ должна быть HashTagRoot.

Примечание. Если не выполнять миграцию, то перед запуском необходимо проверить соответствие (см. выше) для корректного отображения в пользовательском интерфейсе свойств, которые ранее были отмечены хештегами. В этом случае ОПИ будут отображены как дочерние для корневой ОПИ HashTagRoot.

Миграция может быть выполнена:

- **Автоматически.** В сервисе zqa-samples переменная HASHTAGS_MIGRATION_ENABLED имеет значение true. При запуске сервиса ОПИ из родительского узла HashTagRoot перемещаются в иерархию с корневым ОПИ QL (Лаборатория качества), получает код, наименование, описание из соответствующей конфигурации ОПИ. Созданные механизмом автосоздания сущностей ОПИ (для пар соответствия, см. выше) по конфигурации будут помечены суффиксом deprecated. При включенной переменной HASHTAGS_MIGRATION_DELETE_DEPRECATED = true ОПИ с суффиксом deprecated будут удалены автоматически.
 - **Вручную.** В сервисе zqa-samples переменная HASHTAGS_MIGRATION_ENABLED имеет значение false. Необходимо выполнить ручное изменение кода, наименования и перемещение в соответствии с конфигурацией (например, через Swagger или пользовательский интерфейс OM)
3. Запустить сервис zqa-samples. При запуске сервиса выполнится создание сущностей по умолчанию, и при HASHTAGS_MIGRATION_ENABLED = true выполнится описанная выше миграция.
 4. Проверить, что функциональность дополнительных свойств образца работает как раньше. В результате выполненных действий в модели ОПИ будет отображена иерархия ОПИ для ZQL.

<input type="checkbox"/> Лаборатория качества	Область применения для Quality Lab
> <input type="checkbox"/> Оборудование	Корневая область применения для оборудования.
<input type="checkbox"/> Оборудование_deprecated	Корневая область применения для оборудования.
<input checked="" type="checkbox"/> Образцы	Корневая ОПИ для иерархий ОПИ Свойств
<input type="checkbox"/> Класс дополнительных свойств образца	Область применения для класса дополнительных свойств образца
<input type="checkbox"/> Персонал	Корневая область применения для персонала
<input type="checkbox"/> Персонал_deprecated	Корневая область применения для персонала
<input type="checkbox"/> ЛИМС, внутрилабораторный контроль качества	ЛИМС, внутрилабораторный контроль качества
> <input type="checkbox"/> Методики	Группирует области применения, связанные с методиками ЛИМС.

Рисунок 0.1 Иерархия ОПИ для ZQL