

ООО «Газинформсервис»

УТВЕРЖДАЮ

« ____ » _____ 2024 г.

ПРОГРАММНЫЙ КОМПЛЕКС ANKEY IDENTITY MANAGER

Руководство администратора

ЛИСТ УТВЕРЖДЕНИЯ

72410666.00054-02 95 01-ЛУ

Представители предприятия-разработчика:

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Исполнитель

Нормоконтролер

2024

Изм.	Подп.	Дата

Литера

ООО «Газинформсервис»

УТВЕРЖДЕН
72410666.00054-02 95 01-ЛУ

ПРОГРАММНЫЙ КОМПЛЕКС ANKEY IDENTITY MANAGER

Руководство администратора

72410666.00054-02 95 01

Листов 300

2024

Изм.	Подп.	Дата

Литера

Инв. № подл.	Подп. и дата	Взам. инв. №	Инв. № дубл.	Подп. и дата

Содержание

1. Общие сведения.....	8
1.1. Область применения.....	8
1.2. Основные функции администратора Комплекса	9
1.3. Общий порядок администрирования.....	10
2. Настройка программного комплекса.....	11
2.1. Настройки запуска приложения.....	11
2.2. Установка коннекторов.....	11
2.2.1. Настройка объекта «Коннектор».....	12
2.2.2. Настройка объекта «Тип ресурса»	22
2.2.3. Настройка объекта «Ресурс»	24
2.2.4. Настройка объекта «Форма ресурса»	30
2.2.5. Настройка объекта «Маппинг»	44
2.3. Управление учетной записью ресурса.....	60
2.3.1. Создание учетной записи ресурса.....	61
2.3.2. Редактирование учетной записи ресурса	63
2.3.3. Связывание УЗ пользователя с ее владельцем	64
2.3.4. Отвязывание владельца от УЗ пользователя.....	66
2.3.5. Удаление УЗ пользователя	67
2.3.6. Изменение типа УЗ пользователя	68
2.3.7. Изменение дополнительной УЗ пользователя на основную	68
2.3.8. Смена парольной политики для УЗ в УЗР	69
2.4. Управление периодическими заданиями.....	70
2.4.1. Создание периодического задания.....	74
2.4.2. Редактирование периодического задания	75
2.4.3. Удаление периодического задания	75
2.4.4. Запуск и остановка периодического задания.....	76
2.4.5. Описание периодических заданий Комплекса	76
2.4.6. Настройка службы запуска периодических заданий	90

Изм.	Подп.	Дата

2.4.7. Описание настроек расписания в формате cron	90
2.4.8. Просмотр списка периодических заданий	91
2.5. Настройка справочников	92
2.5.1. Создание справочника.....	93
2.5.2. Удаление справочника	93
2.5.3. Управление значениями справочника	94
2.5.4. Экспорт/импорт значений справочника из Excel	95
2.5.5. Управление файлом в справочнике	97
2.5.6. Изменение видимости и поиска настраиваемых полей в справочнике	98
2.6. Настройка рабочих потоков.....	99
2.6.1. Настройка модуля «Activiti».....	106
2.6.2. Настройка рабочего потока для заявки на назначение роли определенной информационной системы	106
2.6.3. Создание рабочего потока	108
2.6.4. Настройка рабочего потока для действий с объектами Комплекса.....	108
2.6.5. Удаление Бизнес Пакета (.bar)	112
2.6.6. Настройка вложений в заявку.....	113
2.6.7. Настройка отзыва заявки заявителем	114
2.6.8. Настройка отображения фотографии в деталях заявки	114
2.7. Управление настраиваемыми полями.....	115
2.7.1. Управление настраиваемым полем.....	115
2.7.2. Создание атрибутов в таблице БД с помощью скриптов Liquidbase.....	118
2.7.3. Настройка локализации настраиваемых полей	119
2.7.4. Настройка локализации существующего поля	121
2.7.5. Поддерживаемые типы настраиваемых полей	122
2.7.6. Настройка динамического фильтра по маске в форме	146
2.8. Настройка безопасности	149

Изм.	Подп.	Дата

2.8.1. Настройка безопасного подключения к веб-консолям	150
2.8.2. Настройка схем аутентификации	150
2.8.3. Административные роли.....	151
2.8.4. Настройка идентификации пользователя через кодовое слово ..	153
2.8.5. Настройка скомпрометированных паролей	154
2.9. Настройка журналов работы.....	155
2.10. Настройка аудита событий	155
2.10.1. Настройка службы аудита.....	156
2.10.2. Описание типов журналов аудита.....	157
2.10.3. Настройка буферизации записей аудита	163
2.10.4. Дополнительная настройка обработчика событий аудита	164
2.11. Настройка парольной политики.....	164
2.11.1. Создание парольной политики	168
2.11.2. Редактирование парольной политики.....	172
2.11.3. Применение парольной политики.....	173
2.11.4. Удаление парольной политики.....	175
2.12. Настройка уведомлений	175
2.12.1. Настройка почтового сервера.....	178
2.12.2. Настройка скрипта отправки уведомлений.....	178
2.12.3. Отправка уведомления с помощью REST-запроса	180
2.12.4. Настройка согласования заявки из уведомления	180
2.12.5. Настройка уведомления о новом инциденте	181
2.12.6. Настройка уведомлений в рабочих потоках с помощью модуля «Activiti»	182
2.12.7. Перевод строки в сообщениях Комплекса	182
2.13. Настройка производительности Комплекса.....	183
2.13.1. Настройка производительности сервера приложений ПК Ankey IDM.....	183
2.13.2. Настройка производительности СУБД.....	189
2.13.3. Настройка производительности службы поиска	190

Изм.	Подп.	Дата

2.14. Настройка пула потоков для асинхронных операций.....	190
2.15. Настройка интеграции с внешними системами	192
2.15.1. Настройка внешней ссылки	193
2.15.2. Проверка нарушений правил разграничения доступа	194
2.16. Настройка службы сервера коннекторов.....	195
2.17. Настройка правил.....	196
2.17.1. Настройка правила автоназначения.....	197
2.17.2. Настройка правила связывания	206
2.17.3. Настройка правила разграничения доступа	207
2.18. Настройка поиска	218
2.18.1. Настройка подключения к поисковой системе.....	218
2.18.2. Настройка поискового индекса	218
2.18.3. Настройка фильтра поиска заместителя.....	219
2.18.4. Настройка области видимости бенефициара в заявке на назначение ролей	219
2.18.5. Настройка фильтра выбора ролей согласующих при переназначении заявки	220
2.19. Настройка политики валидации.....	220
2.20. Настройка валидации для пользовательской функции в системной роли.....	225
2.21. Управление фотографиями пользователей	226
2.22. Управление лицензиями	231
2.23. Управление незавершенными задачами	232
2.24. Настройка автоматической генерации атрибутов	236
2.24.1. Генерация логина.....	236
2.24.2. Генерация идентификатора трудоустройства.....	237
2.24.3. Генерация пароля.....	237
2.24.4. Экспорт конфигурации	239
2.24.5. Восстановление конфигурации	240
2.24.6. Импорт конфигурации	240

Изм.	Подп.	Дата

2.24.7. Удаление конфигурации	242
2.25. Настройка импорта ролей через коннектор «Excel Matrix Upload»	243
2.26. Настройка для отображения столбцов, полей фильтрации и поиска в интерфейсе.....	243
2.26.1. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Пользователи/Учетные записи»	249
2.26.2. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Пользователи».....	250
2.26.3. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Входящие», «Мои заявки», «Входящие/История», «Журнал заявок»	251
2.26.4. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Роли».....	252
2.26.5. Настройка полей фильтрации и поиска по спискам ИС, ролей в меню «Пользователи», «Мой профиль» при запросе доступа	253
2.27. Настройка полей экспорта/импорта объектов пользователи/пользователь-роль/дополнительные трудоустройства	254
2.28. Управление коннекторами	255
2.28.1. Создание коннектора.....	255
2.28.2. Редактирование коннектора.....	256
2.28.3. Активация коннектора	256
2.29. Дополнительные настройки в интерфейсе заявок на запрос ролей	257
2.30. Управление рисками.....	258
3. Настройка согласований.....	262
3.1. Создание настройки согласования	262
3.2. Редактирование настройки согласования	263
3.3. Удаление настройки согласования.....	263
4. Системные настройки	264

Изм.	Подп.	Дата

4.1. Настройки аутентификации	269
4.2. Настройки видимости и доступности объектов в заявках и замещениях.....	270
4.3. Настройки генераторов значений	271
4.4. Добавление настраиваемого поля для управляемых объектов и заявок	272
4.5. Настройки загружаемых файлов.....	273
4.6. Настройки импорта/экспорта объектов системы	274
4.7. Настройки интерфейса	275
4.8. Настройки логирования и массовых операций	276
4.9. Настройки подключений	277
4.10. Настройки производительности	279
4.11. Настройки работы полнотекстового поиска	280
4.12. Настройки синхронизации с целевыми системами	281
4.13. Настройка приоритета полей поискового запроса	281
5. Управление типами УЗ пользователей.....	283
5.1. Создание типа УЗ пользователя.....	285
6. Настройка типов блокировок	287
6.1. Создание типа блокировки.....	287
6.2. Редактирование типа блокировки.....	288
6.3. Удаление типа блокировки	288
7. Настройка типов блокировок	289
8. Настройки виджетов.....	290
Приложение 1	291
Перечень сокращений	298

Изм.	Подп.	Дата

1. ОБЩИЕ СВЕДЕНИЯ

1.1. Область применения

Наименование программы: программный комплекс Ankey Identity Manager (далее – Комплекс, ПК Ankey IDM).

Обозначение программы: 72410666.00054-02.

Комплекс предназначен для автоматизации процессов централизованного управления учетными записями (УЗ) и привилегиями пользователей в целевых системах (ЦС).

Комплекс функционирует с использованием следующего программного обеспечения:

- 1) Операционные системы (ОС):
 - Astra Linux Special Edition 1.6, 1.7 (сертификат ФСТЭК России № 2557 от 27.01.2012);
 - Альт 8 СП (сертификат ФСТЭК России № 3866 от 10.08.2018);
 - Astra Linux Common Edition 2.12;
 - Ubuntu 22.10, 23.04, 23.10;
 - Альт Сервер 8, 9, 10;
 - Simply Linux 10;
 - РЕД ОС 7.3 (сертификат ФСТЭК России № 4060 от 12.01.2019);
 - ОС РОСА «КОБАЛЬТ» 12 (сертификат ФСТЭК России № 4039 от 07.12.2018);
 - Red Hat Enterprise Linux 7, 8, 9;
 - CentOS 7, 8;
 - Fedora Linux 30 и выше.
- 2) Система управления базами данных (СУБД) Jatoba (сертификат ФСТЭК России № 4327 от 19.11.2020) для сертифицированной версии Комплекса, в остальных случаях возможно использование PostgreSQL версии 10-11 (из состава сертифицированной ОС);

Изм.	Подп.	Дата

- 3) Комплект разработчика приложений Java Development Kit (JDK) Axiom JDK 11;
- 4) Служба поиска OpenSearch 1.3.4 и выше.

Комплекс состоит из следующих компонентов:

- сервер приложений ПК Ankey IDM – может устанавливаться вместе или отдельно от СУБД;
- сервер СУБД – может устанавливаться вместе или отдельно от сервера приложений;
- сервер коннекторов – обеспечивает выполнение функций по интеграции с ЦС средствами коннекторов. Может устанавливаться совместно с сервером приложений или на выделенном сервере.

1.2. Основные функции администратора Комплекса

Администратор Комплекса выполняет следующие основные функции:

- установка программных компонент;
- установка коннекторов;
- управление учетными записями ресурсов (УЗР);
- управление периодическими заданиями;
- настройка справочников;
- настройка рабочих потоков;
- управление настраиваемыми полями;
- настройка безопасности;
- настройка журналов работы;
- настройка аудита событий.

Действия администратора Комплекса по установке программных компонент описаны в документе «Руководство по инсталляции» (72410666.00054-02 94 01).

Порядок выполнения остальных функций администратора Комплекса приведен в данном документе.

Изм.	Подп.	Дата

1.3. Общий порядок администрирования

Комплекс позволяет администратору выполнять настройки с помощью:

- 1) REST-запросов для управления объектами Комплекса. Описание поддерживаемых REST-запросов приведено в приложении 1.
- 2) Консоли администрирования.

Для работы в консоли администрирования требуется веб-браузер. Требования к применяемым веб-браузерам определены в документе «Описание применения» (72410666.00054-02 31 01).

Язык интерфейса консоли администрирования зависит от установленного языка веб-браузера. Консоль администрирования поддерживает следующие языки:

- русский;
- английский.

Чтобы войти в консоль администрирования, необходимо выполнить следующие действия:

- 1) Перейти по следующей ссылке в веб-браузере: `http://<адрес сервера>:<порт>`. Пример: `http://localhost :8080/`.
- 2) Пройти аутентификацию, указав идентификатор (логин) и пароль пользователя с правами администратора. При первоначальной установке по умолчанию создается служебный пользователь с идентификатором – `ankey`, паролем – `ankey`.

Изм.	Подп.	Дата

2. НАСТРОЙКА ПРОГРАММНОГО КОМПЛЕКСА

2.1. Настройки запуска приложения

При загрузке Комплекс использует три следующих файла, которые загружаются в указанном порядке, а значения устанавливаются в обратном порядке:

- system.properties;
- config.properties;
- boot.properties.

Если значение для одного и того же параметра установлено в system.properties и boot.properties, то будет использовано значение из boot.properties.

В файле boot.properties, расположенном в директории ankey/conf/boot, указываются порты для соединения по протоколам http/https и порт для взаимной проверки подлинности с помощью сертификата.

2.2. Установка коннекторов

Коннекторы к ЦС устанавливаются одним из двух способов:

- 1) Локально в каталог сервера приложений ПК Ankey IDM в папку «connectors».
- 2) Удаленно на сервер коннекторов.

Служба сервера коннекторов необходима для запуска исполняемого кода коннекторов к ЦС в следующих случаях:

- 1) Коннектор должен запускаться в окружении, не поддерживаемом сервером приложений ПК Ankey IDM, например, в ОС Windows.
- 2) Программные библиотеки, используемые коннектором, могут вызывать конфликт с библиотеками, используемыми сервером приложений ПК Ankey IDM.

Установку коннекторов к ЦС администратору необходимо производить, руководствуясь документацией к коннектору, в следующей последовательности:

- 1) Размещение библиотек коннектора в папке «connectors» сервера приложений ПК Ankey IDM или удаленно на сервере коннекторов.

Изм.	Подп.	Дата

- 2) Размещение внешних библиотек, используемых коннектором, в папке «lib» сервера приложений ПК Ankey IDM или удаленно на сервере коннекторов.
- 3) Настройка типовых объектов коннектора в ПК Ankey IDM:
 - объект «Коннектор»;
 - объект «Тип ресурса»;
 - объект «Ресурс»;
 - объект «Форма ресурса»;
 - объект «Маппинг»;
 - объект «Учетная запись ресурса» согласно подразделу 2.3;
 - периодические задания согласно подразделу 2.4.

Список установленных коннекторов доступен в меню «Настройка ресурсов»/«Коннекторы». Поиск в списке коннекторов осуществляется по атрибутам: «Название», «Описание». В случае, если Комплекс не отображает результат поиска, следует запустить периодическое задание «reindextask».

2.2.1. Настройка объекта «Коннектор»

Объект «Коннектор» («connector») содержит настройки схемы объектов ЦС, а также другие параметры для интеграции с ней.

В параметре «connectorName» указывается наименование коннектора (например, ADConnector).

В параметре «connectorDesc» указывается описание коннектора (например, ADConnector).

Параметр «statusAttr» отвечает за настройку атрибута коннектора, значение которого изменяет статус объекта в ЦС (блокирует, разблокирует).

Настройка параметра «statusAttr» включает следующие атрибуты:

- «account» – объект коннектора. В качестве объекта коннектора поддерживается только значение «account»;
- «name» – имя атрибута;
- «enableValue» – значение для операции разблокировать;
- «disableValue» – значение для операции заблокировать.

Изм.	Подп.	Дата

Пример настройки параметра «statusAttr»:

```
{ "account": {
  "name": "enable",
  "enableValue": true,
  "disableValue": false } }
```

В параметре «connectorBody» выполняются настройки объектов коннектора.

В параметре «connectorBody/name» указывается внутреннее наименование коннектора (например, ADConnector).

В параметре «connectorBody/connectorRef» указываются сведения о версии и библиотеке коннектора, которые соответствуют содержимому файла MANIFEST.MF внутри архива «jar» библиотеки коннектора.

Настройка параметра «connectorBody/connectorRef» включает следующие обязательные атрибуты:

- «bundleName» – соответствует параметру «ConnectorBundle-Name» файла MANIFEST.MF;
- «productName» – название продукта, используется для активации лицензии на коннектор. Должен совпадать с названием продукта в системе лицензирования;
- «bundleVersion» – соответствует параметру «ConnectorBundle-Version» файла MANIFEST.MF;
- «connectorName» – класс Java, в котором реализован коннектор.

Пример настройки параметра «connectorBody/connectorRef»:

```
"connectorRef": {
  "bundleName": "com.gis.openicf.sapbo",
  "productName": "Ankey IDM connector SAP BO",
  "bundleVersion": "[0.0,2.0)",
  "connectorName": "com.gis.openicf.sapbo.SAPBOConnector"
}
```

В параметре «connectorBody/objectTypes» выполняются настройки типов объектов, поддерживаемых коннектором (например, account, group, role и т.д.). Наименования могут задаваться любые.

Пример настройки параметра «connectorBody/objectTypes»:

```
"objectTypes": {
  "account": {...},
```

Изм.	Подп.	Дата

```
"group": {...}
}
```

В параметре «connectorBody/objectTypes/» выполняются настройки типа объекта.

Настройка параметра «connectorBody/objectTypes/» включает следующие атрибуты:

- id – идентификатор объекта Open Identity Connector Framework (ICF). По умолчанию коннектор использует идентификатор «ACCOUNT». Можно вводить свои идентификаторы, при наличии их поддержки в коде коннектора;
- type – тип объекта, по умолчанию «object»;
- \$schema – схема, по которой проводится валидация JavaScript Object Notation (JSON). По умолчанию «http://json-schema.org/draft-03/schema»;
- nativeType – тип объекта внутри коннектора. По умолчанию коннектор использует тип «ACCOUNT». Можно вводить свои типы, при наличии их поддержки в коде коннектора. При использовании нескольких объектов коннектора типы объектов должны быть уникальны;
- properties – перечень атрибутов объекта.

Пример настройки параметра «connectorBody/objectTypes/»:

```
"objectTypes": {
  "account": {
    "id": "__ACCOUNT__",
    "type": "object",
    "$schema": "http://json-schema.org/draft-03/schema",
    "nativeType": "__ACCOUNT__",
    "properties": { ... }
  }.....
}
```

В параметре «connectorBody/objectTypes//properties» перечисляются настройки атрибутов, поддерживаемых коннектором.

Пример настройки параметра «connectorBody/objectTypes//properties»:

```
{
  "<Имя атрибута 1>": {
    "type": "<Тип данных>",
    "required": <Флаг обязательности (true/false)>,
    "nativeName": "<Имя атрибута в целевой системе>",
```

Изм.	Подп.	Дата

```
"nativeType": "<Тип данных в целевой системе>",  
"flags": [ "<Дополнительные флаги>" ]  
},  
"<Имя атрибута 2>": {..... }  
..... }
```

Атрибут «type» может принимать следующие значения:

- string;
- object;
- array;
- integer;
- boolean.

Атрибут «nativeType» может принимать следующие значения:

- string;
- object;
- array;
- integer;
- boolean;
- JAVA_TYPE_BIGDECIMAL;
- JAVA_TYPE_BIGINTEGER;
- JAVA_TYPE_BYTE;
- JAVA_TYPE_BYTE_ARRAY;
- JAVA_TYPE_CHAR;
- JAVA_TYPE_CHARACTER;
- JAVA_TYPE_DATE;
- JAVA_TYPE_DOUBLE;
- JAVA_TYPE_FILE;
- JAVA_TYPE_FLOAT;
- JAVA_TYPE_GUARDEDBYTEARRAY;
- JAVA_TYPE_GUARDEDSTRING;
- JAVA_TYPE_INT;
- JAVA_TYPE_INTEGER;
- JAVA_TYPE_LONG;
- JAVA_TYPE_OBJECT;

Изм.	Подп.	Дата

- JAVA_TYPE_PRIMITIVE_BOOLEAN;
- JAVA_TYPE_PRIMITIVE_BYTE;
- JAVA_TYPE_PRIMITIVE_DOUBLE;
- JAVA_TYPE_PRIMITIVE_FLOAT;
- JAVA_TYPE_PRIMITIVE_LONG;
- JAVA_TYPE_STRING.

Дополнительные флаги предоставляют возможность задавать особый порядок обработки атрибута. Атрибут «flags» может принимать значения:

- NOT_CREATABLE – атрибут не может задаваться ПК Ankey IDM при создании объекта в ЦС (операция «CREATE»);
- NOT_UPDATEABLE – атрибут не может меняться ПК Ankey IDM при обновлении объекта в ЦС (операция «UPDATE»);
- NOT_READABLE – атрибут не может быть прочитан ПК Ankey IDM при синхронизации из ЦС, например, поле с паролем;
- NOT_RETURNED_BY_DEFAULT – атрибут ЦС не будет прочтен по умолчанию, а только по требованию (отдельным вызовом, если реализовано в коннекторе), например, если атрибут содержит большое значение.

Пример настройки атрибутов, поддерживаемых коннектором:

```
{
  "userPrincipalName": {
    "type": "string",
    "required": false,
    "nativeName": "userPrincipalName",
    "nativeType": "string"
  },
  "Password": {
    "type": "string",
    "required": false,
    "nativeName": "__Password__",
    "nativeType": "JAVA_TYPE_GUARDEDSTRING",
    "flags": [
      "NOT_READABLE",
      "NOT_RETURNED_BY_DEFAULT"
    ]
  },
  "cn": {
    "type": "string",
    "required": true,
    "nativeName": "cn",
```

Изм.	Подп.	Дата

```

"nativeType": "string",
"flags": [
"NOT_UPDATEABLE"
]
},
"__NAME__": {
"type": "string",
"required": true,
"nativeName": "__NAME__",
"nativeType": "string"
},
"objectGUID": {
"type": "string",
"required": false,
"nativeName": "objectGUID",
"nativeType": "JAVA_TYPE_BYTE_ARRAY",
"flags": [
"NOT_CREATABLE",
"NOT_UPDATEABLE"
]
}}

```

В параметре «connectorBody/operationOptions» выполняется настройка специальной обработки для каждой операции коннектора. Например, можно настроить запрет изменений типа объекта. Настройка параметра «connectorBody/operationOptions» включает следующие параметры:

- denied – включение/выключение запрета на выполнение операции. Например, для включения указывается значение «true»;
- onDeny – действие, выполняемое в случае, когда параметр denied включен. По умолчанию принимает значение «DO_NOTHING», при котором ничего к запрету действия дополнительно не выполняется. Также может принимать значение THROW_EXCEPTION, при котором в журнал работы будет выведена ошибка.

Пример настройки параметра «connectorBody/operationOptions»:

```

"operationOptions" : {
{
"SYNC" :
{
"denied" : true,
"onDeny" : "DO_NOTHING"
}
}
}

```

Изм.	Подп.	Дата

В параметре «connectorBody/operationTimeout» выполняется настройка времени ожидания завершения для отдельных операций коннектора в миллисекундах. В случае значения «-1» ожидание не ограничено.

Пример настройки параметра «connectorBody/operationTimeout»:

```
"operationTimeout": {
  "GET": -1,
  "SYNC": -1,
  "TEST": -1,
  "CREATE": -1,
  "DELETE": -1,
  "SCHEMA": -1,
  "SEARCH": -1,
  "UPDATE": -1,
  "VALIDATE": -1,
  "AUTHENTICATE": -1,
  "SCRIPT_ON_RESOURCE": -1,
  "SCRIPT_ON_CONNECTOR": -1 }
```

В параметре «connectorBody/poolConfigOption» выполняется настройка пула соединений и включает следующие параметры:

- maxIdle – максимальное количество простаивающих экземпляров коннектора (по умолчанию 1);
- maxWait – максимальное время ожидания получения данных объекта коннектора из ЦС в миллисекундах (по умолчанию 150000);
- minIdle – минимальное количество простаивающих экземпляров коннектора (по умолчанию 1);
- maxObjects – максимальное количество простаивающих и активных экземпляров коннектора (по умолчанию 1);
- minEvictableIdleTimeMillis – минимальное время, которое объект может быть простаивающим в пуле до его удаления в миллисекундах (по умолчанию 120000).

Пример настройки параметра «connectorBody/poolConfigOption»:

```
"poolConfigOption": {
  "maxIdle": 1,
  "maxWait": 150000,
  "minIdle": 1,
  "maxObjects": 1,
  "minEvictableIdleTimeMillis": 120000
}
```

Изм.	Подп.	Дата

В параметре «connectorBody/producerBufferSize» выполняется настройка размера буфера. По умолчанию принимает значение 100. Например, «producerBufferSize»: 100.

В параметре «connectorBody/syncFailureHandler» выполняется настройка повтора неверно завершившихся операций коннектора. Включает следующие параметры:

- maxRetries – максимальное число повторов;
- postRetryAction – действие в случае завершения повторов операций (например, запись в лог).

Пример настройки параметра «connectorBody/syncFailureHandler»:

```
"syncFailureHandler": {
  "maxRetries": 5,
  "postRetryAction": "logged-ignore"
}
```

В параметре «connectorBody/connectorPoolingSupported» выполняется настройка включения поддержки пула соединений коннектором. Например, для включения указывается «connectorPoolingSupported»: true.

В параметре «connectorBody/enableCaseInsensitiveFilter» выполняется настройка включения поиска объектов в коннекторе без учета регистра. Если параметр выключен, то сравнение идентификаторов на стороне Комплекса и в ЦС ведется с учетом регистра. Например, для включения указывается «enableCaseInsensitiveFilter»: true.

Пример использования REST-запроса для создания коннектора:

```
{
  "connectorName": "ADConnector",
  "statusAttr": {
    "account": {
      "name": "enable",
      "enableValue": true,
      "disableValue": false
    }
  },
  "connectorDesc": "ADConnector",
  "connectorBody": {
    "name": "ADConnector",
    "objectTypes": {
      "account": {
        "id": "__ACCOUNT__",

```

Изм.	Подп.	Дата

```

"type": "object",
"$schema": "http://json-schema.org/draft-03/schema",
"nativeType": "__ACCOUNT__",
"properties": {
  "userPrincipalName": {
    "type": "string",
    "required": false,
    "nativeName": "userPrincipalName",
    "nativeType": "string"
  },
  "Password": {
    "type": "string",
    "required": false,
    "nativeName": "__Password__",
    "nativeType": "JAVA_TYPE_GUARDEDSTRING",
    "flags": [
      "NOT_READABLE",
      "NOT_RETURNED_BY_DEFAULT"
    ]
  },
  "cn": {
    "type": "string",
    "required": true,
    "nativeName": "cn",
    "nativeType": "string",
    "flags": [
      "NOT_UPDATEABLE"
    ]
  },
  "__NAME__": {
    "type": "string",
    "required": true,
    "nativeName": "__NAME__",
    "nativeType": "string"
  },
  "memberOf": {
    "type": "array",
    "items": {
      "type": "string",
      "nativeType": "string"
    },
    "nativeName": "memberOf",
    "nativeType": "string"
  },
  "whenChanged": {
    "type": "string",
    "required": false,
    "nativeName": "whenChanged",
    "nativeType": "JAVA_TYPE_DATE",
    "flags": [
      "NOT_CREATABLE",
      "NOT_UPDATEABLE"
    ]
  }
}
},
"group": {

```

Изм.	Подп.	Дата

```

"$schema": "http://json-schema.org/draft-03/schema",
"id": "__GROUP__",
"type": "object",
"nativeType": "__GROUP__",
"properties": {
  "dn": {
    "type": "string",
    "required": false,
    "nativeName": "distinguishedName",
    "nativeType": "string"
  },
  "objectGUID": {
    "type": "string",
    "required": false,
    "nativeName": "objectGUID",
    "nativeType": "JAVA_TYPE_BYTE_ARRAY"
  }
}
},
"connectorRef": {
  "bundleName": "com.gis.openicf.ad",
  "bundleVersion": "1.1-SNAPSHOT",
  "connectorName": "com.gis.openicf.ad.ADCConnector"
},
"operationOptions": {},
"operationTimeout": {
  "GET": -1,
  "SYNC": -1,
  "TEST": -1,
  "CREATE": -1,
  "DELETE": -1,
  "SCHEMA": -1,
  "SEARCH": -1,
  "UPDATE": -1,
  "VALIDATE": -1,
  "AUTHENTICATE": -1,
  "SCRIPT_ON_RESOURCE": -1,
  "SCRIPT_ON_CONNECTOR": -1
},
"poolConfigOption": {
  "maxIdle": 1,
  "maxWait": 150000,
  "minIdle": 1,
  "maxObjects": 1,
  "minEvictableIdleTimeMillis": 120000
},
"producerBufferSize": 100,
"syncFailureHandler": {
  "maxRetries": 5,
  "postRetryAction": "logged-ignore"
},
"connectorPoolingSupported": true
}
}

```

Изм.	Подп.	Дата

При создании коннектора осуществляются следующие проверки:

- 1) Наличие обязательных параметров в «connectorBody»/«connectorRef». Если отсутствует обязательный параметр, выводится ошибка, в деталях которой указано «failedPolicy»: «isValidConnectorRef».
- 2) Уникальность «nativeType» среди всех объектов коннектора. Если типы объектов не уникальные, выводится ошибка, в деталях которой указано «failedPolicy»: «isValidConnectorObjectType».

Пример ошибки, когда не указаны обязательные параметры:

```
{
  "code": 403,
  "reason": "Forbidden",
  "message": "Policy validation failed: resource=managed/connector/*, property=connectorBody,
policyId=isValidConnectorBody, value=<.>",
  "detail": {
    "result": false,
    "localizedMessage": "policy.validation.failed",
    "policyId": "isValidConnectorBody",
    "resource": "managed/connector/*",
    "property": "connectorBody",
    "appErrorCode": 5000,
    "detail": {
      "failedPolicy": "isValidConnectorRef"
    },
    "message": null
  }
}
```

2.2.2. Настройка объекта «Тип ресурса»

Объект «Тип ресурса» («restype») содержит в себе перечень атрибутов подключения к ЦС, является определением для создаваемого в дальнейшем ресурса.

Для управления типами ресурсов администратор выполняет следующие действия:

- создание типа ресурса;
- редактирование типа ресурса;
- удаление типа ресурса.

Управление типом ресурса выполняется администратором в меню «Настройка ресурсов»/«Типы ресурсов».

Изм.	Подп.	Дата

Для создания/редактирования типов ресурса администратору требуется выполнить:

- 1) Нажать кнопку «Создать»/«Редактировать». Откроется окно «Редактор кода».
- 2) Добавить/изменить объект «Тип ресурса» в виде json. Значения атрибута «resTypeName» должно быть уникальными.
- 3) Нажать кнопку «Создать».

Возможные ошибки:

- 1) «Ошибка при проверке JSON схемы» – валидация объектов по json-схеме.
- 2) «Ошибка при создании типа ресурса» и «Ошибка при изменении типа ресурса» – при дублировании объектов, значение атрибута «resTypeName» не уникально.

Удаление формы ресурса выполняется с помощью REST-запроса.

Параметры объекта «Тип ресурса» приведены в таблице 2.1.

Таблица 2.1 – Параметры объекта «Тип ресурса»

Параметр	Описание	Обязательность
resTypeName	Имя типа ресурса	да
resTypeBody	Основные настройки, включает в себя подразделы	да
resTypeBody/properties	Перечень атрибутов ресурса, их типов («type») и значений по умолчанию («default»). Поддерживаемые типы: – string; – integer; – boolean; – "\$ref": "#/definitions/stringArray" (специальный тип для указания массивов значений string)	да
resTypeBody/order	Порядок, в котором будут отображаться	да

Изм.	Подп.	Дата

Параметр	Описание	Обязательность
	атрибуты в ресурсе. Должен содержать все атрибуты, указанные в resTypeBody/properties	
resTypeBody/private	Массив атрибутов, для которых значения в ресурсе будут маскироваться знаком «*»	да

Пример настройки «resTypeBody/properties»:

```
"language": {
  "type": "string",
  "default": "EN"
},
"reconcileDeletedOrganizations": {
  "type": "boolean",
  "default": false
},
"topOrganizations": {
  "$ref": "#/definitions/stringArray",
  "default": []
}
```

Пример настройки «resTypeBody/order»:

```
"order": ["configurationFilePath",
"appServerHost" ]
```

2.2.3. Настройка объекта «Ресурс»

Объект «Ресурс» в Комплексе содержит параметры подключения к ЦС.

Для управления ресурсами администратор выполняет следующие действия:

- создание ресурса;
- редактирование ресурса;
- удаление ресурса.

Управление ресурсами выполняется администратором двумя способами:

- с помощью REST-запроса;
- в меню «Настройка ресурсов».

Параметры объекта «Ресурс» приведены в таблице 2.2.

Изм.	Подп.	Дата

Таблица 2.2 – Параметры объекта «Ресурс»

Параметр	Описание	Обязательность	Пример
resName	Имя ресурса	да	1С Database
resDesc	Описание ресурса	нет	Параметры подключения к кадровой системе
restype_id	Ссылка на тип ресурса	да	1
connector_id	Ссылка на используемый коннектор	нет в случае, когда выбран тип ресурса «ICF Connector Server»	2
connectorServer_id	Ссылка на используемый сервер коннекторов	нет	3
resBody	Значения параметров подключения	нет, в случае автономного типа ресурса «Offline resource», т.к. параметров соединения с системой может не быть	–

Пример настройки для параметра «resBody»:

```
"host": {
  "type": "string",
  "default": "localhost"
},
"port": {
  "type": "integer",
  "default": 389
},
"useSsl": {
  "type": "boolean",
  "default": false
},
"baseContexts": {
```

Изм.	Подп.	Дата

```
"$ref": "#/definitions/stringArray",  
"default": ["ou=Users,dc=gis,dc=lan", "ou=ServiceAccounts,dc=gis,dc=lan"]  
}  
}
```

Кроме типов ресурсов для подключения к ЦС Комплекс поддерживает следующие специальные типы ресурсов:

- 1) Коннектор сервер «ICF Connector Server» – ресурсы данного типа содержат параметры подключения к серверу коннекторов ICF, через который выполняется подключение к ЦС.
- 2) Автономный ресурс «Offline resource» – ресурсы данного типа не содержат параметров подключения. Они могут использоваться, когда удаленное управление УЗ в ЦС невозможно (например, отсутствует Application Programming Interface (API)), но при этом необходимо вести учет выделенных УЗ пользователям. Offline-ресурс выделяется пользователям аналогичным образом, как для обычных ресурсов ЦС. Операции синхронизации для таких ресурсов не предусмотрены, т.к. фактического подключения к ЦС нет.

2.2.3.1. Создание ресурса

При создании ресурса выполняется следующее условие: для типа ресурса «ICF Connector Server» игнорируется значение, указанное в параметре «Коннектор».

Для создания ресурса в меню «Настройка ресурсов» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Ресурсы», затем нажать кнопку «Создать». Появится страница создания ресурса.
- 2) Заполнить обязательные поля «Название» и «Тип». При выборе типа ресурса отображаются параметры конфигурации коннектора, которые можно отредактировать в соответствии с параметрами конкретной системы. Выбрать необходимый коннектор в поле «Коннектор». При необходимости заполнить описание ресурса.
- 3) Если для подключения к системе используется коннектор сервер, то его необходимо выбрать в поле «Сервер коннекторов».

Изм.	Подп.	Дата

- 4) Нажать кнопку «Создать». Чтобы увидеть созданный ресурс, в «Списке ресурсов» следует нажать кнопку «Обновить».

Для каждого ресурса в списке отображается статус текущего подключения.

Возможные статусы подключения:

- «Активен»;
- «Ошибка»;
- «Отключен»;
- «Нет данных».

Примечание. Статусы подключения определяются на основе тестового запроса соответствующего коннектора.

Для статусов «Ошибка», «Отключен», «Нет данных» в деталях ресурса отображается блок «Ошибка» с текстом сообщения и причины возникновения ошибки/конфликта при подключении ресурса.

В случае подключения к ЦС через сервер коннекторов администратору необходимо сначала создать ресурс для сервера коннекторов, а затем создать ресурс для ЦС.

При создании ресурса для ЦС необходимо выбрать в поле «Сервер коннекторов» один из ресурсов с типом «ICF Connector Server».

2.2.3.2. Редактирование ресурса

При редактировании ресурса выполняются следующие условия:

- 1) Параметры «Тип ресурса» и «Коннектор» недоступны для редактирования. При необходимости их изменения должен создаваться новый ресурс.
- 2) Параметр «Коннектор» при редактировании не отображается.

Для редактирования ресурса в меню «Настройка ресурсов» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Ресурсы» и выбрать из списка ресурсов необходимый ресурс.
- 2) Нажать кнопку «Редактировать». Откроется карточка ресурса с параметрами.

Изм.	Подп.	Дата

- 3) Изменить параметры ресурса и нажать кнопку «Сохранить».

2.2.3.3. Удаление ресурса

При удалении ресурса выполняются следующие условия:

- 1) Нельзя удалить ресурс, связанный с объектом «Учетная запись ресурса». Для этого сначала должен быть удален сам объект «Учетная запись ресурса».
- 2) Нельзя удалить ресурс сервера коннекторов (тип ресурса «ICF Connector Server»), если он связан с другим ресурсом. Для этого сначала должна быть удалена связь в ресурсе с коннектор сервером.
- 3) При удалении ресурса автоматически удаляются все связанные с ним записи из справочников.

Для удаления ресурса в меню «Настройка ресурсов» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Ресурсы» и выбрать из списка один или несколько ресурсов.
- 2) Нажать кнопку «Удалить». Откроется окно подтверждения с перечнем удаляемых ресурсов.
- 3) Нажать кнопку «Удалить».


2.2.3.4. Отключение и активация ресурса

Комплекс предоставляет администратору возможность вручную отключить/активировать ресурс.

Возможность отключения ресурса от управления Комплексом позволяет:


- исключить ресурс из процесса инициализации;
- отключить взаимодействие коннектора и УЗР с ресурсом.

Для отключения ресурса в меню «Настройка ресурсов» необходимо выполнить следующие действия:

- 1) Перейти в раздел «Ресурсы» и выбрать из списка ресурс в статусе «Активен». Поддерживается множественный выбор.
- 2) Нажать кнопку . Отобразится сообщение «Выбранные ресурсы успешно отключены».

Изм.	Подп.	Дата

Для активации ресурса в меню «Настройка ресурсов» необходимо выполнить следующие действия:

- 1) Перейти в раздел «Ресурсы» и выбрать из списка ресурс в статусе «Отключен». Поддерживается множественный выбор.
- 2) Нажать кнопку . Отобразится сообщение «Выбранные ресурсы успешно активированы».

2.2.3.5. Настройка ресурса с типом «ICF Connector Server»

Комплекс поддерживает встроенный не редактируемый служебный тип ресурса «ICF Connector Server», используемый для подключения ПК Ankey IDM к серверу коннекторов. Параметры объекта «Ресурс» с типом «ICF Connector Server» приведены в таблице 2.3.

Таблица 2.3 – Параметры объекта «Ресурс» с типом «ICF Connector Server»

Параметр	Описание	Пример
host	Имя хоста или IP-адрес компьютера, на котором установлен сервер коннекторов	localhost
port	Номер порта, на котором прослушивается сервер коннекторов	8759
useSSL	Параметр, отвечающий за безопасную передачу данных по протоколу Secure Sockets Layer (SSL). Параметр имеет значение «true», если настроен SSL между ПК Ankey IDM и сервером коннекторов, в противном случае «false»	false
timeout	Целое значение, указывающее количество миллисекунд, по истечении которых соединение между сервером коннекторов и ПК Ankey IDM превысит лимит времени. Если параметр установлен в значение «0», то лимит времени отсутствует	0
protocol	Протокол взаимодействия с сервером коннекторов. По умолчанию необходимо задавать значение «websocket»	websocket
key	Ключ для аутентификации на сервере коннекторов	*****

Изм.	Подп.	Дата

2.2.4. Настройка объекта «Форма ресурса»

Объект «Форма ресурса» («resform») представляет собой шаблон УЗ в ЦС с перечнем атрибутов.

Для управления формами ресурсов администратор выполняет следующие действия:

- создание формы ресурса;
- редактирование формы ресурса;
- удаление формы ресурса.

Управление формой ресурса выполняется администратором в меню «Настройка ресурсов»/«Формы ресурсов».

Для создания/редактирования формы ресурса администратору требуется выполнить:

- 1) Нажать кнопку «Создать»/«Редактировать». Откроется окно «Редактор кода».
- 2) Добавить/изменить объект «Форма ресурса» в виде json. Значения атрибутов «resFormTable» и «resFormName» должны быть уникальными.
- 3) Нажать кнопку «Создать».

Возможные ошибки:

- 1) «Ошибка при проверке JSON схемы» – валидация объектов по json-схеме.
- 2) «Ошибка при создании формы ресурса» и «Ошибка при изменении формы ресурса» – при дублировании объектов, значения атрибутов «resFormTable» и «resFormName» не уникальны.

Удаление формы ресурса выполняется с помощью REST-запроса.

Параметры объекта «Форма ресурса» приведены в таблице 2.4.

Таблица 2.4 – Параметры объекта «Форма ресурса»

Параметр	Описание	Пример
resFormName	Имя формы ресурса	ADForm
resFormDesc	Описание формы ресурса	AD Form
resFormTable	Имя таблицы, в которой будут храниться	rf_ad

Изм.	Подп.	Дата

Параметр	Описание	Пример
	УЗ	
displayField	<p>Атрибут УЗ, значение которого будет ее отображаемым именем в системе.</p> <p>Например, при отображении списка УЗ будет отображаться значение этого атрибута. С целью идентификации разных УЗ в списках, рекомендуется использовать уникальный атрибут. В случае, если настройка не задана, будет использоваться идентификатор УЗ в ЦС («UID»)</p>	userprincipalname
resFormBody	Настройка полей формы ресурса	–
resFormBody/properties	<p>Настройка полей формы ресурса.</p> <p>Поддерживаемые типы полей «fieldType»:</p> <ul style="list-style-type: none"> – textField – однострочное текстовое поле; – textArea – многострочное текстовое поле; – number – однострочное поле для ввода цифр; – passwordField – однострочное поле, скрывающее введенный текст; – flag – флаг, позволяющий управлять параметром с двумя состояниями — «включено» и «отключено»; – lookup – текстовое поле с возможностью поиска и выбора из predetermined значений; – date – дата с отображением в виде календаря; – timestamp – дата со временем, с 	–

Изм.	Подп.	Дата

Параметр	Описание	Пример
	отображением в виде календаря и возможностью указывать время; – multivaluedField – многозначное поле с отображением в виде таблицы	
resFormBody/order	Порядок отображения атрибутов формы ресурсов в интерфейсе Комплекса	–
resFormBody/required	Список обязательных атрибутов формы ресурсов	–

Пример настройки полей формы ресурса:

```
{
  "resFormName": "ADForm",
  "resFormDesc": "AD Form",
  "resFormTable": "rf_ad",
  "displayField": "userprincipalname",
  "resFormBody": {
    "properties": {
      "userprincipalname": {
        "fieldType": "textField",
        "displayName": "UserPrincipaName"
      },
      "password": {
        "fieldType": "passwordField",
        "displayName": "Password"
      },
      "samaccountname": {
        "fieldType": "textField",
        "displayName": "User logon name",
        "maxLength": 20
      },
      "cn": {
        "fieldType": "textField",
        "displayName": "Common name (CN)",
        "maxLength": 64
      },
      "sn": {
        "fieldType": "textField",
        "displayName": "Last name",
        "maxLength": 64
      },
      "memberof": {
        "fieldType": "multivaluedField",
        "displayName": "Member of",
        "properties": {
          "memberof": {
            "fieldType": "lookup",
```

Изм.	Подп.	Дата

```

"displayName": "AD Groups",
"keyField": true,
"object": "managed/referencebook",
"filter": "/refBookType eq \"AD Groups\"",
"searchStartLength": 3,
"resultsDisplayAmount": 5,
"lookupField": "refBookCode",
"placeholder": "templates.managed.form.approleresaccount.placeholder",
"lookupDisplayFields": "refBookName",
"lookupDisplayFields2": "refBookCode"
}
},
"order": [
"memberof"
],
"required": [
"memberof"
]
},
"orgunit": {
"filter": "/refBookType eq \"AD Organizations\"",
"object": "managed/referencebook",
"fieldType": "lookup",
"displayName": "Org Unit",
"lookupDisplayFields": "refBookName",
"lookupField": "refBookCode"
}
},
"order": [
"userprincipalname",
"samaccountname",
"cn",
"password",
"orgunit",
"sn",
"memberofs"
],
"required": [
"cn",
"samaccountname"
]
}
}
}

```

2.2.4.1. Настройка поискового поля с типом «lookup»

Поисковые поля с типом «lookup» позволяют для выбора значения выполнять поиск по определенному настройками справочнику. Поиск при этом выполняется как по коду, так и наименованию значения из справочника. Поисковые поля могут настраиваться как отдельно в форме ресурса, так и в составе многозначного атрибута формы ресурса.

Изм.	Подп.	Дата

Ограничение при использовании поисковых полей: при добавлении нового объекта, в поле типа «lookup» значение должно быть обязательно с типом «string» (не number, boolean, и т.д.).

Пример настройки отдельного поискового поля в форме ресурса:

```
"usertype": {
  "fieldType": "lookup",
  "displayName": "Тип пользователя",
  "maxLength": 100,
  "object": "managed/referencebook",
  "filter": "/refBookType eq \"UserType\"",
  "lookupField": "refBookCode",
  "lookupDisplayFields": "refBookName"
}
```

Параметры настройки отдельного поискового поля приведены в таблице 2.5.

Таблица 2.5 – Параметры настройки отдельного поискового поля

Параметр	Описание	Пример
fieldType	Тип атрибута формы ресурса	lookup
displayName	Отображаемое наименование атрибута	Тип пользователя
maxLength	Максимальная длина хранимого значения атрибута	100
object	Имя ссылочного объекта системы, из которого будут выбираться значения	managed/referencebook
filter	Фильтр «queryFilter», накладываемый на значения объекта. В случае, если в объекте присутствует поле идентификатор ресурса «resource_id», к фильтру автоматически добавится условие «and resource_id eq значение»	/refBookType eq \"UserType\"
lookupField	Атрибут ссылочного объекта, значение которого будет записано в данное поле	refBookCode
lookupDisplayFields	Атрибут ссылочного объекта, являющийся отображаемым значением поля, так же показывается при выборе	refBookName

Изм.	Подп.	Дата

Параметр	Описание	Пример
	объекта из списка, как основной	

Список поддерживаемых операторов при поиске в фильтре «filter» приведен в таблице 2.6.

Таблица 2.6 – Список поддерживаемых операторов при поиске в фильтре «filter»

Оператор	Описание	Примеры
eq	Поиск объектов с указанным значением атрибута без учета регистра	lastName eq 'Глазырин'
in	Содержит значение из списка	lastName in 'Иванов,Адуев'
and	Логическая операция объединения двух условий	lastName eq 'Макаров' and firstName eq 'Анатолий'
pr	У атрибута задано значение	description pr
!	Отрицание	!(description pr)
boolean- фильтр	Фильтр для выборки либо всех объектов, либо ни одного	true/false

2.2.4.2. Настройка многозначных полей с типом «multivaluedField»

Многозначные поля позволяют задавать множество уникальных значений, связанных с УЗ, например, ее полномочия (роли, группы и т.п.). Многозначные поля могут включать в свой состав поисковые поля.

Ограничение при использовании поисковых полей: при добавлении нового объекта, в поле типа «lookup» значение должно быть обязательно с типом «string» (не number, boolean, и т.д.).

Пример настройки многозначного поля:

```
"groups": {
  "fieldType": "multivaluedField",
  "displayName": "Группы",
  "properties": {
    "groupname": {
      "fieldType": "lookup",
      "displayName": "Имя группы",
      "keyField": true,
      "ignoreEvaluate": true,

```

Изм.	Подп.	Дата

```

"object": "managed/referencebook",
"filter": "/refBookType eq \"Groups\"",
"lookupField": "refBookCode",
"placeholder": "templates.managed.form.approleresaccount.placeholder",
"searchStartLength": 3,
"resultsDisplayAmount": 5,
"lookupDisplayFields": "refBookName",
"lookupDisplayFields2": "refBookCode"
},
"read": {
  "fieldType": "flag",
  "displayName": "Чтение"
},
"write": {
  "fieldType": "flag",
  "displayName": "Запись"
},
"startDate": {
  "fieldType": "date",
  "displayName": "Дата начала"
},
"endDate": {
  "fieldType": "date",
  "displayName": "Дата окончания"
}
},
"order": [
  "groupname",
  "read",
  "write",
  "startDate",
  "endDate"
],
"required": [
  "groupname"
]
}

```

Параметры настройки многозначного поля приведены в таблице 2.7.

Таблица 2.7 – Параметры настройки многозначного поля

Параметр	Описание	Пример
fieldType	Тип атрибута формы ресурса	multivaluedField
displayName	Отображаемое наименование атрибута	Группы
properties	Содержит список атрибутов, в многозначное поле, в формате properties/<Имя атрибута>/	"properties": { "groupname":{...}, "read" {...}, "write" {...},

Изм.	Подп.	Дата

Параметр	Описание	Пример
		"startDate"{...}, "endDate"]
order	Порядок отображения атрибутов, включенных в многозначное поле	["groupname", "read", "write", "startDate", "endDate"]
required	Список атрибутов, включенных в многозначное поле, для которых обязательно должно быть задано значение	["groupname"]

Параметры атрибутов, включенных в многозначное поле (properties/Имя атрибута/Параметры), приведены в таблице 2.8.

Таблица 2.8 – Параметры атрибутов, включенных в многозначное поле (properties/Имя атрибута/Параметры)

Параметр	Описание	Пример
fieldType	Тип атрибута формы ресурса	lookup
displayName	Отображаемое наименование атрибута	Имя группы
keyField	Параметр keyField у атрибутов многозначных полей используется для проверки существования значения многозначного поля. По совпадению	true

Изм.	Подп.	Дата

Параметр	Описание	Пример
	<p>атрибутов с выставленным свойством keyField определяется существует значение или нет. Значение считается существующим, если одновременно совпадают значения всех ключевых полей. Если хотя бы значение одного из ключевых атрибутов отличается, значение считается новым</p>	
ignoreEvaluate	<p>Параметр ignoreEvaluate у атрибутов многозначных полей позволяет игнорировать значения атрибутов при изменении полномочий в УЗ. При установленном значении «true» в параметре «ignoreEvaluate»,</p>	true

Изм.	Подп.	Дата

Параметр	Описание	Пример
	<p>параметр «keyField» обязательно должен быть в значении «false». В противном случае при создании такой формы ресурса будет получена ошибка валидации «Игнорирование ключевых полей мультиатрибутов при пересчете невозможно».</p> <p>Примечание. Изменение ключевого поля полномочия меняет его целиком не зависимо от параметра «ignoreEvaluate»</p>	
object	Имя ссылочного объекта системы, из которого будут выбираться значения	managed/referencebook
filter	<p>Фильтр «queryFilter», накладываемый на значения объекта.</p> <p>Если в объекте</p>	/refBookType eq \"Groups\"

Изм.	Подп.	Дата

Параметр	Описание	Пример
	<p>присутствует поле идентификатор ресурса «resource_id», к фильтру автоматически добавится условие «and resource_id eq значение»</p>	
searchStartLength	<p>Количество вводимых символов, после которых начинает отправляться поисковый запрос. Сообщение берется по указанному ключу для данной локализации. Если ключ локализации не найден, он и будет отображаться в форме</p>	3
resultsDisplayAmount	<p>Количество отображаемых результатов поиска (вариантов) в поисковом поле</p>	5
lookupField	Атрибут ссылочного	refBookCode

Изм.	Подп.	Дата

Параметр	Описание	Пример
	объекта, значение которого будет записано в данное поле	
placeholder	Ключ локализации для подсказки, отображаемой при незаполненном поле при создании/редактировании объекта	templates.managed.form.approleresaccount.placeholder
lookupDisplayFields	Основной атрибут ссылочного объекта, отображаемый в интерфейсе при выборе значения в поле	refBookName
lookupDisplayFields2	Настраиваемое поле ссылочного объекта, отображаемый в интерфейсе при выборе значения в поле	refBookCode

2.2.4.3. Настройка параметра шифрования значения атрибута формы ресурса «encryption»

Для всех типов полей может задаваться параметр шифрования значения «encryption». Пример настройки параметра «encryption» для текстового поля:

```
"properties":
{
```

Изм.	Подп.	Дата

```

"userprincipalname":
{
  "fieldType": "textField",
  "displayName": "UserPrincipaName",
  "encryption":
  {
    "cipher": "AES/CBC/PKCS5Padding",
    "key": "custom-key"
  }
},
....
}

```

Параметр «encryption» содержит следующие настройки:

- key – имя ключа шифрования в хранилище ключей Комплекса;
- cipher – алгоритм шифрования, поддерживаемый виртуальной машиной Java. По умолчанию используется алгоритм «AES /CBC/PKCS5Padding».

2.2.4.4. Настройка для создания новых полномочий в Active Directory через заявку

Комплекс поддерживает настройку только для Active Directory.

Для настройки создания новых полномочий администратору необходимо выполнить следующие действия:

- 1) Включить создание роли по заявке. Описание приведено в подпункте 2.6.5.3.
- 2) Создать форму ресурса. Описание приведено в пункте 2.2.4.
- 3) Создать УЗР, указав в ней созданную форму ресурса. Описание приведено в пункте 2.3.1.
- 4) Обновить созданную форму ресурса, добавив для многозначного поля «multivaluedField» дополнительные параметры.

Список дополнительных параметров поля «multivaluedField» для создания полномочий в Active Directory:

- entitlementsResAccId – числовой идентификатор УЗР для управления группами. Указывает на УЗР, в которой хранятся УЗ создаваемых полномочий;
- entitlementsField – название поля в форме ресурса, в которое создаваемые полномочия записываются для передачи в ЦС;

Изм.	Подп.	Дата

- entitlementsLookupField – необязательное поле для поиска полномочий в справочнике. По умолчанию поиск работает по полю «refBookName».

Добавление выполняется с помощью REST-запроса.

Пример дополнительных параметров поля «multivaluedField»:

```
"memberofs": {
  "fieldType": "multivaluedField",
  "displayName": "Member of",

  "properties": {
    "memberof": {
      "fieldType": "lookup",
      "displayName": "AD Groups",
      "keyField": true,
      "object": "managed/referencebook",
      "filter": "/refBookType eq \"AD Groups\"",
      "searchStartLength": 3,
      "resultsDisplayAmount": 5,
      "lookupField": "refBookCode",
      "placeholder": "templates.managed.form.approleresaccount.placeholder",
      "lookupDisplayFields": "refBookName",
      "lookupDisplayFields2": "refBookCode",
      "entitlementsResAccId": 2,
      "entitlementsField": "dn"
    }
  },
  "order": [
    "memberof"
  ],
  "required": [
    "memberof"
  ]
}
```

2.2.4.5. Настройка динамического фильтра по маске при добавлении полномочий УЗР

Динамический фильтр по маске используется при добавлении полномочий УЗР в процессе создания/редактирования роли.

Для настройки динамического фильтра по маске следует выполнить следующие действия:

- 1) Создать справочник, используя название справочника вида <префикс><имя УЗР><постфикс>, где <имя УЗР> – существующее значение имени УЗР, а <префикс> и <постфикс> – любые значения.

Допускаются различные комбинации использования маски с

Изм.	Подп.	Дата

префиксом/постфиксом, в том числе и их отсутствие. Например, справочник «Offline_access».

- 2) Обновить значение фильтра в форме ресурса, заменив имя УЗР на маску `${resAccName}`. Фильтр должен быть определен в соответствии с созданным справочником. Например, заменить `/refBookType eq 'Offline_access'` на `/refBookType eq '${resAccName}_access'`.

В интерфейсе Комплекса фильтр по маске будет доступен в поле «Имя группы» в окне «Добавление прав».

Пример PATCH-запроса на обновление объекта «/managed/resform/{resourceId}» с постфиксом "_access":

```
[
{
"operation": "replace",
"field": "resFormBody/properties/groups/properties/groupname/filter",
"value": "/refBookType eq '${resAccName}_access'"
}
]
```

, где:

- «{resourceId}» – идентификатор ресурса;
- «field» – путь к полю полномочия;
- «value» – фильтр по маске с постфиксом.

2.2.5. Настройка объекта «МAPPING»

Объект «МAPPING» (далее – маппинг) в Комплексе содержит параметры преобразования атрибутов объектов (пользователей, ролей, организаций и т.д.) при синхронизации из ЦС или при распространении в ЦС.

Для управления маппингом администратор выполняет следующие действия:

- просмотр маппинга;
- создание маппинга. Приведено в подпункте 2.2.5.1;
- редактирование маппинга. Приведено в подпункте 2.2.5.2;
- удаление маппинга. Приведено в подпункте 2.2.5.3.

Для просмотра настроенных маппингов в Комплексе администратору необходимо перейти в меню «Настройка ресурсов»/«МAPPING».

Изм.	Подп.	Дата

Список отображает следующие данные:

- название;
- тип;
- источник;
- получатель.

Поиск в списке маппингов осуществляется по атрибутам: «Название», «Источник», «Получатель». Фильтрация выполняется по атрибуту «Тип».

Для фильтрации списка маппингов по типу администратору необходимо выполнить следующие действия:

- 1) Нажать «Фильтр» в правом углу строки поиска.
- 2) В выпадающем списке выбрать один или несколько статусов (recon, rrepopulate, provision).
- 3) Нажать кнопку «Применить».

Для сброса всех фильтров следует нажать «Очистить все», затем кнопку «Применить».

Если Комплекс не отображает результат поиска, следует запустить периодическое задание «reindextask».

Для просмотра настроек маппинга администратору необходимо выбрать маппинг в списке, после чего в правой части экрана отобразятся вкладки «Карточка маппинга», «Параметры», «Политики», «Скрипты». Для маппинга типа «rrepopulate» вкладки «Политики», «Скрипты» недоступны.

Вкладка «Карточка маппинга» отображает следующие настройки:

- название;
- описание;
- тип;
- источник;
- получатель;
- правило корреляции. В правом верхнем углу поля приводится тип скрипта правила корреляции.

Вкладка «Параметры» отображает правила заполнения атрибутов получателя. Справа от правила приводится действие, для которого применяется данное правило

Изм.	Подп.	Дата

(create, update, delete). Для просмотра настроек правила следует нажать по его наименованию.

Настройки правила содержат следующие параметры:

- источник. В случае отсутствия источника отображается значение «Отсутствует»;
- получатель;
- код преобразования. В правом верхнем углу поля приводится тип скрипта преобразования;
- код условия. В правом верхнем углу поля приводится тип скрипта условия выполнения преобразования.

Вкладка «Политики» отображает следующие настройки политик:

- ситуация;
- действие. Значение действия прописано в виде текста, кода скрипта, либо может быть прикреплен файл. Для просмотра кода скрипта следует нажать на кнопку «Код».

Для маппинга с типом «rgreropulate» настройка политик не поддерживается.

Вкладка «Скрипты» отображает настройки скриптов для следующих действий:

- «onCreate», «onUpdate», «onDelete», «result» для маппинга типа «recon»;
- «onCreate», «onUpdate», «onDelete» для маппинга типа «provision».

Настройки скриптов отображаются в виде кода, либо файла скрипта. Для просмотра файла скрипта следует перейти в директорию проекта в папке «script».

Для маппинга с типом «rgreropulate» настройка скриптов не поддерживается.

Настройки маппинга включают следующие параметры:

- mapName – имя маппинга;
- mapType – тип маппинга;
- mapDesc – описание маппинга;
- mapSource – ресурс-источник;
- mapTarget – ресурс-получатель;
- correlationQuery – правило связывания объектов ресурса-источника и ресурса-получателя;
- mapBody – описание правил маппинга;

Изм.	Подп.	Дата

- scripts – скрипты, вызываемые при различных событиях.

Типы маппингов mapType включают следующие типы:

- prepopulate – обеспечивает предзаполнение формы ресурса перед распространением данных в ЦС;
- provision – обеспечивает преобразование данных при распространении данных в ЦС;
- rescon – обеспечивает преобразование данных при синхронизации из ЦС.

В качестве ресурсов-источников и ресурсов-получателей могут использоваться следующие объекты:

- любые управляемые (managed) объекты, например, пользователь (managed/user);
- формы ресурсов, например, форма ресурса службы каталога Active Directory (resform/aduser);
- специальный объект коннектора «connector/auto», он используется для передачи данных непосредственно в коннектор при распространении или синхронизации данных в/из ЦС;
- конкретный тип объекта коннектора в формате записи «connector/<Ресурс>/<Тип объекта коннектора>» (пример, connector/xml/account), такая запись используется при доверенной синхронизации сведений из внешних источников (например, из кадрового модуля).

Возможные сценарии использования маппингов приведены в таблице 2.9.

Таблица 2.9 – Возможные сценарии использования маппингов

Сценарий	Тип маппинга	Ресурс-источник	Ресурс-получатель
Создание новой УЗ в ЦС	– prepopulate; – provision	– managed/user; – resform/aduser	– resform/aduser; – connector/auto
Обновление атрибутов УЗ ЦС при смене атрибутов управляемого	provision	managed/user	resform/aduser

Изм.	Подп.	Дата

Сценарий	Тип маппинга	Ресурс-источник	Ресурс-получатель
(managed) объекта			
Обновление атрибутов управляемого (managed) объекта данными из ЦС	provision	resform/aduser	managed/user
Синхронизация УЗ из ЦС	recon	connector/auto	resform/aduser
Синхронизация сведений об управляемом (managed) объекте из доверенного источника	recon	connector/xml/accoun t	managed/user

Описание правил маппинга mapBody включает следующие параметры:

- policies – необязательные политики обработки различных ситуаций;
- properties – правила маппинга.

Политики (policies) могут присутствовать для типов «provision» и «recon».

Политики включают следующие параметры:

- situation – ситуация (событие), которая возникает в процессе сверки данных из ресурса-источника с ресурсом-получателем;
- action – действие, которое необходимо выполнить Комплексу в определенной ситуации.

Ситуации (situation) могут быть следующих типов:

- CONFIRMED – записи в ресурсе-источнике и ресурсе-получателя найдены и имеют связь;
- FOUND – найдены записи в ресурсе-источнике и есть соответствующий объект в ресурсе-получателе, при этом связи между ними не было;
- FOUND_ALREADY_LINKED – найдены записи в ресурсе-источнике и есть соответствующий объект в ресурсе-получателе, при этом связи

Изм.	Подп.	Дата

между ними не было, и ресурс-получатель уже связан с другим объектом ресурса-источника;

- ABSENT – связь объекта из ресурса-источника не может быть построена с объектом ресурса-получателя, например, в случае, когда у объекта ресурса-получателя уже есть связь с другим объектом ресурса-источника;
- AMBIGUOUS – объект ресурса-источника соответствует нескольким объектам ресурсов-получателей;
- MISSING – у ресурса-источника есть связь, но отсутствует запись в ресурсе-получателе (например, УЗ была удалена в ЦС в обход Комплекса);
- UNQUALIFIED – объект ресурса-получателя не прошел валидацию (в случае если настроен скрипт «validTarget» в маппинге), но у него есть связь с объектом ресурса-источника;
- SOURCE_IGNORED – объект ресурса-источника не прошел валидацию (в случае если настроен скрипт «validSource» в маппинге), связь с объектами получателями не найдена;
- TARGET_IGNORED – объект ресурса-получателя не прошел валидацию (в случае если настроен скрипт «validTarget» в маппинге);
- UNASSIGNED – найден валидный объект ресурса-получателя, но у него нет связи с другими объектами;
- SOURCE_MISSING – найден объект ресурса-получателя, и у него есть связь с объектом ресурса-источника, но самого объекта ресурса-источника нет.

Действия (action) могут быть следующих типов:

- CREATE – создает объект ресурса-получателя и связь с объектом ресурса-источника;
- UPDATE – обновляет атрибуты объекта ресурса-получателя;
- EXCEPTION – помечает событие, как исключение;
- DELETE – удаляет объект ресурса-получателя;
- IGNORE – игнорирует изменения;

Изм.	Подп.	Дата

- REPORT – не выполняется никаких действий кроме вывода в журнал работы Комплекса тех действий, которые могли бы быть при выполнении действия по умолчанию;
- NOREPORT – ничего не выполнять и не формировать отчет;
- ASYNC – запущен асинхронный процесс, поэтому не выполнять никакие действия с событием и не формировать отчет;
- UNLINK – разорвать связь между ресурсом-источником и ресурсом-получателем;
- LINK – создает связь между ресурсом-источником и ресурсом-получателем.

Возможные действия в зависимости от ситуации для типа маппинга «recon» приведены в таблице 2.10.

Таблица 2.10 – Возможные действия в зависимости от ситуации для типа маппинга «recon»

Ситуация	Возможные действия
SOURCE_IGNORED	REPORT, IGNORE, EXCEPTION, NOREPORT, ASYNC
UNASSIGNED	EXCEPTION, DELETE, IGNORE
AMBIGUOUS	REPORT, IGNORE, EXCEPTION, NOREPORT, ASYNC
FOUND_ALREADY_LINKED	REPORT, IGNORE, EXCEPTION, NOREPORT, ASYNC
CONFIRMED	UPDATE, IGNORE, REPORT, NOREPORT, ASYNC, RESTORE
UNQUALIFIED	DELETE, EXCEPTION, IGNORE, REPORT, NOREPORT, ASYNC
LINK_ONLY	EXCEPTION, UNLINK, IGNORE, REPORT, NOREPORT, ASYNC
SOURCE_MISSING	REPORT, DELETE, CREATE_SOURCE, EXCEPTION

Изм.	Подп.	Дата

Ситуация	Возможные действия
TARGET_IGNORED	REPORT, IGNORE, DELETE, UNLINK, EXCEPTION, NOREPORT, ASYNC
ABSENT	CREATE, EXCEPTION, IGNORE, REPORT, NOREPORT, ASYNC, DELETE
MISSING	EXCEPTION, CREATE, UNLINK, IGNORE, REPORT, NOREPORT, ASYNC
FOUND	UPDATE, EXCEPTION, IGNORE, REPORT, NOREPORT, ASYNC
ALL_GONE	NOREPORT, IGNORE, EXCEPTION, REPORT, ASYNC

Возможные действия в зависимости от ситуации для типа маппинга «provision» приведены в таблице 2.11.

Таблица 2.11 – Возможные действия в зависимости от ситуации для типа маппинга «provision»

Ситуация	Возможные действия
SOURCE_IGNORED	REPORT
UNASSIGNED	EXCEPTION, IGNORE, REPORT, NOREPORT, ASYNC
AMBIGUOUS	EXCEPTION
FOUND_ALREADY_LINKED	EXCEPTION
CONFIRMED	UPDATE, IGNORE, REPORT, NOREPORT
UNQUALIFIED	DELETE, UNLINK, EXCEPTION, IGNORE, REPORT, NOREPORT, ASYNC
LINK_ONLY	NOREPORT
SOURCE_MISSING	DELETE, EXCEPTION, UNLINK, IGNORE, REPORT, NOREPORT, ASYNC
TARGET_IGNORED	REPORT, IGNORE, DELETE, UNLINK, NOREPORT,

Изм.	Подп.	Дата

Ситуация	Возможные действия
	ASync
ABSENT	CREATE, EXCEPTION, IGNORE
MISSING	CREATE, EXCEPTION
FOUND	UPDATE
ALL_GONE	NOREPORT

Если политика не указана, используются правила обработки различных ситуаций по умолчанию, которые приведены в таблице 2.12.

Таблица 2.12 – Правила обработки различных ситуаций по умолчанию

Ситуация	Тип маппинга	Действие по умолчанию
CONFIRMED	provision/recon	UPDATE
FOUND_ALREADY_LINKED	provision/recon	EXCEPTION
ABSENT	provision/recon	CREATE
AMBIGUOUS	provision/recon	EXCEPTION
MISSING	recon	EXCEPTION
MISSING	provision	CREATE
UNQUALIFIED	provision/recon	DELETE
SOURCE_IGNORED	provision/recon	REPORT
TARGET_IGNORED	provision/recon	REPORT
UNASSIGNED	provision/recon	EXCEPTION
SOURCE_MISSING	provision/recon	REPORT

Правила маппинга (properties) включают следующие параметры:

- action – действие, для которого применяется данное правило;
- source – имя поля объекта ресурса-источника (может быть пустым);
- target – имя поля объекта ресурса-получателя (обязательный параметр);
- transform – скрипт преобразования поля данных из ресурса-источника в ресурс-получатель;

Изм.	Подп.	Дата

- condition – скрипт условия выполнения преобразования для конкретного атрибута.

Если условие для преобразования атрибута, заданное в параметре «condition», не выполняется, то значение для целевого атрибута (target) не задается.

Если действие (action) указано, то это преобразование выполняется только для этого действия (action). Если действие (action) не указано, то это преобразование выполняется для любых действий. Для маппингов с типом «prepopulate» запрещено указывать действие (action).

Скрипты (scripts) маппинга, вызываются при наступлении ситуации (события) перед их фактическим выполнением (action). Скрипты могут использоваться для дополнительного формирования, преобразования атрибутов после применения маппинга, в том числе уведомлений.

Скрипты могут настраиваться на следующие действия:

- onCreate – скрипт вызывается перед созданием объекта;
- onUpdate – скрипт вызывается перед обновлением объекта;
- onDelete – скрипт вызывается перед удалением объекта;
- result – скрипт вызывается по результату выполнения всех преобразований маппинга.

Поддерживаются следующие типы скриптов («type»):

- «text/javascript» – скрипты, написанные на языке JavaScript;
- «groovy» – скрипты, написанные на языке Groovy.

Для скриптов, выполняемых на действия onCreate, onUpdate, onDelete, доступны следующие данные:

- mappingConfig – сведения о настройке маппинга;
- sourceId – сведения об идентификаторе ресурса-источника;
- source – сведения о ресурсе-источнике (например, значения атрибутов);
- targetId – сведения об идентификаторе ресурса-получателя;
- target – сведения о ресурсе-получателе (например, значения атрибутов);
- oldTarget – предыдущее состояние ресурса-получателя (например, предыдущее значение атрибута);
- situation – сведения о типе ситуации.

Изм.	Подп.	Дата

Правило связывания ресурса-источника и ресурса-получателя «correlationQuery» может быть задано в следующих вариантах:

- в виде строки фильтра – параметр correlationQuery;
- в виде скрипта, возвращающего фильтр значений – параметр correlationQuery;
- в виде скрипта, возвращающего объект ресурса-получателя непосредственно – параметр correlationScript.

Если правило не задано явно, то для синхронизации используется связывание по совокупности полей служебного идентификатора объекта (systemObjectId) и идентификатора УЗР пользователя (resAccountId). Атрибуты correlationQuery и correlationScript не могут быть заданы в маппинге одновременно.

Пример настройки атрибута correlationQuery в виде строки фильтра:

```
"correlationQuery" : "\"userName eq \${userName}\""
```

Настройка атрибута correlationQuery задается в виде строки в формате «queryFilter». Для подстановки значений из объекта ресурса-источника используются разделители вида: \${имя-свойства}. Строки необходимо помещать в кавычки.

Пример настройки атрибута correlationQuery в виде скрипта, возвращающего фильтр значений:

```
{
...
  "correlationQuery" : {
    "type" : "text/javascript",
    "source" : "var qry = { '_queryFilter': '_id eq \'' + source.userName + '\'' }; qry;"
  },
...
}
```

Скрипт, возвращающий фильтр значений в качестве аргументов, может использовать поля «_queryFilter» или «_queryId» для запроса.

Пример настройки атрибута correlationScript, возвращающего объект ресурса-получателя непосредственно:

```
{
...
  "correlationScript" : {
    "type" : "text/javascript",
    "file" : "script/correlateScript.js"
  },
}
```

Изм.	Подп.	Дата

...
}

2.2.5.1. Создание объекта «Маппинг»

Создание маппинга выполняется администратором двумя способами:

- 1) Через меню «Настройка ресурсов».
- 2) С помощью REST-запроса. Примеры приведены в приложении 1.

Для создания маппинга в разделе «Маппинги» администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать». Откроется окно «Создание маппинга». В левой части окна расположены вкладки «Карточка маппинга», «Параметры», «Политики», «Правило корреляции», «Скрипты». Некоторые вкладки могут быть недоступными в зависимости от заданного значения параметра «Тип».
- 2) На вкладке «Карточка маппинга» задать значения обязательных полей: «Название», «Тип», «Тип источника», «Источник», «Тип получателя», «Получатель». При указании значений следует ориентироваться на возможные сценарии использования маппингов, которые приведены в таблице 2.9.
- 3) Перейти на вкладку «Параметры». Нажать кнопку «Добавить параметр» и задать значения следующих параметров:
 - источник;
 - получатель;
 - действие. По умолчанию выбрано значение «Отсутствует». Поле не отображается для маппингов с типом «rpropulate»;
 - пары источник – получатель можно дублировать, при условии, что для них указаны разные действия; получатель, для которого уже указаны все возможные действия, не отображается в списке получателей; получатель, для которого выбрано «отсутствует» в списке действий, больше не отображается в списке получателей;

Изм.	Подп.	Дата




- код преобразования. Нажать «Добавить преобразование». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже;
 - код условия. Нажать «Добавить условие». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже.
- 4) Перейти на вкладку «Политики» и добавить политику маппинга:
- ситуация. Выбрать ситуацию из выпадающего списка;
 - простое действие. Нажать кнопку «Простое действие» и заполнить поле «Действие». Кнопка «Простое действие» неактивна пока не выбрана ситуация;
 - действие по скрипту. Нажать кнопку «Действие по скрипту». Общие действия при выборе типа скрипта описаны ниже.
- 5) Нажать кнопку «Сохранить».
- 6) Перейти на вкладку «Правило корреляции». Добавить правило корреляции, указав один из типов скрипта «Java Script», «Groovy», «Query». Общие действия при выборе типа скрипта «Java Script» или «Groovy» описаны ниже.
- 7) Нажать кнопку «Применить».
- 8) Перейти на вкладку «Скрипты». Нажать кнопку «Создать скрипт» и задать значения следующих параметров:
- действие. Выбрать действие из выпадающего списка. Возможные действия приведены в пункте 2.2.5;
 - скрипт. Нажать кнопку «Назначить скрипт». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже.
- 9) Нажать кнопку «Сохранить». В случае ошибки отображается сообщение «Ошибка при создании маппинга».

Общие действия при выборе типа скрипта:

- 1) Нажать кнопку «Java Script» или «Groovy» для выбора типа скрипта.
- 2) Выбрать одно из двух действий:

Изм.	Подп.	Дата

- нажать кнопку «Текст», чтобы внести значение в виде кода в поле редактора кода;
- нажать кнопку «Файл», чтобы внести значение в формате файла скрипта. Следует указать название файла, который находится в директории проекта в папке «script». Если файл отсутствует, то при сохранении изменений отобразится ошибка.

При ручном вводе кода скрипта выполняется подсветка синтаксиса, валидация и автодополнение. Для типов скрипта «Java Script», «Groovy» доступно копирование и форматирование кода. Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

При создании маппинга следует обратить внимание на следующие моменты:

- 1) При смене типа маппинга будут сброшены все настройки, кроме имени и описания.
- 2) При смене значения «Тип» отображается предупреждение «При изменении основных настроек маппинга, все остальные данные будут сброшены! Сбросить выбранное?».
- 3) При смене источника/получателя будут сброшены все параметры, если они были заполнены. Все остальные настройки сохраняются.
- 4) Нельзя создать маппинг, если уже существует маппинг с такими же настройками параметров «mapType», «mapSource» и «mapTarget» одновременно. Описание параметров приведено в пункте 2.2.5.
- 5) Для значения connector/auto в источнике/получателе необходимо указывать форму ресурса (resform). Если у указанной формы ресурса отсутствует УЗР, то на вкладке «Параметры» отобразится предупреждение «Не создана УЗР для выбранной формы ресурса».

2.2.5.2. Редактирование объекта «Маппинг»

Для редактирования маппинга необходимо в разделе «Маппинги» выполнить следующие действия:

- 1) Выбрать маппинг в списке.

Изм.	Подп.	Дата

- 2) Нажать кнопку «Редактировать». Откроется окно «Редактирование маппинга». В левой части окна расположены вкладки «Карточка маппинга», «Параметры», «Политики», «Правило корреляции», «Скрипты». Некоторые вкладки могут быть недоступными в зависимости от заданного значения параметра «Тип».
- 3) Перейти на вкладку «Карточка маппинга» для изменения поля «Описание».
- 4) Перейти на вкладку «Параметры» для изменения параметров маппинга. Нажать кнопку «Добавить параметр», в случае если параметр отсутствует или кнопку «Редактировать» напротив параметра. Откроется окно «Редактирование параметра». Для всех параметров, где в «Карточке маппинга» явно указан тип источника/получателя, будет выпадающий список с доступными значениями. В списке действий отображаются только те действия, которые были указаны при создании маппинга. Для значения «Форма ресурса», кроме пользовательских параметров, доступно также системное поле «Status». Доступные поля для изменения:
 - источник;
 - получатель;
 - действие. По умолчанию выбрано значение «Отсутствует». Поле не отображается для маппингов с типом «rrepopulate»;
 - код преобразования. Нажать «Добавить преобразование». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже;
 - код условия. Нажать «Добавить условие». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже.
- 5) Нажать кнопку «Сохранить».
- 6) Перейти на вкладку «Политики». Вкладка «Политики» недоступна для маппингов с типом «rrepopulate». Нажать кнопку «Добавить политику маппинга», в случае если политика отсутствует или кнопку

Изм.	Подп.	Дата

«Редактировать» напротив политики. Откроется окно «Редактирование политики». Доступные поля для изменения:




- ситуация. Для изменения выбрать ситуацию из выпадающего списка;
 - простое действие. Для изменения нажать кнопку «Простое действие» и заполнить поле «Действие». Кнопка «Простое действие» неактивна, пока не выбрана ситуация;
 - действие по скрипту. Для изменения нажать кнопку «Действие по скрипту». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже.
- 7) Нажать кнопку «Сохранить».
 - 8) Перейти на вкладку «Правило корреляции». Вкладка «Правило корреляции» доступна только для маппингов с типом «geson». Нажать кнопку «Добавить правило корреляции» в случае, если правило отсутствует или «Изменить правило корреляции» в случае, если правило добавлено ранее. Внести изменение, выбрав один из типов скрипта «Java Script», «Groovy», «Query». Общие действия при выборе типа скрипта «Java Script» или «Groovy» описаны ниже.
 - 9) Нажать кнопку «Применить».
 - 10) Перейти на вкладку «Скрипты». Нажать кнопку «Редактировать» и задать значения следующих параметров:
 - действие. Для изменения выбрать действие из выпадающего списка. Возможные действия приведены в пункте 2.2.5;
 - скрипт. Внести изменения в существующие значения или нажать кнопку «Удалить скрипт», затем нажать кнопку «Назначить скрипт». Выбрать тип скрипта, а затем выбрать текст или файл. Общие действия при выборе типа скрипта описаны ниже.
 - 11) Нажать кнопку «Сохранить». В случае ошибки отображается сообщение «Ошибка при обновлении маппинга».

Общие действия при выборе типа скрипта:

- 1) Нажать кнопку «Java Script» или «Groovy» для выбора типа скрипта.
- 2) Выбрать одно из двух действий:

Изм.	Подп.	Дата

- нажать кнопку «Текст», чтобы внести значение в виде кода в поле редактора кода;
- нажать кнопку «Файл», чтобы внести значение в формате файла скрипта. Следует указать название файла, который находится в директории проекта в папке «script». Если файл отсутствует, то при сохранении изменений отобразится ошибка.

При ручном вводе кода скрипта выполняется подсветка синтаксиса, валидация и автодополнение. Для типов скрипта «Java Script», «Groovy» доступны копирование и форматирование кода. Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

2.2.5.3. Удаление объекта «Маппинг»

Для удаления маппинга необходимо в разделе «Маппинги» выполнить следующие действия:

- 1) Выбрать маппинг из списка.
- 2) Нажать кнопку «Удалить». Откроется окно «Удаление маппингов».
- 3) Нажать кнопку «Удалить». После успешного удаления отобразится сообщение «Маппинги успешно удалены». В случае ошибки удаления отобразится сообщение «Ошибка при удалении маппингов».

2.3. Управление учетной записью ресурса

УЗР в Комплексе – это объект, который обеспечивает связку УЗ ЦС с их владельцами. После установки связи владелец может выполнять действия со своей УЗ ЦС из интерфейса Комплекса. Связь с владельцем (бенефициаром) может строиться в следующих режимах:

- 1) Автоматически, при совпадении ключевых атрибутов УЗ ЦС и владельца.
- 2) Вручную администратором для УЗ ЦС, которые нельзя однозначно связать с владельцем по ключевым атрибутам.

Управление УЗР включает в себя следующие действия администратора:

- создание УЗР;

Изм.	Подп.	Дата

- редактирование УЗР;
- связывание УЗ с ее владельцем;
- удаление УЗ пользователя;
- изменение типа УЗ;
- изменение дополнительной УЗ пользователя на основную;
- смена парольной политики УЗР;
- запуск процесса сертификации УЗР.

При обновлении Комплекса с предыдущей версии на версию ПК Ankey IDM 1.9 необходимо создать и однократно запустить периодическое задание «resformsmigrationtask». Задание создает новое поле для типа УЗ в таблице "rf_*" формы ресурса. В противном случае не будет отображаться список УЗ в УЗР. Действия по созданию периодического задания приведены в пункте 2.4.1.

2.3.1. Создание учетной записи ресурса

Создание УЗР выполняется администратором двумя способами:

- 1) С помощью REST-запроса.
- 2) Через меню «Настройка ресурсов».

Для создания УЗР в меню «Настройка ресурсов» администратору необходимо выполнить следующие действия:

- 1) Выбрать раздел «Учетные записи ресурсов», затем нажать кнопку «Создать». Откроется окно «Создание УЗР».
- 2) Заполнить обязательные поля: «Название», «Бенефициар», «Ресурс» и «Тип объекта коннектора».
- 3) При необходимости заполнить необязательные поля: «Описание», «Форма ресурса», «Тип УЗ по умолчанию», «Тип правила связывания», «Правило связывания».
- 4) Нажать кнопку «Создать».

Для просмотра УЗР администратору необходимо зайти в раздел «Учетные записи ресурсов» и нажать кнопку «Обновить».

Пример заполнения атрибутов «Учетной записи ресурса» приведен в таблице 2.13.

Изм.	Подп.	Дата

Таблица 2.13 – Пример заполнения атрибутов «Учетной записи ресурса»

Атрибут	Описание	Пример
Имя	Имя УЗР	CRM
Описание	Описание УЗР	Сетевое рабочее место
Бенефициар	<p>Владелец УЗ в случае синхронизации с ЦС или синхронизируемый объект в случае доверенной синхронизации.</p> <p>По умолчанию можно выбрать одного из владельцев:</p> <ol style="list-style-type: none"> 1) Пользователь. 2) Организация. 3) Роль. 4) Заместитель. 5) Информационная система. <p>При необходимости расширить список поддерживаемых объектов нужно указать дополнительные значения через расширение enum-поля <code>beneficiaryObject</code> у объекта <code>managed/resaccount</code></p>	Пользователь
Ресурс Форма ресурса	<p>Параметры соединения с системой (имя ресурса). Поиск осуществляется по имени и описанию ресурса. Для создания УЗР для автономного ресурса необходимо выбрать ресурс «Offline resource» или другой ресурс созданный по типу ресурса «Offline resource». Форма ресурса заполняется в случае целевой синхронизации.</p> <p>Поиск осуществляется по имени и</p>	AD gis.lan adform

Атрибут	Описание	Пример
	описанию формы ресурса	
Тип УЗ по умолчанию	Типы привязываются к УЗ и определяют их настройки. В поле указывается тип, который будет назначаться по умолчанию к вновь создаваемым УЗ	Базовый
Тип объекта коннектора	Объект системы, с которым будет работать коннектор	account
Тип правила связывания	Тип правила связывания. По умолчанию можно выбрать один из типов: 1) CORRELATION_QUERY_STRING 2) CORRELATION_QUERY. 3) CORRELATION_QUERY_SCRIPT	CORRELATION_QUERY_STRING
Правило связывания	Правила связывания УЗ с владельцем в случае синхронизации с ЦС. Описание настройки правила связывания приведено в пункте 2.16.2	userName eq '\${sAMAccountName}'

2.3.2. Редактирование учетной записи ресурса

Для редактирования доступны следующие атрибуты:

- «Название»;
- «Описание»;
- «Тип правила связывания»;
- «Правило связывания».

Изм.	Подп.	Дата

Для редактирования УЗР через меню «Настройка ресурсов» администратору необходимо выполнить следующие действия:

- 1) В разделе «Учетные записи ресурсов» выбрать УЗР и нажать кнопку «Редактировать». Откроется окно редактирования УЗР.
- 2) Выполнить редактирование необходимых параметров и нажать кнопку «Сохранить».

2.3.3. Связывание УЗ пользователя с ее владельцем

Связывание УЗ с владельцем доступно как для УЗ без владельца, так и для УЗ, у которых владелец уже определен – замена владельца.

Связывание УЗ с владельцами выполняется в меню «Настройка ресурсов»/«Учетные записи ресурсов» в подразделе «Учетные записи».

При отсутствии доступа к деталям УЗР в подразделе «Карточка УЗР», подраздел «Учетные записи» недоступен в интерфейсе Комплекса.

Список УЗ доступен для тех УЗР, для которых настроена форма ресурса. Информация, которую содержит список УЗ в подразделе «Учетные записи», приведена в таблице 2.14.

Таблица 2.14 – Информация, которую содержит список УЗ в подразделе «Учетные записи»

Параметр	Описание	Примечание
Учетная запись	Отображаемое имя УЗ	Настройка выполняется для формы ресурса
Идентификатор	Идентификатор УЗ в ЦС	Определяется атрибутом « UID » коннектора
Владелец	Идентификатор владельца УЗ	
Основная	Признак для основной (primary) УЗ	Для основной УЗ устанавливается флаг, отображающий состояние «да». У пользователя может быть только одна основная УЗ. На основную УЗ по умолчанию назначаются

Изм.	Подп.	Дата


Параметр	Описание	Примечание
		<p>привилегии. Если УЗ не является основной, то возможны следующие варианты:</p> <p>1) Неопределенная (undefined) – когда УЗ без владельца.</p> <p>2) Дополнительная (other) – во всех других случаях</p>
Название парольной политики	Отображаемое имя парольной политики	Отображает связь УЗ пользователя с парольной политикой
Статус корреляции	Статус связи УЗ с владельцем	<p>Возможные значения:</p> <ul style="list-style-type: none"> – связана (в случае успешного связывания); – соответствий не найдено (владелец не найден); – множественное соответствие (несколько соответствий найдено)
Тип УЗ	Тип УЗ	<p>Тип УЗ привязывается к УЗ и определяет ее настройки. Управление типами УЗ описано в разделе 6 документа «Руководство администратора» (72410666.00054-02 95 01). Изменение типа УЗ пользователя приведено в пункте 2.3.7. По умолчанию в Комплексе существуют следующие типы:</p> <ul style="list-style-type: none"> – базовая – тип для стандартных УЗ пользователя, назначаемый по умолчанию; – сервисная – тип для сервисных УЗ пользователя

Поиск в списке УЗ ведется по колонкам «Учетная запись» и «Идентификатор».

Для поиска по УЗ необходимо запустить периодическое задание «reindex» после создания формы ресурсов.

Изм.	Подп.	Дата

Для связывания УЗ с ее владельцем администратору необходимо выполнить следующие действия:

- 1) Выбрать необходимую УЗР в разделе «Учетные записи ресурсов». Откроется карточка УЗР с параметрами.
- 2) Перейти в подраздел «Учетные записи».
- 3) В списке УЗ выбрать одну запись и нажать кнопку «Выбрать владельца». Откроется окно «Выбор владельца».
- 4) В списке пользователей выбрать пользователя или выполнить его поиск. Поиск выполняется по атрибутам «Имя для входа» и «ФИО». Для просмотра дополнительных деталей пользователя следует раскрыть структуру. Отобразится карточка деталей пользователя. Для просмотра структуры подразделений необходимо в карточке деталей пользователя нажать кнопку .
- 5) Нажать кнопку «Выбрать».

2.3.4. Отвязывание владельца от УЗ пользователя

Для отвязывания владельца от УЗ пользователя администратору необходимо выполнить следующие действия:

- 1) Выбрать УЗР в разделе «Учетные записи ресурсов». Откроется карточка УЗР с параметрами.
- 2) Перейти в подраздел «Учетные записи».
- 3) Выбрать запись в списке УЗ.
- 4) Нажать кнопку «Отвязать владельца».

Кнопка активна только для УЗ со статусом «Активна» или «Заблокирована». Если владелец отсутствует или статус УЗ «Создается», то кнопка будет заблокирована.

Если отвязать владельца у УЗ пользователя, то автоматически УЗ перестает быть основной. Тип такой УЗ становится неопределенной и в этом случае очищается поле «Причина блокировки» – поле «disableReason» в БД Комплекса. В случае пустого значения поля «Причина блокировки» не будет выполняться автоматическая блокировка/активация УЗ.

Изм.	Подп.	Дата

Изменение дополнительной УЗ пользователя на основную описано в пункте 2.3.7.

2.3.4.1. Массовое отвязывание владельца от УЗ пользователей

Для массового отвязывания владельца от УЗ пользователей администратору необходимо выполнить следующие действия:

- 1) Выбрать УЗР в разделе «Учетные записи ресурсов». Откроется карточка УЗР с параметрами.
- 2) Перейти в подраздел «Учетные записи».
- 3) Выбрать несколько УЗ в списке. Отобразится количество выбранных УЗ пользователей.
- 4) Нажать кнопку «Отвязать владельца». Откроется окно «Отвязать владельца».
- 5) Нажать кнопку «Отвязать».

Массовое отвязывание УЗ пользователя от владельца доступно в карточке пользователя. Описание приведено в подпункте 2.3.6.5 документа «Руководство пользователя» (72410666.00054-02 96 01).

2.3.5. Удаление УЗ пользователя

Для удаления УЗ пользователя администратору необходимо выполнить следующие действия:

- 1) Выбрать необходимую УЗР в разделе «Учетные записи ресурсов». Откроется карточка УЗР с параметрами.
- 2) Перейти в подраздел «Учетные записи». В списке УЗ выбрать одну или несколько записей и нажать кнопку «Удалить».
- 3) В окне подтверждения действия по удалению УЗ нажать кнопку «Удалить».

Администратор может удалить УЗ пользователей в любых статусах.

Изм.	Подп.	Дата

2.3.6. Изменение типа УЗ пользователя

Для изменения типа УЗ пользователя администратору необходимо выполнить следующие действия:

- 1) Выбрать необходимую УЗР в разделе «Учетные записи ресурсов». Откроется карточка УЗР с параметрами.
- 2) Перейти в подраздел «Учетные записи».
- 3) В списке УЗ выбрать одну запись и нажать кнопку «Сменить тип УЗ».
- 4) В окне смены типа УЗ выбрать один из доступных типов.
- 5) Нажать кнопку «Сохранить».

По умолчанию в Комплексе доступны типы «Базовая» и «Сервисная». Возможность назначения типа на УЗ любых УЗР определяется состоянием флага «Доступен для всех УЗР». Если флаг выключен, то доступность определяется через список конкретных УЗР. Подробное описание приведено в разделе 6.

Создание других типов выполняется администратором и описание приведено в подразделе 5.1.

При изменении типа УЗ, необходимо убедиться, что новый тип поддерживает полномочия текущей УЗ. Если флаг «Разрешить выделение полномочий УЗ данного типа по роли» выключен, то новый тип не поддерживает полномочия, выданные текущей УЗ по роли. В этом случае возникает ошибка «Выбранный тип УЗ не поддерживает наличие полномочий, выданных по роли» и Комплекс не даст сменить тип УЗ.

2.3.7. Изменение дополнительной УЗ пользователя на основную

Основная УЗ это УЗ, на которую по умолчанию назначаются привилегии.

В Комплексе у пользователя может быть только одна основная УЗ. Нельзя изменить УЗ на основную, если у нее не определен владелец.

Предусмотрены автоматические изменения для УЗ в следующих случаях:

- 1) УЗ становится основной после того, как для нее был определен владелец.
- 2) УЗ перестает быть основной после того, как был отвязан владелец.
- 3) УЗ перестает быть основной после того, как был изменен ее тип на «Сервисная».

Изм.	Подп.	Дата

- 4) Создается дубликат основной УЗ, в случае если она была выделена пользователю через роль и перестала быть основной. Подробнее о периодическом задании «evaluatetask» приведено в пункте 2.4.5.

Если у УЗ определен владелец, но она не основная, такая УЗ считается дополнительной.

Для того чтобы сделать УЗ основной администратору необходимо в подразделе «Учетные записи» выполнить следующие действия:

- 1) В списке УЗ выбрать запись.
- 2) Нажать кнопку «Сделать основной». Откроется окно «Смена основной УЗ».
- 3) Нажать «Подтвердить».

УЗ не может быть основной, если она имеет тип «Сервисная» или у ее типа не установлен флаг «Может устанавливаться для основных УЗ». В противном случае отображается сообщение «Учетная запись с текущим типом не может быть основной».

При изменении УЗ с основной на дополнительную удаляются только привилегии у УЗ.

2.3.8. Смена парольной политики для УЗ в УЗР

УЗ пользователя в УЗР может быть связана с парольной политикой по умолчанию. В этом случае эта связь отображается в столбце «Название парольной политики». Связать УЗ пользователя с политикой можно только сменив политику по умолчанию на ту, которая была создана для этой УЗР. Если в Комплексе отсутствуют политики по умолчанию для УЗ, то столбец «Название парольной политики» содержит значение «Отсутствует». Подробнее о парольной политике приведено в подразделе 2.11.

Для смены парольной политики УЗ администратору необходимо в меню «Настройка ресурсов»/«Учетные записи ресурсов» выполнить следующие действия:

- 1) Выбрать УЗР из списка и перейти в подраздел «Учетные записи».
- 2) Выбрать УЗ, для которой нужно сменить парольную политику.
- 3) Нажать кнопку «Сменить парольную политику».

Изм.	Подп.	Дата

- 4) Выбрать из списка парольную политику. Список отображает парольные политики, созданные для данной УЗР со следующими ограничениями:
 - нельзя выбрать парольную политику по умолчанию, так как она назначается автоматически;
 - нельзя выбрать уже назначенную парольную политику.
- 5) Нажать кнопку «Сохранить». Отобразится сообщение «Парольная политика УЗ успешно изменена».

Для возврата к парольной политике по умолчанию администратору необходимо выполнить следующие действия:

- 1) Выбрать УЗР из списка и перейти в подраздел «Учетные записи».
- 2) Выбрать УЗ.
- 3) Нажать кнопку «Отвязать парольную политику», чтобы удалить связку между УЗ и парольной политикой.
- 4) Нажать кнопку «Отвязать». Отобразится сообщение «Парольная политика успешно отвязана». После чего для УЗ сменится парольная политика, назначенная вручную, на парольную политику по умолчанию.

2.4. Управление периодическими заданиями

Периодические задания Комплекса предназначены для выполнения действий с УЗ пользователей и другими объектами Комплекса по расписанию.

Управление периодическими заданиями включает в себя следующие действия администратора:

- создание периодического задания;
- редактирование периодического задания;
- удаление периодического задания;
- запуск и остановка периодического задания.

Управление периодическими заданиями выполняется администратором двумя способами:

- с помощью REST-запроса;
- в меню «Админ. системы»/«Периодические задания».

Изм.	Подп.	Дата

Экземпляры периодического задания создаются на основании шаблона (тип задания), содержащего специфичные для задания атрибуты. Любое поле периодического задания можно настроить, как поле типа «lookup», которое позволяет выполнять поиск атрибута. Для этого следует изменить шаблон и включить в него параметры согласно подпункту 2.7.1.2.3.

Параметры периодического задания включают следующее:

- общие сведения о периодическом задании;
- атрибуты задания, определяемые его шаблоном.

Общие сведения о периодическом задании приведены в таблице 2.15.

Таблица 2.15 – Общие сведения о периодическом задании

Атрибут	Описание	Пример
Название (_id)	Идентификатор (имя) запланированной задачи. Обязательный к заполнению атрибут	AD User Recon
Описание (description)	Описание запланированной задачи	Синхронизация пользователей из Active Directory
Включено (enabled)	Значение «true» активирует расписание задания. Когда это свойство имеет значение «false», Комплекс не запускает задание	true
Тип расписания (type)	В настоящее время Комплекс поддерживает только cron-формат. По умолчанию свойство имеет значение «cron»	cron
Дата начала (startDate)	Используется для планирования запуска запланированной задачи. Если этот параметр не задан, запланированная задача запускается по значению «cron» без ограничения по дате. Для указания даты и	–

Изм.	Подп.	Дата

Атрибут	Описание	Пример
	<p>времени используется следующий формат: YYYY-MM-DD hh:mm:ss</p>	
<p>Дата окончания (endDate)</p>	<p>Используется для планирования окончания запланированной задачи. Задание выполняется по указанную дату включительно. Если этот параметр не задан, запланированная задача запускается по значению «stop» без ограничения по дате</p>	—
<p>Расписание задания (schedule)</p>	<p>Принимает синтаксис выражений cron. Задаваться может в двух режимах в меню «Админ. системы»/«Периодические задания»:</p> <ul style="list-style-type: none"> – обычный режим, в котором указывается период повторений в единицах времени секунды, минуты, часы, дни, месяцы; – в формате cron, в котором можно задавать расширенные сценарии запуска заданий, которые также учитывают день недели, год. <p>Расписание не отображается в обычном режиме, если ранее было задано в формате cron с расширенным сценарием запуска. Обязательный к заполнению атрибут</p>	0/1 * * * * ?
<p>Режим восстановления (misfirePolicy)</p>	<p>Для постоянных запланированных задач этот параметр задает их поведение, если запланированная задача была пропущена. Возможные значения:</p>	fireAndProceed

Изм.	Подп.	Дата

Атрибут	Описание	Пример
	<ul style="list-style-type: none"> – «Продолжение выполнения» («fireAndProceed») – запуск упавшего задания сразу же после запуска Комплекса; – «Запуск нового задания» («doNothing») – запуск нового задания по расписанию. <p>По умолчанию принимает значение «doNothing»</p>	
Конкурентное выполнение (concurrentExecution)	<p>Указывает, могут ли несколько экземпляров одной и той же запланированной задачи выполняться одновременно. По умолчанию принимает значение «false». Несколько экземпляров одной и той же запланированной задачи не могут выполняться одновременно по умолчанию. При этом не допускается, чтобы новая запланированная задача была запущена прежде, чем такая же запущенная задача была завершена. Чтобы запустить несколько запланированных задач одновременно, следует установить параметр в значение «true»</p>	false
Тип задания (invokeService)	<p>Определяет используемый шаблон запланированного задания. Описание типовых заданий приведено в пункте 2.4.5. Обязательный к заполнению атрибут</p>	recontask
invokeContext	<p>Параметры, определяемые типом задания (значение параметра «invokeService»)</p>	–
Уровень логирования	<p>Определяет уровень детализации сведений</p>	info

Изм.	Подп.	Дата

Атрибут	Описание	Пример
(invokeLogLevel)	о работе задания в системных журналах Комплекса. По умолчанию уровень журнала имеет значение «info». Параметр может быть установлен в следующие значения: – trace; – debug; – info; – warn; – error; – fatal	
timeZone	Если не установлен, Комплекс использует часовой пояс системы	–
persisted	Значение «true» является признаком сохранения задания в базе данных (БД). По умолчанию свойство имеет значение «true»	true

2.4.1. Создание периодического задания

Для периодического задания в меню «Админ. системы» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Периодические задания», затем нажать кнопку «Создать». Появится страница создания периодического задания.
- 2) Заполнить обязательные поля «Название» и «Расписание задания».
- 3) Расписание задания может задаваться в двух режимах: с указанием частоты повторов и единиц измерения времени, или в расширенном формате cron. Для выбора режима необходимо нажать кнопку «cron».
- 4) При выборе «Тип задания» отображаются дополнительные параметры задания, определяемые выбранным типом.
- 5) Включить задание можно отметив параметр «Активность задания».

Изм.	Подп.	Дата

- б) Нажать кнопку «Сохранить». Чтобы увидеть созданное задание, в списке следует нажать кнопку «Обновить».

2.4.2. Редактирование периодического задания

При редактировании периодического задания выполняются следующие условия:

- 1) Недоступно для редактирования название периодического задания.
- 2) Недоступно для редактирования значение атрибута «Тип задания». Для использования другого типа задания необходимо создавать новое периодическое задание.
- 3) Выполненные изменения применяться при очередном запуске задания.

Для редактирования периодического задания в меню «Админ. системы» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Периодические задания» и выбрать из списка необходимое задание.
- 2) Нажать кнопку «Редактировать». Откроется карточка периодического задания с параметрами.
- 3) Изменить параметры периодического задания и нажать кнопку «Сохранить».

2.4.3. Удаление периодического задания

При удалении периодического задания выполняется следующее условие: удаление задания не прерывает его работу, если оно выполняется, а только удаляет последующие запуски этого задания.

Для удаления периодического задания в меню «Админ. системы» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Периодические задания» и выбрать из списка одно или несколько периодических заданий. Откроется окно подтверждения с перечнем удаляемых заданий.
- 2) Нажать кнопку «Удалить».


Изм.	Подп.	Дата

2.4.4. Запуск и остановка периодического задания

Запуск периодического задания выполняется только для остановленных/отключенных заданий. Для задач в статусе «выполняется» функция недоступна.

Комплекс поддерживает пакетный запуск заданий.


Для запуска периодического задания в меню «Админ. системы» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Периодические задания» и выбрать из списка одно или несколько остановленных/отключенных заданий.
- 2) Нажать кнопку . Задание будет запущено в течение некоторого времени.

При остановке периодического задания выполняются следующие условия:

- 1) Остановка периодического задания доступна только для заданий, которые в настоящий момент имеют статус «Выполняется».
- 2) Выполнение действия остановки должно обеспечиваться периодическим заданием, в противном случае задание будет продолжать выполнение. Все задания, включенные в состав Комплекса, поддерживают остановку.
- 3) Для периодических заданий, имеющих поддержку функции остановки. Сама остановка может происходить не мгновенно, а по прошествии какого-то времени, необходимого для завершения минимальных изменений уже начатых на момент остановки.

Для остановки периодического задания в меню «Админ. системы» администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Периодические задания» и выбрать из списка одно выполняемое задание.
- 2) Нажать кнопку . Задание будет приостановлено в течение некоторого времени.

2.4.5. Описание периодических заданий Комплекса

2.4.5.1. Тип задания «*evaluatetask*»

Задание «*evaluatetask*» выполняется для пересчета привилегий, выделенных пользователю через роль. Параметр задания «*batch*» – количество одновременно

Изм.	Подп.	Дата

пересчитываемых ролей (например, 500). При назначении, отзыве или изменении роли у пользователя данные помечаются, как требующие пересчета привилегий. Задание просматривает эти данные и пересчитывает состав привилегий, при этом добавляя и/или удаляя их. Период запуска по умолчанию – 1 минута.

Правила пересчета привилегий:

- если назначение «Роль» и у пользователя нет УЗ, то ему выделяется УЗ с привилегиями выделяемой роли;
- если назначение «Роль», уже выделена УЗ, и у УЗ нет каких-то привилегий, которые дает данная роль, то к УЗ добавляются данные привилегии;
- если у привилегии типа назначения «Роль», то при отзыве роли, у УЗ удаляются привилегии, которые назначались данной ролью (при условии, что нет других ролей, которые бы давали данные привилегии);
- если выделена роль и у УЗ есть привилегии с типом назначения не «Роль», то при выполнении задания у тех привилегий, которые даются ролью, тип назначения изменится на «Роль»;
- если «Дата начала» сегодня или в прошлом, а «Дата окончания» в будущем, то роль применяется к пользователю (выделяется или обновляется УЗ);
- если «Дата начала» в будущем, то роль не применяется к пользователю. Применение произойдет по наступлении «Дата начала»;
- если «Дата окончания» в прошлом, то привилегии, выделяемые по роли, отзываются;
- если «Дата окончания» сегодня или в будущем, то привилегии, выделяемые по роли, не отзываются. Отзыв произойдет по наступлении «Даты окончания».

2.4.5.2. Тип задания «reindextask»

Задание «reindextask» выполняется для построения/обновления индекса объектов службы поиска. Сейчас индекс создается только при создании нового объекта в Комплексе и обновлении существующего. Если в Комплексе уже

Изм.	Подп.	Дата

существуют объекты, то для них индекс не создается. В случае сбоя/недоступности службы поиска, при создании нового объекта индекс также не будет создан, и данные, полученные через полнотекстовый поиск, будут неактуальными.

Комплекс позволяет настроить списки объектов, которые будут обрабатываться периодическим заданием «reindextask». Настройки выполняются в веб-консоли «Swagger UI» через REST API Комплекса.

Объекты исключения/включения указываются в параметре «invokeContext». Не следует одновременно указывать объекты исключения и включения, в этом случае задание «reindextask» выполняться не будет.

Варианты настройки задачи «reindextask»:

- 1) Если в параметре «invokeContext» отсутствуют значения, запуск задачи выполняется на все объекты Комплекса. Вариант используется по умолчанию.
- 2) Если в параметре «invokeContext» указаны списком объекты исключения в параметре «excludedResources», запуск задачи выполняется на все объекты, кроме исключенных.
- 3) Если в параметре «invokeContext» указаны списком объекты включения в параметре «includeResources», запуск задачи выполняется на все указанные объекты.

Дополнительно в параметре «includeResources» можно указать:

- фильтр «filter» – ограничивает выборку записей, попадающих в индекс при переиндексации. Например, filter: /timestamp gte <>;
- параметр «batch» – определяет размер пакета записей.

Фильтр «filter» и параметр «batch» работают со всеми доступными полями объектов «managed/» и «repo/», хранящимися в БД.

Список объектов, которые индексируются по умолчанию:

- «managed/restype»;
- «repo/audit/activity»;
- «managed/rule»;
- «managed/is»;
- «managed/resform»;

Изм.	Подп.	Дата

- «managed/user»;
- «managed/substitution»;
- «repo/accessfile»;
- «managed/resaccount»;
- «systemAuthzFunction»;
- «managed/organization»;
- «managed/approle»;
- «repo/incident»;
- «repo/passwordpolicy»;
- «scheduler-service-group»;
- «config»;
- «managed/connector»;
- «repo/accessrole»;
- «informationSystemRequestAppRoles».

Пример REST-запроса создания задачи «reindextask» с исключением объектов из переиндексации:

```
{
  "_id": "reindex2",
  "enabled": false,
  "persisted": true,
  "misfirePolicy": "fireAndProceed",
  "schedule": "30 0/1 * * * ?",
  "type": "cron",
  "invokeService": "org.forgerock.openidm.reindextask",
  "invokeContext": {
    "excludedResources": [
      "repo/audit/activity",
      "managed/user",
      "scheduler-service-group"
    ]
  },
  "invokeLogLevel": "info",
  "timeZone": null,
  "startDate": null,
  "endDate": null,
  "concurrentExecution": false,
  "lastStartTime": null,
  "lastEndTime": null,
  "lastStatus": "notstarted",
  "lastErrorDescription": null,
  "description": ""
}
```

Изм.	Подп.	Дата

Пример REST-запроса создания задачи «reindextask» с включением объектов в переиндексацию:

```
{
  "_id": "reindex3",
  "enabled": false,
  "persisted": true,
  "misfirePolicy": "fireAndProceed",
  "schedule": "30 0/1 * * * ?",
  "type": "cron",
  "invokeService": "org.forgerock.openidm.reindextask",
  "invokeContext": {
    "includedResources": [
      {
        "resource": "repo/audit/activity",
        "filter": "timestamp gte '2022-11-17'",
        "batch": 5000
      },
      {
        "resource": "managed/user",
        "batch": 5000
      },
      {
        "resource": "scheduler-service-group"
      }
    ]
  },
  "invokeLogLevel": "info",
  "timeZone": null,
  "startDate": null,
  "endDate": null,
  "concurrentExecution": false,
  "lastStartTime": null,
  "lastEndTime": null,
  "lastStatus": "notstarted",
  "lastErrorDescription": null,
  "description": ""
}
```

Пример фильтра, который ограничивает выборку из таблицы аудита только теми записями, которые участвуют в полнотекстовом поиске:

```
"filter": "(/objectId sw 'managed/user' or /objectId sw 'managed/usrorg' or /objectId sw 'managed/usrapprole' or /objectId sw 'resform/' and ((/message in 'create,delete') OR (/message in 'patch,update' and /changedFields co 'beneficiar_objectid'))) or /objectId sw 'managed/substitution' or /objectId sw 'managed/approle' or /objectId sw 'managed/orgapprole' or /objectId sw 'managed/approleresaccount') and !(runAs eq 'SYSTEM')
```

2.4.5.3. Тип задания «failedsynctask»

Задание «failedsynctask» выполняется для операций выделения/обновления/удаления УЗ в ЦС, ранее завершённые с ошибкой. При повторе

Изм.	Подп.	Дата

операции обновления УЗ, если в форме ресурса определено поле «пароль», то оно не будет изменено. Параметры задания отсутствуют.

2.4.5.4. Тип задания «workflowrequesttask»

Задания «workflowrequesttask» выполняется для исполнения согласованных заявок. Если заявка была успешно согласована на всех этапах, то задание выполняет действия из данного типа заявки. Например, назначает пользователю ранее запрошенные роли. В случае успешного выполнения заявка переводится в статус «Исполнена».

Параметр задания «batch» – количество одновременно обрабатываемых заявок (например, 500).

2.4.5.5. Тип задания «clearpasswordhistorytask»

Задание «clearpasswordhistorytask» выполняется для очищения истории паролей от неиспользуемых данных, относящихся к тем ресурсам, для которых нет в конфигурации парольной политики правила контроля истории паролей. Чтобы задание отработало, необходимо отключить историю паролей, убрав параметр «passwordHistoryCount» в файле passwordpolicy.json. Параметры задания отсутствуют.

2.4.5.6. Тип задания «recontask»

Задание «recontask» выполняется для синхронизации УЗ из ЦС или доверенного источника в режиме получения всех записей (полная синхронизация).

Параметры задания «recontask»:

- «resAccName» – имя УЗР ЦС, для которой выполняется синхронизация УЗ. Например, XMLResAccount;
- «allowEmptySourceSet» – предотвращает ошибку запуска задания, если из ресурса-источника (ЦС или доверенного источника) приходит пустой набор данных. Возможные значения true/false. Чтобы синхронизация пустого ресурса-источника продолжалась, необходимо установить для параметра «allowEmptySourceSet» значение «true»;
- «runTargetPhase» – запускает фазу реконсильации для объектов ресурса-получателя (Комплекса). Возможные значения true/false. По умолчанию принимает значение false, в таком случае выполняется фаза реконсильации

Изм.	Подп.	Дата

только для объектов ресурса-источника. Чтобы запросы на синхронизацию объектов выполнялись как для ресурса-источника, так и для ресурса-получателя, необходимо установить для параметра «runTargetPhase» значение «true». При включенном параметре "runTargetPhase" : true должны быть настроены запросы targetQueryFullEntry и targetQuery.

Для выборки данных из ресурса-источника/получателя используются запросы sourceQueryFullEntry, targetQueryFullEntry, sourceQuery, targetQuery, где:

- sourceQueryFullEntry, sourceQuery – запросы для ресурса-источника;
- targetQueryFullEntry, targetQuery – запроса для ресурса-получателя.

Запросы sourceQueryFullEntry, targetQueryFullEntry позволяют настроить выборку данных на возврат полных объектов. Возможные значения true/false. По умолчанию принимает значение false, при котором забираются все уникальные идентификаторы и по каждому выполняется запрос в ресурс-источник/получатель. Для возврата полных объектов необходимо установить значение true.

Запросы sourceQuery, targetQuery позволяют ограничить выборку данных из ресурса-источника/получателя только необходимыми объектами.

Запросы sourceQuery, targetQuery включают дополнительные параметры:

- 1) `_queryFilter` – фильтр для объекта-источника. Список доступных условий, поддерживаемых фильтром, зависит от типа коннектора и приведен в пункте 2.16.1. Например, для ADConnector фильтр будет задан `dn eq 'CN=XX,DC=domen'`, а для SAP-системы – `username eq 'MYNAME'`.
- 2) `_queryId` – predeterminedенные в Комплексе query-запросы. Из доступных по умолчанию:
 - `query-all` – для получения всех объектов со всеми атрибутами одним запросом;
 - `query-all-ids` – возвращает уникальные идентификаторы всех объектов.

2.4.5.7. Тип задания «livesynctask»

Задание «livesynctask» выполняется для синхронизации УЗ из ЦС или доверенного источника в режиме получения последних измененных записей

Изм.	Подп.	Дата

(инкрементальная синхронизация). Параметр задания «resAccName» – имя УЗР ЦС, для которой выполняется синхронизация УЗ (например, XMLResAccount).

2.4.5.8. Тип задания «lookiprecon»

Задание «lookiprecon» выполняется для синхронизации данных в справочники из ЦС или доверенного источника. Например, применяется для синхронизации справочника полномочий (групп, ролей, прав доступа) из ЦС.

Параметры задания «lookiprecon»:

- «resName» – имя ресурса, из которого выполняется синхронизация справочника. Например, XMLResource;
- «connectorObjectType» – тип объекта коннектора, данные которого будут синхронизироваться в справочник. Например, group;
- «refBookType» – наименование справочника, в который будут синхронизироваться данные. Например, XML Groups;
- «refBookCode» – атрибут объекта ЦС, значение которого будет записано в виде кода в справочник. Может содержать два значения: source – название поля объекта из ресурса и transform – правило, по которому надо преобразовать значение перед записью в справочник. Например, {"source" : "id"};
- «refBookName» – атрибут объекта ЦС, значение которого будет записано в виде наименования в справочник. Может содержать два значения: source – название поля объекта из ресурса и transform – правило, по которому надо преобразовать значение перед записью в справочник. Например, {"source" : "groupName"};
- «deleteIfUnassigned» – признак удаления неактуальных справочных данных при синхронизации. Например, true. Неактуальными считаются те справочники, у которых имя ресурса («resName») и тип справочника («refBookType») отличаются от соответствующих параметров в периодическом задании.

Изм.	Подп.	Дата

Для того чтобы ограничить выборку данных из систем только необходимыми объектами или же заставить систему вернуть все поля объектов, следует указывать запросы для source/target систем.

Для этого используются запросы sourceQueryFullEntry, targetQueryFullEntry, где source/targetQueryFullEntry – параметры для источника (ЦС) и получателя (Комплекс) соответственно:

- sourceQueryFullEntry – возможные значения true/false;
- targetQueryFullEntry – возможные значения true/false.

Если параметр принимает значение false, то Комплекс всегда считает полученный объект неполным и данный объект будет заново запрашиваться из системы, при необходимости. Если параметр принимает значение true, то Комплекс всегда считает полученный объект полным и данный объект будет всегда считываться из памяти.

Для source/target систем можно задавать различные query-запросы.

Из доступных по умолчанию:

- query-all: для получения всех объектов системы со всеми атрибутами одним запросом;
- query-all-ids: возвращает только идентификаторы объектов.

Если задать sourceQueryFullEntry = true, не указывая sourceQuery, то выполняется только запрос query-all-ids, а эти данные будут считаться полным объектом и никакие дополнительные запросы не будут отправляться.

Дополнительные поля можно настроить в периодическом задании. При этом дополнительное поле должно быть определено в конфигурационном файле extend.json с указанием policy схемы. Если поле определено как обязательное в схеме policy, то оно также будет обязательным к указанию при создании/редактировании периодической задачи «lookupreson».

Пример extend конфигурации:

```
{
  "managed/referencebook": {
    "notAvailableForRequest": {
      "managed": {
        "required": false,
        "display": [
```

Изм.	Подп.	Дата

```

        "create",
        "view",
        "update"
    ],
    "fieldType": "flag"
},
"policy": {
    "schema": {
        "type": "boolean"
    }
},
"repo": {
    "column": "na",
    "type": "BOOLEAN"
}
}
}
}

```

Пример периодического задания с дополнительными правами:

```

{
  "_id": "schedule-pgsql-roles-lookup-recon",
  "enabled": true,
  "persisted": true,
  "type": "cron",
  "schedule": "*/10 * * * * ?",
  "concurrentExecution": false,
  "invokeService": "lookuprecon",
  "invokeContext": {
    "resName": "PostgreSql",
    "connectorObjectType": "account",
    "refBookType": "PostgreSql Roles",
    "refBookCode": {
      "source": "LoginName"
    },
    "refBookName": {
      "source": "LoginName"
    },
    "notAvailableForRequest": {
      "source": "ENCRYPTED"
    }
  }
}
}

```

2.4.5.9. Тип задания «warnbeforeexpiredaccountstask»

Задание «warnbeforeexpiredaccountstask» выполняется для уведомления пользователя о просроченном пароле его УЗ ЦС в день просрочки и за несколько дней до момента наступления события.

Изм.	Подп.	Дата

Параметры задания:

- «warnBeforeExpired (days)» – количество дней до момента наступления события, когда будет направлено предупреждение о предстоящей просрочке времени действия пароля;
- «emailSubject» – тема уведомления (например, «Требуется сменить пароль ваших УЗ»);
- «emailBodyBeforeExpired» – текст уведомления, направляемого за несколько дней до момента наступления события (например, <div>Ваш пароль для УЗ \${resAccName} устареет через \${expireDays} дн..</div>);
- «emailBodyExpired» – текст уведомления, направляемого в день просрочки (например, <div>Ваш пароль для УЗ \${resAccName} устарел.</div>).

2.4.5.10. Тип задания «disableinactiveaccountstask»

Задание «disableinactiveaccountstask» выполняется для блокировки УЗ пользователей в ЦС, если пользователь не входил в ЦС несколько дней.

Параметры задания:

- «resFormName» – имя формы ресурса ЦС, в которой присутствует атрибут «Дата последнего входа» (lastLogonDate). Например, ADResForm;
- «inactivePeriod (days)» – период неактивности пользователя в днях.

Задача блокирует УЗ в заданной форме ресурса, если:

- статус УЗ active («Активна»);
- УЗ (accountType): primary (основная) или other (дополнительная); «Дата последнего входа» (lastLogonDate) заполнена и с этой даты прошло не меньше заданного в задаче числа дней (inactivePeriod (days)).

Для использования задания необходимо, чтобы ЦС хранила дату последнего входа/использования УЗ. Должна быть настроена синхронизация даты последнего входа/использования УЗ в служебный атрибут lastLogonDate (в формате уууу-ММ-dd). После активации УЗ (например, администратором вручную или по заявке) пользователь должен осуществить под ней вход в ЦС до следующего запуска задания, иначе УЗ снова заблокируется.

Изм.	Подп.	Дата

2.4.5.11. Tun задания «roleminingtask»

Задача «roleminingtask» выполняется для анализа полномочий, которые назначены пользователям. В результате анализа формируются предложения (отчет) по составу и содержанию новых ролей, с включенными в них полномочиями.

Параметры задания:

- «miningAlgorithm» – алгоритм анализа ролевой модели. По умолчанию поддерживается только алгоритм «HPRoleMinimization» (например, HPRoleMinimization);
- «wd», «wu», «wr», «wp», «wh» – весовые коэффициенты для использования алгоритмом. По умолчанию принимают значение 1.

2.4.5.12. Tun задания «usercertificationtask»

Задание «usercertificationtask» выполняется для формирования заявок на сертификацию ролей подчиненных сотрудников и направление заявок линейным руководителям. Для работы задания должен быть настроен рабочий поток «certificateUserAppRoles» для действия «verify» на объект «endpoint/user-certification».

Параметры задания:

- «ifManagerNotDefined» – действие в случае, если линейный руководитель не найден. Может принимать значения: «do-nothing» и «assign-to-default-role». Если задано «do-nothing», то при отсутствии руководителя у пользователя, заявка на сертификацию не будет создаваться. Если задано «assign-to-default-role», то при отсутствии руководителя у пользователя, заявка на сертификацию будет назначаться на роль, указанную в параметре задания «defaultRole»;
- «defaultRole» – роль по умолчанию, на которую будут назначаться заявки на сертификацию, в случае отсутствия линейного руководителя.

2.4.5.13. Tun задания «aprolecertificationtask»

Задание «aprolecertificationtask» выполняется для формирования заявок на сертификацию настроек ролей определенной информационной системы (ИС) и направление заявок участникам роли сертификатора, заданной в ИС. Для работы задания должен быть настроен рабочий поток «certificateAppRoles» для действия

Изм.	Подп.	Дата

«verify» на объект "endpoint/approle-certification/*". Параметр задания «targetInfoSystem» – имя ИС, для ролей которой выполняется сертификация. По умолчанию принимает значение «default-is», для которого сертификация не выполняется.

2.4.5.14. Tun задания «*email.request.processing.task*»

Задание «email.request.processing.task» читает почтовый ящик, указанный в файле конфигурации email.request.processing.json, и выполняет требуемые действия с заявкой. Если действие успешно выполнено, письмо удаляется.

2.4.5.15. Tun задания «*approlesbyrulemessagetask*»

Задание назначает или отзывает роли пользователю по установленному для роли правилу автоназначения. Параметр задания «batch» – количество одновременно обрабатываемых задач по автоназначению ролей.

2.4.5.16. Tun задания «*deferredchangestatustask*»

Задание «deferredchangestatustask» изменяет статус (активация и/или блокировка) пользователя в зависимости от наступления даты:

- «Дата принудительной активации»;
- «Дата принудительной блокировки».

Запуск осуществляется на первой секунде каждых суток.

2.4.5.17. Tun задания «*closeNonActualIncidentsTask*»

Задание «closeNonActualIncidentsTask» выполняется для закрытия неактуального инцидента:

- при изменении и удалении правила;
- при удалении роли ответственного за инцидент;
- при изменении или удалении нарушителя;
- при изменении причины инцидента (учитываются вложенные роли);
- если инцидент создан на роль, которую уже вручную отозвали;
- если инцидент создан на пользователя, который удален.

2.4.5.18. Tun задания «*scriptexecution*»

Задание «scriptexecution» позволяет запускать скрипты JavaScript.

Изм.	Подп.	Дата

Параметры задания:

- «file» – путь к файлу скрипта;
- «input» – параметры, передаваемые скрипту (например, "first": 2, "second": 3);
- «source» – код скрипта, в случае если не используется файл (например, `java.lang.System.out.println("Test input:" + input.first + ' ' + input.second);`);
- «type» – тип скрипта, поддерживается только «text/javascript» (например, text/javascript).

2.4.5.19. Tun задания «*applyAllSodRulesTask*»

Задание «*applyAllSodRulesTask*» позволяет запускать все правила SoD. После запуска создаются инциденты и назначаются на ответственного за инциденты.

2.4.5.20. Tun задания «*update-user-assignments-materialized-view*»

Задание «*update-user-assignments-materialized-view*» обновляет информацию о ролях пользователя в подразделе «Роли» в меню «Пользователи». Период запуска по умолчанию – 20 минут.

2.4.5.21. Tun задания «*updateRiskValueTask*»

Задание «*updateRiskValueTask*» обновляет уровень риска/общего риска роли.

Параметры задания:

- 1) «filter» – фильтр запроса, при помощи которого осуществляется поиск ролей для обновления их собственного уровня риска;
- 2) «newValue» – значение риска, которое будет присвоено после выполнения периодического задания;
- 3) «resource» – выбор ресурса - роли;
- 4) «batch» – количество одновременно обновляемых ролей (например, 500).

2.4.5.22. Tun задания «*userRiskActualizationTask*»

Задание «*userRiskActualizationTask*» обновляет уровень риска пользователя.

Параметры задания:

- 1) «filter» – фильтр запроса, который ограничивает выборку пользователей для обновления их риска.

Изм.	Подп.	Дата

- 2) «batch» – количество одновременно обновляемых пользователей.

2.4.6. Настройка службы запуска периодических заданий

Настройка службы запуска периодических заданий выполняется в файле `conf/scheduler.json`.

Описание конфигурационного файла `scheduler.json` приведено в разделе 26 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

2.4.7. Описание настроек расписания в формате cron

Расписание периодического задания задается в формате выражения `cron` и состоит из следующих полей:

- 1) Секунды (0-59).
- 2) Минуты (0-59).
- 3) Часы (0-23).
- 4) День месяца (1-31).
- 5) Месяц (1-12 или JAN-DEC).
- 6) День недели (1-7 или MON-SUN).
- 7) Год (текущий-2099).

Поля могут содержать следующие специальные символы:

- символ косая черта «/» обозначает приращение значения. Например, «5/15» в поле «секунды» означает каждые 15 секунд, начиная с пятой секунды;
- знак вопроса «?» означает, что в поле не должно быть указанной величины. Таким образом, если устанавливается день недели, можно вставить «?» в поле «день недели» для обозначения того, что значение «день недели» несущественно;
- знак астериска «*» обозначает, что любое возможное значение может быть принято для данного отдельного поля;
- знак дефиса «-» обозначает интервал значений.

Примеры:

- 1) `0 */15 * * * ?` (запуск каждые 15 минут).
- 2) `0 0 0 */1 * ?` (запуск каждый день).

Изм.	Подп.	Дата

- 3) 0 0 0 1 3 ? (запуск 01 марта в 00:00).
- 4) 0 0 11-13 * * ? (запуск с 11:00 по 13:00 часов).
- 5) 0 30 10 ? * * (запуск в 10:30 каждый день).

2.4.8. Просмотр списка периодических заданий

В списке задач в разделе «Периодические задания» отображаются все периодические задачи в Комплексе. Список отсортирован по столбцу «Название».

В списке отображаются следующие атрибуты периодической задачи:

- 1) «Название» – отображается имя задачи.
- 2) «Последний запуск» – отображается период времени и статус, с которым задача выполнялась в последний раз.
- 3) Статусы задач:
 - «Не выполнялась» – означает, что задача ни разу не запускалась;
 - «Успешно (<время с последнего выполнения>» – означает, что задача была успешно выполнена;
 - «Ошибка (<время с последнего выполнения>» – означает, что задача завершена с ошибкой, а на вкладке «Информация» в блоке «Ошибка» выводится сообщение об ошибке;
 - «Выполняется» – означает, что задача выполняется непосредственно в текущий момент просмотра.
- 4) «Включена» или «Выключена» – показывает состояние активности задания.

Возможные значения переменной <время с последнего выполнения>:

- ru:«Менее минуты назад»/en«Less minute ago» – означает, что последнее выполнение задачи производилось менее 60 секунд назад;
- <количество минут числом> ru:" мин. назад«/en:» min. ago" – означает, что последнее выполнение задачи производилось менее 60 минут назад, например: «43 мин. назад»/en:«43 min. ago»;
- <количество часов назад числом> ru:" ч. назад«/en:» hr. ago" – означает, что последнее выполнение задачи производилось менее 24 часов назад;

Изм.	Подп.	Дата

- <количество дней назад числом> ru:" дн. назад"/en:"d. ago" – означает, что последнее выполнение задачи производилось менее 30 дней назад;
- <количество месяцев назад числом> ru:<мес. назад>/en:<mo. ago> – означает, что последнее выполнение задачи производилось не более 12 месяцев назад;
- ru:<Больше года назад>/en:<Over year ago> – означает, что последнее выполнение задачи производилось более 12 месяцев назад.

Округление в меньшую сторону производится при ≤ 0.5 единицы измерения, а при > 0.5 единицы измерения округление производится в большую сторону.

2.5. Настройка справочников

Справочник (referencebook) – это набор значений с атрибутами определенного типа, поступающими из ЦС при синхронизации справочников. Используется при создании полей с predetermined набором значений.

В меню «Админ. системы»/«Справочники» администратор может выполнить следующие действия:

- 1) Создание справочника.
- 2) Удаление справочника.
- 3) Редактирование справочника.
- 4) Экспорт/импорт значений справочника из Excel.
- 5) Управление файлом в справочнике.
- 6) Изменение видимости и поиска настраиваемых полей в справочнике.

Список справочников отсортирован по имени ресурса и содержит следующие данные:

- «Ресурс» – имя ресурса («resName») ЦС, для которой создан справочник;
- «Название справочника» – тип справочника («refBookType»).

Для одного ресурса в списке может быть указано несколько типов справочников, а один и тот же тип справочника может быть указан в нескольких ресурсах.

Детали справочников содержат следующую информацию:

- поиск;

Изм.	Подп.	Дата

- код и имя;
- настраиваемые поля и прикрепленные файлы в случае изменения видимости и поиска настраиваемых полей в справочнике.

Изменение видимости и поиска настраиваемых полей в справочнике приведено в пункте 2.5.6.

Администратор может производить поиск по деталям и по группе справочников. Поиск по деталям производится по коду и по имени. Поиск по группе справочников производится по ресурсу, по типу, по коду, по имени.

Действия по управлению файлом в справочнике приведены в пункте 2.5.5.

2.5.1. Создание справочника

Для создания справочника следует выполнить следующие действия:

- 1) Нажать кнопку «Действие».
- 2) Нажать кнопку «Создать». Откроется окно «Создание справочника».
- 3) Заполнить обязательные поля «Название справочника», «Ресурс», «Код/Имя».
- 4) Нажать кнопку «Создать». Отобразится сообщение «Справочник успешно создан».

Действия по загрузке файла в справочник приведены в пункте 2.5.5.

При заполнении полей следует учитывать следующие особенности:

- 1) Название справочника на один и тот же ресурс должно быть уникальным. В противном случае отобразится ошибка «Для выбранного ресурса уже существует справочник с таким названием».
- 2) Значение в поле «Код» должно быть уникальным. В противном случае отобразится ошибка «Такое значение уже существует».
- 3) Кнопка «Создать» не будет доступна пока все поля не заполнены корректно.

2.5.2. Удаление справочника

Для удаления справочника следует выполнить следующие действия:

- 1) Выбрать справочник из списка.
- 2) Нажать кнопку «Действие».

Изм.	Подп.	Дата

- 3) Нажать кнопку «Удалить». Откроется окно «Удаление справочника».
- 4) Нажать кнопку «Удалить» для подтверждения. После удаления отобразится сообщение «Справочник успешно удален».

При удалении справочника удаляются все его значения.

2.5.3. Управление значениями справочника

Управление значениями справочника выполняется в правой части экрана на вкладке «Значения» и включает в себя следующие события:

- редактирование;
- добавление;
- удаление.

Для редактирования записи справочника администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Редактировать запись». Откроется окно «Редактирование записи».
- 2) Изменить значения полей «Код» и «Имя».
- 3) Нажать кнопку «Сохранить».

Действия по управлению файлом в справочнике приведены в пункте 2.5.5.

Для добавления новой записи в справочник администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Добавить записи». Откроется окно «Добавление записей». При этом не отображаются существующие значения.
- 2) Нажать кнопку «Добавить значение».
- 3) Ввести значения в поля «Код» и «Имя».
- 4) Нажать кнопку «Создать».

При изменении полей следует учитывать особенности их заполнения, которые приведены в пункте 2.5.1.

Для удаления записи в справочнике администратору необходимо выполнить следующие действия:

- 1) Отметить флагом одну или несколько записей.

Изм.	Подп.	Дата

- 2) При необходимости нажать кнопку «Выбранные», Комплекс отобразит выбранные записи.
- 3) Нажать кнопку «Удалить записи». Откроется окно «Удаление записей».
- 4) Нажать кнопку «Удалить».

Поиск по списку записей осуществляется по всем атрибутам. Управление записями доступно для результатов поиска.

2.5.4. Экспорт/импорт значений справочника из Excel

2.5.4.1. Экспорт значений справочника из Excel

Для экспорта значений справочника необходимо в разделе «Справочники» выполнить следующие действия:

- 1) Нажать кнопку «Действие».
- 2) Нажать кнопку «Экспорт». Откроется окно «Экспорт справочников».
- 3) Ввести название ресурса в поле «Ресурс».
- 4) Выбрать при необходимости «Название справочника» для экспорта одного справочника. Для экспорта всех справочников следует оставить поле пустым.
- 5) Выбрать формат сохранения xls/xlsx.
- 6) Нажать кнопку «Сохранить». Начнется стандартная загрузка файла. Отобразится сообщение «Справочники успешно экспортированы».

Файл экспорта содержит следующие столбцы:

- `_rev` – количество изменений записи;
- `resource_id` – id-ресурса;
- `refBookType` – название справочника;
- `_id` – id-записи справочника;
- `refBookCode` – код записи справочника;
- `_oid` – числовой id-записи справочника;
- `refBookName` – имя записи справочника;
- `status` – статус записи «active»/«deleted»;
- `document` – для настраиваемого поля отображается только название загруженного файла.

Изм.	Подп.	Дата

Файл экспорта поддерживает настраиваемые поля.

2.5.4.2. Импорт значений справочника из Excel

Для импорта значений справочника необходимо в разделе «Справочники» выполнить следующие действия:

- 1) Нажать кнопку «Действие».
- 2) Нажать кнопку «Импорт». Откроется окно «Импорт справочников».
- 3) Ввести название ресурса в поле «Ресурс».
- 4) Нажать кнопку «Добавить файл».
- 5) Выбрать файл импорта в стандартном диалоговом окне.
- 6) Нажать кнопку «Импортировать». Отобразится сообщение «Файл загружен без ошибок».

После загрузки файла отображается статистика:

- 1) Новые справочники.
- 2) Обновленные справочники.
- 3) Удаленные справочники.

Нулевая статистика отображается при импорте файла, в который не было внесено изменений. При этом кнопка «Импортировать» недоступна.

На загрузку файла действуют следующие ограничения:

- можно импортировать только один файл;
- доступный тип файла: xls/xlsx.

Обязательные столбцы при импорте справочников:

- resource_id – id-ресурса;
- refBookType – название справочника;
- _id – id-записи справочника;
- refBookCode – код записи справочника;
- refBookName – имя записи справочника;
- status – статус записи «active»/«deleted».

При подготовке файла импорта следует учитывать следующее:

- при создании и редактировании записи справочника следует указывать значение «active» в столбце «status»;

Изм.	Подп.	Дата

- при удалении записи справочника необходимо указывать значение «deleted» в столбце «status»;
- новая запись не должна содержать значение в столбце _id;
- столбец с типом «document» будет проигнорирован при импорте.

Файл импорта поддерживает настраиваемые поля.

Перечень возможных предупреждений при импорте из-за ошибок в импортируемом файле:

- 1) Отсутствует лист «Справочники».
- 2) Требуемый заголовок отсутствует в файле – отсутствует обязательный столбец.
- 3) Не заполнены обязательные поля в файле.
- 4) Ошибка идентификатора справочника – файл содержит неверный id-записи справочника.
- 5) Ошибка идентификатора ресурса – значение «resource_id» в файле не совпадает с id-ресурса, указанного в поле «Ресурс» при импорте.
- 6) Дублирование записи в справочнике – файл содержит новые записи справочника с одинаковым значением в столбце refBookCode и разными значениями в столбце refBookName.
- 7) Справочник уже существует – файл содержит новую запись справочника, которая уже существует.
- 8) Неверный тип файла.
- 9) Неверный формат файла.
- 10) Ошибка в типе значения.

2.5.5. Управление файлом в справочнике

Управление файлом в справочнике включает в себя следующие события:

- загрузка;
- изменение;
- удаление;
- скачивание.


Изм.	Подп.	Дата

Для управления файлом администратору необходимо выполнить следующие действия:

- 1) Отобразить настраиваемое поле типа «document» в конфигурационном файле conf/extend.json. Описание приведено в подразделе 2.7.5.8.
- 2) Настроить формат и допустимый размер загружаемого файла в конфигурационном файле conf/document.json. Описание приведено в подразделе 4.5.


Если выполнено изменение поиска по настраиваемому полю, то поиск будет работать по названию загруженного файла.

Для загрузки файла в справочник администратору необходимо выполнить следующий действия:

- 1) Выполнить создание или редактирование записи справочника согласно пунктам 2.5.1 и 2.5.3.
- 2) В настраиваемом поле типа «document» нажать значок .
- 3) Загрузить файл. В случае ошибки отобразится значок с соответствующим сообщением.
- 4) Нажать кнопку «Создать»/«Сохранить».

Для изменения файла следует удалить загруженный файл и повторить действия по загрузке.

Для удаления файла следует нажать значок .

Для скачивания файла следует нажать значок . После чего начнется стандартная загрузка файла.

2.5.6. Изменение видимости и поиска настраиваемых полей в справочнике


Перед настройкой видимости и поиска настраиваемых полей в справочнике необходимо настроить их отображение и поиск в конфигурационном файле extend.json. Описание настройки приведено в пункте 2.7.1.2.7.

При создании справочника по умолчанию отображаются поля «Код» и «Имя», по которым осуществляется поиск. Для настраиваемых полей необходимо выполнить

Изм.	Подп.	Дата

изменение, чтобы они стали видимыми и доступными для поиска. Изменение видимости и поиска настраиваемых полей задается после создания справочника.

Для изменения видимости и поиска настраиваемых полей в справочнике администратору необходимо выполнить следующие действия:

- 1) Нажать значок . Откроется окно «Настройка видимости и поиска».
- 2) Отметить видимость и поиск для настраиваемых полей из списка. В списке будут отображаться только те настраиваемые поля, которые были указаны в качестве видимых в конфигурационном файле extend.json. В противном случае отображается предупреждение «Отсутствуют дополнительные поля».
- 3) Нажать «Сохранить».

Настройка поиска влияет только на поиск в деталях справочника. В общем списке справочников поиск осуществляется по всем настраиваемым полям. Сообщение «Ошибка при изменении области видимости» возникает при настройке поиска поля, для которого не выполнена настройка поиска в конфигурационном файле extend.json.

Добавление настраиваемых полей в справочник выполняется после изменения видимости и поиска. Для этого необходимо в деталях справочника нажать кнопку «Добавить запись». Для редактирования настраиваемых полей необходимо нажать кнопку «Редактировать запись».

2.6. Настройка рабочих потоков

Рабочий поток (workflow) – это последовательность действий (бизнес-процессов, подпроцессов) прохождения электронной заявки.

Комплекс позволяет работать с рабочими потоками с помощью модуля «Activiti», с поддержкой стандарта Business Process Model and Notation (BPMN) 2.0. Более подробная информация о модуле «Activiti» изложена на сайте [«http://www.activiti.org/»](http://www.activiti.org/).

В случае нестандартных цепочек согласований с множеством шагов и сложной логикой может потребоваться настройка пула потоков асинхронных операций для

Изм.	Подп.	Дата

увеличения производительности Комплекса. Настройка пула потоков для асинхронных операций описана в пункте 2.13.4.

В настоящем подразделе описаны действия, выполняемые администратором при подключении, отключении, создании и настройке рабочих потоков, а также дополнительные настройки для электронной заявки.

Комплекс поддерживает запуск рабочего потока для следующих действий:

- 1) Сертификация пользователей подразделения.
- 2) Сертификация ролей ИС.
- 3) Создание роли.
- 4) Удаление роли.
- 5) Изменение роли.
- 6) Отзыв роли у пользователя.
- 7) Сброс пароля пользователю администратором или оператором.
- 8) Активация пользователя.
- 9) Удаление пользователя.
- 10) Назначение роли пользователю.
- 11) Изменение пользователем собственных атрибутов.
- 12) Активация УЗ пользователя.
- 13) Создание пользователя.
- 14) Редактирование пользователя администратором или оператором.
- 15) Блокировка пользователя.
- 16) Блокировка УЗ пользователя.
- 17) Сброс пароля УЗ пользователя.
- 18) Удаление УЗ пользователя.
- 19) Создание подразделения.
- 20) Удаление подразделения.
- 21) Редактирование подразделения.
- 22) Изменение срока действия роли.
- 23) Создание записи делегирования.
- 24) Импорт ролей.
- 25) Импорт пользователей.

Изм.	Подп.	Дата

При создании заявки происходит поиск подходящего бизнес-процесса согласования в следующей последовательности:

- 1) По заданным сущности, действию и ИС ищется настройка в объекте «managed/workflowsetting».
- 2) Если настройка найдена:
 - в случае заполненного поля workflowKey, используется указанный процесс согласования;
 - в случае незаполненного поля workflowKey, действие над объектом выполняется сразу, без запуска процесса согласования.
- 3) Если настройка не найдена, по заданным сущности и действию ищется настройка в файле конфигурации request.json:
 - в случае наличия настройки, используется указанный процесс согласования;
 - в случае отсутствия настройки, действие над объектом выполняется сразу, без запуска процесса согласования.

Комплекс поддерживает следующие основные возможности рабочих потоков «Activiti»:

- 1) Последовательное и параллельное согласования заявки.
- 2) Отображение и валидация полей формы заявки, настроенной на разных этапах согласования.
- 3) События, выполняемые по таймеру (например, Timer Boundary Event).
- 4) Уведомления на электронную почту участников согласования.

Комплекс также поддерживает возможность вложения файлов в электронную заявку.

Для просмотра списка бизнес-процессов Комплекса администратору следует перейти в меню «Админ. системы»/«Бизнес-процессы».

Для создания бизнес-процесса в меню «Админ. системы»/«Бизнес-процессы» администратору следует выполнить следующие действия:

- 1) Нажать кнопку «Создать». Откроется визуальный конструктор, который в центре окна отображает бизнес-процесс в виде графической схемы.

Изм.	Подп.	Дата

- 2) Заполнить уникальное значение параметра «Идентификатор» на вкладке «Основные». По умолчанию параметр «Идентификатор» принимает значение «Process_1» и не может быть отредактирован после сохранения бизнес-процесса. Если введено не уникальное значение, при сохранении отображается сообщение «Бизнес-процесс с таким идентификатором уже существует».
- 3) Выполнить действия по созданию схемы бизнес-процесса.
- 4) Нажать кнопку «Сохранить».

Для удаления бизнес-процесса в меню «Админ. системы»/«Бизнес-процессы» администратору необходимо выполнить следующие действия:

- 1) Выбрать нужный бизнес-процесс из списка.
- 2) Нажать на кнопку «Удалить». Откроется окно подтверждения «Удаление бизнес-процесса».
- 3) Нажать на кнопку «Удалить» в окне подтверждения. В случае успешного удаления отобразится сообщение «Бизнес-процесс успешно удален». В случае ошибки отображается сообщение «Ошибка при удалении бизнес-процесса».

Типовые ошибки при попытке удаления бизнес-процесса:

- бизнес-процесс связан по крайней мере с одной ИС;
- в Комплексе присутствуют незавершенные заявки (в процессе согласования), связанные с удаляемым бизнес-процессом.

Для редактирования бизнес-процесса в меню «Админ. системы»/«Бизнес-процессы» администратору следует выбрать бизнес-процесс из списка и нажать кнопку «Редактировать». Откроется визуальный конструктор.

Графическая схема бизнес-процесса ориентирована в направлении слева направо.

Панель с инструментами и элементами графической схемы расположена в левой части экрана. Информация о каждом элементе бизнес-процесса отображается в правой части окна конструктора. Содержимое этой области меняется в зависимости от того, какой элемент был выбран.

Изм.	Подп.	Дата

Графические элементы определяют ход бизнес-процесса и делятся на следующие категории:

- события;
- действия;
- шлюзы.

Чтобы установить новый элемент бизнес-процесса, следует выбрать элемент на панели и нажать на пустую область конструктора. После чего элемент будет установлен. Элемент можно передвигать по области конструктора.

Некоторые элементы имеют несколько типов. Для того чтобы выбрать нужный тип элемента, необходимо выделить элемент на схеме и нажать на инструмент «Изменить тип». Отобразится выпадающее меню для выбора типа.

Основные элементы конструктора приведены в таблице 2.16.

Таблица 2.16 – Основные элементы конструктора

Наименование элемента	Описание
Создать начальное событие	Событие, обозначающее начало процесса
Создать промежуточное/граничное событие	Промежуточное событие. Промежуточные события могут использоваться в качестве граничных событий для задач. В этом случае они могут быть прерывающими или непрерывными
Создать конечное событие	Событие, обозначающее конец процесса
Шлюз	Шлюз, который определяет вариант потока операций (эксклюзивный, параллельный, на основе событий и другие)
Создать задачу	Задача, обозначающая простое действие. Внутри элемента в виде блока помещается наименование процесса
Создать развернутый подпроцесс	Несколько задач, выделенные в отдельную подзадачу или подпроцесс
Создать объект данных	Объект данных, представляющий информацию о том,

Изм.	Подп.	Дата

Наименование элемента	Описание
	какие действия необходимо выполнить или результат этих действий
Создать хранилище данных	Хранилище данных
Создать пул/участника	Пул, отображающий исполнителей (задач, организаций, пользователей). Используется для разграничения ответственности
Создать группу	Группа – элемент для группировки графических элементов
Добавить TextAnnotation	Примечание. Используется для добавления пояснения к элементу
Удалить	Используется для удаления элемента

Основные инструменты конструктора приведены в таблице 2.17.

Таблица 2.17 – Основные инструменты конструктора

Наименование инструмента	Описание
Использовать инструмент «Рука»	Инструмент, позволяющий захватить элемент схемы (всю схему) и передвигать по полю конструктора
Использовать инструмент «Лассо»	Инструмент, позволяющий выделить элемент схемы (всю схему)
Использовать инструмент «Сократить/Растянуть»	Инструмент, позволяющий «раздвинуть» или «сжать» схему
Использовать инструмент «Потоки и ассоциации»	Инструмент, представляющий собой соединяющие элементы

Комплекс поддерживает создание формы заявки заявителя и согласующего с помощью конструктора.

Изм.	Подп.	Дата

Для создания формы заявки заявителя необходимо выполнить следующие действия:

- 1) В конструкторе выбрать элемент «Создать начальное событие».
- 2) Перейти на вкладку «Формы» в правой части окна конструктора. Вкладка «Формы» содержит следующие поля:
 - ключ формы (необязательное поле);
 - поля формы – для создания полей формы заявки.
- 3) Нажать кнопку «+» для создания нового поля формы заявки. Отобразятся следующие поля:
 - «Идентификатор» – идентификатор поля;
 - «Тип» – тип поля. Выбор доступен из выпадающего списка: «string», «long», «boolean», «date», «enum», «custom type» (только значение «text»);
 - «Ярлык» – отображаемое имя поля в форме заявки;
 - «Значение «по умолчанию»» – значение, которое автоматически будет отображаться в поле формы заявки;
 - «Свойства» – свойство поля. Для добавления свойства нажать кнопку «+» и заполнить поля «Идентификатор» и «Значение». У поля с типом «enum» идентификатор может начинаться только с латинских букв или символа нижнего подчеркивания, может содержать цифры.

Список свойств поля формы заявки приведен в таблице 2.18.

Таблица 2.18 – Список свойств поля формы заявки

Идентификатор	Описание	Значение
required	Признак обязательности поля	true/false
writable	Признак имеется ли возможность заполнить	true/false
readable	Признак отображается ли поле в форме	true/false
hidden	Позволяет скрыть поле в форме, при этом отправлять его в запросе. По умолчанию принимает значение «false». При установленном значении «true» в поле «hidden», поле «readable» обязательно должно быть в	true/false

Изм.	Подп.	Дата

Идентификатор	Описание	Значение
	значении «true»	
datePattern	Позволяет задать «маску» для поля с типом «date»	например, dd-MM- уууу

Для создания формы заявки согласующего необходимо выполнить следующие действия:

- 1) В конструкторе выбрать элемент «Создать задачу».
- 2) Нажать «Изменить тип» и выбрать «Задача, выполняемая пользователем».
- 3) Выполнить аналогичные действия, приведенные для формы заявки заявителя в пунктах 2-3.

2.6.1. Настройка модуля «Activiti»

Модуль «Activiti» настраивается в файле, расположенном в каталоге ankey/conf/workflow.json. Если этот файл отсутствует в конфигурации, модуль рабочего потока недоступен для использования.

По умолчанию модуль рабочего потока находится во включенном состоянии «enabled: true». Администратор может отключить модуль, указав в параметре значение «false».

Описание файла приведено в разделе 35 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

2.6.2. Настройка рабочего потока для заявки на назначение роли определенной информационной системы

Настройка рабочего потока для заявки на назначение роли определенной ИС может выполняться администратором Комплекса следующими средствами:

- в меню «Инф. системы» для связывания рабочего потока с ИС, описание приведено в подразделе 2.7 документа «Руководство пользователя» (72410666.00054-02 96 01);
- REST-запросом для объекта «managed/workflowsetting».

Изм.	Подп.	Дата

Параметры REST-запроса для объекта «managed/workflowsetting» приведены в таблице 2.19.

Таблица 2.19 – Параметры REST-запроса для объекта «managed/workflowsetting»

Атрибут	Описание	Обязательный	Поддерживаемые значения	Пример
Сущность (entity)	Объект системы	Да	Объект системы «managed/usrapproval»	managed/usrapproval
Действие (action)	Действие над объектом	Да	create, batchCreate, delete, batchDelete, patch, enable	batchCreate
Информационная система (is_id)	Идентификатор ИС	Нет	Идентификатор существующей ИС	2
Процесс согласования (workflowKey)	Идентификатор процесса согласования. В случае отсутствия значения, действие над объектом выполняется без согласования	Нет	Идентификатор существующего в системе процесса согласования	autoApproval

Пример содержимого REST-запроса для объекта «managed/workflowsetting»:

Method: POST
 URL: ankey/managed/workflowsetting?_action=create
 Body:
 {
 "action": "batchCreate",

Изм.	Подп.	Дата

```
"entity": "managed/usrapprole",  
"is_id": 2,  
"workflowKey": "autoApproval"  
}
```

2.6.3. Создание рабочего потока

Для создания рабочего потока используется специальный плагин для интегрированной среды разработки Eclipse, который может быть использован для построения графической модели, тестирования и развертывания процессов BPMN 2.0. Инструкции по установке Eclipse Designer приведены в документации на сайте [«http://www.activiti.org/»](http://www.activiti.org/).

Рабочий поток можно создать с помощью диаграмм, добавляя необходимые элементы. Нарисованная диаграмма преобразуется в XML-файл формата `.bpmn20.xml`. Рабочий поток может содержать в себе форму процесса (например, форма заявки), которая содержит в себе перечень атрибутов процесса, их типов и значений. В форме процесса могут присутствовать любые из следующих типов атрибутов:

- строка;
- текст;
- целочисленный тип;
- логический тип;
- дата;
- перечисляемый тип.

Правила работы с формой процесса приведены в документации на сайте [«http://www.activiti.org/»](http://www.activiti.org/), в разделе «Forms».

Примеры готовых рабочих потоков расположены в каталоге `ankey/samples/workflow`.

2.6.4. Настройка рабочего потока для действий с объектами Комплекса

После создания рабочего потока xml-файл рабочего потока следует переименовать с расширением `.bar` и поместить в каталог `ankey/workflow`.

В файле `request.json`, расположенном в каталоге `ankey/conf`, указаны действия, производимые с идентификатором рабочего потока для конкретной сущности.

Изм.	Подп.	Дата

В нижеприведенном примере для сущности «managed/usrapprole» (назначение ролей пользователю) указаны действия создания («create»), массового создания («batchCreate») с использованием рабочего потока «managerApproval».

Пример содержимого файла request.json с демонстрационными рабочими потоками:

```
{
  "entities": {
    "managed/usrapprole": [
      {
        "action": "create",
        "workflowKey": "managerApproval"
      },
      {
        "action": "batchCreate",
        "workflowKey": "managerApproval"
      }
    ],
    "managed/user": [
      {
        "action": "patch",
        "workflowKey": "seflEdit"
      }
    ],
    "resform/*": [
      {
        "action": "enable",
        "workflowKey": "autoApproval"
      }
    ]
  },
  "resources": {
    "managed/user": {
      "title": "fullName",
      "fields": [
        "mail",
        "managerFullName",
        "organizationTree"
      ]
    },
    "managed/approle": {
      "title": "appRoleName",
      "fields": [
        "appRoleName",
        "appRoleDesc"
      ]
    },
    "repo/internal/user": {
      "title": "_id",
      "fields": [
        "_id"
      ]
    }
  }
}
```

Изм.	Подп.	Дата

```

}
}

```

В разделе *entities* для пар (сущность, действие) задается процесс согласования по умолчанию (используется в случае, если нет настройки в Комплексе).

В разделе *resource* определяются заголовок и отображаемые атрибуты объектов в пользовательском интерфейсе. Для добавления/удаления атрибута необходимо отредактировать строку «fields».

При переключении рабочего потока на другой, необходимо указать в файле название нового рабочего потока в строках «workflowKey».

Рабочие потоки, которые поставляются в составе Комплекса в каталоге ankey/workflow, приведены в таблице 2.20.

Таблица 2.20 – Рабочие потоки, которые поставляются в составе Комплекса в каталоге ankey/workflow

Идентификатор	Описание
autoApproval	Рабочий поток с автоматическим согласованием заявки. Форма заявки содержит только поле «Обоснование»
autoApprovalWithForm	Демонстрационный рабочий поток с автоматическим согласованием заявки
certificateAppRoles	Рабочий поток для согласования заявок на сертификацию ролей ИС
certificateUserAppRoles	Рабочий поток для согласования заявок на сертификацию ролей пользователей
dependentEnumApproval	Аналог managerApproval. Демонстрирует возможности фильтрации значений зависимого enum-поля. В состав входит настраиваемое enum-поле в форме заявки, зависимое от другого поля
incidentResolution	Рабочий поток для обработки инцидентов нарушения правил разграничения доступа
managerApproval	Демонстрационный рабочий поток с согласованием заявки в два этапа:

Изм.	Подп.	Дата

Идентификатор	Описание
	1) Согласование руководителем бенефициара. 2) Согласование участниками роли с названием «Администратор ИБ»
multiplyApproval	Демонстрационный рабочий поток реализует последовательное согласование из двух этапов: 1) Этап «Прямой руководитель или его заместитель». На данном этапе два согласующих: руководитель сотрудника и его действующий заместитель. Если заместитель отсутствует, согласующий этапа только руководитель. Если руководитель отсутствует, заявку согласовывает пользователь «ankey». 2) Этап «Согласование администраторами». На этапе две группы согласующих: «Администратор 1» и «Администратор 2». В случае, если одна из групп отсутствует в Комплексе, согласуют только участники существующей группы. Если отсутствуют обе группы, заявку согласовывает пользователь «ankey». Для успешного согласования заявки требуется согласование обоих этапов. Заявка будет отклонена, если ее отклонят на одном из этапов
newUserManagerApproval	Аналог managerApproval. Используется для создания пользователя по заявке
parallelApproval	Демонстрационный рабочий поток реализует параллельное согласование из двух этапов: 1) Этап «1-й этап»: согласуют сотрудники роли «Role 1» или пользователь «ankey», в случае отсутствия роли в Комплексе. 2) Этап «2-й этап»: согласуют сотрудники роли «Role 2» или пользователь «ankey», в случае отсутствия роли в

Изм.	Подп.	Дата

Идентификатор	Описание
	<p>Комплексе.</p> <p>Для успешного согласования заявки требуется согласование обоих этапов. Заявка будет отклонена, если ее отклонят на одном из этапов. При отклонении на любом этапе оставшийся этап, если он еще не был обработан, будет «удален» из списка заданий заявки, ожидающих обработки</p>
threeStageApproval	<p>Демонстрационный рабочий поток реализует согласование заявки в три этапа:</p> <ol style="list-style-type: none"> 1) Согласование руководителем бенефициара. 2) Согласование участниками роли с названием «Администратор ИБ». 3) Согласование участниками роли с названием «Ankey Admins».

2.6.5. Удаление Бизнес Пакета (.bar)

Для полного и корректного удаления Бизнес Пакета (БП), следует выполнить следующие шаги:

- шаг 1. Проверка связей БП с ИС;
- шаг 2. Удаление БП на файловом уровне;
- шаг 3. Удаление БП через REST API;
- шаг 4. Проверить связь БП в меню «Админ. системы»/«Настройка согласований».

2.6.5.1. Проверка связей БП с информационными системами

Удалить БП можно лишь при полном отсутствии связей БП с ИС.

В противном случае при отправке REST-запроса будет выброшена ошибка о неудачной попытке удаления. Ошибка содержит в себе информацию о количестве ИС, которые на момент удаления БП имеют с ним связь.

Изм.	Подп.	Дата

Для успешного удаления БП, данные ИС необходимо предварительно отредактировать через интерфейс ПК Ankey IDM, очистив ссылку на БП. После этого можно приступать к удалению (см. следующие шаги).

2.6.5.2. Удаление БП на файловом уровне

Удалить БП на файловом уровне можно как на активном, так и на не запущенном ПК Ankey IDM.

Для этого надо:

- 1) Перейти в директорию workflow: ankey/workflow.
- 2) Удалить необходимый БП (.bar файл).

2.6.5.3. Удаление БП через REST API

Удалить БП через REST API можно исключительно на уже запущенном ПК Ankey IDM.

Для этого необходимо выполнить следующие действия:

- 1) Отправить REST-запрос на получение списка идентификаторов всех Process Definition:
GET: `http://<hostname>:<port>/ankey/workflow/processdefinition?_queryId=query-all-ids.`
- 2) Выбрать подходящий идентификатор Process Denition (атрибут «_id»).
- 3) Удалить необходимый Process Denition:
DELETE: `http://<hostname>:<port>/ankey/workflow/processdefinition/<id>.`

Примечания:

- 1) Каждый Process Definition имеет ссылку на «deploymentId». Это идентификатор развернутого БП.
- 2) Один БП может содержать в себе сразу несколько Process Definition.
- 3) Удаляя любой из Process Definition из пакета, уничтожается весь пакет целиком, в том числе все связанные с ним процессы Process Definition.

2.6.6. Настройка вложений в заявку

Настройка вложений в заявку выполняется в конфигурационном файле `conf/document.json`.

Изм.	Подп.	Дата

Описание настройки загружаемых файлов через интерфейс Комплекса приведено в подразделе 4.5.

2.6.7. Настройка отзыва заявки заявителем

Для корректной работы отзыва заявки в рабочем потоке потребуется добавить новый поток (sequenceFlow) от промежуточного шлюза «gateway» до завершающего шлюза «gateway» и дать название данному потоку, например, «отменено».

Пример графической настройки демонстрационного рабочего потока «managerApproval» приведен на рисунке 2.1.

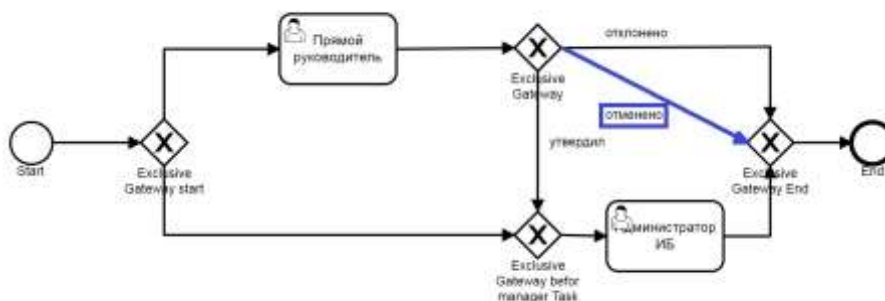


Рисунок 2.1 – Пример графической настройки демонстрационного рабочего потока «managerApproval»

Дополнительно необходимо задать условие срабатывания данного потока, которое зависит от переменной «ankey_outcome», и должно срабатывать если переменная принимает значение «cancelled». Пример настройки условия срабатывания для демонстрационного рабочего потока «managerApproval»:

```
<sequenceFlow id="sf_cancel_before_eg_end" name="отменено" sourceRef="eg_after_manager"
targetRef="eg_end">
  <conditionExpression xsi:type="tFormalExpression"><![CDATA[/${ankey_outcome ==
'cancelled'}]]></conditionExpression>
</sequenceFlow>
```

2.6.8. Настройка отображения фотографии в деталях заявки

Для отображения фотографии пользователя в деталях заявки необходимо в каталоге ankey/conf в файле request.json добавить в блок «resources» поле «photoPreview».

Изм.	Подп.	Дата

Пример содержимого файла request.json с настройкой для отображения фотографии пользователя:

```
"resources": {  
  "managed/user": {  
    "title": "fullName",  
    "fields": ["photoPreview"]  
    ...  
  }  
  ...  
}
```

2.7. Управление настраиваемыми полями

Комплекс предоставляет возможность администратору добавлять настраиваемые поля для любого управляемого объекта.

Для добавления настраиваемого поля администратору необходимо выполнить следующие действия:

- 1) Добавить настраиваемое поле в конфигурационный файл extend.json согласно пункту 2.7.1.
- 2) Создать атрибуты в таблицы БД с помощью скриптов Liquidbase. Подробнее приведено в пункте 2.7.2.
- 3) Настроить локализацию настраиваемого поля. Подробнее приведено в пункте 2.7.3.

2.7.1. Управление настраиваемым полем

Управление настраиваемым полем выполняется в конфигурационном файле extend.json в каталоге ankey/conf/ и включает в себя следующие действия администратора:

- 1) Добавление/редактирование нового поля.
- 2) Добавление политики валидации для поля. Описание приведено в пункте 2.7.1.1.
- 3) Определение порядка отображения настраиваемых полей и полей, настроенных по умолчанию. Описание приведено в пункте 2.7.1.2.

Описание конфигурационного файла extend.json приведено в разделе 18 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

Новое настраиваемое поле следует добавить в область управляемого объекта и выполнить:

- 1) Указать в параметре «геро» настройки хранения поля в БД. Настройки параметра «геро» приведены в подразделе 18.1 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 2) Указать параметры для отображения поля в интерфейсе в параметре «managed». Настройки параметра «managed» приведены в подразделе 18.3 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

После добавления/редактирования настраиваемых полей управляемого объекта «managed/approle» необходимо запустить периодическое задание «reindexTask», выполнив полный или частичный запуск для объекта «informationSystemRequestAppRoles».

Для добавления настраиваемых полей в форму «Документы» при создании/редактировании подразделения необходимо в качестве управляемого объекта указывать объект «Документы подразделения» – repo/orgdocusage.

Поддерживаемые типы настраиваемых полей приведены в пункте 2.7.5.

Описание настройки для отображения столбцов, полей фильтрации и поиска в интерфейсе Комплекса приведено в подразделе 2.25.

Примечания:

- 1) При настройке параметра «viewRequestCard»:
 - в «managed» → «display» требуется дополнительно прописать поля в request.json: «resources» → «managed/user»;
 - для полей типа «lookup» требуется дополнительно прописать поле в request.json, обозначенное как «displayField» в lookup-поле.
- 2) Для расширения видимости существующих полей Комплекса достаточно прописать видимость поля только в «managed» → «display» (исключая «геро» и «policy»).
- 3) При настройке параметра «audit» настраиваемое поле автоматически добавляется в audit.json в массив «watchedFields», что позволяет

Изм.	Подп.	Дата

отображать изменения настраиваемого поля. Подробнее приведено в подразделе 5.8.

2.7.1.1. Политика валидации настраиваемого поля

Для настраиваемого поля в параметре «policy» → «policies» можно задать политики валидации.

Можно использовать как поставляемые с Комплексом политики, так и политики собственной разработки. Список политик валидации Комплекса приведен в пункте 18.2.1 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Для настройки нужно указать имя политики в параметре «policyId» и при необходимости задать ей конфигурацию в параметре «params».

Все политики, кроме required, разрешают отсутствие значения у проверяемого поля (null).

2.7.1.2. Определение порядка отображения настраиваемых полей и полей, настроенных по умолчанию

При добавлении настраиваемых полей можно менять порядок полей, определенных по умолчанию. Порядок отображения полей зависит от порядка следования перечисленных полей.

Пример настройки отображения полей в объекте «repo/orgdocusage»:

```
{
  "repo/orgdocusage": {
    "documentType": {
      "repo": {
        "column": "documenttype",
        "type": "STRING",
        "init": {
          "file": "conf/liquibase.json",
          "type": "liquibase/json"
        }
      }
    },
    "policy": {
      "schema": {
        "type": "string",
        "required": true,
        "example": "enum123"
      }
    },
    "policies": [
      {
```

Изм.	Подп.	Дата

```

    "policyId": "maxLength",
    "params": {
      "numChars": 512
    }
  },
  {
    "policyId": "required"
  }
]
},
"managed": {
  "required": true,
  "searchable": true,
  "fieldType": "enum",
  "values": [
    "Соглашение о конфиденциальности",
    "Договор о сотрудничестве"
  ],
  "display": [
    "create",
    "view",
    "update"
  ]
}
},
"documents": {
  "managed": {
    "fieldType": "document",
    "display": [
      "create",
      "update",
      "view"
    ],
    "required": true
  }
}
},
"_id": "extend"
}

```

2.7.2. Создание атрибутов в таблице БД с помощью скриптов

Liquidbase

Для добавления атрибутов в таблицы БД администратору необходимо создать файл, на базе которого будет создана колонка для хранения значений настраиваемого поля.

Файл со скриптом Liquidbase создается со следующими данными:

- стандартный блок XML Schema;
- «tableName» – название таблицы, в которой создается колонка;
- «schemaName» – по умолчанию принимает значение «ankey»;

Изм.	Подп.	Дата

- «column name» – название колонки, которое указывается в параметре «repo» в файле extend.json;
- «type» – тип, принятый для БД.

Администратору следует указать название таблицы, название колонки и тип при создании файла.

Пример скрипта, в котором создается колонка «orgunit» в таблице «organization»:

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:ext="http://www.liquibase.org/xml/ns/dbchangelog-ext"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.1.xsd
  http://www.liquibase.org/xml/ns/dbchangelog-ext http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-
ext.xsd">
  <changeSet id="100" author="gis">
    <addColumn tableName="organization" schemaName="ankey">
      <column name="orgunit" type="character varying(128)"/>
    </addColumn>
  </changeSet>
</databaseChangeLog>
```

2.7.3. Настройка локализации настраиваемых полей

При создании настраиваемого поля требуется задать для него локализацию. Локализация создается для отображения элементов интерфейса.

Комплекс позволяет настроить русскоязычную и англоязычную локализацию.

Администратору необходимо выполнить следующие шаги:

Шаг 1. Создать директорию localization.

Шаг 2. Создать директорию localization/i18n.

Шаг 3. В директории localization/i18n создать файлы для отображения названия создаваемого поля:

- файл translation_en.properties – для англоязычной локализации;
- файл translation_ru.properties – для русскоязычной локализации.

В файлах задаются следующие значения:

- ключ templates.managed.form – для окон редактирования и создания;
- ключ templates.managed.card – для страниц просмотра.

Пример содержания файла для англоязычной локализации поля «customAttr»:

Изм.	Подп.	Дата

templates.managed.form.user.customAttr=Custom User Attribut
 templates.managed.card.user.customAttr=Custom User Attribute

Для русскоязычной локализации требуется выполнить конвертацию текста для <Название атрибута в интерфейсе> в кодировку JS/JAVA.

Пример содержания файла русскоязычной локализации для поля «customAttr»:

```
templates.managed.form.user.customAttr=\\u041A\\u0430\\u0441\\u0442\\u043E\\u043C\\u043D\\u044B\\u0439
\\u043F\\u043E\\u043B\\u044C\\u0437\\u043E\\u0432\\u0442\\u0435\\u043B\\u044C\\u0441\\u043A\\u04
38\\u0439 \\u0430\\u0442\\u0440\\u0440\\u0438\\u0431\\u0443\\u0442
```

```
templates.managed.card.user.customAttr=\\u041A\\u0430\\u0441\\u0442\\u043E\\u043C\\u043D\\u044B\\u0439
\\u043F\\u043E\\u043B\\u044C\\u0437\\u043E\\u0432\\u0442\\u0435\\u043B\\u044C\\u0441\\u043A\\u04
38\\u0439 \\u0430\\u0442\\u0440\\u0440\\u0438\\u0431\\u0443\\u0442
```

В значениях локализации недопустимо использование круглых скобок.

Шаг 4. Создать Bundle локализации.

Bundle локализации представляет собой архив Java ARchive (JAR). Внутри архива JAR содержатся следующие данные:

- каталог i18n с файлами translation_en.properties, translation_ru.properties;
- каталог META-INF с файлом MANIFEST.MF;
- файл manifest-addition.mf.

Создание Bundle локализации выполняется через системы сборки (например, maven, gradle, ant и другие) или утилиту jar.

При создании Bundle локализации необходимо находиться в папке localization.

Для создания Bundle администратору необходимо выполнить следующие шаги:

Шаг 4.1. Создать файл manifest-addition.mf с дополнительными настройками для манифест-файла следующего содержания:

```
Bundle-Name: Localization bundle
Bundle-SymbolicName: localization
Bundle-Version: 0.0.1
I18N: i18n.translation
```

Пример создания файла с дополнительными параметрами к манифест-файлу:

```
cd ../../
echo 'Bundle-Name: Localization bundle
Bundle-SymbolicName: localization
Bundle-Version: 0.0.1
I18N: i18n.translation' > manifest-addition.mf
```

Изм.	Подп.	Дата

Шаг 4.2. Создать jar-файл `localization.jar` с дополнительными настройками для манифест-файла, выполнив команду `jar cfm localization.jar manifest-addition-path localization`, где «`manifest-addition-path`» – путь к файлу с дополнительными параметрами к манифесту, а `localization` – название директории, откуда будет собираться Bundle.

Для выполнения команды в среде Windows следует использовать абсолютный путь к файлу, выполнив команду: `jar cfm localization.jar manifest-addition.mf localization`.

После выполнения команды получен Bundle локализации: `localization.jar`.

Каталог META-INF создается автоматически и содержит в себе манифест MANIFEST.MF, созданный на основе `manifest-addition.mf`. Перед загрузкой Bundle в Ankey следует убедиться, что манифест MANIFEST.MF содержит в себе строку «`I18N: i18n.translation`».

Шаг 5. Загрузить Bundle локализации в Комплекс.

Для загрузки файла `localization.jar` необходимо сохранить его в директорию `ankey/extensions` и перезагрузить Комплекс. Подробная информация о перезапуске Комплекса приведена в документе «Руководство по инсталляции» (72410666.00054-02 94 01).

2.7.4. Настройка локализации существующего поля

Для настройки локализации существующего поля необходимо выполнить шаги, описанные в пункте 2.7.3.

Настройка локализации вкладки «Информация» возможна для следующих объектов в соответствующем конфигурационном файле:

- 1) «Входящие»/«Заявки», «Входящие»/«История», «Мои заявки», «Журнал заявок» – `templates.request.detail.tabs.info`.
- 2) «Подразделения» – `templates.modules.organizations.detail.tabs.info`.
- 3) «Инциденты» – `templates.modules.incident.detail.tabs.info`.
- 4) «Инф.системы» – `templates.modules.is.detail.tabs.info`.
- 5) «Админ.системы»:
 - «Ресурсы» – `templates.modules.administration.resource.tab.detail.info`;

Изм.	Подп.	Дата

- «Правила» – templates.modules.administration.rules.detail.tabs.info;
- «Парольные политики» – templates.modules.administration.passwordPolicies.tab.detail.info.

В файлах локализации следует указать конфигурационный файл объекта и его новое значение, как показано на примерах ниже.

Пример англоязычной локализации для вкладки «Информация» в файле translation_en.properties:

```
templates.request.detail.tabs.info=About
templates.modules.organizations.detail.tabs.info=Info
templates.modules.incident.detail.tabs.info=My info
templates.modules.is.detail.tabs.info=Custom info
templates.modules.administration.resource.tab.detail.info=My custom info
templates.modules.administration.passwordPolicies.tab.detail.info=Info about object
templates.modules.administration.rules.detail.tabs.info=New name info
```

Пример русской локализации для вкладки «Информация» в файле translation_ru.properties:

```
templates.request.detail.tabs.info=\u041e\u0431\u044a\u0435\u043a\u0442\u0435
templates.modules.organizations.detail.tabs.info=\u0418\u043d\u0444\u043e\u043c\u0430\u0446\u0438\u044f
templates.modules.incident.detail.tabs.info=\u041c\u043e\u0438 \u0438\u043d\u0444\u043e\u0440\u043c\u0430\u0446\u0438\u044f
templates.modules.is.detail.tabs.info=\u041a\u0443\u0441\u043e\u043c\u043e\u0432\u0438\u0435 \u0438\u043d\u0444\u043e\u0440\u043c\u0430\u0446\u0438\u044f
templates.modules.administration.resource.tab.detail.info=\u041c\u043e\u0438 \u043a\u0443\u0441\u043e\u043c\u043e\u0432\u0438\u0435 \u0438\u043d\u0444\u043e\u0440\u043c\u0430\u0446\u0438\u044f
templates.modules.administration.passwordPolicies.tab.detail.info=\u0418\u043d\u0444\u043e\u0440\u043c\u0430\u0446\u0438\u044f \u043e\u0431\u044a\u0435\u043a\u0442\u0430
templates.modules.administration.rules.detail.tabs.info=\u041d\u043e\u0432\u043e\u0435 \u043d\u0430\u0437\u0432\u0430\u043d\u0438\u0435
```

2.7.5. Поддерживаемые типы настраиваемых полей

Тип настраиваемого поля указывается в настройках параметра «геро» конфигурационного файла extend.json.

Для добавления доступны поля следующих типов:

- string;
- integer;
- object;
- boolean.

Изм.	Подп.	Дата

Поддерживаемые типы настраиваемых полей представлены в таблице 2.21.

Таблица 2.21 – Поддерживаемые типы настраиваемых полей

Тип поля управляемого объекта (managed/fieldType)	Тип поля (policy/schema/type)	Тип в БД (repo/type)	Описание
textField	string	STRING	Строковое поле
textArea	string	STRING	Многострочное поле
flag	boolean	BOOLEAN	Boolean поле
number	integer	INTEGER	Числовое поле, поддерживаются значения от -2147483648 до 2147483647
passwordField	string	STRING	Поле для хранения пароля
tree	integer	BIGINT	Иерархия
date	string	DATETIME	Поле для хранения даты
enum	string	STRING	Перечисляемый тип
enumForm	string	STRING	Перечисляемый тип, который позволяет отображать или скрывать настраиваемые поля
lookup	string, integer	STRING, BIGINT	Поле с возможностью поиска объекта
multilookup	array	JSONB	Поле поиска объекта с множественным

Изм.	Подп.	Дата

Тип поля управляемого объекта (managed/fieldType)	Тип поля (policy/schema/type)	Тип в БД (repo/type)	Описание
			выбором
document	object	JSONB	Поле с возможностью загрузки, скачивания, удаления файла
multiGroup	array	JSONB	Поле, состоящее из групп lookup-полей с множественным выбором этих групп

2.7.5.1. Настраиваемое поле типа «string»

Настраиваемое поле типа «string» – строковое поле для хранения текстовой информации.

Для добавления нового поля «customAttr» типа «string» в карточку пользователя администратору необходимо выполнить следующие шаги:

Шаг 1. Изменить конфигурационный файл extend.json в каталоге ankey/conf/, добавив в параметры «repo», «policy», «managed» следующие строки:

```
{
  "managed/user": {
    "customAttr" : {
      "repo": {
        "column": "customattr",
        "type": "STRING",
        "init" : {
          "file" : "db/postgresql/liquibase/custom.xml",
          "type" : "liquibase/xml"
        }
      }
    },
    "policy" : {
      "schema" : {
        "type" : "string",
        "example" : "My super custom attr!",
        "required" : true
      }
    },
    "policies": [
      {
```

Изм.	Подп.	Дата

```

    "policyId": "required"
  },
  {
    "policyId": "maxLength",
    "params": {
      "numChars": 512
    }
  }
]
},
"managed" : {
  "required" : true,
  "fieldType" : "textField",
  "display" : ["create","update","view","selfEdit"]
}
}
}
}
}

```

Шаг 2. В каталоге `ankey/db/postgresql/liquibase` создать скрипт `custom.xml` и добавить в него следующие строки, оставив стандартный блок со схемами:

```

<?xml version="1.0" encoding="UTF-8"?>

<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.1.xsd">
  <changeSet id="100" author="gis">

    <addColumn tableName="usr" schemaName="ankey">
      <column name="customattr" type="character varying(128)"/>
    </addColumn>
  </changeSet>
</databaseChangeLog>

```

Шаг 3. Создать Bundle локализации. В файлы локализаций добавить следующие строки:

– русская локализация (`translation_ru.properties`):

```

templates.managed.form.user.category=\u0414\u043e\u0432\u044b\u0439\u0020\u0430\u0442\u0442\u0440\u0438\u0431\u0443\u0442

```

```

templates.managed.card.user.category=\u0414\u043e\u0432\u044b\u0439\u0020\u0430\u0442\u0442\u0440\u0438\u0431\u0443\u0442

```

– английская локализация (`translation_en.properties`):

```

templates.managed.form.user.category=customAttr
templates.managed.card.user.category=customAttr

```

Изм.	Подп.	Дата

2.7.5.2. Настраиваемое поле типа «enum»

Настраиваемое поле типа «enum» – это перечисляемый тип данных, в котором множество значений представляет собой ограниченный список идентификаторов. Ниже приведены примеры конфигурационного файла extend.json для заведения полей с типом «enum».

Пример настраиваемого поля для объекта «Пользователь»:

```
{
  "managed/user": {
    "address": {
      "repo": {
        "column": "address",
        "type": "STRING",
        "init": {
          "file": "liquibase/custom_usr.xml",
          "type": "liquibase/xml"
        }
      }
    },
    "policy": {
      "schema": {
        "type": "string",
        "example": "ул. Васи Алексева д.3"
      }
    },
    "policies": [{
      "policyId": "maximumLength",
      "params": {
        "numChars": 64
      }
    }
  ]
},
  "managed": {
    "required": false,
    "fieldType": "enum",
    "values": [
      "ул. Кронштадтская, д.10, литера А",
      "ул. Намёткина, д. 10А"
    ],
    "display": [
      "create",
      "update",
      "view",
      "viewUserCard"
    ]
  }
}
```

Пример расширения значений существующего поля типа «enum»:

Изм.	Подп.	Дата

```
{
  "managed/resaccount": {
    "beneficiarObject": {
      "managed": {
        "values": [
          "managed/approperesaccount"
        ]
      }
    }
  }
}
```

В файл translation_ru.properties для локализации нового значения требуется добавить перевод, например, «Привилегии роли»:

```
templates.managed/approperesaccount.title=\u041f\u0440\u0438\u0432\u0438\u043b\u0435\u0433\u0438\u0438\u0440\u043e\u043b\u0438
```

Пример расширения значений настраиваемых полей типа «enum» для дальнейшей настройки фильтрации:

```
"managed/approle": {
  "landscape": {
    "managed": {
      "fieldType": "enum",
      "values": [
        "Продуктив",
        "Разработка",
        "Тестирование"
      ],
      "display": ["create",
        "update",
        "view"
      ],
      "search": [
        ],
      "searchable": false
    },
    "policy": {
      "schema": {
        "example": "",
        "required": false,
        "type": "string"
      }
    },
    "repo": {
      "column": "landscape",
      "type": "STRING",
      "init": {
        "file": "conf/liquibase.json",
        "type": "liquibase/json"
      }
    }
  },
}
```

Изм.	Подп.	Дата

```

"risk": {
  "managed": {
    "fieldType": "enum",
    "values": [
      "Высокий",
      "Средний",
      "Низкий"
    ],
    "display": ["create",
      "update",
      "view"
    ],
    "search": [
    ],
    "searchable": false
  },
  "policy": {
    "schema": {
      "example": "",
      "required": false,
      "type": "string"
    }
  },
  "repo": {
    "column": "risk",
    "type": "STRING",
    "init": {
      "file": "conf/liquibase.json",
      "type": "liquibase/json"
    }
  }
}
}
}

```

Необходимо указать, что настраиваемое поле типа «enum» является поисковым: "searchable":true. Для настройки фильтрации и поиска необходимо определить поля через endpoint/ui.form. Описание настройки приведено в подразделе 2.25.

Пример конфигурационного файла extend.json для ИС:

```

{
  "managed/is": {
    "isfield": {
      "managed": {
        "searchable": true,
        "required": false,
        "display": [
          "create",
          "update",
          "view",
          "selfEdit",
          "viewMyProfile"
        ],

```

Изм.	Подп.	Дата

```

"fieldType": "enum",
"values": [
"ул. Кронштадтская, д.10, литера А",
"ул. Намёткина, д. 11"
]
},
"policy": {
"schema": {
"type": "string"
}
},
"hero": {
"column": "isfield",
"type": "STRING"
}
}
}
}

```

2.7.5.3. Настраиваемое поле типа «lookup»

Настраиваемое поле типа «lookup» – это текстовое поле с возможностью поиска и выбора из predetermined значений.

Для добавления нового поля «org_kurator» типа «lookup» в карточку «Подразделения» администратору необходимо выполнить следующие шаги:

Шаг 1. Изменить конфигурационный файл extend.json в каталоге ankey/conf/, добавив в параметры «hero», «policy», «managed» следующие строки:

```

{
"managed/organization": {
"orgKurator": {
"managed": {
"fieldType": "lookup",
"display": [
"create",
"update",
"view"
],
"displayField": "orgKuratorName",
"object": "managed/user",
"searchStartLength": 3,
"resultsDisplayAmount": 5,
"lookupDisplayFields": [
"user Name"
],
"lookupDisplayFields2": [
"personnelNumber"
],
"lookupField": "_ouid",
"queryId": null,
"filter": "position eq 'Начальник'"
}
},
}
}

```

Изм.	Подп.	Дата

```

"policy": {
  "schema": {
    "example": "MANAGER",
    "required": false,
    "type": "integer"
  }
},
"repo": {
  "column": "org_kurator",
  "init": {
    "file": "liquibase/custom_organization.xml",
    "type": "liquibase/xml"
  },
  "type": "BIGINT"
}
}
}
}

```

При редактировании конфигурационного файла следует учесть следующие моменты:

- 1) Тип атрибута `type` должен обязательно совпадать с типом атрибута «lookupField». В данном примере `lookupField=oid`, `oid` имеет тип `bigint`, поэтому `"type": "BIGINT"`.
- 2) Фильтр должен быть настроен так, чтобы для каждого «lookupField» он был уникальным. Например, имеется справочник `referencebook` для разных ресурсов (`resource_id=1`, `resource_id=2`) с одинаковыми значениями `refBookType="Сотрудник"`. По фильтру `"filter": "refBookType eq 'Сотрудник' and resource_id eq 1"` возвращается одно значение для «lookupField» – из первого ресурса и это будет правильной настройкой. Настройка `"filter": "refBookType eq 'Сотрудник'"` не будет являться корректной, так как вернет сотрудников из разных ресурсов.
- 3) Поля типа «lookup» не должны возвращать удаленные объекты. Для этого в качестве объекта атрибута («object») предпочтительнее использовать значение `"object": "endpoint/lookup/{name}"` (Например, `"endpoint/lookup/approle"`). В случае указания объекта атрибута («object») в виде «`managed/{name}`» следует пользоваться параметром «filter». Для запроса пользователей используется настройка `"filter": "!(/accountStatus eq 'deleted')"`, для остальных сущностей используется `"filter": "!(/status eq 'deleted')"`.

Изм.	Подп.	Дата

- 4) В качестве объекта атрибута («object») для корректного отображения пользователям с ролями «Оператор», «Администратор ролей ИС» необходимо указывать следующие значения:
- "object": "endpoint/lookup/user" – для поиска по доступным для просмотра пользователям;
 - "object": "endpoint/lookup/organization" – для поиска по доступным для просмотра подразделениям;
 - "object": "endpoint/lookup/approle" – для поиска по доступным для просмотра ролям.
- 5) Следует указать уникальные между собой имя настраиваемого поля и его «отображаемое» имя («displayName»). В вышеупомянутом примере эти имена, как раз, отличаются: «orgKurator» и «orgKuratorName». Несоблюдение этого требования может повлечь за собой поломку отображения заявок на редактирование данного настраиваемого поля.

Для lookup/multilookup-полей в Комплексе предусмотрены следующие возможности:

- 1) Отображение дополнительной информации по выбранному объекту, которая доступна только для объекта «Роли». Дополнительная информация будет выводиться под строкой поиска и отображаться в деталях роли. Для этого необходимо задать атрибут «infoFields» с типом массив.
- 2) Настройка динамического фильтра по маске в любой форме. Описание приведено в пункте 2.7.6.

Пример конфигурационного файла extend.json с атрибутом «infoFields» в параметре «managed» для lookup/multilookup-полей:

```
"customOrgLookupE": {
  "fieldType": "lookup",
  "required": false,
  "custom": true,
  "object": "endpoint/lookup/organization",
  "displayField": "customOrgLookupEDisplay",
  "searchStartLength": 3,
  "resultsDisplayAmount": 5,
  "lookupDisplayFields": [
```

Изм.	Подп.	Дата

```

    "name"
  ],
  "lookupDisplayFields2": [
    "code"
  ],
  "lookupField": "_oid",
  "filter": "true",
  "infoFields": [
    "code",
    "name"
  ]
}
"customUserMultilookupE": {
  "fieldType": "multilookup",
  "required": false,
  "custom": true,
  "object": "endpoint/lookup/user",
  "displayField": "customUserMultilookupEDisplay",
  "lookupDisplayFields": [
    "fullName"
  ],
  "lookupDisplayFields2": [
    "userName"
  ],
  "lookupField": "_oid",
  "filter": "true",
  "infoFields": [
    "fullName",
    "position",
    "organizationName"
  ]
}
},

```

Шаг 2. Добавить колонку в БД, в которой будет храниться `oid` выбранного объекта в `lookup`-поле. Для этого в каталоге `ankey/db/postgresql/liquibase` создать скрипт `custom_organization.xml` и добавить в него следующие строки, оставив стандартный блок со схемами:

```

<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
  xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.1.xsd">

  <changeSet id="100" author="gis">
    <addColumn tableName="organization" schemaName="ankey">
      <column name="org_kurator" type="bigint"/>
    </addColumn>
  </changeSet>
</databaseChangeLog>

```

Изм.	Подп.	Дата

Шаг 3. Создать объект «handler» для заполнения виртуального поля. Поле «orgKuratorName», которое указано в «displayField», не существует и необходимо создать виртуальное поле для объекта. Пример объекта «handler» на основе идентификатора из поля «ogrKurator», который предзаполняет поле «orgKuratorName» для корректного отображения в интерфейсе Комплекса:

```

package com.gis.idm.integration.handlers;
import com.gis.idm.api.config.ConfigurationHelper;
import com.gis.idm.api.managed.HandlerResult;
import com.gis.idm.api.managed.ManagedObjectHandler;
import com.gis.idm.api.model.Organization;
import com.gis.idm.api.service.data.UserService;
import org.apache.felix.scr.annotations.*;
import org.forgerock.json.JsonValue;
import org.forgerock.json.resource.Request;
import org.forgerock.json.resource.ResourceException;
import org.forgerock.openidm.util.ContextUtil;
import org.forgerock.services.context.Context;
import org.forgerock.util.promise.Promise;

import java.util.Map;

import static org.osgi.framework.Constants.SERVICE_DESCRIPTION;

/**
 * Handler для показа lookup UDF полей
 * Заполнение виртуальных полей, для отображения в UI
 */
@Service(ManagedObjectHandler.class)
@Component(
    immediate = true,
    name = OrganizationHandler.PID
)
@Properties({
    @Property(name = SERVICE_DESCRIPTION, value = "GiS.IDM :: OrganizationIntegrationHandler
Service"),
    @Property(name = ManagedObjectHandler.PROPERTY_RESOURCE, value = Organization.MANAGED),
    @Property(name = ManagedObjectHandler.PROPERTY_ORDER, value =
ManagedObjectHandler.DEFAULT_USER_LEVEL)
})
public class OrganizationHandler implements ManagedObjectHandler {
    static final String PID = ConfigurationHelper.DEFAULT_SERVICE_RDN_PREFIX +
"OrganizationIntegrationHandler";
    private static final String SUB_ORG_ID = "orgKurator";
    private static final String VIRTUAL_ORG_KURATOR = "orgKuratorName";

    @Reference
    private UserService userService;

/**
 * При получении объекта из БД заполняем необходимые виртуальные поля
 *
 * @param context контекст сервиса

```

Изм.	Подп.	Дата

```

* @param request объект Request
* @param object полученный объект, в данном случае организация
* @param args
* @return
*/
@Override
public Promise<HandlerResult, ResourceException> onRetrieve(Context context, Request request, JsonValue
object, Map<String, Object> args) {
    return addSubOrgName(context, object).then(ignore -> HandlerResult.OK);
}

private Promise<JsonValue, ResourceException> addSubOrgName(Context context, JsonValue object) {
    return executeIf(ContextUtil.isExternal(context), () -> {
        Long orgKuratorOuid = object.get(SUB_ORG_ID).asLong();
        return executeIf(orgKuratorOuid != null, () -> userService.findByOuid(context, orgKuratorOuid)
            .then(user -> user != null
                ? object.put(VIRTUAL_ORG_KURATOR, user.getUserName())
                : object));
    });
}
}
}

```

Необходимо проверять, является ли контекст внешним, так как данные виртуальные поля используются только для отображения. В противном случае данные поля будут заполняться при всех запросах, что влечет повышение нагрузки на Комплекс.

Пример настройки поля «targetInfoSystem» типа «lookup» в шаблоне периодического задания «approlecertificationtask».

```

"org.forgerock.openidm.approlecertificationtask": {
  "id": "http://json-schema.org/draft-04/schema#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description": "AppRole Certification",
  "type": "object",
  "properties": {
    "targetInfoSystem": {
      "type": "string",
      "example": "Сетевое рабочее место",
      "description": "Наименование ИС, роли которой подлежат сертификации",
      "fieldType": "lookup",
      "displayField": "name",
      "object": "endpoint/lookup/is",
      "managedObject": "managed/is",
      "searchStartLength": 3,
      "resultsDisplayAmount": 5,
      "placeholder": "templates.managed.form.approle.lookup.placeholder.targetInfoSystem",
      "lookupDisplayFields": [
        "name"
      ],
      "lookupDisplayFields2": [
        "description"
      ]
    }
  }
}

```

Изм.	Подп.	Дата

```

    "lookupField": "name",
    "queryId": null,
    "filter": "true"
  }
},
"required": [
  "targetInfoSystem"
],
"additionalProperties": false
}

```

2.7.5.4. Настраиваемое поле типа «multilookup»

Настраиваемое поле типа «multilookup» – это текстовое поле с возможностью поиска и множественного выбора из predetermined значений. Поле «multilookup» принимает значение массива – «array», а колонка в БД имеет тип «jsonb».

Шаги по управлению настраиваемым полем «multilookup» выполняются аналогичным образом, как и для простого поля «lookup». Описание настройки приведено в подпункте 2.7.5.3.

Настройка «multilookup» поддерживается для следующих объектов Комплекса:

- «substitution» – замещение;
- «requestForm» – форма заявки;
- «organization» – подразделение;
- «user» – пользователь;
- «usrOrg» – трудоустройства.

В деталях настроенных объектов и в истории изменений пользователя Комплекс отображает множество значений «multilookup» поля.

Пример файла extend.json в каталоге ankey/conf для поля «Поиск объекта» с множественным выбором:

```

{
  "managed/substitution": {
    "informationSystemIds": {
      "managed": {
        "fieldType": "multilookup",
        "displayField": "informationSystemDisplay",
        "object": "managed/is",
        "placeholder": "templates.substitution.informationSystemIds",
        "lookupField": "_ouid",
        "filter": "true",
        "lookupDisplayFields": [
          "name"
        ]
      },

```

Изм.	Подп.	Дата

```

"lookupDisplayFields2": [
  "description"
],
"display": [
  "create",
  "view",
  "requestView"
],
"search": [],
"searchable": false
},
"policy": {
  "schema": {
    "example": [ 1, 2, 3 ],
    "required": false,
    "type": "array"
  },
  "policies": [
    {
      "policyId": "resourceExists",
      "params": {
        "resource": "managed/is"
      }
    }
  ]
},
"repo": {
  "column": "informationsystemids",
  "type": "JSONB",
  "init": {
    "file": "conf/custom_substitution.xml",
    "type": "liquibase/xml"
  }
}
}
}
}

```

В конфигурационном файле extend.json можно не указывать следующие строки для политики:

```

"policies":[
  {
    "policyId":"array"
  }
]

```

Политика применяется автоматически и запрещает:

- значения, которые не являются массивом;
- массив, содержащий «null»;
- массив, содержащий дубликаты.

Изм.	Подп.	Дата

Пример скрипта `custom_substitution.xml` в каталоге `ankey/db/postgresql/liquibase` для создания колонки в БД с типом «`jsonb`»:

```
<?xml version="1.0" encoding="UTF-8"?>
<databaseChangeLog
xmlns="http://www.liquibase.org/xml/ns/dbchangelog"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog
http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-3.1.xsd">

  <changeSet id="1" author="gis">
    <addColumn schemaName="ankey" tableName="substitution">
      <column name="informationsystemids" type="jsonb"></column>
    </addColumn>
  </changeSet>
</databaseChangeLog>
```

Для настройки обязательности заполнения поля «`multilookup`» необходимо в конфигурационном файле `extend.json` указать:

- в блоке «`managed`»: `"required": true;`
- в блоке «`policy`»/«`schema`»: `"required": true;`
- политику в блоке «`policy`»/«`policies`»: `"policyId": "required".`

Пример настройки обязательности заполнения поля «`multilookup`» для объекта «`referencebook`» в файле `extend.json` в каталоге `ankey/conf`:

```
{
  "managed/organization": {
    "ext_org": {
      "managed": {
        "fieldType": "multilookup",
        "displayField": "ext_org_display",
        "required": true,
        "object": "managed/referencebook",
        "placeholder": "templates.organization.ext_org",
        "lookupField": "_oid",
        "filter": "true",
        "lookupDisplayFields": [
          "refBookName"
        ],
        "lookupDisplayFields2": [
          "refBookCode"
        ],
        "display": [
          "audit",
          "create",
          "update",
          "view"
        ],
        "search": [],
        "searchable": false
      }
    }
  }
}
```

Изм.	Подп.	Дата

```

    },
    "policy": {
      "schema": {
        "example": "",
        "required": true,
        "type": "array"
      },
      "policies": [
        {
          "policyId": "array"
        },
        {
          "policyId": "required"
        }
      ]
    },
    "repo": {
      "column": "ext_org",
      "type": "JSONB",
      "init": {
        "file": "conf/liquibase.json",
        "type": "liquibase/json"
      }
    }
  }
}

```

2.7.5.5. Настраиваемое поле типа «multiGroup»

Настраиваемое поле типа «multiGroup» – поле, состоящее из групп lookup-полей с множественным выбором этих групп.

Настройка «multiGroup» поддерживается для следующих объектов Комплекса:

- «approle» – роль;
- «information_system» – ИС.

Для добавления нового поля типа «multiGroup» администратору следует выполнить действия, приведенные в разделе 2.7.

В конфигурационном файле extend.json указать, что атрибут «fieldType» принимает значение «multiGroup». Название поля типа «multiGroup» должно отличаться от названия, указанного в параметре «displayField». В противном случае возникнет ошибка при сохранении конфигурационного файла extend.json.

В атрибуте «rootGroupField» определяется имя lookup-поля, которое будет главным (корневым) при отображении в разделе информации об объекте, в деталях заявки, в аудите. Поле является обязательным для заполнения.

Изм.	Подп.	Дата

В атрибуте «groupFields» определяются два и более lookup-поля, по которым будет происходить группировка. Эти поля не имеют связи между собой. Результатом является поле, состоящее из lookup-полей, с возможностью добавления нескольких групп одновременно.

Порядок отображения lookup-полей задается в параметре «groupFieldsOrder». Указываются lookup-поля, которые перечисляются в атрибуте «groupFields». Например, "groupFieldsOrder": ["organization", "user"]. Поле является обязательным для заполнения.

В Комплексе выполняется валидация для предотвращения полных дубликатов.

Настройки lookup-полей выполняются аналогично настройке поля типа «multiGroup». Описание настройки приведено в подпункте 2.7.5.3.

Пример файла extend.json в каталоге ankey/conf для поля «multiGroup», где в атрибуте «groupFields» указаны lookup-поля на объекты «organization» и «user»:

```
{
  "managed/is": {
    "multiGroupField": {
      "managed": {
        "fieldType": "multiGroup",
        "rootGroupField": "organization",
        "groupFieldsOrder": ["organization", "user"],
        "groupFields": {
          "organization": {
            "object": "managed/organization",
            "lookupField": "_ouid",
            "fieldType": "lookup",
            "displayField": "orgDisplayField",
            "filter": "true",
            "lookupDisplayFields": [
              "name"
            ],
            "lookupDisplayFields2": [
              "code"
            ],
            "infoFields": [
              "code",
              "name"
            ],
            "search": [],
            "searchable": false,
            "searchStartLength": 3,
            "resultsDisplayAmount": 5,
            "placeholder": "templates.managed.form.user.lookup.placeholder.organization",
            "required": false
          },
          "user": {
```

Изм.	Подп.	Дата

```

"object": "managed/user",
"lookupField": "_oid",
"fieldType": "lookup",
"displayField": "userDisplayField",
"filter": "true",
"lookupDisplayFields": [
  "lastName"
],
"lookupDisplayFields2": [
  "firstName"
],
"infoFields": [
  "userName",
  "mail",
  "position"
],
"search": [],
"searchable": false,
"searchStartLength": 3,
"resultsDisplayAmount": 5,
"placeholder": "templates.managed.form.user.lookup.placeholder.user",
"required": false
}
},
"displayField": "multiGroupField",
"display": [
  "create",
  "update",
  "view"
]
},
"policy": {
"schema": {
"example": [
  {
    "organization": 1,
    "user": 2
  },
  {
    "organization": 3,
    "user": 4
  }
]
},
"required": false,
"type": "array"
}
},
"repo": {
"column": "multigroupfield",
"type": "JSONB",
"init": {
"file": "conf/liquibase.json",
"type": "liquibase/json"
}
}
}
}

```

Изм.	Подп.	Дата

```
}
}
```

2.7.5.6. Настройка отображения изменений пользователя в деталях массовых операций

Для отображения изменений у пользователя в деталях массовых операций используется схема `viewBulkOperation`, имеющая следующий набор полей:

```
"viewBulkOperation": [
  "userName",
  "accountStatus",
  "enableDate",
  "disableDate",
  "lastName",
  "firstName",
  "middleName",
  "mail",
  "phone",
  "organization_id",
  "position",
  "personnelNumber",
  "usrOrgTabNum",
  "managerObjectId",
  "locked",
  "avatarId"
]
```

При добавлении поля в `extend.json` необходимо прописать указанную схему в поле `display` раздела `managed` для того чтобы настраиваемое поле отображалось в деталях массовых операций.

Например:

```
...
"managed" : {
  "fieldType" : "textField",
  "display" : ["view", "viewBulkOperation"]
}
```

Более подробное описание приведено в пункте 2.7.1.

2.7.5.7. Отображение настраиваемого поля в справочнике

Возможные типы настраиваемых полей для отображения в справочниках: «`textField`», «`number`», «`flag`», «`date`», «`timestamp`», «`enum`», «`lookup`», «`document`».

Настройка отображения поля в справочнике выполняется в конфигурационном файле `extend.json` для управляемого объекта «`managed/referencebook`». Общее описание управления настраиваемыми полями приведено в подразделе 2.7.

Изм.	Подп.	Дата

Для изменения отображения настраиваемого поля администратору необходимо выполнить:

- 1) Изменение отображения настраиваемых полей в интерфейсе Комплекса.
- 2) Изменение поиска по настраиваемому полю.

Для отображения настраиваемых полей в интерфейсе Комплекса необходимо в конфигурационном файле `extend.json` в параметре «`managed`» выполнить:

- 1) Добавить новый атрибут «`title`». В атрибуте «`title`» следует указать шаблон или текст, который будет отображаться в заголовке настраиваемого поля.
- 2) Указать в параметре «`managed`»/«`display`»:
 - «`update`»;
 - «`view`».

Пример конфигурационного файла `extend.json` с атрибутом «`title`»:

```
"managed": {
  "required": false,
  "title": "текстовое поле",
  "searchable": false,
  "fieldType": "textField",
  "display": [
    "update",
    "view"
  ]
}
```

В параметре «`managed`» значение атрибута «`displayField`» должно отличаться от названия настраиваемого поля.

Для управления поиском по настраиваемому полю параметр «`searchable`» должен иметь значение «`true`».

После выполнения настроек необходимо запустить периодическое задание «`reindextask`».

Пример настройки отображения поля «`customInformationSystemLookupM`» типа «`lookup`» в справочнике:

```
{
  "customInformationSystemLookupM":{
    "repo":{
      "column":"custominformationsystemlookupm",
      "type":"BIGINT",
      "init":{
        "file":"conf/liquibase.json",
```

Изм.	Подп.	Дата

```

    "type": "liquibase/json"
  }
},
"policy": {
  "schema": {
    "type": "integer",
    "required": false
  }
},
"managed": {
  "searchable": true,
  "displayField": "customInformationSystemLookupMDf",
  "required": false,
  "fieldType": "lookup",
  "display": [
    "update",
    "view"
  ],
  "object": "managed/is",
  "searchStartLength": 3,
  "resultsDisplayAmount": 5,
  "lookupDisplayFields": [
    "name"
  ],
  "lookupDisplayFields2": [
    "description"
  ],
  "lookupField": "_ouid",
  "filter": "true"
}
}
}

```

Пример отображения настраиваемого поля типа «document» в справочнике:

```

{"customDocumentT": {
  "repo": {
    "column": "customdocument",
    "type": "JSONB",
    "init": {
      "file": "conf/liquibase.json",
      "type": "liquibase/json"
    }
  },
  "policy": {
    "schema": {
      "type": "object",
      "required": false
    }
  },
  "managed": {
    "required": false,
    "searchable": true,
    "fieldType": "document",
    "display": [
      "update",

```

Изм.	Подп.	Дата

```

    "view"
  ]
}
}
}

```

2.7.5.8. Настройка видимости перечисляемых полей

Настройка видимости перечисляемых полей позволяет выбирать, какие поля отображать или скрывать в формах создания и редактирования ИС и подразделений. Для этого используется тип «enumForm», который определяет перечисляемые поля.

Каждое перечисляемое поле может быть связано с настраиваемыми полями с помощью атрибута «valuesFields». Если атрибут не указан или его значение пусто, поле будет автоматически скрыто.

Видимость связанных настраиваемых полей в формах создания и редактирования определена в блоке «display» для перечисляемого поля с типом «enumForm»:

```

"display": [
  "audit",
  "create",
  "update",
  "view"
]

```

Для таких полей необходимо указывать параметры для их отображения, исключая «create» и «update». Иначе настраиваемые поля будут дублироваться в формах создания и редактирования.

Ниже приведен пример настройки видимости перечисляемого поля с типом «enumForm» в конфигурационном файле extend.json, где поле «customUsrEnumForm» связано с настраиваемыми полями «customUsrText» и «customUsrNumber». Видимость для поля «customUsrEnumForm» настроена в блоке «display», для каждого настраиваемого поля блок «display» не включает в себя параметры «create» и «update».

```

{
  "managed/usrorg": {
    "customUsrEnumForm": {
      "repo": {
        "column": "customusrorgenumform",
        "type": "STRING",
        "init": {
          "file": "conf/liquibase.json",
          "type": "liquibase/json"
        }
      }
    }
  }
}

```

Изм.	Подп.	Дата

```

}
},
"policy": {
  "schema": {
    "type": "string",
    "required": false,
    "example": "enum123"
  },
  "policies": [
    {
      "policyId": "maximumLength",
      "params": {
        "numChars": 512
      }
    }
  ]
},
"managed": {
  "required": false,
  "fieldType": "enumForm",
  "values": [
    "enum1",
    "enum2",
    "enum3"
  ],
  "valuesFields": {
    "enum1": [
      "customUsrText"
    ],
    "enum2": [
      "customUsrNumber"
    ],
    "enum3": []
  },
  "display": [
    "audit",
    "create",
    "update",
    "view"
  ]
}
},
"customUsrText": {
  "repo": {
    "column": "customusrorgtext",
    "type": "STRING",
    "init": {
      "file": "conf/liquibase.json",
      "type": "liquibase/json"
    }
  },
  "policy": {
    "schema": {
      "type": "string",
      "example": "My super custom attr!",
      "required": false
    }
  },

```

Изм.	Подп.	Дата

```

"policies": [
  {
    "policyId": "maxLength",
    "params": {
      "numChars": 64
    }
  }
],
"managed": {
  "required": false,
  "fieldType": "textField",
  "display": [
    "audit",
    "view"
  ]
},
"customUsrNumber": {
  "repo": {
    "column": "customusrorgnumber",
    "type": "INTEGER",
    "init": {
      "file": "conf/liquibase.json",
      "type": "liquibase/json"
    }
  },
  "policy": {
    "schema": {
      "type": "integer",
      "required": false
    }
  },
  "managed": {
    "required": false,
    "fieldType": "number",
    "display": [
      "audit",
      "view"
    ]
  }
}
}
}
}
}

```

2.7.6. Настройка динамического фильтра по маске в форме

Настройка динамического фильтра по маске выполняется в конфигурационном файле `conf/extend.json` для `lookup/multilookup`-полей.

Маска поддерживается только для параметра «filter».

Администратору необходимо прописать в настройках поля маску для параметра «filter».

Изм.	Подп.	Дата

Параметры, используемые в масках, могут быть динамическими и статическими:

- 1) Динамические параметры – поля, которые существуют и могут быть изменены в форме.
- 2) Статические параметры – поля, которых не существует в форме, значение предопределено и передается в форму заранее. Варианты статических параметров:
 - константа. Например, (/documentType eq 'Соглашение о конфиденциальности').
 - параметр, связанный с объектом Комплекса через его идентификатор.

Если статические параметры применяются при создании нового объекта, доступ к его атрибутам становится невозможным. Возможность обращения к атрибутам проявляется лишь при редактировании данного объекта.

Структура фильтра может быть как динамической, так и статической, но она может выглядеть одинаково. Например, «(/organizationId eq \${organizationId})». От того, какие параметры используются в маске – динамические или статические, зависит тип фильтра.

Пример динамического фильтра по маске в форме заявки Комплекса приведен в подпункте 2.6.5.25.

Поддержка маски в управляемых объектах Комплекса приведена в таблице 2.22.

Таблица 2.22 – Поддержка маски в управляемых объектах Комплекса

Управляемый объект Комплекса	Динамические параметры	Статические параметры: константа	Статические параметры: идентификатор изменяемого объекта (id, _oid)
managed/approle	поддерживается	поддерживается	не поддерживается
managed/user	поддерживается	поддерживается	не

Изм.	Подп.	Дата

Управляемый объект Комплекса	Динамические параметры	Статические параметры: константа	Статические параметры: идентификатор изменяемого объекта (id, _ouid)
			поддерживается
managed/rule	поддерживается	поддерживается	не поддерживается
managed/resaccount	поддерживается	поддерживается	не поддерживается
managed/resource	поддерживается	поддерживается	не поддерживается
managed/referencebook (добавление значений справочника)	не поддерживается	поддерживается	не поддерживается
managed/referencebook (изменение)	поддерживается	поддерживается	не поддерживается
managed/information_system	поддерживается	поддерживается	не поддерживается
managed/organization	поддерживается	поддерживается	поддерживается
managed/userOrg	поддерживается	поддерживается	поддерживается
managed/substitution	поддерживается	поддерживается	не поддерживается
managed/requestForm	поддерживается	поддерживается	поддерживается на отзыв роли

Изм.	Подп.	Дата

Комплекс позволяет в зависимости от выбранного значения в другом поле автоматически передавать:

- 1) Статические параметры в форму заявки на отзыв ролей. Например, "filter": "`(/beneficiaryId eq '{_id}') and (/beneficiaryOuid eq '{_ouid}') and (/appRoleOuids in '{appRoleOuids}')`".
- 2) Атрибуты пользователя в качестве статических параметров при добавлении/редактировании трудоустройства в меню «Пользователи». Например, "filter": "`(/organizationId eq '{organizationId}') and (/udfEmployeeType eq '{udfEmployeeType}')`".

Маски с поддержкой статических параметров доступны в форме заявки на отзыв роли и при выборе добавлении/редактировании трудоустройства.

Если в маске указано поле типа `boolean`, то при значении «false» зависимое поле будет заблокировано, а при значении «true» – разблокировано.

2.8. Настройка безопасности

Настройка безопасности выполняется в конфигурационных файлах. Комплекс позволяет выполнить прямое редактирование конфигурационных файлов через интерфейс. Описание настроек через интерфейс Комплекса приведено в подразделе 4.1. Описание конфигурационных файлов приведено в документе «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Администратор может выполнять следующие настройки безопасности:

- 1) Настройка безопасного подключения к веб-консолям. Описание приведено в пункте 2.8.1.
- 2) Настройка схем аутентификации. Действия приведены в пункте 2.8.2.
- 3) Настройка политики блокировки УЗ. Описание приведено в подразделе 4.1.
- 4) Настройка повторного использования идентификатора удаленного пользователя. Описание приведено в подразделе 4.1.
- 5) Настройка белого списка для проверки клиентского сертификата. Описание приведено в подразделе 4.1.
- 6) Настройка идентификации пользователя через кодовое слово. Описание приведено в пункте 2.8.4.

Изм.	Подп.	Дата

- 7) Настройки генераторов значений. Описание приведено в подразделе 4.3.
- 8) Настройки логирования и массовых операций. Описание приведено в подразделе 4.8.

В Комплексе существуют административные роли, которые являются внутренним механизмом. Список административных ролей приведен в пункте 2.8.3.

2.8.1. Настройка безопасного подключения к веб-консолям

Для настройки безопасного подключения к веб-консолям администратору необходимо выполнить следующие действия:

- 1) В конфигурационном файле `jetty.xml` закомментировать или удалить блок кода `<<Call name = "addConnector";>`, который включает свойство `openidm.port.http`.
- 2) Оставить блоки кода `<<Call name = "addConnector">`, которые содержат свойства `openidm.port.https` и `openidm.port.mutualauth`. Значения для этих свойств задаются в файле `conf/boot/boot.properties`.
- 3) В конфигурационном файле `config.properties` установить для свойства `<<org.osgi.service.http.enabled>` значение `<false>`, как показано в следующем примере:

```
\# Enable pax web http/https services to enable jetty
```

```
org.osgi.service.http.enabled=false
```

```
org.osgi.service.http.secure.enabled=true
```

2.8.2. Настройка схем аутентификации

Для настройки схем аутентификации администратору необходимо в конфигурационном файле `authentication.json` выполнить следующие действия:

- 1) Добавить схемы аутентификации в секцию `<authModules>` файла настроек, если они отсутствуют.
- 2) Установить порядок применения схем аутентификации, указав их последовательно в секции `<authModules>`.
- 3) Активировать схему аутентификации, установив параметр `<enabled>` в значение `<true>`.

Изм.	Подп.	Дата

Настройки файла authentication.json приведены в разделе 2 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Схемы аутентификации, поддерживаемые Комплексом и их настройки, приведены в подразделе 2.1 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

2.8.3. Административные роли

Административные роли Комплекса предназначены для разграничения доступа к объектам и действиям с объектами. Административные роли назначаются пользователю Комплексом автоматически при наступлении определенных условий.

Присвоенные административные роли перечисляются в атрибуте «roles» объекта «Пользователь» («managed/user»). Административные роли являются внутренним механизмом Комплекса, поэтому не отображаются в веб-интерфейсе Комплекса.

Перечень административных ролей Комплекса приведен в таблице 2.23.

Таблица 2.23 – Перечень административных ролей Комплекса

Идентификатор роли	Способ назначения	Полномочия роли
ankey-admin	Присвоение роли «Ankey Admins»	Полные полномочия на все объекты Комплекса
ankey-authorized	Присваивается всем пользователям в случае успешной аутентификации	Полномочия пользователя с доступом к пунктам меню «Входящие», «Мои заявки», «Мой профиль»
ankey-adminsod	Присваивается пользователям роли, указанной в параметре «Роль ответственного за инциденты» правила разграничения доступа	Доступ к меню «Инциденты», «Входящие»/«Инциденты»

Изм.	Подп.	Дата

Идентификатор роли	Способ назначения	Полномочия роли
ankey-operator	Присваивается пользователям роли, указанной в параметре «Роль оператора подразделения»	<p>Доступ к меню:</p> <ul style="list-style-type: none"> – «Пользователи»; – «Подразделения»; – «Журнал заявок». <p>Права на выполнение действий:</p> <ul style="list-style-type: none"> – все действия с пользователями организации, для которой указан в качестве оператора; – все действия с подразделениями, которые входят в структуру организации, для которой указан в качестве оператора; – переназначение заявок для бенефициаров из организации, для которой указан в качестве оператора
ankey-approlesadmin	<p>Присваивается пользователям при следующих условия:</p> <ul style="list-style-type: none"> – одна из ролей пользователя указана в параметре «Администратор ролей ИС»; – присвоена роль «Ankey Roles Managers» 	Доступ к меню «Роли». Все права на действия с ролями в рамках своей ИС
ankey-assignmentsadmin	Присваивается пользователям роли,	Доступ к меню «Пользователи» и к меню «Инф. системы» и права на

Изм.	Подп.	Дата

Идентификатор роли	Способ назначения	Полномочия роли
	указанной в параметре «Администратор назначений» ИС	просмотр деталей ИС, списка ролей и пользователей, с возможностью отзыва ролей в рамках ИС

2.8.4. Настройка идентификации пользователя через кодовое слово

Комплекс позволяет настроить идентификацию пользователя через кодовое слово. После выполнения настройки будет доступна следующая функциональность:

- 1) Кнопка «Задать код идентификации» в карточке пользователя. Кодовое слово задается и изменяется только пользователем под своей УЗ.
- 2) Кнопка «Идентифицировать» в карточке пользователя для идентификации кодового слова. По умолчанию проверка доступна для администратора через REST API.

Настройка функциональности может выполняться с помощью:

- 1) Создания системных ролей.
- 2) Создания периодического задания для генерации кодового слова.

Создание системных ролей для объекта «Пользователь» выполняется:

- 1) Для пользователя, который задает кодовое слово с правом доступа `user-secretword-create` – создание кодового слова пользователя.
- 2) Для пользователя, который выполняет идентификацию с правами доступа:
 - `user-read` – чтение пользователя;
 - `user-list-read` – чтение списка пользователей;
 - `user-secretword-verify` – идентификация по кодовому слову пользователя.

Описание создания системной роли приведено в пункте 2.10.1 документа «Руководство пользователя» (72410666.00054-02 96 01).

Периодическое задание выполняет генерацию кодового слова для всех пользователей.

Генератор настраивается с помощью конфигурационного файла `secretgenerator.json`, в котором задается регулярное выражение. Описание конфигурационного файла `secretgenerator.json` приведено в разделе 29 документа

Изм.	Подп.	Дата

«Описание конфигурационных файлов» (72410666.00054-02 97 01). Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.3.

Генерация кодового слова не относится к типовым периодическим заданиям Комплекса. Поэтому необходимо создать задание с типом «secretwordgeneratingtask», как приведено в пункте 2.4.1. Запуск периодического задания создаст или заменит текущее кодовое слово пользователя.

События идентификации записываются в журнал аудита audit.json.

Пример запроса через REST API для идентификации кодового слова пользователя:

```
POST /endpoint/user/secretword/identification
request body:
{
"userId": "362d1df8-8187-408e-bcd0-da86297363f9",
"secret": "строка с верифицируемым кодом"
}
```

Комплекс возвращает следующий ответ:

```
{
"identificationStatus": "SUCCESS" // или "FAILURE"
}
```

2.8.5. Настройка скомпрометированных паролей

Настройка скомпрометированных паролей позволяет задать требования для пароля. Настройка применяется в Комплексе только совместно с парольной политикой. Когда парольная политика активна, Комплекс проводит обязательную проверку нового пароля пользователя/УЗ пользователя на соответствие списку скомпрометированных паролей. Если пароль найден в списке, то Комплекс не позволит его установить.

Для настройки скомпрометированных паролей администратору необходимо выполнить следующие действия:

- 1) Настроить список скомпрометированных паролей в конфигурационном файле compromisedpasswords.json. Описание приведено в разделе 32 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

- 2) Настроить парольную политику «Пароль не должен быть из списка скомпрометированных паролей» для пользователя/УЗР. Описание приведено в подразделе 2.11.

2.9. Настройка журналов работы

Журналирование работы Комплекса производится отдельно от аудита. Для настройки системных журналов администратору необходимо отредактировать файл `logging.properties`, расположенный в каталоге `ankey/conf`.

Описание конфигурационного файла `logging.properties` приведено в разделе 20 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

2.10. Настройка аудита событий

Служба аудита Комплекса может записывать всю системную активность одной или нескольких ЦС, включая активности с локальными файлами данных, хранилищем Комплекса. Комплекс записывает данные о следующих событиях аудита:

- 1) События доступа.
- 2) События системной активности.
- 3) События аутентификации.
- 4) События изменений конфигурации.
- 5) События синхронизации.
- 6) События, связанные с заявками:
 - `claim` – взять в работу заявку;
 - `unclaim` – вернуть заявку из работы;
 - `approved` – согласовать заявку;
 - `rejected` – отклонить заявку;
 - `delegate` – вернуть заявителю для уточнения;
 - `resolve` – отправить возвращенную заявителю заявку на согласование;
 - `reassign` – переназначить согласующего заявки. Фиксируется от кого и на кого выполняется переназначение;
 - `cancel` – отменить заявку.

Изм.	Подп.	Дата

2.10.1. Настройка службы аудита

Комплекс предоставляет конфигурацию журналов аудита в файле, расположенном в каталоге `ankey/conf/audit.json`.

Чтобы настроить службу аудита для записи события, администратору необходимо включить его в список событий для обработчика событий аудита, используемого для запросов. Администратору следует выбрать один любой обработчик событий аудита для управления запросами на журналах аудита. По умолчанию обработчиком запросов аудита является хранилище Комплекса.

Для того чтобы указать обработчик событий аудита, который должен использоваться для запросов, администратору необходимо указать параметр «`handlerForQueries`» в файле `audit.json` следующим образом:

```
{
  "auditServiceConfig" : {
    "handlerForQueries" : "repo",
    "availableAuditEventHandlers" : [
      "org.forgerock.audit.events.handlers.csv.CSVAuditEventHandler",
      "org.forgerock.openidm.audit.impl.RepositoryAuditEventHandler",
      "org.forgerock.openidm.audit.impl.RouterAuditEventHandler"
    ]
  }
}
```

Параметр «`availableAuditEventHandlers`» содержит массив обработчиков событий аудита, доступных для Комплекса.

В `audit.json` можно выполнить следующие настройки:

- 1) Настройка обработчика событий аудита в хранилище Комплекса.
- 2) Настройка обработчика событий аудита в CSV-файле.
- 3) Настройка обработчика событий аудита «маршрутизатор».
- 4) Настройка темы событий.
- 5) Настройка обработки исключений.
- 6) Настройка логирования.

Описание настройки файла `audit.json` приведено в разделе 1 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

2.10.2. Описание типов журналов аудита

2.10.2.1. Журнал доступа

В журнал доступа Комплекс записывает сообщения, касающиеся доступа к REST-интерфейсу.

Файл журнала доступа по умолчанию расположен в каталоге ankey/audit/access.csv.

Параметры журнала доступа представлены в таблице 2.24.

Таблица 2.24 – Параметры журнала доступа

Параметр	Описание
roles	Роли Комплекса, связанные с запросом
_id	Идентификатор для объекта сообщения, например «0419d364-1b3d-4e4f-b769-555c3ca098b0»
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита
transactionId	Идентификатор транзакции
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта
server.ip	IP-адрес для сервера Комплекса
server.port	Номер порта, используемый сервером Комплекса
client.ip	IP-адрес клиента
client.port	Номер порта клиента
request.protocol	Протокол для запроса, как правило, REST
request.operation	Типичная REST-операция
request.detail	Типичные детали для запроса действия
http.request.secure	Логическое значение для запроса безопасности
http.request.method	HTTP-метод, запрошенный клиентом
http.request.path	Путь HTTP-запроса
http.request.queryParameters	Параметры, отправленные в HTTP-запросе

Изм.	Подп.	Дата

Параметр	Описание
http.request.headers	HTTP-заголовки для запроса (необязательно)
http.request.cookies	HTTP-куки для запроса (необязательно)
http.response.headers	HTTP-заголовки ответа (необязательно)
response.status	Normally, SUCCESSFUL, FAILED или null
response.statusCode	SUCCESS в response.status приводит к нулевому значению response.statusCode, FAILURE в response.status приводит к ошибке 400 уровня
response.detail	Сообщение, связанное с ответом response.statusCode, таким как Not Found или Internal Server Error
response.elapsedTime	Время для выполнения события доступа
response.elapsedTimeUnits	Единица времени отклика

2.10.2.2. Журнал активности

В журнал активности Комплекс записывает операции на внутренних и внешних объектах, включая события по работе с заявками.

Файл по умолчанию расположен в каталоге ankey/audit/activity.csv.

Параметры журнала активности представлены в таблице 2.25.

Таблица 2.25 – Параметры журнала активности

Параметр	Описание
passwordChange	True/False запись об изменениях пароля
_id	Идентификатор для объекта сообщения, например, «0419d364-1b3d-4e4f-b769-555c3ca098b0»
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита
transactionId	Идентификатор транзакции
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта

Изм.	Подп.	Дата

Параметр	Описание
runAs	Идентификатор пользователя, под УЗ которого выполнялось действие
objectId	Идентификатор объекта, например, /managed/user/0419d364-1b3d-4e4f-b769-555c3ca098b0
operation	Типичная REST-операция
before	JSON-представление объекта перед выполнением действия
after	JSON-представление объекта после выполнения действия
changedFields	Поля, которые были изменены
revision	Число ревизий объекта
status	Результат (например, SUCCESS)
message	Сообщение о действии

2.10.2.3. Журнал синхронизации с ЦС

В журнал синхронизации с ЦС Комплекс записывает результаты выполнения синхронизации.

Файл по умолчанию расположен в каталоге ankey/audit/recon.csv.

Параметры журнала синхронизации с ЦС представлены в таблице 2.26.

Таблица 2.26 – Параметры журнала синхронизации с ЦС

Параметр	Описание
reconId	UUID для операции синхронизации
_id	Идентификатор для объекта сообщения, например, «0419d364-1b3d-4e4f-b769-555c3ca098b0»
transactionId	Идентификатор транзакции
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта
action	Действие синхронизации, отображено как REST-действие

Изм.	Подп.	Дата

Параметр	Описание
exception	Трассировка стека исключения
linkQualifier	Определитель связи, примененный к действию
mapping	Имя маппинга, используемое для операций синхронизации, заданных в файле ankey/conf/recon.json
message	Описание действия синхронизации
messageDetail	Детали запуска синхронизации, представленные в виде REST-запроса
situation	Ситуация, возникающая при синхронизации
sourceObjectId	Идентификатор объекта в исходной системе, например, «managed/user/d03a29e8-c95f-4637-82f6-cc9b94d0dfb7»
status	Статус результата синхронизации, например, «SUCCESS» или «FAILURE»
targetObjectId	Идентификатор объекта в ЦС, например, «system/xmlfile/account/dkruglov»
reconciling	Значение того, что Комплекс синхронизировал: «source» для первого этапа и «target» для второго
ambiguousTargetObjectIds	Когда параметр «situation» имеет значение «AMBIGUOUS» или «UNQUALIFIED», и Комплекс не может отличить более одного объекта-получателя, Комплекс регистрирует идентификаторы таких объектов
reconAction	Действие синхронизации (как правило «recon» или «null»)
entryType	Тип записи журнала синхронизации (как правило «start», «entry» или «summary»)

2.10.2.4. Журнал аутентификации

Комплекс записывает результаты операций аутентификации в журнал аутентификации.

Файл по умолчанию расположен в каталоге ankey/audit/authentication.csv.

Параметры журнала аутентификации представлены в таблице 2.27.

Изм.	Подп.	Дата

Таблица 2.27 – Параметры журнала аутентификации

Параметр	Описание
entries	JSON-представление сессии аутентификации
_id	Идентификатор для объекта сообщения, например, «0419d364-1b3d-4e4f-b769-555c3ca098b0»
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита
transactionId	Идентификатор транзакции
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта
result	Результат транзакции («SUCCESSFUL» или «FAILED»)
principal	Массив УЗ, используемых для аутентификации, например, «ankey»
context	Полная безопасность операции аутентификации, включая аутентификацию ID, целевой конечной точки, ролей, IP-адресов, с которых был сделан запрос аутентификации

2.10.2.5. Журнал конфигурации

В журнал конфигурации Комплекс записывает изменения, выполненные в настройках Комплекса.

Файл по умолчанию расположен в каталоге ankey/audit/config.csv.

Параметры журнала конфигурации представлены в таблице 2.28.

Таблица 2.28 – Параметры журнала конфигурации

Параметр	Описание
revision	Число ревизий объекта
_id	Идентификатор для объекта сообщения, например, «0419d364-1b3d-4e4f-b769-555c3ca098b0»
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита

Изм.	Подп.	Дата

Параметр	Описание
transactionId	Идентификатор транзакции
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта
runAs	Идентификатор пользователя, под УЗ которого выполнялось действие
objectId	Идентификатор объекта, например, «ui»
operation	Типичная REST-операция
before	JSON-представление объекта до выполнения действия
after	JSON-представление объекта после выполнения действия
changedFields	Поля, которые были изменены

2.10.2.6. Журнал инкрементальной синхронизации с ЦС

В журнал инкрементальной синхронизации с ЦС Комплекс записывает результаты выполнения инкрементальной синхронизации.

Файл по умолчанию расположен в каталоге ankey/audit/sync.csv.

Параметры журнала инкрементальной синхронизации с ЦС представлены в таблице 2.29.

Таблица 2.29 – Параметры журнала инкрементальной синхронизации с ЦС

Параметр	Описание
targetObjectId	Идентификатор объекта в ЦС, например, «uid=dkruglov,ou=People,dc=example,dc=com»
_id	Идентификатор для объекта сообщения, например, «0419d364-1b3d-4e4f-b769-555c3ca098b0»
transactionId	Идентификатор транзакции
timestamp	Время, когда Комплекс зарегистрировал сообщение
eventName	Имя события аудита
userId	Идентификатор пользователя
trackingIds	Уникальное значение для отслеживаемого объекта

Изм.	Подп.	Дата

Параметр	Описание
action	Действие синхронизации, отображено как REST-действие
exception	Трассировка стека исключения
linkQualifier	Определитель связи, примененный к действию
mapping	Имя маппинга, используемое для операций синхронизации, заданных в файле ankey/conf/sync.json
message	Описание действия синхронизации
messageDetail	Детали запуска синхронизации, представленные в виде REST-запроса
situation	Ситуация синхронизации
sourceObjectId	Идентификатор объекта в исходной системе, например, «managed/user/d03a29e8-c95f-4637-82f6-cc9b94d0dfb7»
status	Статус результата синхронизации (например, «SUCCESS» или «FAILURE»)

2.10.3. Настройка буферизации записей аудита

Комплекс поддерживает буферизацию, чтобы уменьшить число записей в ПК Ankey IDM. Для настройки буферизации администратору необходимо отредактировать файл «audit.json» и задать следующие параметры:

- «enabled» – включает/выключает буферизацию (True/false);
- «autoFlush» – определяет, будет ли служба аудита автоматически удалять события после их записи на диск (True/false).

Пример настройки буферизации:

```
"eventHandlers" : [
  {
    "config" : {
      ...
      "buffering" : {
        "autoFlush" : false,
        "enabled" : false
      }
    }
  }
]
```

Комплекс записывает данные в журналы аудита асинхронно, в то время как функция «autoFlush» позволяет службе аудита записывать данные в журналы на регулярной основе.

Изм.	Подп.	Дата

2.10.4. Дополнительная настройка обработчика событий аудита

Дополнительная настройка обработчика событий аудита приведена в пункте 1.2.4 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

2.11. Настройка парольной политики

Настройка парольной политики выполняется в меню «Админ. системы»/«Парольные политики».

Администратор выполняет следующие действия:

- создание парольной политики. Описание приведено в пункте 2.11.1;
- редактирование парольной политики. Описание приведено в пункте 2.11.2;
- применение парольной политики. Описание приведено в пункте 2.11.3;
- удаление парольной политики. Описание приведено в пункте 2.11.4.

Парольная политика создается для следующих объектов:

- 1) «Пользователи».
- 2) «Учетная запись ресурсов».

Для объекта «Пользователи» создается одна парольная политика, которая автоматически применяется ко всем пользователям для входа в Комплекс. Политика будет действовать для всех новых пользователей. Для ранее созданных пользователей политика будет действовать только на смену пароля.

Для объекта «Учетная запись ресурсов» имеется возможность создавать различные варианты политик, которые автоматически применяются к УЗ пользователей в УЗР. Поскольку каждому пользователю в УЗР определен тип УЗ, политику можно настроить таким образом, чтобы она применялась только к определенным типам УЗ.

Варианты парольной политики для объекта «Учетная запись ресурсов»:

- 1) Политика, которая автоматически применяется только к УЗ всех типов/определенного типа в УЗР.
- 2) Политика, которая автоматически применяется ко всем УЗ существующих УЗР. Кроме тех, на которые действуют более приоритетные политики.

Изм.	Подп.	Дата

- 3) Политика, которая применяется к конкретной УЗ всех типов/определенного типа в УЗР.

Особенности применения парольных политик и их приоритет в зависимости от флагов приведены в пункте 2.11.3.

Список парольных политик содержит следующую информацию:

- название политики;
- назначение.

Детали парольных политик содержат следующие сведения:

- 1) «Данные»:

- название политики;
- назначение: «Пользователь» или «Учетные записи ресурсов»;
- флаги «Применить ко всем УЗ пользователя в выбранных УЗР», «Применить ко всем УЗ ресурсов» установлен/нет. Отображаются только для парольной политики с назначением «Учетные записи ресурсов»;

- 2) «Настройки политики»:

- число последних использованных паролей. Отображается, если было задано значение;
- максимальное время действия пароля. Отображается, если было задано значение;
- минимальное время действия пароля. Отображается, если было задано значение.

- 3) «Объекты применения» – отображаются детали применения политики для объекта «Пользователи».

- 4) «Связки учетных записей ресурсов с типами УЗ» – отображаются детали применения политики для объекта «Учетная запись ресурсов».

Список правил для поддерживаемых парольных политик приведен в таблице 2.30.

Таблица 2.30 – Список поддерживаемых правил для парольных политик

Параметр	Назначение	Примечание
minLength	Минимальная длина	Если значение минимальной

Изм.	Подп.	Дата

Параметр	Назначение	Примечание
	пароля	длины пароля не указано, то количество символов приравнивается к значению максимальной длины пароля
maximumLength	Максимальная длина пароля	Если значение минимальной длины пароля не указано, то количество символов приравнивается к 12 по умолчанию в соответствии с параметрами генерации пароля
disallowUserLogin	Пароль не должен содержать логин пользователя	–
minAlphabetChars	Минимальное число букв в пароле	–
minNumericChars	Минимальное число цифр в пароле	–
minSpecialChars	Минимальное число специальных символов в пароле	–
minLowercaseChars	Минимальное число символов в нижнем регистре	–
minUppercaseChars	Минимальное число символов в верхнем регистре	–
startWithAlphabet	Пароль должен начинаться с буквы	–
cannotContainCharacter	Пароль не должен	Массив отображаемых символов,

Изм.	Подп.	Дата

Параметр	Назначение	Примечание
s	содержать перечисленные символы	в кавычках, перечисленных через запятую. Например, ["@", "\$", "1", "Я"]
minPwdAge	Минимальное время действия пароля (в сутках)	В случае смены пароля администратором для пользователя или УЗ ЦС пользователя (операция сброса пароля), политика не применяется
expirePeriod	Максимальное время действия пароля	Если срок действия пароля равен 0 или не установлен, считается, что срок его действия неограничен
minNewChars	Минимальное количество измененных символов при создании нового пароля	В случае пакетной смены пароля, политика применяется к каждой УЗ. В случае смены пароля администратором для пользователя или УЗ ЦС пользователя (операция сброса пароля), политика не применяется
passwordHistoryCount	Число последних использованных паролей	В случае смены пароля администратором для пользователя или УЗ ЦС пользователя (операция сброса пароля), политика не применяется
isCompromisedPassword	Пароль не должен быть из списка скомпрометированных паролей	Список скомпрометированных паролей задается в конфигурационном файле compromisedpasswords.json в веб- консоли «Swagger UI» через REST

Изм.	Подп.	Дата

Параметр	Назначение	Примечание
		API Комплекса методом PUT

Перед созданием/редактированием парольной политики следует учесть следующие моменты:

- 1) Нельзя создать более одной парольной политики с назначением «Пользователи». Назначение «Пользователи» будет недоступно при наличии в Комплексе такой политики.
- 2) Нельзя создать более одной парольной политики по умолчанию на одну и ту же УЗР, заданную на все типы УЗ пользователя.
- 3) Нельзя создать более одной парольной политики по умолчанию на одну и ту же УЗР, указав один и тот же тип УЗ пользователя.
- 4) Нельзя создать более одной парольной политики с флагом «Применить ко всем УЗ ресурсов». Флаг будет недоступен при наличии в Комплексе такой политики.
- 5) Название парольной политики должно быть уникальное. В противном случае отобразится подсказка «Такое значение уже существует».
- 6) Для всех числовых полей допустимо вводить целые значения от 1 до 2147483647 (включительно). Не допускается вводить формулы, отрицательные и дробные числовые значения. В противном случае в зависимости от формата вводимых значений они будут либо сброшены, либо числовое поле будет подсвечено красным цветом.

Какие парольные политики относятся к политикам по умолчанию приведено в таблице 2.31 подпункта 2.11.3.

2.11.1. Создание парольной политики

Важные моменты, которые следует учесть перед созданием парольной политики, приведены в подразделе 2.11.

Для создания парольной политики с назначением «УЗР» администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать». Откроется окно «Создание парольной политики».

Изм.	Подп.	Дата

- 2) Перейти на обязательную вкладку «Параметры» и выполнить:
 - заполнить обязательное поле «Название парольной политики». В противном случае отобразится подсказка «Не заполнено обязательное поле»;
 - выбрать в обязательном поле «Назначение» значение «Учетные записи ресурсов»;
 - заполнить при необходимости остальные поля.
- 3) Установить при необходимости один из двух флагов «Применить ко всем УЗ пользователям в выбранных УЗ ресурсах» или «Применить ко всем УЗ ресурсам», чтобы политика применялась по умолчанию. Подробнее о политике по умолчанию приведено в пункте 2.11.3.
- 4) Перейти при необходимости на вкладку «Правила политики» и выполнить действия, приведенные в подпункте 2.11.1.1.
- 5) Перейти при необходимости на вкладку «Пользовательские правила» и выполнить действия, приведенные в подпункте 2.11.1.2.
- 6) Перейти на обязательную вкладку «Список УЗ ресурсов» и выбрать нужные УЗР. По умолчанию с УЗР связаны все типы УЗ. Кнопка «Ограничить по типам УЗ» позволяет настроить политику на определенный тип УЗ, подробнее в подпункте 2.11.1.3.
- 7) Нажать кнопку «Создать».

Следует учитывать следующее поведение:

- 1) Вкладка «Список УЗ ресурсов» недоступна, если был установлен флаг «Применить ко всем УЗ ресурсам».
- 2) Если не выбраны УЗР на вкладке «Список УЗ ресурсов», отображается сообщение «Ошибка при создании парольной политики из-за отсутствия УЗР».
- 3) Смена флага «Применить ко всем УЗ ресурсам» приводит к сбросу выбранных УЗР в списке. После чего требуется повторно выбрать УЗР на вкладке «Список УЗ ресурсов».
- 4) Поле «Назначение» и флаги будут недоступны при редактировании.

Изм.	Подп.	Дата

Для создания парольной политики с назначением «Пользователи» администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать». Откроется окно «Создание парольной политики».
- 2) Перейти на обязательную вкладку «Параметры» и выполнить:
 - заполнить обязательное поле «Название парольной политики». В противном случае отобразится подсказка «Не заполнено обязательное поле»;
 - выбрать в обязательном поле «Назначение» значение «Пользователи». При этом вкладка «Список УЗ ресурсов» станет недоступна;
 - заполнить при необходимости остальные поля.
- 3) Перейти при необходимости на вкладку «Правила политики» и выполнить действия, приведенные в подпункте 2.11.1.1.
- 4) Перейти при необходимости на вкладку «Пользовательские правила» и выполнить действия, приведенные в подпункте 2.11.1.2.
- 5) Нажать кнопку «Создать».

2.11.1.1. Добавление правила политики для парольной политики

Добавление правил политики выполняется при создании или редактировании парольной политики.

Для добавления правил политики необходимо:

- 1) Нажать кнопку «Добавить правило».
- 2) Выбрать нужное правило из выпадающего списка. При добавлении правила «Пароль не должен содержать перечисленные символы» добавлять можно только по одному символу. Для ввода каждого следующего символа нужно нажимать кнопку «Ввод».
- 3) Ввести при необходимости в поле «Значение» числовое значение.
- 4) Нажать кнопку «Добавить».

2.11.1.2. Добавление пользовательских правил для парольной политики

Комплекс позволяет расширить существующие правила парольной политики.

Изм.	Подп.	Дата

Для этого требуется создать пользовательский java-класс. После этого выполнить добавление пользовательского правила, в котором следует описать свой алгоритм проверки пароля.

Добавление пользовательских правил выполняется при создании или редактировании парольной политики.

Для добавления пользовательских правил необходимо:

- 1) Перейти на вкладку «Пользовательские правила».
- 2) Нажать «Добавить правило».
- 3) Заполнить идентификатор правила. Указывается значение созданного пользовательского java-класса.
- 4) Заполнить блок с параметрами со своим алгоритмом проверки пароля.
- 5) Нажать кнопку «Добавить».

Возможные сообщения об ошибках при сохранении политики:

- «Невалидный JSON в пользовательских правилах» – при заполнении блока с параметрами была допущена синтаксическая ошибка;
- «Ошибка в пользовательских правилах» – идентификатор правила не соответствует созданному java-классу или не существует.

Пример создания java-класса для пользовательских правил парольной политики:

```
public class CannotBeNeighborsCharactersPolicy implements PasswordPolicy
```

Пример блока параметров со своим алгоритмом проверки пароля:

```
{code:java}
{
  "neighbourChars": [
    "qwerty",
    "123456"
  ],
  "numChars": 5
}
{code}
```

2.11.1.3. Изменить связь УЗР с типами УЗ для парольной политики

Комплекс предоставляет возможность ограничить и выбрать типы УЗ пользователей для парольной политики.

Изм.	Подп.	Дата

Чтобы ограничить по типам УЗ, необходимо выполнить следующие действия при создании/редактировании парольной политики:

- 1) На вкладке «Список УЗ ресурсов» нажать кнопку «Ограничить по типам УЗ». Откроется окно «Ограничение по типам УЗ».
- 2) Выбрать один или несколько типов УЗ доступных из списка. Если тип УЗ недоступен, отображается значок информации.
- 3) Нажать «Сохранить».
- 4) Завершить создание/редактирование парольной политики.

Если не выбран ни один тип, то парольная политика будет действовать на все существующие и будущие типы пользовательских УЗ в данной УЗР.

Связки УЗР с типами УЗ отображаются в деталях парольной политики на вкладке «Информация». Для просмотра выбранных типов УЗ следует раскрыть структуру УЗР.

2.11.2. Редактирование парольной политики

Важные моменты, которые следует учесть перед редактированием парольной политики, приведены в подразделе 2.11.

Для редактирования парольной политики администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Парольные политики»:

- 1) Выбрать парольную политику.
- 2) Нажать кнопку «Редактировать». Откроется окно «Редактирование парольной политики».
- 3) На вкладке «Параметры» выполнить редактирование поля «Название парольной политики». Поле «Назначение» недоступно.
- 4) На вкладке «Правила политики» выполнить редактирование, удаление или добавление правил:
 - нажать кнопку «Редактировать» для редактирования правила. Внести изменения и нажать кнопку «Изменить»;
 - нажать кнопку «Удалить» для удаления правила;
 - нажать кнопку «Добавить правило» для добавления нового правила.Выполнить действия, приведенные в подпункте 2.11.1.1.

Изм.	Подп.	Дата

- 5) На вкладке «Пользовательские правила» выполнить добавление пользовательских правил:
- нажать кнопку «Редактировать» для редактирования правила. Внести изменения и нажать кнопку «Изменить»;
 - нажать кнопку «Удалить» для удаления правила;
 - нажать кнопку «Добавить правило», откроется окно «Пользовательское правило». Выполнить действия, приведенные в подпункте 2.11.1.2.
- 6) На вкладке «Список УЗ ресурсов» выполнить редактирование УЗР. Вкладка недоступна, если был установлен флаг «Применить ко всем УЗ ресурсов».
- 7) Нажать кнопку «Сохранить».

2.11.3. Применение парольной политики

Способы применения парольных политик:

- 1) Вручную.
- 2) По умолчанию (автоматически).

Комплекс применяет парольные политики в зависимости от их приоритета.

Способы применения парольных политик и приоритет приведены в таблице 2.31.

Таблица 2.31 – Способы применения парольных политик и приоритет

Назначение политики	Объекты, к которым применяется политика	Способ применения	Приоритет	Установка флагов
Пользователи	Все пользователи	По умолчанию	–	Отсутствует
Учетная запись ресурса	Конкретные УЗ всех типов/определенного типа в выбранных УЗР	Вручную	Самый высокий	Не установлены
Учетная	Конкретные УЗ	По	Высокий	«Применить ко

Изм.	Подп.	Дата

Назначение политики	Объекты, к которым применяется политика	Способ применения	Приоритет	Установка флагов
запись ресурса	определенного типа в выбранных УЗР	умолчанию		всем УЗ пользователям в выбранных УЗ ресурсов»
Учетная запись ресурса	Все УЗ в выбранных УЗР	По умолчанию	Средний	«Применить ко всем УЗ пользователям в выбранных УЗ ресурсов»
Учетная запись ресурса	Все УЗР Комплекса	По умолчанию	Низкий	«Применить ко всем УЗ ресурсов». Устанавливаются оба флага

В Комплексе может применяться только одна парольная политика на все УЗР Комплекса. Она автоматически будет применяться и к новым УЗР.

Парольная политика так же будет применяться для тех паролей, которые приходят через синхронизацию из ЦС.

Парольную политику, которая создана для конкретных УЗ/типов УЗ в выбранных УЗР, необходимо вручную связать с УЗ пользователей. Действия подробно описаны в пункте 2.3.7.

Для применения более строгой политики для конкретной УЗР следует:

- 1) Создать строгую политику на эту же УЗР с выключенными флагами.
- 2) Сменить политику по умолчанию для УЗ на более строгую. Смена парольной политики описана в пункте 2.3.7. В противном случае будет действовать парольная политика по умолчанию.

Изм.	Подп.	Дата

2.11.4. Удаление парольной политики

Для удаления парольной политики администратору необходимо выполнить в меню «Админ. системы»/«Парольные политики» следующие действия:

- 1) Выбрать из списка парольную политику.
- 2) Нажать кнопку «Удалить». Появится окно «Удаление парольной политики».
- 3) В окне «Удаление парольной политики» отображается информация:
 - название парольной политики и область ее применения;
 - информационное сообщение «При удалении парольной политики будут удалены все ее связки с аккаунтами».
- 4) Нажать кнопку «Удалить».

2.12. Настройка уведомлений

Комплекс позволяет настраивать уведомления на электронную почту пользователей следующими средствами:

- 1) Настройка уведомлений на события (создание, изменение, удаление) управляемых объектов в файле managed.json. Отправка уведомлений осуществляется скриптом (JavaScript). Описание конфигурационного файла managed.json приведено в разделе 22 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 2) Настройка уведомлений на события (создание, изменение, удаление) УЗ в объекте «Маппинг». Отправка уведомлений осуществляется скриптом (JavaScript).
- 3) Настройка уведомлений в рабочих потоках с помощью модуля «Activiti». Отправка уведомлений осуществляется задачей рабочего потока «Mail Task».
- 4) Настройка уведомлений при согласовании заявки из письма.

Поддерживаемые атрибуты письма приведены в таблице 2.32.

Изм.	Подп.	Дата

Таблица 2.32 – Поддерживаемые атрибуты письма

Атрибу т	Описание	Призна к обязательности	Значение по умолчанию	Пример	Комментарии
subject	Тема письма	Нет	<no subject>	Ankey IDM: создание УЗ	–
from	От кого	Нет	Значение атрибута from из файла настроек external.email.js on	ankeynoreply@gaz-is.ru	Если атрибут from не заполнен и не указано значение по умолчанию в файле настроек external.email.js on, произойдет ошибка
to	Кому	Да	–	aib@gaz-is.ru,ankeyadmin@gaz-is.ru	В случае нескольких получателей их адреса указываются через запятую
cc	Копия	Нет	–	user@gaz-is.ru,manager@gaz-is.ru	В случае нескольких получателей их адреса указываются через запятую

Изм.	Подп.	Дата

Атрибу	Описание	Признак обязательности	Значение по умолчанию	Пример	Комментарии
bcc	Скрытая копия	Нет	–	user@gaz-is.ru,manager@gaz-is.ru	В случае нескольких получателей их адреса указываются через запятую
type	Тип письма	Нет	text/html	text/plain	Возможные значения: text/plain, text/html, text/xml. В случае указания не поддерживаемого типа произойдет ошибка
body	Тело письма	Нет	<empty message>	В ЦС Microsoft Active Directory создана УЗ aib@gaz-is.ru	–

Изм.	Подп.	Дата

2.12.1. Настройка почтового сервера

Настройка почтового сервера, через который будут осуществляться уведомления, выполняется в следующих конфигурационных файлах:

- 1) `workflow.json` – в случае уведомлений из рабочего потока. Администратору следует добавить блок «mail». Описание файла приведено в разделе 35 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 2) `external.email.json` – в случае уведомлений для управляемых объектов и объекта «Маппинг». Описание файла приведено в разделе 19 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Конфигурационные файлы расположены в каталоге установки Комплекса `ankey/conf/`.

2.12.2. Настройка скрипта отправки уведомлений

Отправка уведомления может выполняться предварительно разработанным скриптом (например, JavaScript). По умолчанию скрипты хранятся в каталоге `ankey/script`.

Пример содержимого скрипта отправки уведомления:

```
var email = {  
    from : "ankeynoreply@domain.local",  
    to : "user@domain.local",  
    cc : "admin@domain.local",  
    subject : "Account created",  
    type : "text/html"  
};  
  
if (openidm.read("config/external.email")) {  
    email.body = "Found created account in target system:<br>"  
    + "<table border='1'><tbody>"  
    + "<tr><td>Account Name</td><td>" + source.username + "</td></tr>"
```

Изм.	Подп.	Дата

```
+ "<tr><td>Last Name</td><td>" + source.lastname + "</td></tr>"
+ "<tr><td>First Name</td><td>" + source.firstname + "</td></tr>"
+ "<tr><td>Middle Name</td><td>" + source.middlename + "</td></tr>"
+ "</tbody></table><br><br>"
+ "Account created in ankey.";
```

```
openidm.action("external/email", "sendEmail", email);
```

```
} else {
    console.log("Email service not configured; report not generated. ");
}
```

В примере скрипта выше вызываются следующие функции:

- 1) `openidm.read ("config/external.email")` – функция считывает настройки подключения к почтовому серверу из файла `external.email.json`.
- 2) `openidm.action ("external/email", "sendEmail", email)` – функция отправки почты. Процедура получает три аргумента на вход: идентификатор объекта (пример, «external/email»), действие («sendEmail») и объект, в котором хранятся необходимые для выполнения действия значения (например, `email`).

Объект, в котором хранятся необходимые для выполнения отправки уведомления данные, включает следующие параметры:

- 1) «from» – почтовый адрес отправителя.
- 2) «to» – почтовые адреса получателей, разделенные запятой.
- 3) «cc» – почтовые адреса получателей, разделенные запятой, в копии письма.
- 4) «bcc» – почтовые адреса получателей, разделенные запятой, в скрытой копии письма.
- 5) «subject» – тема письма.
- 6) «body» – содержимое письма.

Изм.	Подп.	Дата

- 7) «type» – формат содержимого письма (MIME-тип). Может принимать значения «text/plain», «text/html», or «text/xml».

2.12.3. Отправка уведомления с помощью REST-запроса

Для отправки уведомления используется следующий REST-запрос:

```
ANKEY@NODE1:~$ curl
--header "Content-Type: application/json" ^
--header "X-ankey-Username: ankey" ^
--header "X-ankey-Password: ankey" ^
--request POST ^
--data '{"from":"ankeynoreply@domain.local","to":"recipient@domain.local", "subject":"Test","body":"Test"}'
"<http://localhost:8080/ankey/external/email?_action=send>"
```

2.12.4. Настройка согласования заявки из уведомления

Комплекс поддерживает возможность направления заявки по электронной почте и согласованию ее в ответном письме.

Настройка решения по согласованию заявки из письма выполняется в следующих объектах:

- 1) Файл настроек параметров подключения к почтовому серверу для отправки сообщений `external.email.json`. Настройка выполняется в подразделе 4.9.
- 2) Файл настроек параметров подключения к почтовому серверу для получения входящих сообщений `email.request.processing.json`. Настройка выполняется в подразделе 4.9.
- 3) Периодическое задание с типом «`email.request.processing.task`» для обработки действий с заявками через электронную почту. Описание приведено в пункте 2.4.5.

Изм.	Подп.	Дата

2.12.5. Настройка уведомления о новом инциденте

Для настройки уведомления о новом инциденте необходимо выполнить следующие действия:

- 1) Настройки параметра почтового сервера в файле `ankey/conf/workflow.json` согласно пункту 2.12.1.
 - 2) Убедиться, что хотя бы у одного ответственного за инциденты, для которого будет назначено рассмотрение инцидента, указан почтовый адрес.
 - 3) Убедиться, что в Комплексе установлена переменная `JAVA_OPTS` со значением `-DbaseUrl=<адрес ankey>`, например `-DbaseUrl=http://www.gaz-is.ru/ankey`. В случае отсутствия этого параметра будет использован адрес по умолчанию: `http://localhost:8080/`
- Шаблон уведомления о новом инциденте приведен в таблице 2.33.

Таблица 2.33 – Шаблон уведомления о новом инциденте

Атрибуты письма	Текст	Пример
Тема	Новый инцидент №<номер инцидента>	Новый инцидент №17
Сообщение	Для Вашей информационной системы <имя информационной системы> имеется новый инцидент №<номер инцидента>: Правило (роль): <имя роли>; Нарушитель: <ФИО нарушителя>. Ознакомиться с деталями инцидента можно по ссылке <адрес ankey>	Для Вашей информационной системы Сетевое рабочее место имеется новый инцидент №17: Правило (роль): Администратор домена; Нарушитель: Барбашев Олег Андреевич. Ознакомиться с деталями инцидента можно по ссылке https://www.gaz-is.ru/ankey

Изм.	Подп.	Дата

2.12.6. Настройка уведомлений в рабочих потоках с помощью модуля «Activiti»

Для настройки уведомлений в рабочих потоках используется типовая задача рабочего потока «Mail Task» модуля «Activiti».

Основные рекомендации по настройке «Mail Task»:

- задача «Mail Task» должна выполняться параллельно другим задачам рабочего потока (например, с помощью компонента «Parallel Gateway»), для исключения задержек в обработке заявки или ошибочного завершения рабочего потока;
- возникающие исключения (например, не определен адрес получателя) при работе задачи «Mail Task» должны игнорироваться, для этого в файл конфигурации рабочего потока добавлен параметр «ignoreException».

Пример добавления параметра «ignoreException» в файл конфигурации рабочего потока:

```
<serviceTask id="mailtask1" name="Mail Task" activiti:type="mail">
  <extensionElements>
    .....
    <activiti:field name="ignoreException">
      <activiti:string>
        <![CDATA[true]]>
      </activiti:string>
    </activiti:field>
  </extensionElements>
</serviceTask>
```

2.12.7. Перевод строки в сообщениях Комплекса

Для настройки перевода строки в сообщениях Комплекса необходимо выполнить следующие действия:

- 1) Добавить символ перевода строки «\n» в файл русскоязычной локализации translation_ru.properties.
- 2) Перезагрузить Комплекс.

Описание файла русскоязычной локализации translation_ru.properties приведено в пункте 2.7.3.

Пример перевода строки в сообщении при создании подразделения:

Изм.	Подп.	Дата

2.13.1.1. Настройка выделяемой памяти Java процессу сервера приложений

По умолчанию размер выделяемой памяти процессу Java сервера приложений Комплекса задается в скрипте `ankey/startup.sh` в переменной окружения: `ANKEYIDM_OPTS="-Xmx1024m -Xms1024m"`.

При необходимости данные параметры могут быть изменены в соответствии с документацией Java.

Дополнительно для оптимизации работы с памятью рекомендуется выбирать подходящий под текущую нагрузку Комплекса сборщик мусора Java.

2.13.1.2. Настройка пула соединений к СУБД

Настройка пула соединений к СУБД выполняется в конфигурационном файле `conf/repo.jdbc.json` в секции «pool». Описание параметров секции «pool» приведено в таблице 2.34.

Таблица 2.34 – Описание параметров секции «pool»

Параметр	Значение по умолчанию	Описание параметра
<code>initialSize</code>	0	Первоначальный размер пула при старте сервера приложений Комплекса
<code>maxTotal</code>	20	Максимальное число подключений к БД
<code>maxIdle</code>	20	Максимальное число простаивающих соединений к БД
<code>maxWaitMillis</code>	5000	Максимальное время ожидания подключения к БД в миллисекундах. Значение по умолчанию «-1» означает, что нет ограничения по времени

Дополнительные параметры, настройка которых может потребоваться, если есть проблема с периодическим отключением соединений к БД. Описание дополнительных параметров приведено в таблице 2.35.

Изм.	Подп.	Дата

Таблица 2.35 – Описание дополнительных параметров

Параметр	Значение по умолчанию	Описание параметра
testOnBorrow	true	Проверка соединения на работоспособность при каждом запросе из пула. При значении «true» должно быть указано значение validationQuery
testOnCreate	false	Проверка на работоспособность перед созданием соединения
testOnReturn	false	Проверка перед возвращением соединения в пул. Если после очередного цикла проверки соединения в пуле: <ul style="list-style-type: none"> – maxIdle, то при попытке возврата соединения в пул оно будет закрыто; – < minIdle, то фоновый процесс создаст дополнительные соединения. Объекты, которые не могут быть проверены, будут удалены из пула
testWhileIdle	false	Проверка неиспользуемых соединений. При значении «true» должно быть указано значение validationQuery
timeBetweenEvictionRunsMillis	-1	Время ожидания в миллисекундах между проверками

Изм.	Подп.	Дата

Параметр	Значение по умолчанию	Описание параметра
		работоспособности и очистки неактивных соединений
numTestsPerEvictionRun	3	Количество неактивных соединений, проверяемых за раз
minEvictableIdleTimeMillis	1000 * 60 * 30	Минимальное время между проверками доступности соединения
removeAbandonedOnBorrow	false	Удаление некорректных соединений при выделении из пула. Соединение считается разорванным и может быть удалено, если оно не использовалось дольше, чем значение параметра removeAbandonedTimeout
removeAbandonedOnMaintenance	false	Удаление из пула некорректных соединений после проверки доступности соединения
removeAbandonedTimeout	300	Время ожидания в секундах, через которое любое простаивающее соединение будет считаться неактивным
validationQuery	–	SQL-запрос проверки соединения. Запрос должен использовать оператор SELECT, который возвращает хотя бы одну строку, например, «SELECT 1»
connectionInitSqls	NULL	SQL-запрос инициализации соединения

Изм.	Подп.	Дата

Пример настройки параметров соединений к СУБД в конфигурационном файле `conf/repo.jdbc.json` в секции «pool»:

```
"pool" : {
  "initialSize": 0,
  "maxTotal": 8,
  "maxIdle": 8,
  "minIdle": 0,
  "maxWaitMillis": -1,
  "testOnBorrow": true,
  "testOnCreate": true,
  "testOnReturn": true,
  "testWhileIdle": true,
  "timeBetweenEvictionRunsMillis": 400,
  "numTestsPerEvictionRun": 4,
  "minEvictableIdleTimeMillis": -1,
  "removeAbandonedOnBorrow": true,
  "removeAbandonedOnMaintenance": true,
  "validationQuery": "SELECT 1",
  "connectionInitSqls": ["SELECT 1", "SELECT 2"]
}
```

2.13.1.3. Отказ от создания точек восстановления перед каждым запросом в БД

После запуска Комплекса в постоянную эксплуатацию (фиксирование настраиваемых полей объектов Комплекса) можно отказаться от создания точек восстановления перед каждым запросом в БД во избежание ошибки «cached plan must not change result type», возникающей после изменения структуры таблиц (добавление/изменение/удаление колонок) при использовании пула соединений в БД PostgreSQL. Это позволит избежать потенциальных задержек при взаимодействии с БД.

Для отключения функции создания точек восстановления необходимо в конфигурационном файле `repo.jdbc.json` убрать параметр `autosave=conservative` из переменной `connection/dataSource/url`.

Например, вместо:

```
"url" : "jdbc:postgresql://localhost:5432/ankey?autosave=conservative"
```

будет

```
"url" : "jdbc:postgresql://localhost:5432/ankey"
```

Изм.	Подп.	Дата

При необходимости поменять настраиваемые поля после отключения параметра, выполнив следующие действия:

- 1) Вернуть первоначальное значение параметра.
- 2) Перезагрузить сервер приложений Комплекса.
- 3) Внести в extend.json изменения и дождаться их применения.
- 4) Отключить функции создания точек восстановления.
- 5) Перезагрузить сервер приложений Комплекса.

2.13.1.4. Настройка асинхронности выполнения операций распространения данных

Настройка асинхронности выполнения операций распространения данных может выполняться в случае долгого выполнения provision-операций (для объекта «Маппинг» параметр «mapType» установлен в «provision»). При включении режима асинхронного выполнения на стороне Комплекса объекты будут создаваться или обновляться мгновенно, вне зависимости выполнилась ли операция или еще нет фактически.

Настройка асинхронного выполнения операций выполняется в конфигурационном файле conf/sync.json. Описание конфигурационного файла sync.json и пример приведены в разделе 30 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.10.

2.13.1.5. Настройка количества потоков выполнения периодических заданий

При большом количестве периодических заданий рекомендуется увеличить количество потоков выполнения «threadCount» в файле conf/scheduler.json, описание приведено в пункте 2.4.6.

2.13.1.6. Настройка частоты обращения к endpoint/notifications/count

Частота обращения к endpoint/notifications/count задается на вкладке «ui/configuration» в меню «Админ. системы»/«Системные настройки»/«Настройки интерфейса».

Изм.	Подп.	Дата

Чтобы изменить частоту обращения к endpoint/notifications/count, необходимо:

- 1) В параметре «notificationCountRequestInterval» указать значение в диапазоне от 10 до 86400. Значение указывается в секундах. По умолчанию значение параметра равно 10 секундам. Нельзя указывать значение меньше 10 секунд.
- 2) Для применения настроек обновить страницу браузера.

2.13.2. Настройка производительности СУБД

Комплекс можно отнести к транзакционным системам (Online Transaction Processing – OLTP). Комплекс может выполнять большое число чтений из БД (например, работа пользователей в интерфейсе), так и большое число записей в БД (например, синхронизация данных, пересчет полномочий (задача evaluate), аудит).

Общие рекомендации по повышению производительности СУБД:

- 1) Для построения отчетов использовать отдельную реплику СУБД (например, Standby реплика в режиме только чтения), чтобы уменьшить нагрузку на оперативные данные Комплекса.
- 2) Для записи и чтения событий аудита использовать отдельную инстанцию СУБД. Описание настройки приведено в пункте 2.10.4.
- 3) Выполнять мониторинг производительности СУБД и повышение ее общей производительности в соответствии с документацией разработчика СУБД.
- 4) Настраивать количество допустимых подключений на стороне СУБД необходимо с учетом настройки пула соединений на стороне сервера приложений (см. подпункт 2.13.1.3).

Для колонок таблиц, используемых в правилах связывания УЗР и правилах корреляции в маппингах синхронизации, рекомендуется создавать индексы в БД. Для создания индекса создается файл настройки «liquibase» и помещается в каталог ankey\db\postgresql\liquibase. Пример файла настройки индекса ankey\db\postgresql\liquibase\snn.xml для настраиваемого поля «Табельный номер SNN» объекта «Пользователь» (managed/user):

Изм.	Подп.	Дата

```
<changeSet id="123" author="gis">
  <createIndex tableName="usr"
    indexName="idx_snn"
    schemaName="ankey"
    tablespace="pg_default"
    unique="true">
    <column name="snn" type="character varying(64)"/>
  </createIndex>
</changeSet>
```

2.13.3. Настройка производительности службы поиска

Настройка производительности службы поиска Opensearch выполняется в соответствии с документацией разработчика.

Комплекс не содержит дополнительных настроек производительности в части службы поиска.

2.14. Настройка пула потоков для асинхронных операций

Агрегированные заявки объединяют несколько отдельных заявок в одну, что может привести к значительному увеличению количества одновременно обрабатываемых задач.

Нестандартные цепочки согласований с множеством шагов и сложной логикой требуют больше времени и ресурсов для обработки.

Для улучшения производительности Комплекса можно использовать настройки пула асинхронных операций.

Комплекс поддерживает следующие независимые настройки пула асинхронных операций:

- 1) «`asyncExecutor`» в конфигурационном файле `conf/workflow.json`.
Настройка отвечает за обработку заявок на уровне `Activiti`, на всех этапах: начало бизнес-процесса, переход заявки по этапам согласования.
Включение «`asyncExecutor`» активирует многопоточную обработку заявок одновременно в несколько потоков, что позволит значительно повысить их пропускную способность и скорость обработки.
- 2) «`ankey.async.execution.service.threads.count`» в конфигурационном файле `conf/request.json`. Настройка отвечает за количество потоков, обрабатывающих заявки на этапе их отправки на согласование. Влияет на производительность на начальном этапе, может ускорить обработку

Изм.	Подп.	Дата

новых заявок, но не повлияет на скорость прохождения уже запущенных заявок через этапы согласования.

Обе настройки могут помочь улучшить производительность Комплекса, как вместе, так и независимо друг от друга. Выбор оптимальной настройки зависит от конкретных требований к производительности Комплекса.

Для настройки «`asyncExecutor`» Администратору необходимо выполнить следующие действия:

- 1) Активировать асинхронный режим операций, изменив значение параметра «`asyncExecutorEnabled`» с «`false`» (по умолчанию) на «`true`».
- 2) Задать количество потоков в пуле асинхронного выполнения в параметре «`asyncExecutorCorePoolSize`». По умолчанию 2.

После сохранения настроек Комплекс будет перезагружен автоматически.

Описание конфигурационного файла `conf/workflow.json` приведено в разделе 35 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Для настройки «`ankey.async.execution.service.threads.count`» Администратору необходимо выполнить следующие действия:

- 1) Активировать асинхронный режим операций, изменив значение параметра `asyncWorkflowInitialization` с «`false`» (по умолчанию) на «`true`».
- 2) Задать количество потоков в пуле асинхронного выполнения:
 - открыть конфигурационный файл `conf/boot/boot.properties`;
 - найти параметр `ankey.async.execution.service.threads.count`;
 - раскомментировать строку, если она закомментирована;
 - указать желаемое количество потоков. По умолчанию 3.

Примечания:

- 1) Пул «`ankey.async.execution.service.threads.count`» активен только при включенном асинхронном режиме.
- 2) Если параметр «`ankey.async.execution.service.threads.count`» не задан или закомментирован, значение пула по умолчанию будет равно количеству ядер CPU машины, на которой запущен Комплекс.

Изм.	Подп.	Дата

- 3) Если параметр «asyncExecutorCorePoolSize» не задан, а «asyncExecutorEnabled» активирован, Комплекс автоматически использует значение пула по умолчанию, равное двум.

2.15. Настройка интеграции с внешними системами

Интеграция с внешними системами осуществляется с использованием программного интерфейса REST API Комплекса. Описание поддерживаемых функций REST API Комплекса доступно в веб-консоли «Swagger UI», которая доступна по умолчанию по адресу: <http://localhost:8080/api>.

В веб-консоли «Swagger UI» доступны следующие действия с объектами Комплекса:

- 1) Просмотр объектов и их деталей. Выполняется запросом «GET» для соответствующих объектов.
- 2) Создание новых объектов. Выполняется запросом «POST» или «PUT» для соответствующих объектов.
- 3) Изменение объектов. Выполняется запросом «PATCH» или «PUT» для соответствующих объектов.
- 4) Удаление объектов. Выполняется запросом «DELETE» для соответствующих объектов.

Все действия выполняются из под служебной УЗ пользователя «ankey» с паролем по умолчанию «ankey». Действия могут быть выполнены из под другого пользователя, для этого необходимо указать имя пользователя и его пароль в параметрах «X-Ankey-Username» и «X-Ankey-Password» соответственно. Описание остальных параметров приведено в веб-консоли «Swagger UI».

Дополнительно администратор Комплекса может настраивать интеграцию со следующими внешними системами:

- 1) Внешние системы формирования отчетов. Интеграция производится за счет публикации внешней ссылки в интерфейсе Комплекса на веб-консоль системы формирования отчетов.

Изм.	Подп.	Дата

- 2) Внешние системы, которым необходимо проверять роли, назначаемые пользователю, на предмет нарушения правил разграничения доступа, настроенных в Комплексе.

2.15.1. Настройка внешней ссылки

Комплекс позволяет добавлять внешнюю ссылку и отображать ее в качестве пункта меню. Пункт меню с внешними ссылками по умолчанию называется «Документы и отчеты» и доступен Администратору всегда, даже когда внешние ссылки не заданы. Пользователь может видеть и получать доступ к этому пункту меню только тогда, когда есть одна или более внешних ссылок.

Администратору доступна кнопка «Редактировать», недоступная пользователям. Кнопка «Редактировать» открывает окно «Управление внешними ссылками», в котором Администратор может добавлять новые ссылки, изменять название внешней ссылки, редактировать и удалять заданные ранее ссылки.

Администратор может изменять название страницы с внешними ссылками в окне «Управление внешними ссылками», но название страницы в списке «Ссылки» изменится только при следующем входе в Комплекс, о чем сообщает информационная надпись в окне.

Добавление новой ссылки начинается с нажатия кнопки «Добавить ссылку» в окне «Управление внешними ссылками». Открывается окно «Создание внешней ссылки». Поля «Название» и «Адрес ссылки» – это обязательные поля. Кнопка «Создать» будет заблокирована до заполнения обязательных полей.

Название внешней ссылки уникально. Адрес ссылки проходит валидацию: при попытке ввести некорректные данные появится сообщение «Адрес ссылки должен начинаться с http, https, ftp».

Не обязательные поля – это поля «Ссылка будет доступна следующим ролям» и «Ссылка будет доступна следующим ролям по маске». В поле «Область видимости ролей» Администратор может выбрать существующую в системе роль. В поле «Ссылка будет доступна следующим ролям по маске» действует всплывающая подсказка о том, какие роли пользователей могут получить доступ по введенной маске. Маска – это часть названия роли, она не зависит от регистра. Например, при введении

Изм.	Подп.	Дата

в это поле значения «бухгалтер» отображаются %% внутри которых можно задать вручную значение маски.

Нажатие кнопки «Сохранить» после заполнения обязательных и, при необходимости, не обязательных полей сохраняет вновь созданную внешнюю ссылку, что подтверждается появлением сообщения «Внешние ссылки обновлены». После этого в списке «Ссылка» появится новая ссылка, доступная пользователям.

Существующую ссылку можно редактировать в окне «Управление внешними ссылками», которое открывается после нажатия кнопки «Редактировать». В этом окне для каждой ссылки есть своя кнопка «Редактировать», нажатие которой открывает для выбранной ссылки окно «Редактирование внешней ссылки», в котором можно добавить несколько ролей (добавленная роль не отображается в списке) или удалить как одну, так и все роли. В этом же окне можно редактировать обязательные поля «Название» и «Адрес ссылки», добавлять, изменять или удалять маски.

Для того, чтобы удалить внешнюю ссылку необходимо в окне «Управление внешними ссылками» нажать на кнопку «Корзина», расположенную рядом с удаляемой ссылкой, а затем нажать кнопку «Сохранить». Удалять ссылки может только Администратор.

2.15.2. Проверка нарушений правил разграничения доступа

Для проверки нарушений правил разграничения доступа из внешней системы необходимо отправить запрос REST API со следующими параметрами:

- метод «POST»;
- адрес «/endpoint/preliminary/incident»;
- тело запроса:

```
{  
  "userName": "<Идентификатор пользователя>",  
  "appRoleNames": ["<Имя роли 1>", "<Имя роли 1>"]  
}
```

В теле запроса в качестве идентификатора пользователя используется значение атрибута «userName» объекта «Пользователь», в качестве списка имен ролей «appRoleNames» значение атрибута «appRoleName» объекта «Роль». Если передается несколько ролей, то они разделяются запятыми.

Изм.	Подп.	Дата

В случае выявления нарушения Комплекс возвращает следующий ответ:

```
{  
  "hasIncidents" : true,  
  "manual": ["<Имя правила 1>","<Имя правила 2>"],  
  "role": ["<Имя роли 1>","<Имя роли 2>"]  
}
```

Содержимое ответа включает:

- «hasIncidents» – флаг наличия («true») или отсутствия («false») нарушения;
- «manual» – список правил разграничения доступа, на которых выявлено нарушение;
- «role» – список ролей, на которых выявлено нарушение, в случае если пользователь находится в другом подразделении чем роль.

2.16. Настройка службы сервера коннекторов

Настройка службы сервера коннекторов ICF Connector Server выполняется следующими средствами:

- 1) Конфигурационный файл <Путь к каталогу с ICF Connector Server>/conf/ConnectorServer.properties.
- 2) Скрипт <Путь к каталогу с ICF Connector Server>/bin/ConnectorServer.sh.

Файл ConnectorServer.properties включают следующие параметры настройки:

- 1) Порт службы сервера коннекторов, на котором будут приниматься подключения «connectorserver.port». По умолчанию имеет значение: «8759».
- 2) Наименование каталога на сервере коннектора, где располагаются внешние библиотеки, необходимые для работы коннекторов «connectorserver.libDir». По умолчанию имеет значение: «lib».
- 3) Параметр, активирующий режим подключения к службе сервера коннекторов по протоколу SSL «connectorserver.usessl». По умолчанию имеет значение: «false». Для включения режима необходимо установить параметр в значение «true».

Изм.	Подп.	Дата

- 4) Наименование каталога на сервере коннектора, где располагаются библиотеки коннекторов «connectorserver.bundleDir». По умолчанию имеет значение: «bundles».
- 5) Наименование используемой библиотеки журналирования работы сервера коннекторов «connectorserver.loggerClass». По умолчанию имеет значение: org.forgerock.openicf.common.logging.slf4j.SLF4JLog.
- 6) Ключ аутентификации при подключении к серверу коннекторов «connectorserver.key». Хранится в виде хеш значения, задаваемого параметром /setKey скрипта ConnectorServer.sh.

Скрипт ConnectorServer.sh имеет следующие параметры запуска:

- 1) Параметр «/run <параметры Java>». Запускает службу сервера коннекторов. Пример выполнения: «./ConnectorServer.sh /run» или «./ConnectorServer.sh /run «-J-Djavax.net.ssl.keyStore=mykeystore.jks» «-J-Djavax.net.ssl.keyStorePassword=changeit»», в случае подключения к службе сервера коннекторов по протоколу SSL.
- 2) Параметр «/setKey <значение ключа>». Задаёт значение ключа аутентификации при подключении к серверу коннекторов. Пример выполнения: «./ConnectorServer.sh /setKey changeit».
- 3) Параметр «/setDefault». Сбрасывает настройки файла ConnectorServer.properties на значения по умолчанию. Пример выполнения: «./ConnectorServer.sh /setDefault».

2.17. Настройка правил

Комплекс содержит следующие типы правил:

- 1) Правило автоназначения роли. Настраивается при создании/редактировании роли.
- 2) Правило связывания. Настраивается при создании/редактировании УЗР, для связывания новых УЗ ЦС с их владельцами (бенефициарами).
- 3) Правило корреляции. Настраивается в объекте «Маппинг» с типом «reson» для синхронизации объектов из доверенного источника. В таком режиме отсутствует форма ресурса и правило связывание не применимо.

Изм.	Подп.	Дата

- 4) Правило разграничения доступа (далее – правило Segregation of Duties (SoD)). Настраивается в разделе «Правила» меню «Админ. системы», для выявления нарушений прав доступа.

2.17.1. Настройка правила автоназначения

Правило автоназначения роли позволяет настроить автоматическое назначение роли пользователю или группе пользователей на основе заданного правила соответствия. Для одной роли допустимо создавать только одно правило.

Если в Комплексе присутствуют пользователи, которые удовлетворяют правилу, выполняется автоматическое назначение роли. Если атрибуты пользователей, которым была автоматически назначена роль, перестали соответствовать правилу, производится автоматический отзыв роли.

Назначение и отзыв ролей по правилу автоназначения выполняется стандартным периодическим заданием Комплекса «*approlesbyrule*».

Создание правила автоназначения выполняется при создании или редактировании роли в меню «Роли».

Для создания правила автоназначения администратору необходимо выполнить следующие действия:

- 1) Перейти на вкладку «Правило автоназначения» в процессе создания/редактирования роли.
- 2) Нажать кнопку «Открыть конструктор правил». Откроется окно «Правило автоназначения» в режиме «Конструктор».
- 3) Добавить правило.
- 4) Нажать кнопку «Сохранить».

Добавление правил подробно описано в подпункте 2.16.1.1.

Комплекс позволяет моделировать применения автоназначения ролей, описание приведено в подпункте 2.16.1.2.

2.17.1.1. Добавление правила автоназначения

Добавление правила доступно в двух режимах: «Конструктор» или «Код». Для переключения между режимами требуется нажать соответствующую кнопку.

Изм.	Подп.	Дата

Правило автоназначения представляет собой фильтр атрибутов для следующих объектов:

- «Пользователи»;
- «Трудоустройства»;
- «Роли»;
- «Учетные записи ресурса».

Для объекта «Полномочия УЗ» правило автоназначения задается в режиме редактора кода, переход в который возможен по кнопке «Перейти в редактор кода». Для создания правила его необходимо ввести в поле «Наличие полномочий УЗ у пользователя» и нажать кнопку «Сохранить». Ниже приведены примеры задания следующих правил автоназначения в режиме редактора кода:

- 1) «/Offline_Resform/groups/groupname eqic “код1”» – правило автоназначения роли по наличию составных полномочий УЗ, где:
 - «Offline_Resform» – имя формы ресурса;
 - «groups» – название мультиатрибута формы ресурса;
 - «groupname» – часть мультиатрибута «groups», имя группы (справочник);
 - «“код 1”» – значение refBookCode справочника в ЦС.
- 2) «/Offline_Resform/username eqic “John”» – правило автоназначения роли по наличию атрибута в УЗ, где:
 - «Offline_Resform» – имя формы ресурса;
 - «username» – атрибут формы ресурса, имя пользователя;
 - «“John”» – значение атрибута «username».

Примечание. В многозначных атрибутах можно задать условие только на типы полей «lookup» и «textField».

Перечень возможных атрибутов для объектов приведен в таблице 2.36.

Таблица 2.36 – Перечень возможных атрибутов для объектов

Объект	Атрибут
Пользователи	<ul style="list-style-type: none"> – «Электронная почта»; – «Фамилия»; – «Имя»;

Изм.	Подп.	Дата

Объект	Атрибут
	<ul style="list-style-type: none"> – «Отчество»; – «Вход в систему отключен»; – «Имя пользователя»; – «Телефон»; – «Табельный номер»; – «Линейный руководитель»; – «Основное подразделение»; – «Основная должность»; – «Идентификатор трудоустройства»; – «Статус»
Трудоустройства	<ul style="list-style-type: none"> – «Пользователь»; – «Подразделение»; – «Линейный руководитель»; – «Должность»; – «Идентификатор трудоустройства»
Роли	<ul style="list-style-type: none"> – «Название»; – «Описание»; – «Информационная система»; – «Правила автоназначения ролей»; – «Тип роли»; – «Запрет делегирования роли»
Учетные записи ресурса	<ul style="list-style-type: none"> – «Название»; – «Описание»; – «Бенефициар»; – «Ресурс»; – «Форма ресурса»; – «Тип объекта коннектора»; – «Тип правила связывания»; – «Тип УЗ по умолчанию»

Изм.	Подп.	Дата

В одном правиле допускается объединение нескольких фильтров. В этом случае будет осуществляться проверка на соответствие правилу по всем фильтрам.

При объединении в правиле фильтров «Пользователи» и «Трудоустройства» подразделение, на которое назначается роль, определяется по следующим условиям:

- 1) Фильтр «Пользователи» используется только для назначения на основное трудоустройство.
- 2) Если в фильтре «Трудоустройства» для пользователя доступно несколько трудоустройств, то роль назначается на все трудоустройства, соответствующие условиям фильтра.

Условия, поддерживаемые фильтром, приведены в таблице 2.37.

Таблица 2.37 – Условия, поддерживаемые фильтром

Условие	Отображение в конструкторе правила	Описание	Пример в режиме кода
eq	Равно	Равенство с учетом регистра	(/lastName eq 'Глазырин')
eqic	Равно без уч. рег.	Равенство без учета регистра	(/userName eqic 'glazyrin-s')
sw	Начинается с	Начинается с	(/userName sw 'GLAZYRIN')
co	Содержит	Содержит	(/userName co 'GLAZ')
tree	В иерархии	В иерархии (условие поддерживается только для атрибута «Основное подразделение» объекта «Пользователи» и атрибута «Подразделение» объекта «Трудоустройства») Проверяет, что переданное значение содержится в иерархии ссылочного объекта	(/organization_id tree 16)

Изм.	Подп.	Дата

Услови е	Отображение в конструктор е правила	Описание	Пример в режиме кода
in	Массив	<p>Среди всех перечисленных значений. Массив данных поддерживается для строковых полей (string и textField) и поля типа «lookup».</p> <p>Если в массиве указана несуществующая роль, то при сохранении роли отображается сообщение «Ошибка при изменении роли». Для строковых полей:</p> <ul style="list-style-type: none"> – поиск ведется по точному значению, включая любые символы, в том числе и символ пробела; – значение массива может состоять из нескольких значений с использованием символов; – несколько значений между собой объединены логическим оператором «ИЛИ» для пользователей, «И» для ролей и УЗР. <p>При использовании символов</p>	<p>Для объекта «Пользователи»:</p> <ol style="list-style-type: none"> 1) (/lastName in 'Глазырин,Барбашев') – выполнится для пользователя с фамилией «Глазырин» или «Барбашев». 2) (/lastName in 'Месье,Глазырин,Барбашев') – выполнится для пользователей с фамилией «Месье» или «Глазырин» или «Барбашев». 3) (/phone in '444, 443') – выполнится для пользователей, у которых телефон «444» или «443». 4) (/position in 'Главный бухгалтер,Аналитик\ , экономист') – выполнится для пользователей с должностью «Главный бухгалтер» или «Аналитик,экономист». <p>Для объекта «Роли»</p>

Изм.	Подп.	Дата

Услови е	Отображение в конструкторе правила	Описание	Пример в режиме кода
		<p>«,» и «\» следует учитывать следующие особенности:</p> <ul style="list-style-type: none"> – в случае использования запятой в режиме «Код» символ запятой отделяется знаком экранирования. В качестве знака экранирования используется двойной символ обратной косой черты (\\). Знак экранирования, предшествующий символу запятой, включает этот символ в результат. В режиме «Конструктор» символ запятой экранировать не требуется, необходимо ввести подряд значения через запятую и нажать «Enter». Например, в конструкторе необходимо указать "111-222", "222-333\\", "555-123", а в коде значение пропишется, как ((/position in "111-222"\\, 	<p>(/appRoleName in 'Аналитик,Бухгалтер') – выполнится для всех пользователей, у которых назначены обе роли «Аналитик» и «Бухгалтер»</p>

Изм.	Подп.	Дата

Услови е	Отображение в конструктор е правила	Описание	Пример в режиме кода
		<p>"222-333\\\\"";</p> <p>– в случае использования символа «\» в режиме «Код» в качестве знака экранирования используется тройной символ обратной косой черты (\\\). В режиме «Конструктор» экранировать символ «\» не требуется. Например, в конструкторе необходимо указывать "Инженер\Аналитик", а в коде значение пропишется, как ((/position in 'Инженер\\\Аналитик')).</p> <p>Для поля типа «lookup» поиск ведется по точному или частичному значению</p>	
pr	Присутствует	Присутствует	(/mail pr) – выполнится для пользователей, у которых указана электронная почта в карточке профиля
lt	Меньше	Меньше	(/organization_id lt 2)
gt	Больше	Больше	(/organization_id gt 3)

Изм.	Подп.	Дата

Услови е	Отображение в конструктор е правила	Описание	Пример в режиме кода
and	И	Одновременное выполнение всех перечисленных условий	(/userName eqic 'GLAZYRIN-S' and lastName eq 'Глазырин')
or	ИЛИ	Выполнение хотя бы одного из перечисленных условий	(/userName eqic 'GLAZYRIN-S' or lastName eq 'Глазырин')
!	Не равно	Операция отрицания	!(/appRoleName eq 'Бухгалтер') – выполнится для всех пользователей, у которых отсутствует роль «Бухгалтер»

В конструктор можно добавить настраиваемое поле типа «date» со следующими условиями:

- «gt» – больше;
- «ge» – больше или равно;
- «lt» – меньше;
- «le» – меньше или равно;
- «gtcd» – позднее текущей даты;
- «gecd» – позднее или равно текущей дате;
- «ltcd» – ранее текущей даты;
- «lecd» – ранее или равно текущей дате.

Описание действий по управлению настраиваемыми полями приведено в подразделе 2.7.

Фильтр может иметь символьный тип или числовой. Символьный необходимо заключать в одиночные кавычки, например, firstName eq 'Анатолий'. Для числового типа кавычки опускают, кроме числового типа в условии «массив».

Изм.	Подп.	Дата

Переоценка соответствия существующим правилам происходит в одном из следующих случаев:

- 1) Изменение атрибутов пользователя (выполняется переоценка конкретного пользователя).
- 2) Изменение атрибутов организации (выполняется переоценка всех пользователей).
- 3) Изменение правила автоматического назначения.

Возможные сообщения об ошибках:

- «Правило некорректно» – допущена синтаксическая ошибка в режиме «Код»;
- «Поле не заполнено» – поле не заполнено в режиме «Конструктор».

Если в условие правила добавлено неактуальное значение, отображается знак «!» и выводится предупреждение «Объект был удален!». Предупреждение отображается только в режиме конструктора. Варианты отображения значения удаленного объекта в условии:

- 1) Значение удаленного объекта отображается в поле. При этом возможно сохранение такого правила, но правило для данного условия не будет выполняться Комплексом. Если правило содержит другие условия с существующими объектами, то правило будет работать только для этих объектов.
- 2) Значение удаленного объекта отсутствует для отображения в поле. Поле считается не заполненным, при этом невозможно перейти в режим «Код» или сохранить правило. Необходимо удалить условие или указать новое значение.

Примечание. Если в условие правила автоназначения роли по наличию УЗР добавлено значение несуществующей в Комплексе формы ресурса, то задание такого правила невозможно, Комплекс отобразит ошибку «Ошибка при изменении роли».

2.17.1.2. Моделирование применения автоназначения ролей

Просмотр предварительного результата применения правила позволяет моделировать применения автоназначения ролей.

Изм.	Подп.	Дата

Для просмотра результата применения правила необходимо выполнить следующие действия:

- 1) Добавить хотя бы одно правило автоназначения, согласно подпункту 2.16.1.1.
- 2) Нажать кнопку «Предварительный результат применения правила». Откроется соответствующее окно, в котором расположены вкладки:
 - «Все владельцы роли» – список всех пользователей, которые будут обладать данной ролью в результате применения правила. Этот список включает как пользователей, у которых роль уже была назначена по данному правилу, так и тех, у кого она будет назначена впервые. При этом из общего количества пользователей будет исключено количество тех, у кого права доступа будут отозваны;
 - «Назначение роли» – список пользователей, которым роль будет назначена;
 - «Отзыв роли» – список пользователей, которым роль будет отозвана.
- 3) Нажать кнопку «Выгрузить в Excel» при необходимости.

Внизу окна «Предварительный результат применения правила» отображается статистика сколько пользователей будет изменено в результате применения правила.

Возможные ошибки:

- 1) «Ошибка в правиле автоназначения» – ошибка валидации.
- 2) «Для просмотра результатов нужно заполнить хотя бы одно правило» – отсутствует правило.

2.17.2. Настройка правила связывания

Правило связывания (далее – запрос `beneficiaryQuery`) настраивается в объекте «Учетная запись ресурса». Описание приведено в пункте 2.3.1. Запрос `beneficiaryQuery` выполняет поиск владельца для УЗ пользователя ПК Ankey IDM по УЗ из коннектора (не по форме ресурса). Правило связывания применяется только для синхронизации с ЦС.

Изм.	Подп.	Дата

Настройка правил разграничения доступа выполняется администратором в разделе «Правила SoD» меню «Админ. системы» и включает в себя следующие действия администратора:

- создание правила;
- редактирование правила;
- удаление правила;
- запуск правила;
- экспорт правил;
- импорт правил.

Правила SoD делятся на следующие типы:

- 1) Конструктор SoD-правил для ролей.
- 2) Конструктор SoD-правил для полномочий.
- 3) Доступность ролей пользователя по подразделению.
- 4) Несоответствие полномочий ролевой модели.
- 5) Несовместимость ролей.

2.17.3.1. Создание правила разграничения доступа

Для создания правила SoD администратору необходимо выполнить следующие действия:

- 1) Перейти в раздел «Правила SoD» меню «Админ. системы».
- 2) Нажать кнопку «Создать». Отобразится окно «Создание правила».
- 3) Заполнить обязательное поле «Название».
- 4) При необходимости заполнить необязательное поле «Описание».
- 5) Выбрать тип правила:
 - конструктор SoD-правил для ролей;
 - конструктор SoD-правил для полномочий;
 - несоответствие полномочий ролевой модели;
 - доступность ролей по подразделению;
 - несовместимость ролей.
- 6) Для типа «Конструктор SoD-правил для ролей» выполнить в конструкторе правил:

Изм.	Подп.	Дата

- задать хотя бы одно обязательное правило в разделе «Правило запрета роли»;
 - задать при необходимости правило в разделе «Правило исключения для указанных пользователей».
- 7) Для типа «Конструктор SoD-правил для полномочий» выполнить в конструкторе правил:
- задать хотя бы одно обязательное правило в разделе «Правило запрета полномочий»;
 - задать при необходимости правило в разделе «Правило исключения для указанных пользователей».
- 8) Для типа «Несоответствие полномочий ролевой модели» обязательно добавить хотя бы одно значение УЗР или подразделения.
- 9) Для типа «Доступность ролей по подразделению» обязательно добавить хотя бы одно значение подразделения.
- 10) Для типа «Несовместимость ролей» обязательно добавить хотя бы одну роль и к ней хотя бы одну несовместимую роль. К одной роли можно добавить несколько несовместимых ролей. Если добавленные роли являются вложенными, то при сохранении правила отобразится ошибка валидации о запрете создания правила несовместимости вложенных ролей.
- 11) Назначить ответственного за инциденты по названию роли. Поле «Роль ответственного за инциденты» обязательное для всех типов правил.
- 12) При необходимости установить флаг «Автоматически исправлять инциденты». Если установлен флаг, то при выявлении нарушения будет автоматически создан и исправлен инцидент. Если не установлен флаг, то инцидент будет создан после запуска правила. В этом случае ответственный за инциденты выполняет действия по его исправлению. Действия по исправлению инцидента приведены в пункте 2.1.12 документа «Руководство пользователя» (72410666.00054-02 96 01).
- 13) При необходимости установить флаг «Информировать о нарушениях в заявках». Флаг отвечает за проверку роли на нарушения по этому правилу

Изм.	Подп.	Дата

при выполнении запроса доступа. Подробнее о проверке роли на нарушения приведено в пункте 2.3.1 документа «Руководство пользователя» (72410666.00054-02 96 01).

14) Нажать кнопку «Создать».

Детали правила SoD отображаются на вкладках «Информация» и «Правило».

Инциденты будут созданы только на пользователей в выбранных объектах Комплекса после запуска правила. При одновременном выборе подразделения и УЗР, инциденты будут созданы только на пользователей, которые присутствуют как в выбранном подразделении, так и в УЗР.

Когда задано правило типа «Конструктор SoD-правил для ролей» на запрет родительской и вложенной роли, то при его запуске будет создан один инцидент на пользователей, у которых присутствуют эти роли. Подробнее про инциденты приведено в подразделе 2.9 документа «Руководство пользователя» (72410666.00054-02 96 01).

Описание действий для запуска правила приведено в подпункте 2.16.3.4.

Конструктор правил позволяет задавать условия в графической форме или в виде кода.

Конструктор в разделе «Правило запрета роли» позволяет создать правила запрета на основе:

- роли в ИС;
- роли.

Конструктор в разделе «Правило запрета полномочий» позволяет создать правила запрета на основе:

- наличия УЗР у пользователя;
- наличия полномочий УЗ у пользователя. Полномочия УЗ у пользователя задаются только в режиме кода.

Конструктор в разделе «Правило исключения для указанных пользователей» позволяет создать правила исключения на основе:

- атрибутов пользователя;
- назначенных пользователю ролей;
- имеющихся у пользователя УЗР;

Изм.	Подп.	Дата

– роли из ИС у пользователя.

Если не задано правило исключения, то созданные правила применяются для всех пользователей Комплекса.

Для задания правила администратору необходимо выполнить следующие действия:

- 1) Нажать «Открыть конструктор правил». Откроется окно конструктора.
- 2) Добавить условие. Значение должно соответствовать типу атрибута.
- 3) Нажать кнопку «Создать».

Управление условиями осуществляется с помощью следующих кнопок:

- «Объединить» – позволяет объединить условия в группы;
- «Разделить» – позволяет разделить группу условий;
- «Удалить» – позволяет удалить условия;
- «Добавить условие» – позволяет добавить условие.

Условия между собой можно объединять логическими операторами «И», «ИЛИ».

Для просмотра кода правила необходимо перейти в раздел «Код». В разделе «Код» также доступно редактирование условий правила. При возврате в раздел «Конструктор» код правила преобразуется в графической форме.

Поддерживаемые условные операторы приведены в таблице 2.38.

Таблица 2.38 – Поддерживаемые условные операторы

Поддерживаемые условные операторы	Код	Назначение
Равно	eq	Находит полное соответствие значению. Поддерживается для всех типов атрибутов
Не равно	!eq	Находит все объекты, которые не соответствуют значению. Поддерживается для всех типов атрибутов
Присутствует	pr	Возвращает значение «ИСТИНА» в случае, если значение атрибута найдено у объекта. Поддерживается для всех типов атрибутов
Отсутствует	!pr	Возвращает значение «ЛОЖЬ» в случае, если

Изм.	Подп.	Дата

Поддерживаемые условные операторы	Код	Назначение
		значение атрибута не найдено у объекта. Поддерживается для всех типов атрибутов
В иерархии	tree	Выполняет поиск значения по всем дочерним связанным объектам. Поддерживается только для атрибута «Подразделение»
Не в иерархии	!tree	Выполняет поиск значения только по текущему уровню объекта. Поддерживается только для атрибута «Подразделение»
Массив	in	Выполняет поиск среди перечисленных значений. Поддерживается для текстовых атрибутов
Содержит	co	Находит соответствие указанной части значения. Поддерживается для текстовых атрибутов
Равно без уч.рег.	eqic	Находит полное соответствие значению, не учитывая регистр. Поддерживается для всех типов атрибутов
Не равно без уч.рег.	!eqi c	Находит все объекты, которые не соответствуют значению, не учитывая регистр. Поддерживается для всех типов атрибутов

В конструктор можно добавить настраиваемое поле типа «date» со следующими условиями:

- «gt» – больше;
- «ge» – больше или равно;
- «lt» – меньше;
- «le» – меньше или равно;
- «gtcd» – позднее текущей даты;
- «gecd» – позднее или равно текущей дате;
- «ltcd» – ранее текущей даты;
- «lecd» – ранее или равно текущей дате.

Изм.	Подп.	Дата

Описание действий по управлению настраиваемыми полями приведено в подразделе 2.7.

Если в фильтр добавлено неактуальное значение, отображается знак и выводится предупреждение «Объект был удален!». Предупреждение отображается только в графическом режиме конструктора. Варианты отображения значения удаленного объекта в условии:

- 1) Значение удаленного объекта отображается в поле. При этом возможно сохранение такого правила, но правило для данного условия не будет выполняться Комплексом. Если правило содержит другие условия с существующими объектами, то правило будет работать только для этих объектов.
- 2) Значение удаленного объекта отсутствует для отображения в поле. Поле считается не заполненным, при этом невозможно перейти в режим «Код» или сохранить правило. Необходимо удалить условие или указать новое значение.

2.17.3.2. Редактирование правила разграничения доступа

Для редактирования правила SoD доступны все атрибуты, кроме типа правила. Администратору доступны все действия при редактировании, как при создании правила.

2.17.3.3. Удаление правила разграничения доступа

Для удаления правила SoD администратору необходимо выполнить следующие действия:

- 1) Отметить одно или несколько правил и нажать кнопку «Удалить». Откроется окно подтверждения удаления правила.
- 2) Исключить правила из списка удаляемых при необходимости, и нажать кнопку «Удалить».

Для просмотра удаленных правил необходимо выбрать в фильтре списка статус «Удалено».

Изм.	Подп.	Дата


2.17.3.4. Запуск правила разграничения доступа

Запуск выполняется только выбранных правил SoD через кнопки запуска и остановки. Обязательным условием для запуска правила является наличие ответственного за инцидент. Ответственный за инцидент указывается при создании/редактировании правила в соответствующем поле.

После запуска правила SoD в Комплексе создаются инциденты и они автоматически назначаются на ответственного. Если в Комплексе уже существует данный инцидент, он не будет создан повторно.

Комплекс позволяет задать расписание запуска правила.

Для этого необходимо выполнить следующие действия:

- 1) Выбрать правило.
- 2) Нажать кнопку . Откроется окно «Расписание запуска правила».
- 3) Задать следующие параметры:
 - даты начала и окончания – позволяют установить дату начала или окончания запуска правила. Если даты не заданы, то правило будет запускаться с текущего дня и до бесконечности;
 - флаг «Активность задания» – позволяет активировать задание, которое будет автоматически запускаться согласно расписанию. По умолчанию выключено;
 - флаг «Cron» – позволяет установить расписание в формате cron. По умолчанию установлено значение «0 0 * * 6 ?»;
 - флаг «Конкурентное выполнение» – позволяет задать очередность выполнения. По умолчанию выключено.
- 4) Нажать кнопку «Сохранить».

Статус последнего запуска, параметры расписания отображаются в деталях правила SoD на вкладке «Информация».

2.17.3.5. Экспорт правил разграничения доступа

Комплекс позволяет экспортировать правила разграничения доступа с типом «Несовместимость ролей» в форматах xls/xlsx.

Изм.	Подп.	Дата

Для экспорта правил необходимо в меню «Админ. системы»/«Правила SoD» выполнить следующие действия:

- 1) Нажать кнопку «Экспорт». Откроется окно «Экспорт правил SOD».
- 2) Выбрать тип правила «Несовместимость ролей».
- 3) Ввести название роли ответственного за инциденты при необходимости. Поле «Роль ответственного за инциденты» не является обязательным к заполнению. Если поле не заполнено, экспортируются правила по всем ответственным за инциденты.
- 4) Выбрать формат сохранения xls или.xlsx. Формат xls установлен по умолчанию.
- 5) Нажать кнопку «Сохранить».

В результате экспорта отобразится сообщение «Правила успешно экспортированы». На локальный компьютер экспортируется один файл в одном из выбранных форматов сохранения.

Экспортируемый файл включает в себя следующие листы:

- 1) «Несовместимость ролей» – содержит таблицу со списком выгруженных правил с типом «Несовместимость ролей» по указанной роли/ролям ответственного за инциденты. Таблица содержит следующие поля:
 - «autoResolve» – признак автоматического исправления инцидентов;
 - «description» – описание правила;
 - «name» – название правила;
 - "_id" – идентификатор объекта (objectId) в БД;
 - «checkBeforeRequestRole» – признак проверки ролей на нарушения;
 - «status» – состояние записи в БД, может принимать значения «active» и «deleted». Экспортируются записи только со статусом «active»;
 - «adminSodAppRoleName» – роль ответственного за инциденты.
- 2) «Несовместимые роли» – содержит таблицу со списком правил несовместимости ролей. Таблица содержит следующие поля:
 - «ruleName» – название правила разграничения доступа;
 - «mainRole» – название основной роли;
 - «incompatibleRole» – название несовместимой роли;

Изм.	Подп.	Дата

- «status» – статус записи. При экспорте статус записи всегда «active».

2.17.3.6. Импорт правил разграничения доступа

Комплекс позволяет импортировать правила разграничения доступа с типом «Несовместимость ролей» в форматах xls/xlsx.

В файле импорта листы с данными импорта обязательно должны быть названы «Несовместимость ролей» и «Несовместимые роли».

Лист «Несовместимость ролей» содержит следующие атрибуты, из которых «description» и "_id" не являются обязательными к заполнению:

- 1) «autoResolve» – признак автоматического исправления инцидентов.
- 2) «description» – описание правила.
- 3) «name» – название правила.
- 4) "_id" – идентификатор объекта (objectId) в БД. По наличию идентификатора можно определить создается или обновляется правило:
 - идентификатор присутствует – правило обновляется;
 - идентификатор отсутствует – правило создается;
- 5) «checkBeforeRequestRole» – признак проверки ролей на нарушения.
- 6) «status» – состояние записи в БД. Возможные значения:
 - «delete» – указывается для удаления правила;
 - «active» – указывается для создания или обновления правила;
- 7) «adminSodAppRoleName» – роль ответственного за инциденты.

Лист «Несовместимые роли» содержит следующие обязательные к заполнению атрибуты:

- 1) «ruleName» – название правила разграничения доступа.
 - 2) «mainRole» – название основной роли.
 - 3) «incompatibleRole» – название несовместимой роли.
 - 4) «status» – статус записи. Возможные значения:
 - «delete» – указывается для удаления несовместимости ролей в правиле;
 - «create» – указывается для создания несовместимости ролей в правиле.
- Для всех новых правил записи о несовместимых ролях должны быть в статусе «create»;

Изм.	Подп.	Дата

- «active» – указывается при наличии имеющихся несовместимостей ролей в правиле. Записи со статусом «active» отдельно не проверяются на наличие в БД.

Для импорта правил необходимо в меню «Админ. системы»/«Правила SoD» выполнить следующие действия:

- 1) Нажать кнопку «Импорт». Откроется окно «Импорт правил SOD».
- 2) Нажать кнопку «Добавить файл» и загрузить файл импорта. После успешной загрузки файла отображается статистика.
- 3) Нажать кнопку «Импортировать».

В результате импорта отобразится сообщение «Правила SoD успешно импортированы».

Перечень возможных предупреждений из-за ошибок в импортируемом файле:

- 1) Отсутствуют обязательные заголовки в файле.
- 2) В файле нет записей, кроме заголовков.
- 3) Найдены дубликаты имен правил (в файле).
- 4) Найдены дубликаты идентификаторов правил.
- 5) Найдены несовместимые роли без заданного для них правила.
- 6) Найдены идентификаторы несуществующих правил.
- 7) Найдены имена несуществующих или удаленных ролей.
- 8) Найдены незаполненные поля типа «Флаг».
- 9) Обязательное поле «status» не заполнено или содержит некорректное значение.
- 10) Не заполнено обязательное поле.
- 11) Найдены идентичные пары несовместимых ролей, имеющие разные статусы, для одного правила.
- 12) Найдены идентичные пары несовместимых ролей в разных правилах.
- 13) Правило не прошло валидацию по одной из стандартных политик.
- 14) Основная роль указана для создания несовместимостей в нескольких правилах.
- 15) Не указан идентификатор для удаляемого правила.
- 16) Найден идентификатор правила с неверным типом.

Изм.	Подп.	Дата

Для копирования деталей ошибки необходимо нажать на значок .

2.18. Настройка поиска

Комплекс обеспечивает поиск объектов в интерфейсе средствами внешней системы Opensearch.

Настройка поиска включает следующие действия администратора:

- 1) Настройка подключения к поисковой системе.
- 2) Настройка поискового индекса.
- 3) Настройка фильтра поиска заместителя.
- 4) Настройка фильтра поиска бенефициара при оформлении заявки.

2.18.1. Настройка подключения к поисковой системе

Настройка подключения к поисковой системе выполняется в файле `conf/search.connection.json`. Описание конфигурационного файла `search.connection.json` приведено в разделе 27 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Для настройки подключения к поисковой системе с включенной аутентификацией по логину и паролю необходимы данные авторизации. В конфигурационный файл `conf/search.connection.json` следует добавить обязательный параметр «auth» и следующие настройки:

- «type» – тип аутентификации. Значение по умолчанию «basic»;
- «principal» – имя пользователя;
- «password» – пароль пользователя.

Описание настройки через интерфейс Комплекса приведено в подразделе 4.9.

2.18.2. Настройка поискового индекса

Настройка поискового индекса Opensearch выполняется в файле `conf/search.settings.json`.

Описание конфигурационного файла `search.settings.json` и пример приведены в разделе 28 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.11.

После выполнения настроек необходимо запустить периодическое задание «reindexTask».

2.18.3. Настройка фильтра поиска заместителя

Настройка фильтра поиска заместителя выполняется в конфигурационном файле `conf/endpoint.lookup.substitution.json`. Фильтр работает только для заявок на назначение ролей. Фильтр поиска заместителя позволяет пользователям выбирать заместителя из определенных подразделений, необязательно своего.

Настройки файла `endpoint.lookup.substitution.json` приведены в разделе 16 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Если решение, предоставляемое Комплексом, не подходит под бизнес-логику заказчика, возможна разработка своего решения по настройке области видимости заместителей.

2.18.4. Настройка области видимости бенефициара в заявке на назначение ролей

Настройка области видимости бенефициара при оформлении заявки на назначение ролей выполняется в конфигурационном файле `conf/beneficiary.picker.provider.json`. Описание конфигурационного файла `beneficiary.picker.provider.json` приведено в разделе 10 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Настройка применяется также при поиске согласующих при переназначении заявки на пользователя.

Настройка области видимости бенефициара позволяет пользователям самостоятельно запрашивать роли в ИС для других пользователей.

Включение возможности добавлять пользователей в заявку и её переназначение можно выполнить через интерфейс Комплекса в подразделе 4.2.

Изм.	Подп.	Дата

2.18.5. Настройка фильтра выбора ролей согласующих при переназначении заявки

Настройка фильтра выбора ролей согласующих при переназначении заявки выполняется в конфигурационном файле `conf/approle.reassign.picker.provider.json`.

Фильтр работает только для заявок на назначение ролей.

Фильтр выбора ролей согласующих позволяет пользователям, имеющим соответствующие полномочия (администратор, оператор или пользователь с системной ролью), переадресовывать входящую заявку на выбранную роль. Полномочия по переназначению заявки на роль задаются соответствующими правами доступа при создании системной роли. Права доступа приведены в пункте 2.10.2 документа «Руководство пользователя» (72410666.00054-02 96 01).

Конфигурационный файл `conf/approle.reassign.picker.provider.json` содержит следующие параметры настройки:

```
{  
  "name": "<идентификатор области видимости>",  
  "enabled": <включение возможности выбора ролей согласующих при создании/переназначении заявки,  
  может принимать значения true или false>  
}
```

Комплекс поддерживает следующую область видимости ролей: «`appRoleFullAccess`». При настройке идентификатора области видимости «`appRoleFullAccess`» доступны для выбора все активные роли в Комплексе.

Если решение, предоставляемое Комплексом, не подходит под бизнес-логику заказчика, возможна разработка своего решения по настройке области видимости ролей.

В случае некорректной загрузки нестандартного решения или отсутствия сервиса с указанным названием, будет включена область видимости по умолчанию «`appRoleFullAccess`».

2.19. Настройка политики валидации

Политика валидации используется для определения условий, которым должно соответствовать значение атрибута объекта.

Изм.	Подп.	Дата

Комплекс проверяет вводимое значение на соответствие политике. Если вводимое значение не соответствует политике, Комплекс выдает ошибку валидации.

Комплекс поддерживает два типа политик:

- «PolicyFunction» вызывается при создании и любом изменении объекта (даже если изменился не тот атрибут, на который была настроена политика);
- «OptionalPolicyFunction» вызывается при заданном условии. По умолчанию это условие создания объекта или изменения значения валидируемого атрибута. Условие можно менять, переопределив соответствующий метод (подробнее в javadoc).

Примечание. Если при изменении состава политик типа «PolicyFunction» (настроенных на атрибут), существующие объекты, которые удовлетворяли старым политикам, перестали удовлетворять новым политикам, изменение состава политик следует выполнять с устранением всех ошибок валидации.

Все политики, кроме required, разрешают отсутствие значения у проверяемого атрибута (null).

Список и описание политик валидации приведен в таблице 2.39.

Таблица 2.39 – Список и описание политик валидации

Имя политики	Описание	Параметры конфигурации
required	Проверка, что значение не null и не пустая строка после удаления всех начальных и конечных пробелов	–
unique	Проверка, что нет другого объекта с таким же значением валидируемого	–

Изм.	Подп.	Дата

Имя политики	Описание	Параметры конфигурации
	атрибута. В случае, если значение строкового типа, проверка осуществляется без учета регистра	
cannotContainCharacters	Проверка, что в значении атрибута нет запрещенных символов	forbiddenChars – массив запрещенных строк. Например, <pre> "params": { "forbiddenChars": ["/"] } </pre>
maximumLength	Проверка максимальной длины	numChars – максимальная длина атрибута. Например, <pre> "params": { "numChars": 255 } </pre>
regexMatches	Проверка соответствия значения регулярному выражению	1) regex – регулярное выражение. 2) flags – флаг (необязательный). Допустимые значения: – \i – регистровая независимость; – \m – многострочный режим. Например, <pre> "params": { </pre>

Изм.	Подп.	Дата

Имя политики	Описание	Параметры конфигурации
		<pre>"regexp": "^([^\s\\@]+)\\@([^\s\\@]+)\$" }</pre>
enum	<p>Проверка, что значение среди множества допустимых</p>	<p>values – массив разрешенных значений. Например,</p> <pre>"params": { "values": ["active", "disabled"] }</pre>
resourceExists	<p>Проверка существования ресурса с заданным числовым идентификатором (oid) </p>	<p>resource – имя ресурса для проверки (обязательный параметр). Например, "params": { "resource": "managed/approle" }</p>
isInteger	<p>Проверка целочисленности значения</p>	–
multivaluedFieldsCombinationUnique	<p>Проверка, что комбинация ключевых атрибутов у полномочий уникальна. При попытке выделить УЗ с двумя и более полномочиями, состав ключевых</p>	<p>Переиспользовать политику нельзя</p>

Изм.	Подп.	Дата

Имя политики	Описание	Параметры конфигурации
	<p>полей которых совпадает (отличий либо нет вообще, либо есть только в неключевых полях), выделение завершится ошибкой валидации. Если данная ошибка произошла во время синхронизации УЗ, скорее всего некорректно настроен состав ключевых полей (не определяет уникальность привилегии)</p>	
isUiFormKeyValueMatchRegExPolicy	<p>Проверка, что значение uiformkey имеет корректный вид в соответствии с регулярным выражением: содержит только буквы любого регистра и цифры,</p>	—

Изм.	Подп.	Дата

Имя политики	Описание	Параметры конфигурации
	разделенные символом '/'	

2.20. Настройка валидации для пользовательской функции в системной роли

Пользовательская функция позволяет накладывать дополнительное ограничение на право доступа в системной роли. Описание системной роли приведено в разделе 2.10 документа «Руководство пользователя» (72410666.00054-02 96 01).

Для пользовательской функции можно настроить вывод собственного сообщения об ошибке.

Для этого администратору следует выполнить следующие действия:

- 1) Создать новый класс пользовательской функции авторизации, реализующий функцию `SystemRoleAuthzFunction`.
- 2) Создать системную роль и указать право доступа на какое-либо действие. Например, для объекта «Заявка» указать право доступа на создание множественной заявки на блокировку/разблокировку («request-batchstart»).
- 3) Указать в системной роли пользовательскую функцию авторизации.

Возможные варианты вывода сообщения об ошибке:

- 1) Если в `message` задан шаблон, то отображается текст из шаблона.
- 2) Если в `message` задан текст сообщения, отображается заданный текст.
- 3) Если `message` отсутствует или его значение отсутствует – отображается стандартное сообщение.

Пример добавления текста сообщения об ошибке в пользовательской функции:

```
@Override
public <RR extends Request> Promise<Boolean, ResourceException> applyAuthz(Context context, RR request)
{
    return new CustomAuthException("Текст ошибки").asPromise();
}
```

Изм.	Подп.	Дата

2.21. Управление фотографиями пользователей

Управление фотографиями пользователей включает в себя следующие события:

- загрузка;
- обновление;
- удаление.

Управление фотографиями пользователей выполняется администратором двумя способами:

- с помощью задач синхронизации. Применяется для управления фотографиями из внешней системы;
- в меню «Пользователи». Применяется для загрузки/редактирования фотографии при создании или редактировании пользователя. Подробнее приведено в подразделе 2.4 документа «Руководство пользователя» (72410666.00054-02 96 01).

Управление фотографиями для всех событий выполняется в следующей последовательности:

- 1) Настройка параметров отображения загружаемых фотографий в файле `conf/photo.json`. Описание приведено в подразделе 4.5.
- 2) Синхронизация фотографий из внешней системы.
- 3) Синхронизация пользователей. Загружаются данные пользователей и осуществляется создание/обновление/удаление связи каждого пользователя с фотографией.

Параметры конфигурационного файла `photo.json` приведены в разделе 13 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). При настройке следует учитывать ограничения параметров конфигурационного файла `photo.json`. В противном случае возникают ошибки «400: Bad Request» или «Размер кадра меньше 80x80!»

Фотографии загружаются/обновляются/удаляются в объекте «`managed/document`». При загрузке фотографии в объекте «`managed/document`»

Изм.	Подп.	Дата

дополнительно генерируется уменьшенная копия изображения, которая используется для предпросмотра.

При удалении в процессе запуска синхронизации фотографий оба документа: фотография и уменьшенная копия изображения удаляются. При удалении в процессе запуска синхронизации пользователей удаляется только уменьшенная копия изображения.

Объект «managed/document» включает следующие поля:

- «content» – в параметр записывается визуальное содержимое в формате UrlBase64;
- «name» – в параметр записывается название изображения (например, «image.jpeg»);
- «mimeType» – в параметр записывается MIME-тип изображения (например, «image/jpeg»);
- «size» – в параметр записывается размер изображения в байтах.

Каждая фотография в объекте «managed/document» уникальна. Идентификация фотографии происходит по совокупности полей объекта «managed/document».

В названии файла нельзя использовать пробелы и спецсимволы. Если их использование необходимо, то название должно быть перед отправкой закодировано стандартной URL кодировкой строк.

Для применения функциональности всех событий синхронизации администратору необходимо изменить настройки маппингов:

- 1) Синхронизации фотографий.
- 2) Синхронизации пользователей.

Пример маппинга синхронизации фотографий для всех событий (загрузки, обновления, удаления):

```
{
  "mapName": "newmappingphoto",
  "mapDesc": "new mapping photo",
  "mapType": "recon",
  "mapSource": "connector/newresource/userphoto",
  "mapTarget": "managed/document",
  "correlationQuery": "/name eqic '${filename}' and /mimeType eqic '${mimetype}' and /content eqic '${content}'",
  "mapBody": {
```

Изм.	Подп.	Дата

```

"properties": [{
  "source": "content",
  "target": "content"
},
{
  "source": "filename",
  "target": "name"
},
{
  "source": "mimetype",
  "target": "mimeType"
},
{
  "source": "",
  "target": "size",
  "transform": {
    "type": "text/javascript",
    "source": "source.content.length"
  }
}
],
"policies": [{
  "situation": "ABSENT",
  "action": "CREATE"
},
{
  "situation": "UNASSIGNED",
  "action": {
    "type": "text/javascript",
    "globals": {},
    "source": "var usage=(openidm.query('repo/documentusage', {'_queryFilter': '/document_id eq ' +
target._oid }, [usageType]).result[0].usageType);if (usage==='avatar') {action='DELETE'} else
{action='IGNORE'};action;"
  }
}, {
  "situation": "FOUND",
  "action": "UPDATE"
}
]
}
}

```

Пример маппинга синхронизации пользователей для создания, обновления, удаления связи с фотографиями:

```

{ "mapName": "newmappingname",
  "mapType": "recon",
  "mapDesc": "new mapping name",
  "mapSource": "connector/newresource/account",
  "mapTarget": "managed/user", "correlationQuery": "userName eqic '${username}'",
  "mapBody": {
    "policies": [
      {
        "action": "IGNORE",
        "situation": "CONFIRMED"
      }
    ]
  }
}

```

Изм.	Подп.	Дата

```

},
{
  "action":"UPDATE",
  "situation":"FOUND"
},
{
  "action":"CREATE",
  "situation":"ABSENT"
},
{
  "action":"EXCEPTION",
  "situation":"AMBIGUOUS"
},
{
  "action":"UNLINK",
  "situation":"MISSING"
},
{
  "action":"DELETE",
  "situation":"SOURCE_MISSING"
},
{
  "action":"IGNORE",
  "situation":"UNQUALIFIED"
},
{
  "action":"DELETE",
  "situation":"UNASSIGNED"
},
{
  "action":"IGNORE",
  "situation":"TARGET_IGNORED"
}
],
"properties":[
  {
    "source":"",
    "target":"userName",
    "transform":{
      "type":"text/javascript",
      "source":"source.username.toUpperCase();"
    }
  },
  {
    "source":"lastname",
    "target":"lastName"
  },
  {
    "source":"firstname",
    "target":"firstName"
  },
  {
    "source":"middlename",
    "target":"middleName"
  },
  {
    "source":"password",

```

Изм.	Подп.	Дата

```

    "target": "password"
  },
  {
    "source": "manager",
    "target": "managerId"
  },
  {
    "source": "",
    "target": "organization_id",
    "transform": {
      "type": "text/javascript",
      "source": "Number(\"${orgId}\");"
    }
  },
  {
    "target": "avatarId",
    "source": "userPhotoFileName",
    "transform": {
      "type": "text/javascript",
      "source": "if (source != null) { openidm.query('managed/document', { '_queryFilter': '/name eq \'' + source
+ '\", [_ouid]').result[0]._ouid;}"
    }
  }
]
}
}

```

Объект «mapBody» содержит блок «policies», в котором описываются возможные события синхронизации.

За связь пользователей с фотографиями отвечает следующая часть в маппинге:

```

{
  "target": "avatarId",
  "source": "userPhotoFileName",
  "transform": {
    "type": "text/javascript",
    "source": "if (source != null) { openidm.query('managed/document', { '_queryFilter': '/name eq \'' + source
+ '\", [_ouid]').result[0]._ouid;}"
  }
}

```

Когда поле «userPhotoFileName» из внешней системы указано в настройках маппинга пользователя, скрипт выполняет запрос документа из объекта «managed/document».

Подставляет его ouid, как «avatarId» у пользователя и осуществляет и создание/обновление/удаление связи каждого пользователя с фотографией.

За обновление фотографий отвечает следующая часть в настройке синхронизации фотографий:

Изм.	Подп.	Дата

```
{  
  "situation": "FOUND",  
  "action": "UPDATE"  
}
```

2.22. Управление лицензиями

Управление лицензиями ПК Ankey IDM включает следующие действия администратора:

- 1) Настройка параметров лицензирования в файле настроек conf/license.activator. Описание настроек приведено в приложении 1.
- 2) Активация пользовательской лицензии.
- 3) Активация лицензии на коннектор. Описание приведено в пункте 2.29.3.

Для активации пользовательской лицензии администратору необходимо выполнить следующие действия в меню «О продукте»:

- 1) Нажать кнопку «Активировать». Откроется форма для ввода ключа лицензии и email-адреса.
- 2) Ввести в обязательные поля ключ лицензии и email-адрес. В случае неверных данных отображаются сообщения об ошибках.
- 3) Нажать кнопку «Далее». После чего выполняется генерация запроса для формирования запроса на активацию.
- 4) Нажать кнопку «Скопировать», чтобы скопировать запрос на активацию.
- 5) Нажать ссылку «Система лицензирования». В новой вкладке откроется страница активации системы лицензирования. Для подключения требуется наличие доступа в Интернет.
- 6) В окне системы лицензирования вставить скопированный запрос на активацию и нажать кнопку «Активировать».
- 7) Вставить ключ активации, который был отправлен на указанный email-адрес и нажать кнопку «Активировать». Сформируется сертификат.
- 8) Нажать кнопку «Скопировать в буфер обмена» и перейти в ПК Ankey IDM для продолжения активации лицензии.
- 9) Нажать кнопку «Далее» и вставить сертификат в поле «Сертификат лицензии».

Изм.	Подп.	Дата

- 10) Нажать кнопку «Активировать». Отобразится окно «Продукт успешно активирован!».
- 11) Нажать кнопку «Закрыть» для закрытия окна. После чего в меню «О продукте» отобразится окно с актуальными данными о лицензии.

Обновление активированной лицензии ПК Ankey IDM при приобретении дополнительных пользовательских лицензий выполняется аналогично действиям выше.

Активацию лицензии в кластере достаточно выполнить на одном узле кластера. При этом требуется наличие одинаковых ключей, хранимых в каталоге «security» во всех узлах кластера. В случае изменения ключей, хранимых в каталоге «security», активированная лицензия не может быть прочитана на другом узле.

2.23. Управление незавершенными задачами

Меню «Незавершенные задачи» доступно администратору Комплекса и отображает список незавершенных задач.

Незавершенные задачи – это задачи создания/обновления/удаления УЗ в ЦС, завершенные с ошибкой. Незавершенные задачи автоматически повторяются периодическим заданием «failedsync». На вкладке «Информация» в блоке «Ошибка» выводится сообщение об ошибке и причина ошибки.

Список незавершенных задач может меняться в фоновом режиме при частом срабатывании периодического задания «failedsync». В подразделе «Незавершенные задачи» есть счетчик, на котором для контроля выполнения задач отображается количество незавершенных задач на текущий момент. Счетчик обновляется после совершения контекстного действия (удаление задач, запуск задач), либо по нажатию кнопки «Обновить». Если незавершенных задач нет, то счетчик отсутствует.

Рекомендуется отключить выполнение задания «failedsync» при работе с незавершенными задачами.

Примеры ситуаций, при которых могут возникать незавершенные задачи, приведены в таблице 2.40.

Изм.	Подп.	Дата

Таблица 2.40 – Примеры ситуаций, при которых могут возникать незавершенные задачи

Ситуация	Операция	Причины	Решение
Такой пользователь уже существует в ЦС	Создание УЗ (CREATE)	Не соблюдена последовательность: 1) Синхронизация существующих УЗ. 2) Выделение новых УЗ	1) Провести синхронизацию с ЦС, после чего привяжется действующая УЗ. 2) Удалить неактуальную УЗ в статусе «Создаётся» у пользователя
ЦС недоступна (отключена, истек срок действия пароля служебной УЗ, заблокирована служебная УЗ)	Все операции	Неверная настройка подключения Комплекса к ЦС	Исправить настройки ресурса в части параметров подключения к ЦС. Дождаться автоматического повтора упавших заданий
ЦС отвергла атрибуты создаваемой/изменяемой УЗ (не заполнены обязательные поля, пароль не прошел парольную политику на	1) Создание УЗ (CREATE). 2) Изменение УЗ (UPDATE)	Неправильная настройка на стороне Комплекса. Форма ресурсов и маппинги не учитывают логику	1) Исправить логику заполнения атрибутов УЗ. 2) Вручную изменить

Изм.	Подп.	Дата

Ситуация	Операция	Причины	Решение
стороне ЦС)		для атрибутов УЗ	атрибуты для проблемной УЗ на стороне Комплекса. Дождаться автоматическог о повтора
Валидация Комплекса отвергла атрибуты создаваемой/изменяемой УЗ (не заполнены обязательные поля)	1) Создание УЗ (CREATE). 2) Изменение УЗ (UPDATE)	Неправильная настройка на стороне Комплекса. Предзаполнение формы не учитывает логику для ее атрибутов	1) Исправить логику заполнения атрибутов УЗ. 2) Вручную изменить атрибуты для проблемной УЗ на стороне Комплекса. Дождаться автоматическог о повтора
Пользователь на стороне ЦС отсутствует (например, был удален напрямую)	1) Изменение УЗ (UPDATE). 2) Удаление УЗ (DELETE)	Возможно неправильная настройка синхронизации на стороне Комплекса или отключенная синхронизация с ЦС	Настроить мапинг синхронизации УЗ, указав для ситуации отсутствующего пользователя действие «Удаление» (DELETE).

Изм.	Подп.	Дата

Ситуация	Операция	Причины	Решение
			Дождаться выполнения синхронизации с ЦС, после чего на отсутствующую запись будет применено действие «Удаление» на стороне Комплекса

Список незавершенных задач содержит следующую информацию:

- «Тип» – тип операции (создание, обновление, удаление);
- «Приложение» – значение атрибута «resAccName» задачи синхронизации. В случае, когда пользователь еще не связан с какой-либо УЗР и значение атрибута «resAccName» не определить, отображается значение «Отсутствует»;
- «Учетная запись» – УЗ пользователя, для которой выполнялась задача;
- «Владелец» – ФИО пользователя, которому принадлежит УЗ;
- «Дата» – дата текущего события.

Незавершенные задачи в списке сортируются по дате события от наиболее ранних к наиболее поздним. Для настройки фильтрации незавершенных задач следует нажать кнопку «Фильтр» в правом углу строки поиска. Все настройки фильтра комбинируются между собой. Доступны следующие настройки фильтра:

- 1) Дата события.
- 2) Тип операции (создание, обновление, удаление).

Возможные действия администратора:

- 1) Просмотр списка событий незавершенных задач.
- 2) Редактирование события незавершенной задачи.

Изм.	Подп.	Дата

- 3) Поиск событий в списке по ФИО владельца.
- 4) Удаление события из списка незавершенных задач.
- 5) Выборочный запуск незавершенных задач.

Для просмотра события незавершенной задачи администратору следует выбрать событие из списка. После чего на вкладке «Информация» в зависимости от типа события отображаются следующие данные:

- наименования атрибутов при операции «создание»;
- наименования атрибутов и разница их значений при операции «обновление».

Для редактирования события незавершенной задачи администратору следует:

- 1) Выбрать событие из списка и нажать кнопку «Редактировать». Кнопка «Редактировать» доступна для задач типа «Создание» (CREATE).
- 2) Внести изменения в форме «Редактирование учетной записи».
- 3) Нажать «Сохранить».

Для удаления событий из списка незавершенных задач администратору следует выбрать одно или несколько событий и нажать кнопку «Удалить». Откроется окно «Удаление незавершенных задач».

Для подтверждения действия нажать кнопку «Удалить».

Удаление события необходимо выполнять после устранения причин, вызвавших ошибку, в противном случае событие будет повторно создано.

Для выборочного запуска незавершенных задач следует выбрать задачу из списка. После выбора задачи кнопка «Запустить» станет активна и на ней появится счетчик. После запуска выбранной задачи в списке незавершенных задач меняется ее тайминг (запись времени). Можно запустить несколько (более одной) задач, выделив их флагом. Когда задача, выбранная из списка незавершенных задач, запускается и выполняется, то список обновляется: из него удаляется выполненная задача.

2.24. Настройка автоматической генерации атрибутов

2.24.1. Генерация логина

Генерация логина выполняется автоматически, если при создании пользователя не заполнено поле «Имя пользователя».

Изм.	Подп.	Дата

Настройки генерации логина содержатся в конфигурационном файле `logingeneration.json`. Описание конфигурационного файла `logingeneration.json` приведено в разделе 21 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Файл расположен в каталоге установки Комплекса `ankey/conf/`.

Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.3.

2.24.2. Генерация идентификатора трудоустройства

При создании нового организационного присвоения Комплекс автоматически сгенерирует уникальный идентификатор, если он не был указан в поле «Идентификатор трудоустройства».

Настройки процесса генерации идентификаторов трудоустройства содержатся в конфигурационном файле `usrorgtabnumgeneration.json`. Подробное описание файла приведено в разделе 34 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.3.

2.24.3. Генерация пароля

Комплекс позволяет выполнить настройку генерации пароля.

Когда настройка включена, в интерфейсе Комплекса при смене пароля пользователя/УЗ пользователя, отобразится кнопка «Сгенерировать пароль».

Пароль будет генерироваться с учетом парольных политик. Когда парольные политики не заданы, пароль будет генерироваться из символов, которые указаны в параметре «`defaultRegExp`».

Настройки генерации пароля содержатся в конфигурационном файле `passwordgenerator.json`.

Описание конфигурационного файла `passwordgenerator.json` приведено в разделе 23 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Файл расположен в каталоге установки Комплекса `ankey/conf/`.

Изм.	Подп.	Дата

Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.3. ## Управление конфигурацией

Управление конфигурациями выполняется в меню «Админ. системы»/«Конфигурации» и включает в себя следующие действия администратора:

- создание конфигурации;
- экспорт конфигурации. Описание приведено в пункте 2.23.1;
- восстановление конфигурации. Описание приведено в пункте 2.23.2;
- импорт конфигурации. Описание приведено в пункте 2.23.3;
- удаление конфигурации. Описание приведено в пункте 2.23.4.

Создание конфигурации позволяет сохранить настройки следующих объектов Комплекса:

- config;
- connector;
- mapping;
- resaccount;
- resform;
- resource;
- restype;
- uiform;
- workflow.

Комплекс поддерживает следующие типы файлов:

- JSON – для объектов config, connector, mapping, resform, resource, uiform, restype;
- XML – для объекта workflow (BPMN-схемы).

При работе с конфигурацией выполняется политика валидации isUiFormKeyValueMatchRegExPolicy для uiFormKey, которая позволяет избежать создание дубликатов.

Описание политики валидации приведено в подразделе 2.18.

Для создания конфигурации администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать».



Изм.	Подп.	Дата

- 2) Ввести название конфигурации, и при необходимости задать описание.
- 3) Нажать кнопку «Далее».
- 4) Выбрать объекты для создания конфигурации.
- 5) Нажать кнопку «Сохранить».

Список конфигураций отображает следующие данные:

- 1) «Название».
- 2) «Дата создания».
- 3) «Статус». Статус может принимать следующие значения:
 - «Изменена» – конфигурация, часть объектов которой не соответствуют настройкам объектов Комплекса;
 - «Соответствует» – конфигурация, все объекты которой соответствуют настройкам объектов Комплекса.
- 4) «Кем создано».

Для просмотра деталей конфигурации необходимо выбрать конфигурацию в списке, после чего детали отобразятся в правой части экрана на вкладке «Информация». Детали конфигурации содержат следующую информацию:

- 1) Основные данные.
- 2) Список объектов конфигурации. Рядом с удаленным объектом отображается значок  и выводится предупреждение «Объект удален». Рядом с измененным объектом отображается значок  и выводится предупреждение «Объект изменен».

2.24.4. Экспорт конфигурации

Для экспорта конфигурации администратору необходимо выполнить следующие действия:

- 1) Выбрать нужную конфигурацию из списка. Экспорт конфигурации не поддерживает множественный выбор.
- 2) Нажать кнопку «Экспорт». Откроется окно «Экспорт конфигурации».
- 3) Дождаться окончания формирования конфигурации. Файл формируется в виде архива .zip.

Изм.	Подп.	Дата

- 4) Дождаться окончания стандартного сохранения zip-архива. Отобразится сообщение «Конфигурация сохранена». В случае ошибки отображается сообщение «Ошибка при экспорте». Для повтора экспорта следует нажать кнопку «Попробовать еще раз».

2.24.5. Восстановление конфигурации

Для восстановления конфигурации необходимо выполнить следующие действия:

- 1) Выбрать конфигурацию в статусе «Изменена».
- 2) Нажать кнопку «Применить конфигурацию». Откроется окно подтверждения.
- 3) Нажать кнопку «Применить». В случае успешной активации отобразится сообщение «Конфигурация успешно применена». Статус конфигурации изменится на «Соответствует». На вкладке «Информация» отобразятся данные о дате и инициаторе последнего применения. В противном случае отобразится сообщение «Ошибка применения конфигурации».

При восстановлении конфигурации объектов «resform» и «mapping» следует учесть:

- 1) Нельзя восстановить конфигурацию, которая содержит удаленные объекты «resform». При попытке восстановить отобразится сообщение «Ошибка применения конфигурации».
- 2) Нельзя восстановить конфигурацию с объектом «mapping», который имеет те же характеристики (по совокупности параметров «mapType», «mapSource» и «mapTarget»), что и объект «mapping» текущей конфигурации. Для восстановления такого объекта необходимо сначала удалить «mapping», а затем восстановить конфигурацию с этим объектом.

2.24.6. Импорт конфигурации

При подготовке файла импорта конфигурации следует учитывать следующее:

- 1) Название папок в zip-архиве должно соответствовать названию импортируемых объектов. Список объектов приведен в подразделе 2.23.

Изм.	Подп.	Дата

- 2) Каждая папка содержит отдельные файлы на объект. Если какой-то тип объектов не импортируется, то соответствующую папку не требуется добавлять в zip-архив.

Для импорта конфигурации администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Импорт». Откроется окно «Импорт конфигурации».
- 2) Заполнить уникальное значение обязательного поля «Название». Если введено не уникальное значение, отображается «Объект уже существует».
- 3) Нажать кнопку «Добавить файл». На загрузку файла действуют следующие ограничения:
 - можно импортировать только один файл;
 - доступный тип файла: zip;
 - размер файла не более 10 Мб.
- 4) После добавления файла отобразится сообщение «Файл загружен без ошибок». Ниже отображаются импортируемые объекты конфигурации.
- 5) Нажать кнопку «Импортировать». Отобразится сообщение «Конфигурация успешно импортирована». В случае ошибки отображается сообщение «Ошибка при импорте». Для повтора импорта следует нажать кнопку «Попробовать еще раз». Конфигурация импортируется без применения. Восстановление конфигурации приведено в пункте 2.23.2.

Перечень возможных предупреждений при импорте конфигурации:

- 1) Объект не прошел валидацию по политике.
- 2) Сообщение «Не поддерживаемый тип файла в объекте» отображается, если файл имеет неверный тип или содержит ошибки открытия архива (например, на архив установлен пароль, нарушена целостность архива).
- 3) Сообщение «Ошибка при загрузке» отображается, если объекты конфигурации внутри файла не соответствуют структуре конфигурации.

После импорта объектов «workflow» рабочие потоки отображаются в меню «Админ. системы»/«Бизнес-процессы». Описание рабочих потоков приведено в подразделе 2.6.

Изм.	Подп.	Дата

При импорте конфигурации объекта «resource» следует учитывать следующие особенности:

- 1) Настройки ресурсов типа «connectorServer» должны быть импортированы первыми. Затем остальные ресурсы могут быть перенесены при последующем импорте. Невозможно перенести все одновременно в одном архиве.
- 2) Импорт ресурсов следует осуществлять только после создания коннекторов и объектов «resType».
- 3) Файл импорта обязательно должен содержать «resTypeName» и «resName».
- 4) В Комплексе должен существовать тип ресурса «resType» с таким же именем, как указанное имя типа ресурса «resTypeName».
- 5) В Комплексе должен существовать ресурс «resource» с таким же именем, как указанное имя ресурса «resName».
- 6) Если в файле ресурса указан атрибут «connectorName», то в Комплексе должен быть коннектор с соответствующим названием.
- 7) Если в файле ресурса указан атрибут «connectorServerName», то в Комплексе должен быть ресурс с указанным именем сервера коннекторов.
- 8) Значения атрибутов «resName» – название ресурса, «resFormName» – имя формы ресурса, «defaultAccountPolicyName» – название типа УЗ пользователей в файле импорта должны соответствовать существующим в Комплексе.

В противном случае при импорте конфигурации возникает ошибка.

2.24.7. Удаление конфигурации

Для удаления конфигурации администратору необходимо выполнить следующие действия:

- 1) Выбрать нужную конфигурацию из списка. Поддерживается множественный выбор.
- 2) Нажать кнопку «Удалить». Откроется окно подтверждения «Удаление конфигураций».

Изм.	Подп.	Дата

- 3) Нажать кнопку «Удалить» в окне подтверждения. В случае успешного удаления отобразится сообщение «Конфигурации успешно удалены». В случае ошибки отображается сообщение «Ошибка при удалении конфигураций». Для повтора удаления следует выполнить действия сначала.

2.25. Настройка импорта ролей через коннектор «Excel Matrix Upload»

Для настройки импорта ролей через коннектор «Excel Matrix Upload» необходимо внести в конфигурационный файл `excelexport.json` следующие изменения:

```
{
  "_id": "excelexport",
  "characterEncoding": "DEFAULT",
  "rbacConfig": {
    "directoryPath": "/home/vagrant/ankey/samples",
    "importRbacModelTaskId": "schedule-excel-import-task",
    "reconRbacModelTaskId": "schedule-excel-target-recon"
  }
}
```

Настройка содержит следующие обязательные параметры:

- «`directoryPath`» – путь к директории с `xls/xlsx` файлами, прописанный в настройках коннектора;
- «`importRbacModelTaskId`» – идентификатор периодического задания для считывания данных из `xls/xlsx` файла и записи их в `rf_` таблицу;
- «`reconRbacModelTaskId`» – идентификатор периодического задания для создания ролей на основе данных в `rf_` таблице.

2.26. Настройка для отображения столбцов, полей фильтрации и поиска в интерфейсе

Настройки применяются в соответствующих разделах интерфейса Комплекса, где данные представлены в виде таблиц. Содержимое списков данных, доступных в самообслуживании веб-консоли, зависит от выбранного меню. Описание интерфейса приведено в пункте 1.2.2 документа «Руководство пользователя» (72410666.00054-02 96 01).

Изм.	Подп.	Дата

Некоторые настройки могут быть недоступны для изменения в зависимости от конкретной таблицы со списком данных.

Настройки выполняются в веб-консоли «Swagger UI» через REST API Комплекса методом PUT через `/endpoint/ui.form/{resourceId}`. При запросе методом PUT, если форма не существует, она будет создана. В ином случае выполняется обновление.

Основные параметры запроса:

- «resourceId» – идентификатор формы интерфейса. Обязательный параметр;
- «uiFormKey» – уникальный ключ формы интерфейса, к которому будут привязаны настройки поиска, фильтрации. Обязательный параметр;
- «mainObject» – управляемый объект, обязательный параметр;
- «description» – описание раздела, где будет применяться форма;
- «type» – тип формы интерфейса, обязательный параметр.

Настройка `endpoint/ui.form` выполняется в следующих параметрах:

- 1) «fields» – настройка отображения полей в таблице. Можно добавлять поля, которые есть в `ui.managed.json`, поля БД, а так же любые настраиваемые поля, за исключением полей с типом «lookup». Можно изменять отображение столбцов: добавлять новые, скрывать стандартные, менять порядок. Порядок отображения полей в таблице зависит от порядка следования перечисленных полей. В интерфейсе столбцы будут отображаться слева направо.
- 2) «filters» – настройка списка полей фильтрации. Можно добавлять любые поля любого типа, кроме виртуальных. В отдельных случаях допустимо добавлять собственные виртуальные поля для настройки переопределения параметров.
- 3) «search» – настройка полей полнотекстового поиска. Можно добавлять любые поля, кроме виртуальных, имеющие тип «textField», «lookup», «multilookup» или «enum». Также любые параметры пользователя, в том числе настраиваемые поля, которые поддерживают полнотекстовый поиск.

Изм.	Подп.	Дата

Для использования поля в фильтрах и поиске, его необходимо сделать поисковым. Для этого нужно установить параметр «searchable» в значение «true». Определение поискового поля происходит предварительно в конфигурационном файле extend.json.

Перевод для заголовка столбца задается в файлах для русской локализации translation_ru.properties и англоязычной локализации translation_en.properties. Файлы перевода расположены в директории «localization/i18n». Для русскоязычной локализации требуется выполнить конвертацию текста в кодировку JS/JAVA. Для изменения заголовка существующего столбца следует изменить его перевод по умолчанию.

Перечень таблиц со списками данных и их настройки приведены в таблице 2.41.

Таблица 2.41 – Перечень таблиц со списками данных и их настройки

Название	resourceId	uiFormKey	Отображаемые поля в таблице (fields)	Поля, по которым выполняется поиск (search)	Поля, по которым выполняется фильтрация (filters)
Таблица со списком пользователей в меню «Пользователи»	users-list	users/list	fullName, username, mail, position, organizationName, accountStatus	lastName, firstName, middleName, username, position	organization_id

Изм.	Подп.	Дата

Название	resourceId	uiFormKey	Отображаемые поля в таблице (fields)	Поля, по которым выполняется поиск (search)	Поля, по которым выполняется фильтрация (filters)
Таблица со списком заявок в меню «Входящие»/«Заявки»	request-incoming	request/incoming	_oid, beneficiaryFullName, requesterFullName, displayType, creationDate, completionDate	firstName, middleName, lastName	displayType, displayStatus, creationDate, completionDate, roleOrInfSystemFilter
Таблица со списком заявок в меню «Журнал заявок»	request-workflow	request/workflow	_oid, beneficiaryFullName, requesterFullName, displayType, displayStatus, creationDate, completionDate	firstName, middleName, lastName	displayType, displayStatus, creationDate, completionDate, roleOrInfSystemFilter
Таблица со списком заявок в меню «Мои	request-own	request/own	_oid, beneficiaryFullName,	firstName, middleName	displayType, displayStatus, creationDate,

Изм.	Подп.	Дата

Название	resourceId	uiFormKey	Отображаемые поля в таблице (fields)	Поля, по которым выполняется поиск (search)	Поля, по которым выполняется фильтрация (filters)
заявки»			requesterFullName, displayType, creationDate, completionDate, displayStatus	name, lastName	completionDate, roleOrInfSystemFilter
Таблица со списком заявок в меню «Входящие»/«История»	request-history	request/history	_oid, beneficiaryFullName, requesterFullName, displayType, displayStatus, creationDate, completionDate	firstName, middleName, lastName	displayType, displayStatus, creationDateHistoryFilter, roleOrInfSystemFilter
Таблица со списком ролей в меню «Роли»	approle-list	approle/list	appRoleName, roleName, appRoleType	appRoleName, roleName	is_id
Таблица со списком заявок в меню	request-aggregate	request/aggregate	number, requesterFullName,	firstName, middleName	creationDate, completionDate

Изм.	Подп.	Дата

Название	resourceId	uiFormKey	Отображаемые поля в таблице (fields)	Поля, по которым выполняется поиск (search)	Поля, по которым выполняется фильтрация (filters)
«Входящие»/«Агрегированные заявки»			displayType, creationDate, completionDate	name, lastName	
Таблица со списком заявок в разделе «Агрегированные заявки»/«Атомарные заявки»	request-nested	request/nested	appRoleName, beneficiaryFullName, accounts, position, organizationName, stageName	firstName, middleName, lastName	creationDate, completionDate
Таблица со списком ИС при запросе доступа	request-is	request/is	недоступно к редактированию	name, description	настройка отсутствует
Таблица со списком ролей при запросе доступа	request-roles	request/roles	organizationName	appRoleName, appRoleDesc	настройка отсутствует
Таблица со списком подразделений в меню	organization-list	organization/list	code, name, parentOrganizationName	name, code	настройка отсутствует

Изм.	Подп.	Дата

Название	resourceId	uiFormKey	Отображаемые поля в таблице (fields)	Поля, по которым выполняется поиск (search)	Поля, по которым выполняется фильтрация (filters)
«Подразделения»					
Таблица со списком ИС в меню «Инф.системы»	is-list	is/list	name, description	name, description	name
Таблица со списком УЗ в меню «Пользователи»/«Учетные записи»	users-account-list	users/accounts/list	resAccName, displayField, accountType, createDate, accountPolicyName, status	displayField	accountPolicyId, resaccount_id, status

После выполнения настроек фильтрации необходимо запустить периодическое задание «reindexTask», выполнив полный или частичный запуск для объекта «informationSystemRequestAppRoles».

2.26.1. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Пользователи/Учетные записи»

Комплекс позволяет настроить отображение столбцов в подразделе «Учетные записи» в меню «Пользователи». Можно изменять отображение столбцов: добавлять новые, скрывать стандартные, менять порядок.

Пример REST-запроса для отображения стандартных столбцов:

```
{
  "uiFormKey": "users/accounts/list",
```

Изм.	Подп.	Дата

```

"mainObject":"orchestration",
"description":"Таблица со списком УЗ пользователя",
"type":"table",
"fields":[
  "resAccName",
  "displayField",
  "accountType",
  "createDate",
  "accountPolicyName",
  "status"
],
"filters":[
  "accountPolicyId",
  "resaccount_id",
  "status"
],
"search":[
  "displayField"
]
}

```

Для настройки можно использовать все поля, доступные в `ui.managed.json` для объекта «orchestration».

Основные варианты полей:

- «resAccName» – название УЗР;
- «displayField» – отображаемое имя УЗ;
- «createDate» – дата создания УЗ;
- «accountType» – признак УЗ, например, основная или дополнительная;
- «status» – статус УЗ пользователя в УЗР;
- «beneficiar_objectid» – идентификатор бенефициара;
- «resaccount_id» – идентификатор УЗР;
- «resAccDesc» – описание УЗР;
- «accountPolicyId» – идентификатор УЗ;
- «accountPolicyName» – тип УЗ, например, базовая.

Общее описание настройки приведено в подразделе 2.25.

2.26.2. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Пользователи»

Комплекс позволяет настроить отображение столбцов в меню «Пользователи». Можно изменять отображение столбцов: добавлять новые, скрывать стандартные, менять порядок.

Изм.	Подп.	Дата

Пример REST-запроса для отображения стандартных столбцов:

```
{
  "uiFormKey": "users/list",
  "mainObject": "managed/user",
  "type": "table",
  "fields": [
    "fullName",
    "userName",
    "mail",
    "position",
    "organizationName",
    "accountStatus"
  ],
  "filters": [
    "organization_id"
  ],
  "search": [
    "lastName",
    "firstName",
    "middleName",
    "userName",
    "mail",
    "position"
  ]
}
```

Общее описание настройки приведено в подразделе 2.25.

2.26.3. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Входящие», «Мои заявки», «Входящие/История», «Журнал заявок»

Комплекс позволяет настроить отображение столбцов в меню «Входящие», «Мои заявки», «Входящие/История», «Журнал заявок». Можно изменять отображение столбцов: добавлять новые, скрывать стандартные, менять порядок.

Пример REST-запроса для отображения стандартных столбцов в меню «Журнал заявок»:

```
{
  "uiFormKey": "request/workflow",
  "mainObject": "managed/request",
  "type": "table",
  "description": "Отображение полей просмотра журнал заявок",
  "fields": [
    "_ouid",
    "beneficiaryFullName",
    "displayType",
    "displayStatus",
  ]
}
```

Изм.	Подп.	Дата

```

"creationDate"
],
"filters": [
  "displayType",
  "displayStatus",
  "creationDate",
  "roleOrInfSystemFilter",
  "requestStageFilter"
],
"search": [
  "firstName",
  "lastName",
  "middleName"
]
}

```

В параметр «filters» можно добавлять собственные виртуальные поля для настройки переопределения параметров отображения поля в фильтре через параметр «extends». Например, для настройки переопределения параметров отображения поля в фильтре необходимо создать виртуальное поле (например, displayTypeFilter), указать поля «extends»:«переопределяемое поле» и «fieldType»:«тип переопределяемого поля» или любой другой.

Пример настройки переопределения параметров отображения поля в фильтре:

```

"displayTypeFilter": {
  "extends": "displayType",
  "fieldType": "status"
},

```

Общее описание настройки приведено в подразделе 2.25.

2.26.4. Настройка для отображения столбцов, полей фильтрации и поиска в меню «Роли»

Комплекс позволяет настроить отображение столбцов в меню «Роли».

Пример REST-запроса для отображения стандартных столбцов:

```

{
  "uiFormKey": "approle/list",
  "mainObject": "managed/approle",
  "type": "table",
  "fields": [
    "appRoleName",
    "isName",
    "appRoleType"
  ],
  "filters": [

```

Изм.	Подп.	Дата

```

"is_id"
],
"search": [
  "appRoleName",
  "isName"
]
}

```

Общее описание настройки приведено в подразделе 2.25.

2.26.5. Настройка полей фильтрации и поиска по спискам ИС, ролей в меню «Пользователи», «Мой профиль» при запросе доступа

Поля для поиска и фильтрации при запросе доступа определяются через endpoint/ui.form.

Комплекс не поддерживает обновление отображения столбцов, за которые отвечает параметр fields.

Пример настройки полей поиска и фильтрации для ИС:

```

{
  "uiFormKey": "request/is",
  "mainObject": "managed/is",
  "description": "Таблица со списком ИС при запросе доступа",
  "type": "table",
  "filters": [
    "name"
  ],
  "search": [
    "name"
  ]
}

```

Пример настройки полей поиска и фильтрации для ролей:

```

{
  "uiFormKey": "request/roles",
  "mainObject": "managed/approle",
  "description": "Таблица со списком ролей при запросе доступа",
  "type": "table",
  "filters": [
    "appRoleName"
  ],
  "search": [
    "appRoleName"
  ]
}

```

Общее описание настройки приведено в подразделе 2.25.

Изм.	Подп.	Дата

2.27. Настройка полей экспорта/импорта объектов пользователи/пользователь-роль/дополнительные трудоустройства

Настройка полей экспорта/импорта объектов пользователи/пользователь-роль/дополнительные трудоустройства выполняется для следующих атрибутов:

- линейный руководитель. В файле экспорта/импорта по умолчанию поле «manager_userName» на листе «Пользователи»;
- основное подразделение. В файле экспорта/импорта по умолчанию поле «organization_code» на листе «Пользователи»;
- владелец роли – пользователь, которому будет назначена роль при создании связки роли с пользователем или пользователь, кому принадлежит роль. В файле экспорта/импорта по умолчанию поле «usr_userName» на листе «Пользователь-Роль»;
- идентификатор трудоустройства. В файле экспорта/импорта по умолчанию поле «usrorg_tabNum» на листе «Пользователь-Роль»;
- дополнительное подразделение. В файле экспорта/импорта по умолчанию поле «organization_code» на листе «Доп. трудоустройства».

Настройка полей выполняется в конфигурационном файле conf/userexportimport.json. Описание конфигурационного файла userexportimport.json приведено в разделе 33 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). Настройка файла может быть выполнена в интерфейсе Комплекса. Описание приведено в подразделе 4.6.

Настраиваемые поля могут содержать несколько значений. Можно заменить значения на другие, но нельзя их удалить или оставить пустыми.

Если во время импорта будут нарушены значения, по которым определяется уникальность, Комплекс отобразит ошибку.

Изм.	Подп.	Дата

2.28. Управление коннекторами

Для управления коннекторами через меню «Настройка ресурсов/Коннекторы» администратор выполняет следующие действия:

- создание коннектора;
- редактирование коннектора;
- активация коннектора.




Список коннекторов отображает следующую информацию:

- «Название» – название коннектора;
- «Описание» – описание коннектора;
- «Класс» – класс коннектора;
- «Статус активации» – статус активации лицензии;
- «Версия» – версия коннектора;
- «Техническая поддержка» – тип технической поддержки («базовая», «гарантийная» или «отсутствует») и сроки гарантийного обслуживания. Лицензии с истекшим сроком гарантийного обслуживания отображаются красным цветом и при наведении выводится подсказка «Срок действия истек», лицензии с действительным сроком гарантийного обслуживания отображаются зеленым цветом и при наведении выводится подсказка «Действительна».

2.28.1. Создание коннектора

Для создания коннектора администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать» в меню «Настройка ресурсов/Коннекторы». Отобразится окно создания коннектора в режиме редактора кода.
- 2) Внести значение в виде кода в текстовое поле.
- 3) Нажать кнопку «Создать».




Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Изм.	Подп.	Дата

2.28.2. Редактирование коннектора

Для редактирования коннектора администратору необходимо выполнить следующие действия:

- 1) В списке коннекторов выбрать коннектор.
- 2) Нажать кнопку «Редактировать». Отобразится окно редактирования коннектора в режиме редактора кода.
- 3) Внести необходимые изменения в текстовое поле.
- 4) Нажать кнопку «Сохранить».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

2.28.3. Активация коннектора

В меню «Настройка ресурсов»/«Коннекторы» выполняется активация лицензии на один класс коннектора.

Возможные статусы активации:

- «Активирован»;
- «Не активирован»;
- «Активация не требуется». Статус отображается для коннекторов, разработанных сторонними организациями.

Для активации лицензии на коннектор администратору необходимо выполнить следующие действия:

- 1) В списке коннекторов выбрать коннектор со статусом «Не активирован».
- 2) Нажать кнопку «Активировать».
- 3) Пройти активацию, выполнив аналогичные действия, приведенные в подразделе 2.20.

После активации лицензии статус активации принимает значение «Активирован». В случае, когда список содержит несколько коннекторов одного класса, достаточно активировать лицензию только одного коннектора. При активации коннектора с одним классом у остальных коннекторов автоматически статус активации принимает значение «Активирован».

Изм.	Подп.	Дата

Активация коннектора, у которого не задан атрибут «productName», невозможна.

Активация лицензии на коннектор в кластере аналогична активации пользовательской лицензии для кластера. Описание приведено в подразделе 2.20.

2.29. Дополнительные настройки в интерфейсе заявок на запрос ролей

Комплекс позволяет настроить отображение полей в интерфейсе заявки при запросе ролей на форме выбора трудоустройств пользователя.

Настройка выполняется в конфигурационном файле extend.json для объектов «managed/user» и «managed/usrorg» в параметре «customDisplays» для следующих атрибутов:

- «viewBeneficiaryFirstStepWizard» - поля с дополнительной информацией под ФИО пользователя;
- «viewBeneficiaryPickerTable» - поля таблицы списка пользователей;
- «viewModal» - поля в профиле пользователя;
- «wizardEmploymentExtended» - поля с дополнительной информацией в списке трудоустройств;
- «wizardEmploymentPopUp» - поля подсказки с дополнительной информацией по трудоустройству.

Пример настройки отображения полей по умолчанию:

```
{
  "managed/user": {
    "customDisplays": {
      "viewBeneficiaryFirstStepWizard": [
        "position", "organizationName", "mail", "phone"
      ],
      "viewBeneficiaryPickerTable": [
        "fullName", "position", "organizationName", "mail", "phone"
      ],
      "viewModal": [
        "userName", "accountStatus", "lastName", "firstName", "middleName", "mail", "organizationTree",
        "managerFullName", "position"
      ]
    }
  },
  "managed/usrorg": {
    "customDisplays": {
```

Изм.	Подп.	Дата

```

"wizardEmploymentExtended": [
],
"wizardEmploymentPopup": [
"organizationTree", "managerFullName"
]
}
}
}
}

```

Для настройки можно использовать все поля, доступные в `ui.managed.json` для объектов «managed/user» и «managed/usrorg».

2.30. Управление рисками

Меню «Управление рисками» доступно только администратору Комплекса. Раздел «Маркеры риска» отображает редактор кода для конфигурации рисков. Маркеры риска помогают оценивать риски, связанные с доступом, управлением, обработкой и хранением информации, а также с выполнением задач. Значение риска будет всегда не отрицательным числом.

Комплекс позволяет настроить уровни для маркеров риска у следующих объектов:

- 1) Роль.
- 2) Пользователь.

Пример настройки уровней для маркеров риска в редакторе кода:

```

{
  "_id": "risk",
  "aprole": {
    "highLevel": 100,
    "middleLevel": 50,
    "defaultLevel": 0
  },
  "user": {
    "highLevel": 300,
    "middleLevel": 100
  }
}

```

Маркеры риска роли – это индикаторы роли, которые указывают значимость роли, связанную с выполнением функциональных обязанностей, предоставляемых ролью.

Маркеры риска роли разделяются два показателя:

Изм.	Подп.	Дата

- 1) Риск роли – отражает собственный риск, связанный непосредственно с данной ролью.
- 2) Общий риск роли – представляет собой совокупный риск, включающий в себя как собственный риск роли, так и риски всех вложенных в нее ролей. Общий риск роли рассчитывается на всех уровнях вложенности.

Риск роли можно задать или изменить в необязательном поле «Риск» как при создании/редактировании роли, так и впоследствии следующими способами:

- 1) Вручную через интерфейс. В случае введения неверного значения Комплекс отобразит ошибку «Допустимый диапазон от 0 до 2 147 483 647».
- 2) При импорте ролей. Значение риска роли указывается в поле «riskValue». Описание импорта приведено в пункте 2.5.6 документа «Руководство пользователя» (72410666.00054-02 96 01).
- 3) При запуске периодического задания «updateRiskValueTask». Описание периодического задания «updateRiskValueTask» приведено в пункте 2.4.5.

Маркеры риска роли отображаются по уровню риска для каждой роли.

Для уже существующих в Комплексе ролей оба поля: общий риск и риск роли будут заполняться значением – 0. По умолчанию у всех новых ролей риск будет равен значению «defaultlevel» указанному в настройках.

Уровни риска роли задаются в блоке «approle» с помощью следующих параметров:

- «defaultlevel»;
- «middlelevel»;
- «highlevel».

При изменении уровней для маркеров риска обновление риска/общего риска роли происходит автоматически.

Для просмотра риска/общего риска существующих ролей следует выбрать:

- 1) Меню «Пользователи»/«Роли» столбец «Общий риск».
- 2) Меню «Роли»:
 - столбец «Общий риск»;
 - вкладка «Информация» в деталях роли;

Изм.	Подп.	Дата

- вкладка «Вложенные роли» в столбце «Общий риск»;
- вкладка «История изменений роли».

3) Меню «Мои заявки» в деталях заявки на изменение роли.

Риск пользователя – это сумма общих рисков ролей, назначенных этому пользователю. Если суммарный риск превысит максимальное допустимое значение (2 147 483 647), то риск пользователя будет ограничен этим максимальным значением. Если пользователь получит дополнительные роли с риском, его риск не сможет превысить установленный предел.

Риски пользователя будут пересчитываться при назначении/отзыве роли и изменении риска роли. Для обновления риска пользователя следует запустить периодическое задание «userRiskActualizationTask». Описание периодического задания «userRiskActualizationTask» приведено в пункте 2.4.5.

Маркеры риска пользователя отображаются по уровню риска для каждого пользователя.

Уровни риска пользователя задаются в блоке «user» с помощью следующих параметров:

- «middlelevel»;
- «highlevel».

Для просмотра риска пользователя следует выбрать:

- 1) Меню «Мой профиль» и профиль пользователя в меню «Пользователи»:
 - иконка с индикатором цвета;
 - подраздел «Карточка пользователя»;
- 2) Меню «Пользователи» столбец «Риск».

Примеры значений уровней риска/общего риска, которые могут возникать, приведены в таблице 2.42, где x – это текущее значение риска пользователя/риска роли.

Таблица 2.42 – Примеры значений уровней риска

Диапазон значений	Уровень риска	Цвет иконки
$\geq \text{highlevel}$	Высокий	Красный
$\text{middlelevel} \leq x < \text{highlevel}$	Средний	Желтый

Изм.	Подп.	Дата

Диапазон значений	Уровень риска	Цвет иконки
$0 < x < \text{middlelevel}$	Минимальный	Зеленый
0	Риск по умолчанию	Серый

Изм.	Подп.	Дата

3. НАСТРОЙКА СОГЛАСОВАНИЙ

Просмотр настройки согласований выполняется в меню «Админ. системы»/«Настройка согласований».

Раздел «Настройка согласований» отображает администратору настроенные бизнес-процессы на действия в Комплексе.

Список «Настройка согласований» содержит следующие данные:

- тип заявки;
- бизнес-процесс.

В списке не отображаются типы заявок, которые формируются Комплексом на периодической основе (периодическим заданием). Типы заявок и их назначение приведены в подразделе 2.1 документа «Руководство пользователя» (72410666.00054-02 96 01).

3.1. Создание настройки согласования

Для создания настройки согласования администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Настройка согласований»:

- 1) Нажать кнопку «Создать». Откроется окно «Создание настройки согласования». Кнопка «Создать» заблокирована, если все возможные типы заявок уже заданы.
- 2) Выбрать из выпадающего списка «Тип заявки». В выпадающем списке с типом заявок отображаются только те типы, которые еще не настроены в Комплексе.
- 3) Выбрать из выпадающего списка «Бизнес-процесс».
- 4) Нажать кнопку «Сохранить».

В случае ошибки отображается сообщение «Ошибка при создании настройки согласования».

Изм.	Подп.	Дата

3.2. Редактирование настройки согласования

Для редактирования настройки согласования администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Настройка согласований»:

- 1) Выбрать бизнес-процесс из списка.
- 2) Нажать кнопку «Редактировать». Откроется окно «Редактирование настройки согласования», в котором представлена информация о типе заявки и бизнес-процессе.
- 3) Выбрать из выпадающего списка «Бизнес-процесс».
- 4) Нажать кнопку «Сохранить».

В случае ошибки отображается сообщение «Ошибка при редактировании настройки согласования».

3.3. Удаление настройки согласования

Для удаления настройки согласования администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Настройка согласований»:

- 1) Выбрать бизнес-процесс из списка.
- 2) Нажать кнопку «Удалить». Откроется окно «Удаление настройки согласования», в котором представлена информация о типе заявки и бизнес-процессе.
- 3) Нажать кнопку «Удалить» для подтверждения удаления.

В случае ошибки отображается сообщение «Ошибка при удалении настройки согласования».




Изм.	Подп.	Дата

4. СИСТЕМНЫЕ НАСТРОЙКИ

Системные настройки выполняются в меню «Админ. системы»/«Системные настройки» и позволяют управлять конфигурационными файлами JSON через интерфейс Комплекса.

Настройки выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

На вкладке «Общие» доступны следующие настройки:

- 1) Настройки аутентификации. Описание приведено в подразделе 4.1.
- 2) Настройки видимости и доступности объектов в заявках и замещениях. Описание приведено в подразделе 4.2.
- 3) Настройки генераторов значений. Описание приведено в подразделе 4.3.
- 4) Настройки добавления настраиваемых полей для управляемых объектов и заявок. Описание приведено в подразделе 4.4.
- 5) Настройки загружаемых файлов. Описание приведено в подразделе 4.5.
- 6) Настройки импорта/экспорта объектов системы. Описание приведено в подразделе 4.6.
- 7) Настройки интерфейса. Описание приведено в подразделе 4.7.
- 8) Настройки логирования и массовых операций. Описание приведено в подразделе 4.8.
- 9) Настройки подключений. Описание приведено в подразделе 4.9.
- 10) Настройки производительности. Описание приведено в подразделе 4.10.

Изм.	Подп.	Дата

- 11) Настройки работы полнотекстового поиска. Описание приведено в подразделе 4.11.
- 12) Настройки синхронизации с целевыми системами. Описание приведено в подразделе 4.12.
- 13) Настройки фильтров. Описание приведено в подразделе 4.13.

Собственные решения по обработчикам событий и пользовательским сервисам можно отобразить в интерфейсе на отдельных вкладках «Обработчики объектов» и «Пользовательские сервисы» и выполнять настройки с помощью редактора кода по аналогии как для вкладки «Общие».

Обработчики событий и пользовательские сервисы можно настроить на различные действия (создание, обновление, удаление, чтение) над любыми управляемыми объектами Комплекса.

Для предоставления доступа к настройке обработчиков событий и пользовательских сервисов необходимо выдать пользователю с системной ролью следующие права:

- settings-handler-list-read – доступ на чтение списка обработчиков событий (без деталей конфигураций);
- settings-service-list-read – доступ на чтение списка пользовательских сервисов (без деталей конфигураций);
- setting-details – доступ на чтение деталей конфигураций обработчиков событий и пользовательских сервисов;
- setting-update – доступ на редактирование конфигураций обработчиков событий и пользовательских сервисов.

Описание системных ролей приведено в подразделе 2.10 документа «Руководство пользователя» (72410666.00054-02 96 01).

Для корректного отображения настроек обработчиков событий создаваемый java-класс должен отвечать следующим требованиям:

- 1) Класс должен реализовывать интерфейс HandlerSetting:
 - класс должен возвращать описание handler (getDescription);
 - класс должен возвращать имя объекта, который обрабатывается handler (getResource);

Изм.	Подп.	Дата

- класс может возвращать имя конфигурации, которую использует handler (getConfigName).
- 2) Класс должен иметь уникальное свойство SystemSetting.NAME в аннотациях OSGI.
- 3) Класс должен содержать интерфейс SystemSetting в списке реализуемых сервисов в аннотациях OSGI.

Пример настройки UserHandler:

```

@Component(
    immediate = true,
    configurationPolicy = ConfigurationPolicy.IGNORE,
    service = {SystemSetting.class},
    property = {
        Constants.SERVICE_PID + "=" + UserHandlerSetting.PID,
        Constants.SERVICE_DESCRIPTION + "=GiS :: user handler setting",
        Constants.SERVICE_VENDOR + "=" + ServerConstants.SERVER_VENDOR_NAME,
        SystemSetting.NAME + "=" + UserHandlerSetting.NAME
    }
)
public class UserHandlerSetting implements HandlerSetting {
    static final String NAME = "user.handler.setting";
    static final String PID = ConfigurationHelper.DEFAULT_SERVICE_RDN_PREFIX + NAME;

    @Reference
    private RequestService requestService;

    @Override
    public String getDescription() {
        return "Конфигурация handler для logingeneration";
    }

    @Override
    public String getResource() {
        return User.MANAGED;
    }

    @Override
    public RequestService getRequestService() {
        return requestService;
    }

    @Override
    public String getConfigName() {
        return "logingeneration";
    }
}

```

Пример настройки UserHandler без конфигурации:

Изм.	Подп.	Дата

```
@Component(  
    immediate = true,  
    configurationPolicy = ConfigurationPolicy.IGNORE,  
    service = {SystemSetting.class},  
    property = {  
        Constants.SERVICE_PID + "=" + NonExistentHandlerSetting.PID,  
        Constants.SERVICE_DESCRIPTION + "=GiS :: no config handler setting",  
        Constants.SERVICE_VENDOR + "=" + ServerConstants.SERVER_VENDOR_NAME,  
        SystemSetting.NAME + "=" + NonExistentHandlerSetting.NAME  
    }  
)  
public class NonExistentHandlerSetting implements HandlerSetting {  
    static final String NAME = "non.configurable.handler.setting";  
    static final String PID = ConfigurationHelper.DEFAULT_SERVICE_RDN_PREFIX + NAME;  
  
    @Reference  
    private RequestService requestService;  
  
    @Override  
    public String getDescription() {  
        return "Несуществующая конфигурация handler";  
    }  
  
    @Override  
    public String getResource() {  
        return User.MANAGED;  
    }  
  
    @Override  
    public RequestService getRequestService() {  
        return requestService;  
    }  
  
    @Override  
    public String getConfigName() {  
        return null;  
    }  
}
```

Для корректного отображения настроек пользовательских сервисов создаваемый java-класс должен отвечать следующим требованиям:

- 1) Класс должен реализовывать интерфейс ServiceSetting:
 - класс должен возвращать описание service (getDescription);
 - класс может возвращать имя конфигурации, которую использует service (getConfigName).
- 2) Класс должен иметь уникальное свойство SystemSetting.NAME в аннотациях OSGI.
- 3) Класс должен содержать интерфейс SystemSetting в списке реализуемых сервисов в аннотациях OSGI.

Изм.	Подп.	Дата

Пример настройки service:

```

@Component(
    immediate = true,
    property = {
        Constants.SERVICE_PID + "=" + RequestServiceSetting.PID,
        Constants.SERVICE_DESCRIPTION + "= GiS :: Request service setting",
        Constants.SERVICE_VENDOR + "=" + ServerConstants.SERVER_VENDOR_NAME,
        SystemSetting.NAME + "=" + RequestServiceSetting.NAME
    },
    service = {SystemSetting.class}
)

public class RequestServiceSetting implements ServiceSetting {
    static final String NAME = "request.service.setting";
    static final String PID = ConfigurationHelper.DEFAULT_SERVICE_RDN_PREFIX + NAME;

    @Reference
    private RequestService requestService;

    @Override
    public String getDescription() {
        return "Конфигурация service для request";
    }

    @Override
    public RequestService getRequestService() {
        return requestService;
    }

    @Override
    public String getConfigName() {
        return "request";
    }
}

```

Пример настройки service без конфигурации:

```

@Component(
    immediate = true,
    configurationPolicy = ConfigurationPolicy.IGNORE,
    service = {SystemSetting.class},
    property = {
        Constants.SERVICE_PID + "=" + NonExistentHandlerSetting.PID,
        Constants.SERVICE_DESCRIPTION + "=GiS :: no config handler setting",
        Constants.SERVICE_VENDOR + "=" + ServerConstants.SERVER_VENDOR_NAME,
        SystemSetting.NAME + "=" + NonExistentHandlerSetting.NAME
    }
)

public class NonExistentHandlerSetting implements HandlerSetting {
    static final String NAME = "non.configurable.handler.setting";
    static final String PID = ConfigurationHelper.DEFAULT_SERVICE_RDN_PREFIX + NAME;

    @Reference
    private RequestService requestService;
}

```

Изм.	Подп.	Дата

```
@Override
public String getDescription() {
    return "Несуществующая конфигурация handler";
}

@Override
public String getResource() {
    return User.MANAGED;
}

@Override
public RequestService getRequestService() {
    return requestService;
}

@Override
public String getConfigName() {
    return null;
}
}
```

4.1. Настройки аутентификации

Комплекс поддерживает следующие настройки аутентификации через интерфейс:

- 1) Настройка блокировки УЗ пользователя на определенный период в случае неуспешных попыток входа выполняется на вкладке «authenticated.lock». По умолчанию для любого пользователя ПК Ankey IDM включено неограниченное количество попыток ввода пароля при входе в Комплекс. Такая настройка дает возможность злоумышленнику подобрать пароль пробным путем. Комплекс позволяет настроить отслеживание числа недопустимых попыток входа и может быть настроен на реагирование на этот вид возможной атаки путем отключения УЗ на заданный период времени. Параметры политики блокировки УЗ контролируют пороговое значение количества последовательных попыток входа с неверным паролем, а также действия, выполняемые по достижении порогового значения. Описание конфигурационного файла authenticated.lock.json приведено в разделе 7 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 2) Настройка белого списка для проверки клиентского сертификата выполняется на вкладке «authenticated.whitelist». В Комплексе

Изм.	Подп.	Дата

используется балансировщик нагрузки с поддержкой взаимной аутентификации Transport Layer Security (TLS). При взаимной аутентификации TLS выполняется проверка сертификатов как клиента, так и сервера. Проверка подлинности клиента через взаимную TLS требует, чтобы сертификат включал в себя параметр Client Authentication. Эта схема проверки подлинности позволяет клиенту подтвердить свой сертификат в центре сертификации. Описание конфигурационного файла `authenticated.whitelist.json` приведено в разделе 8 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

- 3) Настройка авторизации через внешние сервисы, такие как Blitz, GitHub, Госуслуги, Одноклассники и другие выполняется на вкладке «`identityProviders`». Описание конфигурационного файла `identityProviders.json` приведено в подпункте 2.1.7.1 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 4) Настройка повторного использования логинов удаленных пользователей выполняется на вкладке «`uniqueLogin`». Описание конфигурационного файла `uniqueLogin.json` приведено в разделе 9 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).
- 5) Настройка аутентификации через схемы OAuth2.0, OpenId Connect, IWA и другие выполняется на вкладке «`authentication`». Описание действий приведено в пункте 2.8.2. Описание конфигурационного файла `authentication.json` приведено в разделе 2 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.2. Настройки видимости и доступности объектов в заявках и замещениях

Для настройки видимости и доступности объектов в заявках и замещениях администратору необходимо выполнить следующие действия:

- 1) В области «Для заместителей» указать, какие пользователи будут доступны при выборе заместителя:
 - все пользователи;

Изм.	Подп.	Дата




- только из области видимости пользователя;
 - собственное значение, если было настроено.
- 2) В области «Добавление пользователей в заявку и ее переназначение» установить один из двух флагов, либо оба:
- флаг «Возможность смены бенефициара в заявках при создании»;
 - флаг «Применить выбранную область видимости для оператора».
- 3) Указать, какие пользователи будут доступны при смене бенефициара в заявке:
- все пользователи;
 - только из области видимости пользователя;
 - собственное значение, если было настроено.
- 4) В области «Переназначение заявки на роль» включить флаг «Возможность переназначения заявки на роль». При этом в поле «Доступные роли» по умолчанию станет активно значение «Все роли». Дополнительно поле может отображать собственное значение, если было настроено.
- 5) Нажать «Сохранить». Отобразится сообщение «Настройки системы успешно сохранены».

Примечание. Собственные решения, разработанные под бизнес-логику заказчика и настроенные в конфигурационных файлах, отобразятся автоматически.

4.3. Настройки генераторов значений

Настройки генераторов значений выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Изм.	Подп.	Дата

Настройка регулярного выражения для генерации пароля в случае отсутствия парольных политик выполняется на вкладке «passwordgenerator». Описание конфигурационного файла passwordgenerator.json и пример приведены в разделе 23 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Настройка регулярного выражения для автоматической генерации кодового слова во время выполнения периодического задания с типом «secretwordgeneratingtask» выполняется на вкладке «secretgenerator». Описание конфигурационного файла secretgenerator.json приведено в разделе 29 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Описание настройки идентификации пользователя через кодовое слово приведено в пункте 2.8.4.

Настройки скрипта для генерации табельного номера в формате JavaScript, и количества попыток генерации табельного номера выполняются на вкладке «usrorgtabnumgeneration». Описание конфигурационного файла usrorgtabnumgeneration.json приведено в разделе 34 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).




Настройки скрипта для генерации логина в формате JavaScript и количества попыток генерации логина выполняются на вкладке «logingeneration». Описание конфигурационного файла logingeneration.json приведено в разделе 21 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.4. Добавление настраиваемого поля для управляемых объектов и заявок

Добавление настраиваемого поля для управляемых объектов и заявок выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

Изм.	Подп.	Дата

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Добавление настраиваемых полей для любого управляемого объекта в системе, конфигурирование существующих полей, управление политиками, поиском, автозаполнением и прочим выполняются на вкладке «extend»:

```
{
  "managed/user": {
    "lastName": {
      "managed": {
        "display": [
          "selfEdit"
        ]
      }
    },
    "phone": {
      "managed": {
        "display": [
          "selfEdit"
        ]
      }
    }
  },
  "_id": "extend"
}
```

Описание конфигурационного файла extend.json и пример приведены в пункте 2.7.1.

Настройка динамического списка в форме заявки выполняется на вкладке «requestform.enum». Описание конфигурационного файла requestform.enum.json приведено в разделе 24 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.5. Настройки загружаемых файлов

Настройка загружаемых в систему документов, их размера, поддерживаемых типов и расширений выполняется на вкладке «document».

Описание конфигурационного файла document.json приведено в разделе 13 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

Применение загружаемых файлов описано в пункте 1.3.1 документа «Руководство пользователя» (72410666.00054-02 96 01).

Настройка загружаемых фотографий, их размера, поддерживаемых типов и расширений выполняется на вкладке «photo».

Описание конфигурационного файла photo.json приведено в разделе 14 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Управление фотографиями пользователей описано в подразделе 2.19.

После сохранения настроек загружаемых фотографий Комплекс будет перезагружен автоматически.

4.6. Настройки импорта/экспорта объектов системы

Настройки импорта/экспорта объектов системы выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Настройка уникальных полей пользователей, руководителей и организаций при экспорте и импорте пользователей выполняется на вкладке «userexportimport». Описание конфигурационного файла userexportimport.json приведено в разделе 33 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Настройка кодировки в Excel-файлах при импорте/экспорте объектов системы выполняется на вкладке «excelexport». Описание конфигурационного файла excelexport.json приведено в разделе 17 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). Дополнительно на вкладке «excelexport» можно настроить импорт ролей через коннектор «Excel Matrix Upload». Описание настройки импорта ролей через коннектор «Excel Matrix Upload» приведено в подразделе 2.24.

Настройка отображаемых значений lookup-полей и multilookup-полей выполняется на вкладке «lookupexportimport»:

```
{
  "lookups": {
    "aprole": {
      "customUserLookupM": "userName"
    }
  }
}
```

Изм.	Подп.	Дата

В параметре «аррrole» указывается:

- 1) «customUserLookupM» – существующее имя lookup-поля в роли.
- 2) «userName» – имя поля, значение которого необходимо указать для роли в файле импорта. По этому значению будет осуществляться поиск при применении изменений.

В параметре «аррrole» нельзя указывать несуществующие и виртуальные поля, в противном случае отобразится ошибка «Ошибка сохранения конфигурации».

Если в конфигурационном файле «lookrexportimport» не указано поле для «lookup»/«multilookup», но в extend.json у них указано виртуальное поле в параметре «lookupDisplayFields», то при загрузке файла с импортируемыми ролями возникнет ошибка «Невозможно обработать отображаемые виртуальные поля».




Настройка разделителей и переносов строк в csv-файлах при импорте/экспорте объектов системы выполняется на вкладке «csvexport».

Описание конфигурационного файла csvexport.json приведено в разделе 12 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.7. Настройки интерфейса

Настройки интерфейса выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Настройки отображения заголовков, информации о продукте и техподдержке, логотипа, фавикона выполняются на вкладке «ui/themeconfig»:

```
{
  "icon": "favicon.png",
  "settings": {
    "logo": {
```

Изм.	Подп.	Дата

```

    "src": "logo.svg",
    "title": "GiS :: Ankey Identity Manager",
    "alt": "GiS :: Ankey Identity Manager"
  },
  "system": {
    "title": "Ankey Identity Manager",
    "logoutUrl": "#!/logout"
  },
  "login": {
    "logo": {
      "src": "/assets/images/icons/login/icons_ankey_logo.svg",
      "title": "GiS :: Ankey Identity Manager",
      "alt": "GiS :: Ankey Identity Manager",
      "height": "70px",
      "width": "70px"
    },
    "footer": {
      "supportContacts": "Возникли вопросы? \n Связь с администратором 8 (800) XXX-XX-XX"
    }
  },
  "version": {
    "name": "Ankey IDM",
    "company": "ООО «Газинформсервис»"
  }
},
"_id": "ui/themeconfig"
}

```

Настройки для управления таблицами в интерфейсе, выбора отображаемых столбцов для различных объектов системы выполняются на вкладке «ui.custom.table». Описание конфигурационного файла ui.custom.table.json приведено в разделе 31 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). Описание настройки для отображения столбцов в меню «Пользователи/Учетные записи» приведено в пункте 2.27.1.

Настройки интерфейса, смены языка интерфейса, формата даты и прочее выполняются на вкладке «ui/configuration». Описание конфигурационного файла ui-configuration.json приведено в разделе 31 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.8. Настройки логирования и массовых операций

Настройка логирования выполняется на вкладке «audit». Включает в себя настройки по созданию записей аудита, выбор типов событий и места их хранения.

Изм.	Подп.	Дата

Особенности настройки аудита при добавлении настраиваемого поля в конфигурационном файле `extend.json`:

- если в массиве `«display»` указано значение `«audit»`, то атрибут будет автоматически добавлен в массив `«watchedFields»` файла `audit.json`;
- если атрибут не был добавлен в массив `«watchedFields»` и в последующем произошло его изменение, то будет создана запись об изменении в таблице `auditactivity` БД обработчика событий аудита. Но такая запись не будет отображаться в разделе `«История изменения пользователя»` в меню `«Пользователи»`, поскольку поле `«changedFields»` в таблице будет пустым. Поэтому если требуется отображать изменения настраиваемого поля на вкладке `«История изменений пользователя»` необходимо в массиве `«display»` указать значение `«audit»`.

Описание конфигурационного файла `audit.json` приведено в разделе 1 документа `«Описание конфигурационных файлов»` (72410666.00054-02 97 01).

Настройка массовых операций выполняется на вкладке `«bulk.operation»` и включает в себя:

- включение/выключение возможности отслеживания массовых операций;
- объекты, для которых выполняется настройка массовых операций.

Возможность отслеживания массовых операций включена по умолчанию, параметр `«enabled»` принимает значение `«true»`.

Описание конфигурационного файла `bulk.operation.json` приведено в разделе 11 документа `«Описание конфигурационных файлов»` (72410666.00054-02 97 01).

4.9. Настройки подключений

Настройки подключений выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Настройка подключения к системе полнотекстового поиска выполняется на вкладке `«search.connection»`. Описание конфигурационного файла `search.connection.json` приведено в разделе 27 документа `«Описание конфигурационных файлов»` (72410666.00054-02 97 01).

После сохранения настроек Комплекс будет перезагружен автоматически.

Изм.	Подп.	Дата

Настройка почтового сервера и шаблонов писем для согласования заявок из почты под различные бизнес-процессы выполняется на вкладке «workflow». Описание конфигурационного файла workflow.json в разделе 35 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). После сохранения настроек Комплекс будет перезагружен автоматически.

Настройка параметров подключения к СУБД выполняется на вкладке «repo.jdbc»:

```
{
  "connection": {
    "dbType": "POSTGRESQL",
    "jndiName": "",
    "driverClass": "org.postgresql.xa.PGXADatasource",
    "jdbcUrl": "jdbc:postgresql://ankey-postgres:5432/ankey",
    "username": "ankey",
    "password": {
      "$crypto": {
        "type": "x-simple-encryption",
        "value": {
          "cipher": "AES/CBC/PKCS5Padding",
          "data": "OPfZ58vUR7wVk+AxTB9MuA==",
          "iv": "xfkc3OUgWbsZ31jnYPoyYg==",
          "key": "openidm-sym-default"
        }
      }
    }
  },
  "defaultCatalog": "ankey",
  "maxBatchSize": 100,
  "maxTxRetry": 5,
  "connectionTimeoutInMs": 30000,
  "enableConnectionPool": true,
  "dataSource": {
    "url": "jdbc:postgresql://ankey-postgres:5432/ankey?autosave=conservative",
    "user": "ankey",
    "password": {
      "$crypto": {
        "type": "x-simple-encryption",
        "value": {
          "cipher": "AES/CBC/PKCS5Padding",
          "data": "ogKn0fic74qDh5xZhszfGw==",
          "iv": "E36mYMU8P4bAJGp8Ml7Fjw==",
          "key": "openidm-sym-default"
        }
      }
    }
  }
},
  "pool": {
    "initialSize": 0,
    "maxTotal": 20,
    "maxIdle": 20,
    "minIdle": 0,
  }
}
```

Изм.	Подп.	Дата

```

    "maxWaitMillis": 5000
  }
},
"_id": "repo.jdbc"
}

```

Описание конфигурационного файла `repo.jdbc.json` и примеры приведены в подпунктах 2.13.1.2 и 2.13.1.3. Для применения настроек подключения к СУБД необходимо перезагрузить Комплекс вручную.

Настройка почтового сервера уведомлений выполняется на вкладке «external.email». Описание конфигурационного файла `external.email.json` приведено в разделе 19 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).




Настройка почтового сервера для получения входящих сообщений и шаблонов писем для согласования заявок из электронной почты выполняется на вкладке «email.request.processing». Описание конфигурационного файла `email.request.processing.json` и примеры приведены в разделе 15 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

После сохранения настроек Комплекс будет перезагружен автоматически.

4.10. Настройки производительности

Настройки производительности выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Настройка количества потоков выполнения периодических заданий выполняется на вкладке «scheduler». Описание конфигурационного файла

Изм.	Подп.	Дата

scheduler.json приведено в разделе 26 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Настройка времени ожидания ответа статуса ресурса выполняется на вкладке «resource». Описание конфигурационного файла resource.json приведено в разделе 25 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).




Настройка асинхронности выполнения операций распространения данных выполняется на вкладке «sync». Описание конфигурационного файла sync.json и пример приведены в разделе 30 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

4.11. Настройки работы полнотекстового поиска

Настройки работы полнотекстового поиска выполняются на соответствующих вкладках с помощью редактора кода. Редактор кода содержит настройки по умолчанию.

Для редактирования настройки администратору необходимо в поле редактора кода ввести изменения и нажать кнопку «Сохранить». В случае невалидного конфигурационного файла изменения не сохраняются и отображается сообщение «Ошибка сохранения конфигурации».

После выполнения настроек необходимо запустить периодическое задание «reindexTask».

Для копирования кода следует нажать на значок . Для форматирования кода следует нажать на значок . Чтобы развернуть редактор кода на весь экран, следует нажать на значок .

Настройка параметров системы полнотекстового поиска выполняется на вкладке «search.settings». Описание конфигурационного файла search.settings.json и пример приведены в разделе 28 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01). После сохранения настроек Комплекс будет перезагружен автоматически.

Изм.	Подп.	Дата

4.12. Настройки синхронизации с целевыми системами

В настройках синхронизации с ЦС имеется возможность настройки, которая позволяет учитывать изменения только в многозначных атрибутах УЗ пользователей при пересчете периодического задания «evaluatetask». Описание периодического задания «evaluatetask» приведено в пункте 2.4.5.

Настройка содержится в конфигурационном файле accounts.setting.json:

```
{  
  "evaluateOnlyMultivaluedFields": false  
}
```

Для включения настройки необходимо перевести флаг «Учитывать только мультиатрибуты при пересчете изменений в учетных записях пользователей» в правое положение и нажать кнопку «Сохранить».

4.13. Настройка приоритета полей поискового запроса

Комплекс поддерживает настройку весов (от 0 до 10) для всех полей, участвующих в поисковых запросах.

Для разделов, указанных ниже, добавлена сортировка по релевантности:

- меню «Инф.системы»;
- меню «Роли»;
- в списках ИС, ролей в форме запроса ролей.

Для настройки полей поискового запроса следует использовать следующие конфигурационные файлы:

- 1) extend.json – для настраиваемых полей.
- 2) managed.json – для полей Комплекса.

По умолчанию все поля имеют значение 1, кроме поля «lastName» в объекте «user», которому установлено значение 2, чтобы сохранить релевантность поиска. При этом поля Комплекса имеют больший приоритет перед дополнительными настраиваемыми полями. Чтобы повысить приоритет настраиваемого поля перед полем Комплекса, необходимо установить вес в диапазоне от 6 до 10. Величина повышения приоритета не является линейной, то есть повышение на два не приведет к тому, что поле будет в два раза приоритетнее. В конкретных случаях различным типам

Изм.	Подп.	Дата

полей может быть присвоен свой индивидуальный вес для увеличения их релевантности.

Значение 0 уменьшает приоритет.

В файле `extend.json` настройка полей полнотекстового поиска задается с помощью флага «`searchable: true`» в блоке «`managed`». Вес поля может быть указан с помощью параметра «`searchWeight`» от 0 до 10. Если вес не указан, то полю автоматически присваивается значение по умолчанию. Если для параметра «`searchable`» установлено значение `false`, то поле не будет участвовать в поиске, даже если установлен вес.

Пример настройки в файле `extend.json`:

```
"managed": {  
  "required": false,  
  "fieldType": "textField",  
  "searchable": true,  
  "searchWeight": 10,  
  "display": [  
    "create",  
    "update",  
    "view",  
    "selfEdit"  
  ]  
}
```

В файле `managed.json`, в блоке «`search`», вес указывается для каждого поля. Также здесь можно указать вес поля от 0 до 10. Описание конфигурационного файла `managed.json` приведено в разделе 22 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Изм.	Подп.	Дата

5. УПРАВЛЕНИЕ ТИПАМИ УЗ ПОЛЬЗОВАТЕЛЕЙ

Управление типами УЗ пользователей выполняется в меню «Настройка ресурсов»/«Типы УЗ пользователей».

Список типов УЗ пользователей содержит следующую информацию:

- название типа;
- описание.

Детали типа УЗ пользователя отображаются на вкладках «Параметры», «Доступность в УЗР» и «Права на создание доп. УЗ».

Вкладка «Доступность в УЗР» отображается, если был выключен флаг «Доступен для всех УЗР» и содержит список выбранных УЗР.

Вкладка «Параметры» может содержать следующие сведения:

- 1) Основные данные. Содержит название и описание, а так же состояние флагов «Доступен для всех УЗР» и «Может устанавливаться для основных УЗ».
- 2) Разрешения. Содержит перечень поддерживаемых разрешений в виде флагов.

Управление флагами выполняется при создании нового типа УЗ. Отображается состояние – «включено» или «отключено».

Описание флагов в зависимости от их состояния:

- 1) Флаг «Доступен для всех УЗР» позволяет определить, будет ли новый тип УЗ доступен для всех УЗР Комплекса или только для определенных. По умолчанию флаг включен. Если флаг выключить, Комплекс предложит выбрать УЗР из списка, после чего тип УЗ будет доступен только для выбранных. Это предоставляет возможность устанавливать данный тип только определенным УЗР и устанавливать его в качестве типа по умолчанию. Если ничего не выбрать, тип УЗ будет недоступен для УЗР и не будет отображаться при попытке смены типа УЗ.
- 2) Флаг «Может устанавливаться для основных УЗ» отвечает за доступность типа УЗ для основной УЗ. По умолчанию флаг выключен, что означает

Изм.	Подп.	Дата

невозможность установки такого типа для основной УЗ. Комплекс позволяет сменить тип, но УЗ автоматически станет дополнительной.

- 3) Флаг «Разрешить создание дополнительных УЗ любым пользователям» отвечает за право разрешать всем пользователям Комплекса создание дополнительной УЗ в УЗР. По умолчанию флаг выключен.

Описание возможных разрешений:

- 1) Разрешить выделение полномочий УЗ данного типа по роли. При отключенном флаге полномочия УЗ не будут назначаться пользователю через роль.
- 2) Разрешить изменение родительской формы УЗ данного типа через маппинги. Если флаг установлен, то УЗ будет связываться с владельцем при синхронизации. При включенном флаге будет разрешено вносить изменения из ЦС при синхронизации. Возможно изменение всех свойств формы УЗ, кроме полномочий. Изменение полномочий выполняется с помощью флага «Разрешить выделение полномочий УЗ данного типа по роли». Кроме этого, если флаг установлен, статус УЗ пользователя в ЦС будет меняться при блокировке/активации пользователя. В противном случае статус УЗ не меняется. Изменить статус такой УЗ можно будет через блокировку/активацию УЗ.
- 3) Очищать значение поля «Причина блокировки» при изменении типа УЗ. Поле «Причина блокировки» – поле «disableReason» в БД Комплекса. Если флаг установлен, то поле будет очищаться. В случае пустого значения поля «Причина блокировки» не будет выполняться автоматическая блокировка/активация УЗ.
- 4) Сохранять УЗ при удалении пользователя. При включенном флаге позволяет удалить пользователя без удаления самой УЗ. В этом случае УЗ будет отвязана при удалении пользователя. При отключенном флаге УЗ будет удалена вместе с пользователем.

Администратор выполняет создание типа УЗ пользователя. Описание приведено в подразделе 5.1.

Управление УЗР приведено в подразделе 2.3.

Изм.	Подп.	Дата

5.1. Создание типа УЗ пользователя

Для создания типа УЗ пользователя администратору необходимо выполнить следующие действия:

- 1) Нажать кнопку «Создать». Откроется окно «Создание типа УЗ пользователя».
- 2) Перейти на обязательную вкладку «Параметры» и выполнить:
 - заполнить обязательное поле «Название». В противном случае отобразится подсказка «Не заполнено обязательное поле»;
 - заполнить при необходимости поле «Описание»;
 - выбрать при необходимости флаг «Может устанавливаться для основных УЗ».
- 3) Перейти на вкладку «Разрешения» и при необходимости установить следующие флаги:
 - разрешить выделение полномочий УЗ данного типа по роли;
 - разрешить изменение родительской формы УЗ данного типа через маппинги. Флаг должен быть установлен, чтобы УЗ привязывалась к владельцу при синхронизации;
 - очищать значение поля «Причина блокировки» при изменении типа УЗ;
 - сохранять УЗ при удалении пользователя.
- 4) Перейти на вкладку «Доступность в УЗР», в которой по умолчанию установлен флаг «Доступен для всех УЗР». При необходимости изменить положение флага. После чего откроется список для выбора УЗР.
- 5) Перейти на вкладку «Права на создание доп. УЗ». На вкладке можно настроить права на создание дополнительных УЗ. Выбрать один из следующих способов:
 - выбор роли – только пользователи с этой ролью смогут создавать дополнительные УЗ.
 - установка флага «Разрешить создание дополнительных УЗ любым пользователям» – право на создание дополнительных УЗ получат все пользователи Комплекса;

Изм.	Подп.	Дата

- отключение права: если флаг неактивен и роли не выбраны, то создание дополнительных УЗ будет недоступно.
- б) Нажать кнопку «Сохранить». Отобразится сообщение «Тип УЗ пользователя успешно создан».

Подробное описание создания дополнительной УЗ в процессе назначения ролей приведено в пункте 2.4.6.1.

Изм.	Подп.	Дата

6. НАСТРОЙКА ТИПОВ БЛОКИРОВОК

По умолчанию администратор, оператор и владелец УЗ имеют полный доступ ко всем типам блокировок. Комплекс позволяет расширить доступ к типам блокировок для других пользователей. Расширение доступа осуществляется через настройки применения и использование ролей.

Настройка типов блокировок выполняется в меню «Админ. системы»/«Типы блокировок» и включает в себя следующие действия администратора:

- создание типа блокировки;
- редактирование типа блокировки;
- удаление типа блокировки.

6.1. Создание типа блокировки

Для создания типа блокировки администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Типы блокировок»:

- 1) Нажать кнопку «Создать». Откроется окно «Создание типа блокировки».
- 2) На вкладке «Параметры» заполнить обязательные поля:
 - «Название» – название типа блокировки;
 - «Объект системы» – по умолчанию «Учетная запись пользователя»;
 - «Бизнес-процесс» – название бизнес-процесса, который определяет последовательность шагов и форму заявки.
- 3) На вкладке «Настройки применения» при необходимости заполнить поля:
 - «Блокировка доступна для ролей» – название роли, для которой доступна блокировка;
 - «Активация доступна для ролей» – название роли, для которой доступна активация.
- 4) Нажать кнопку «Сохранить». Для отмены действий нажать кнопку «Отменить».

Флаг «Блокировка и активация доступны руководителю» по умолчанию выключен. При включении доступ к типу блокировки получает и руководитель.

Изм.	Подп.	Дата

6.2. Редактирование типа блокировки

Для редактирования типа блокировки администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Типы блокировок»:

- 1) Выбрать тип блокировки из списка.
- 2) Нажать кнопку «Редактировать». Откроется окно «Редактирование типа блокировки».
- 3) Внести изменения в существующих полях.
- 4) Нажать кнопку «Сохранить». Для отмены действий нажать кнопку «Отменить».

6.3. Удаление типа блокировки

Для удаления типа блокировки администратору необходимо выполнить следующие действия в меню «Админ. системы»/«Типы блокировок»:

- 1) Выбрать тип блокировки из списка.
- 2) Нажать кнопку «Удалить». Откроется окно «Удаление типа блокировки».
- 3) Нажать кнопку «Удалить» для подтверждения удаления. Для отмены действий нажать кнопку «Отменить».

Нельзя удалить тип блокировки, если в Комплексе есть УЗ, заблокированные по данному типу. В таком случае отображается сообщение «Удаление невозможно, пока есть действующие блокировки по выбранному типу блокировки».

Изм.	Подп.	Дата

7. НАСТРОЙКА ТИПОВ БЛОКИРОВОК

По умолчанию администратор, оператор и владелец УЗ имеют полный доступ ко всем типам блокировок. Комплекс позволяет расширить доступ к типам блокировок для других пользователей. Расширение доступа осуществляется через настройки применения и использование ролей.

Настройка типов блокировок выполняется в меню «Админ. системы»/«Типы блокировок» и включает в себя следующие действия администратора:

- создание типа блокировки;
- редактирование типа блокировки;
- удаление типа блокировки.

Изм.	Подп.	Дата

8. НАСТРОЙКИ ВИДЖЕТОВ

Список виджетов содержит следующую информацию:

- название;
- описание.

Настройка виджетов выполняется в меню «Админ. системы»/«Настройки виджетов» и включает в себя следующие действия администратора:

- 1) Выбрать виджет из списка.
- 2) Нажать кнопку «Настроить». Откроется окно «Настройка виджета».
- 3) Ввести название роли, которой будет доступен виджет. Для добавления нескольких ролей нажать кнопку «Добавить».
- 4) Нажать кнопку «Сохранить». После чего отобразится сообщение «Настройки виджета успешно изменены».

Настроенные виджеты отображаются пользователям в меню «Мониторинг». Описание приведено в подразделе 2.11 документа «Руководство пользователя» (72410666.00054-02 96 01).

Изм.	Подп.	Дата

ПРИЛОЖЕНИЕ 1

В данном приложении приводятся примеры выполнения запросов с использованием REST API.

Пример 1. Самостоятельная смена пароля пользователем.

Обязательные параметры запроса:

- «oldPassword» – значение старого пароля;
- «newPassword» – значение нового пароля.

Пример REST-запроса для самостоятельной смены пароля пользователем, например, petrov-p:

```
C:\> curl--cacert self-signed.crt ^
--header "Content-Type: application/json" ^\
--header "X-Ankey-Username: ankey" ^\
--header "X-Ankey-Password: ankey" ^\
--request POST ^
--data "{
  "oldPassword": "qwe",
  "newPassword": "newPasswordValue"}" ^
'http://localhost:port/ankey/endpoint/user?_action=changePassword'
```

Пример 2. Добавление фотографии пользователя в профиль.

Основные параметры запроса при добавлении фотографии в профиль:

- 1) «df8123b3-4206-4caf-bd6d-6f2c17083982» – ID пользователя.
- 2) 'file=@/home/vagrant/photo.jpg' – путь к файлу с фотографией. Путь до файла указывается либо полный, либо относительно директории откуда curl запускается.

Пример REST-запроса добавления фотографии пользователя в профиль:

```
curl -X POST
--header 'X-Ankey-Username: ankey'
--header 'X-Ankey-Password: ankey'
--form 'json={
  "name": {"$ref": "cid:file#filename"},
  "mimeType": {"$ref": "cid:file#mimetype"},
  "content": {"$ref": "cid:file#content"},
  "size": {"$ref": "cid:file#size"}
};type=application/json'
--form 'file=@/home/ankey/photo.jpg;type=image/jpeg' 'http://localhost:port/ankey/endpoint/user/df8123b3-4206-4caf-bd6d-6f2c17083982/photo'
```

При добавлении фотографии недопустимо менять или устанавливать фотографию у удаленного сотрудника, в противном случае возникает ошибка «404: Not Found».

Изм.	Подп.	Дата

Другие ограничения приведены в подразделе 2.19.

Пример 3. Процесс лицензирования ПК Ankey IDM.

Перед выполнением действий по лицензированию ПК Ankey IDM следует убедиться, что сервер приложений ПК Ankey IDM развернут на ОС Linux x64. Все REST-запросы выполняются строго на той машине, где развернут активируемый ПК Ankey IDM. В противном случае полученный сертификат может оказаться недействительным.

Администратору следует выполнить следующие шаги:

- 1) Проверить настройки конфигурации активатора.
- 2) Сгенерировать offline-запрос для проведения активации.
- 3) Получить сертификат на сервере лицензирования.
- 4) Применить полученный сертификат в ПК Ankey IDM.
- 5) При необходимости осуществить проверку лицензии.

Шаг 1. Для проверки настройки конфигурации активатора следует выполнить REST-запрос проверки актуальности настроек активатора:

GET: `http://host:port/ankey/config/license.activator`

Ответ содержит следующие настройки активатора:

- `activationResource` – ресурс для проведения завершающих этапов по активации ключа и принимает значение «`https://license.gaz-is.ru/offlineActivate`»;
- `productName` – имя продукта, совпадает с именем продукта, на который выписан ключ на сервере лицензирования и принимает значение «Ankey IDM»;
- `maxActiveUsersPropertyName` – имя свойства лицензии, хранящее максимально допустимое количество пользователей, совпадает со свойством ПК Ankey IDM на сервере лицензирования и принимает значение «`user_count`».

Шаг 2. Для генерации offline-запроса для проведения активации следует выполнить следующий REST-запрос:

POST: `http://host:port/ankey/endpoint/license/offline?_action=activate`

Body:

```
{
  "email": "admin-name@gaz-is.ru",
  "licenseKey": "Y7AL1-6SEHE-COLOE-GE9"
}
```

Основные параметры запроса на генерацию offline-запроса:

- `email` – электронная почта администратора;
- `licenseKey` – ключ активации.

Изм.	Подп.	Дата

Ответ на запрос по активации ключа:

```
{
  "certificationRequest": {
    "email": "admin-name@gaz-is.ru",
    "license_request": "--CERTIFICATE REQUEST--"},
  "activationResource": "https://license.gaz-is.ru/offlineActivate"
}
```

Запрос по активации ключа содержит следующие параметры:

- certificationRequest – содержит сгенерированный offline-запрос, который используется для проведения активации на сервере лицензирования;
- activationResource – содержит ссылку для перехода на ресурс для проведения завершающих этапов по активации ключа.

Шаг 3. Получить сертификат на сервере лицензирования, используя данные полученные на шаге 2.

Шаг 4. Выполнить REST-запрос применения сертификата в ПК Ankey IDM. Перед выполнением запроса следует выполнить форматирование полученного сертификата, заменив все переносы строк на пробелы. После чего выполнить запрос применения сертификата в ПК Ankey IDM:

```
PUT: http://host:port/ankey/config/license
Body:
{
  "certificate": "---BEGIN CERTIFICATE--- SAMPLE ---END CERTIFICATE--- "
```

Шаг 5. Для проверки лицензии необходимо выполнить следующий REST-запрос:

GET: http://host:port/ankey/endpoint/license

Ответ на запрос проверки лицензии содержит следующие параметры:

- productName – имя продукта;
- activeUsers – текущее количество активных пользователей в ПК Ankey IDM;
- maxActiveUsers – максимально допустимое количество активных пользователей ПК Ankey IDM (количество пользовательских лицензий), заданное при создании/редактировании лицензии на сервере лицензирования.

Пример ответа на запрос проверки лицензии:

```
{
  "_id": "",
  "_rev": "",
  "productName": "Ankey IDM",
  "activeUsers": 0,
```

Изм.	Подп.	Дата

```
"maxActiveUsers": 10
}
```

Ниже приведены примеры для создания маппинга.

Пример использования REST-запроса для создания маппинга распространения данных в ЦС:

```
{
  "mapName": "xml_connectorAuto_create",
  "mapType": "provision",
  "mapDesc": "rf_xml to xml",
  "mapSource": "resform/XML1",
  "mapTarget": "connector/auto",
  "mapBody": {
    "policies": [],
    "properties": [{
      "source": "username",
      "target": "username"
    }]
  }
}
```

Пример использования REST-запроса для создания маппинга синхронизации УЗ из ЦС:

```
{
  "mapName": "xml_recon",
  "mapType": "recon",
  "mapDesc": "xml to rf_xml",
  "mapSource": "connector/auto",
  "mapTarget": "resform/XML1",
  "correlationQuery": {
    "type": "text/javascript",
    "source": "var qry = { '_queryFilter': 'username eq \'' + source.username + '\'' }; qry;"
  },
  "mapBody": {
    "policies": [{
      "situation": "CONFIRMED",
      "action": "UPDATE"
    },
    {
      "situation": "FOUND",
      "action": "UPDATE"
    },
    {
      "situation": "ABSENT",
      "action": "CREATE"
    },
    {
      "situation": "AMBIGUOUS",
      "action": "EXCEPTION"
    },
    {
      "situation": "MISSING",
```

Изм.	Подп.	Дата

```

    "action": "UNLINK"
  },
  {
    "situation": "SOURCE_MISSING",
    "action": "EXCEPTION"
  },
  {
    "situation": "UNQUALIFIED",
    "action": "IGNORE"
  },
  {
    "situation": "UNASSIGNED",
    "action": "IGNORE"
  },
  {
    "situation": "TARGET_IGNORED",
    "action": "IGNORE"
  }
],
"properties": [{
  "source": "username",
  "target": "username"
}]
},
"scripts": {
  "result": {
    "type": "text/javascript",
    "file": "script/reconresult.js"
  },
  "onCreate": {
    "type": "text/javascript",
    "file": "script/onCreate.js"
  },
  "onUpdate": {
    "type": "groovy",
    "source": "println \"onUpdateScript\";"
  },
  "onDelete": {
    "type": "text/javascript",
    "source": "console.log(\"onDeleteScript\");"
  }
}
}
}

```

Пример использования REST-запроса для создания маппинга распространения атрибутов из объекта «Пользователь» в форму ресурса с заданными условиями (condition):

```

{
  "mapName": "ad_provision_user_to_resform",
  "mapType": "provision",
  "mapDesc": "Mapping for provision from managed/user to aresform/ADForm on Update",
  "mapSource": "managed/user",
  "mapTarget": "resform/ADForm",
  "mapBody": {
    "policies": [
      .....
    ]
  }
}

```

Изм.	Подп.	Дата

```

],
"properties": [{
  "source": "lastName",
  "target": "sn",
  "condition": {
    "type": "text/javascript",
    "source": "(object.lastName!=null && !object.lastName=="
  }
}],
{
  "source": "firstName",
  "target": "givenname",
  "condition": {
    "type": "text/javascript",
    "source": "(object.firstName!=null && !object.firstName=="
  }
}
]
}
}

```

Пример маппинга с указанием действия (action) в правиле маппинга (properties):

```

{
  "mapName": "xml_connectorAuto_create",
  "mapType": "provision",
  "mapDesc": "rf_xml to xml",
  "mapSource": "resform/XML1",
  "mapTarget": "connector/auto",
  "mapBody": {
    "policies": [],
    "properties": [{
      "source": "username",
      "target": "username"
    }, {
      "action": "create",
      "source": "password",
      "target": "password"
    }, {
      "action": "update",
      "source": "password",
      "target": "password",
      "transform": {
        "type": "text/javascript",
        "source": "openidm.decrypt(source)"
      }
    }
  ]
}
}

```

Пример маппинга с указанием скриптов (scripts):

```

{
  "mapName": "xml_recon",
  "mapType": "recon",

```

Изм.	Подп.	Дата

```

"mapDesc": "xml to rf_xml",
"mapSource": "connector/auto",
"mapTarget": "resform/XML1",
"mapBody": {
  "policies": [],
  "properties": [.....]
},
"scripts": {
  "result": {
    "type": "text/javascript",
    "file": "script/reconresult.js"
  },
  "onCreate": {
    "type": "text/javascript",
    "file": "script/onCreate.js"
  },
  "onUpdate": {
    "type": "groovy",
    "source": "println \"onUpdateScript\";"
  },
  "onDelete": {
    "type": "text/javascript",
    "source": "console.log(\"onDeleteScript\");"
  }
}
}
}

```

Пример 5. Убрать видимость кнопки «Сменить пароль» в меню «Мой профиль».

Пример REST-запроса на изменение конфигурационного файла ui-configuration.json:

PATCH: /ankey/config/ui/configuration

```

[
{
"operation": "replace",
"field": "configuration/hideChangePassword",
"value": true
}
]

```

Описание конфигурационного файла ui-configuration.json приведено в разделе 31 документа «Описание конфигурационных файлов» (72410666.00054-02 97 01).

Возможные значения параметра «value»:

- false – включает видимость кнопки «Сменить пароль» (установлено по умолчанию);
- true – выключает видимость кнопки «Сменить пароль».

Изм.	Подп.	Дата

Перечень сокращений

API	– Application Programming Interface
BPMN	– Business Process Model and Notation
CSV	– Comma-Separated Values
DBT	– Database Table Connector
HTTP	– HyperText Transfer Protocol
ICF	– Identity Connector Framework
IP	– Internet Protocol
IWA	– Integrated Windows Authentication
JAR	– Java Archive
JDK	– Java Development Kit
JSON	– JavaScript Object Notation
JWT	– JSON Web Tokens
MIME	– Multipurpose Internet Mail Extensions
OSGI	– Open Services Gateway Initiative
SQL	– Structured Query Language
SSL	– Secure Sockets Layer
TLS	– Transport Layer Security
UI	– User Interface
UID	– User identifier
URL	– Uniform Resource Locator
UUID	– Universally unique identifier
XML	– eXtensible Markup Language
БД	– база данных
БП	– бизнес пакет
ИБ	– информационная безопасность
ИС	– информационная система
ОС	– операционная система
СУБД	– система управления базами данных

Изм.	Подп.	Дата

- УЗ – учетная запись
- УЗР – учетная запись ресурса
- ФСТЭК – Федеральная служба по техническому и экспортному контролю
России России
- ЦС – целевая система

Изм.	Подп.	Дата

