

# Creatio IDE

Разработка в Creatio IDE

Версия 8.0



Эта документация предоставляется с ограничениями на использование и защищена законами об интеллектуальной собственности. За исключением случаев, прямо разрешенных в вашем лицензионном соглашении или разрешенных законом, вы не можете использовать, копировать, воспроизводить, переводить, транслировать, изменять, лицензировать, передавать, распространять, демонстрировать, выполнять, публиковать или отображать любую часть в любой форме или посредством любые значения. Обратный инжиниринг, дизассемблирование или декомпиляция этой документации, если это не требуется по закону для взаимодействия, запрещены.

Информация, содержащаяся в данном документе, может быть изменена без предварительного уведомления и не может гарантировать отсутствие ошибок. Если вы обнаружите какие-либо ошибки, сообщите нам о них в письменной форме.

# Содержание

<b>Разработка в Creatio IDE</b>	<b>4</b>
Переход в раздел Конфигурация (Configuration)	4
Переключение между версиями интерфейса раздела Конфигурация (Configuration)	4
Управление доступом к разделу Конфигурация (Configuration)	5
Описание интерфейса раздела Конфигурация (Configuration)	6
<b>Создать клиентскую схему</b>	<b>17</b>
Создание новой схемы невизуального модуля	17
Создание новой схемы модели представления	19
Создание замещающей схемы	20
<b>Создать схему объекта</b>	<b>22</b>
Создать пользовательскую схему объекта	23
Указать родительский объект	25
Добавить пользовательскую колонку в объект	27
Добавить индексы в объект	29
Создать замещающую схему объекта	29
Сохранить и опубликовать объект	30
<b>Создать схему [Исходный код]</b>	<b>30</b>

# Разработка в Creatio IDE



Раздел [ *Конфигурация* ] ([ *Configuration* ]) предназначен для управления конфигурационными элементами, при помощи которых реализована функциональность системы.

Используя инструменты раздела [ *Конфигурация* ] ([ *Configuration* ]) можно выполнять следующие действия:

- Управлять пакетами, из которых состоит конфигурация, а также их содержимым.
- Расширять и модифицировать функциональность Creatio.
- Организовывать работу с хранилищами системы контроля версий.
- Управлять разработкой функциональности и переносом изменений между рабочими средами.

## Переход в раздел [ *Конфигурация* ] ([ *Configuration* ])

В Creatio реализованы следующие способы перехода в раздел [ *Конфигурация* ] ([ *Configuration* ]):

- Через дизайнер системы по кнопке . В блоке [ *Конфигурирование разработчиком* ] ([ *Admin area* ]) перейдите по ссылке [ *Управление конфигурацией* ] ([ *Advanced settings* ]).
- По ссылке [Адрес приложения Creatio]/0/ClientApp/#/WorkspaceExplorer. Например, <http://my.creatio.com/0/ClientApp/#/WorkspaceExplorer>.
- По алиасу /we. Например, <http://my.creatio.com/0/we>.
- По алиасу /conf. Например, <http://my.creatio.com/0/conf>.
- По алиасу /dev. Например, <http://my.creatio.com/0/dev>.

**На заметку.** Для фреймворка [ASP.NET Core](#) приставку /0 использовать не нужно.

Раздел [ *Конфигурация* ] ([ *Configuration* ]) будет открыт в новой вкладке.

## Переключение между версиями интерфейса раздела [ *Конфигурация* ] ([ *Configuration* ])

По умолчанию раздел [ *Конфигурация* ] ([ *Configuration* ]) отображается в интерфейсе 5.X. Для Creatio версии 7.16.3 реализована возможность отображения раздела в интерфейсе 7.X.

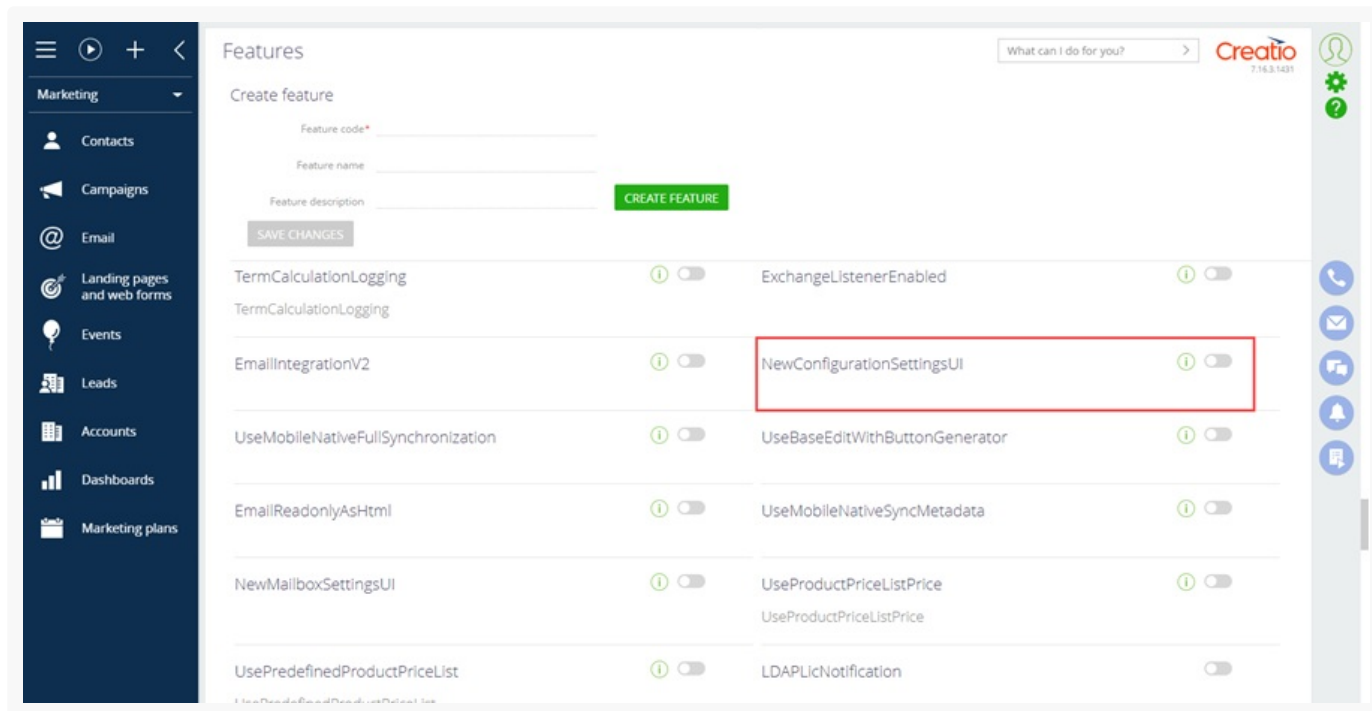
Чтобы отобразить раздел в интерфейсе 7.X, необходимо включить опцию дополнительной функциональности `NewConfigurationSettingsUI`, которая переключает пункт меню [ *Управление конфигурацией* ] ([ *Advanced settings* ]) дизайнера системы.

Для включения опции `NewConfigurationSettingsUI` необходимо выполнить следующие действия:

1. Перейти на страницу дополнительной функциональности [FeaturesPage](#) по ссылке

[Адрес приложения Creatio]/0/ViewModule.aspx#BaseSchemaModuleV2/FeaturesPage . Например, <http://my.creatio.com/0/Nui/ViewModule.aspx#BaseSchemaModuleV2/FeaturesPage> .

2. Проверить значение опции `NewConfigurationSettingsUI` . Опция должна быть включена. Если опция отключена, измените ее значение с помощью переключателя и сохраните изменения нажатием [ Сохранить изменения ] ([ Save changes ]).



**На заметку.** Чтобы при включенной опции дополнительной функциональности `NewConfigurationSettingsUI` отобразить раздел [ Конфигурация ] ([ Configuration ]) в интерфейсе 5.X, используйте ссылку [Адрес приложения Creatio]/0/dev\_old . Например, [http://my.creatio.com/0/dev\\_old](http://my.creatio.com/0/dev_old) .

## Управление доступом к разделу [ Конфигурация ] ([ Configuration ])

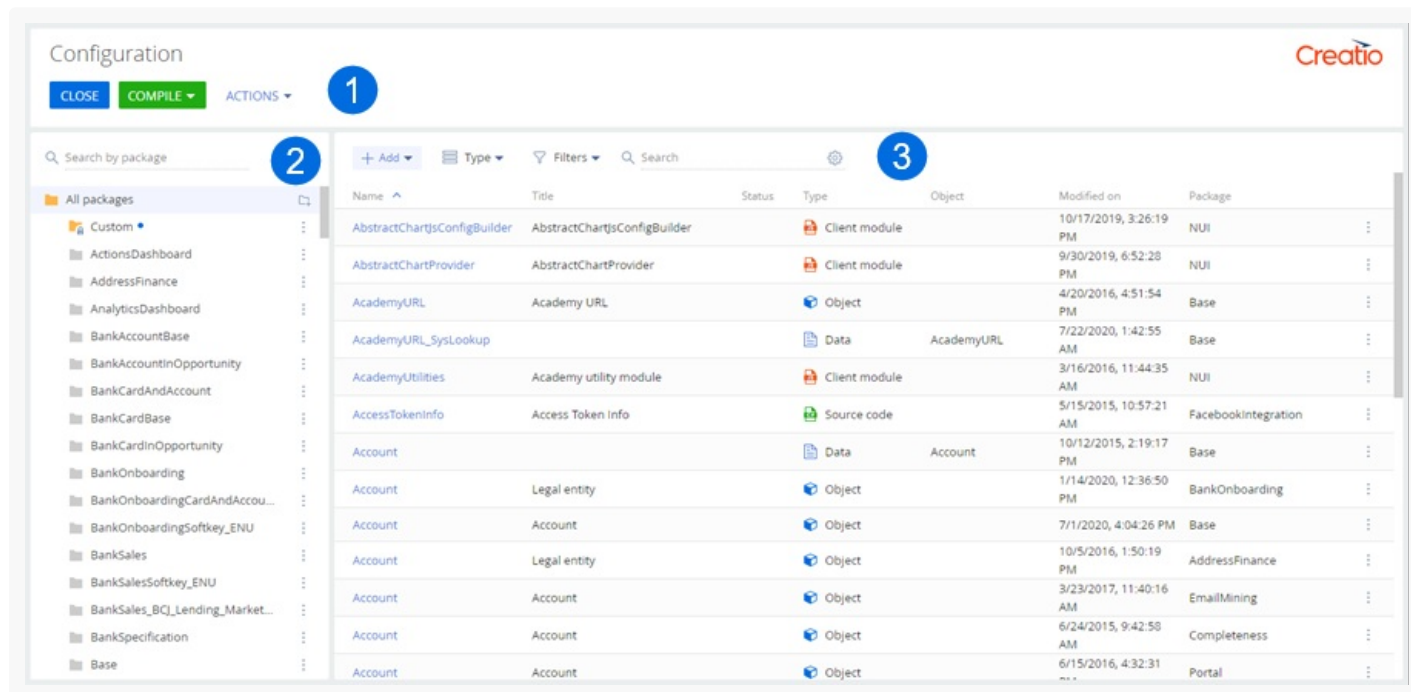
По умолчанию доступ к основным системным операциям есть только у администраторов системы. Но его можно [настроить](#) для определенных пользователей или групп пользователей. Чтобы предоставить доступ к разделу [ Конфигурация ] ([ Configuration ]), выполните следующие действия:

1. Перейдите в дизайнер системы по кнопке . В блоке [ Пользователи и администрирование ] ([ Users and administration ]) перейдите по ссылке [ Права доступа на операции ] ([ Operation permissions ]).
2. Выберите системную операцию [ Доступ к разделу "Конфигурация" ] ([ Access to "Configuration" section ]) (код [ CanManageSolution ]).
3. На детали [ Доступ к операции ] ([ Operation permission ]) нажмите и укажите получателя прав. Запись появится на детали в колонке [ Уровень доступа ] ([ Access level ]) со значением [ Да ] ([ Yes ]), а пользователи, входящие в указанную роль, получают доступ к системной операции [ Доступ к разделу "Конфигурация" ] ([ Access to "Configuration" section ]).

**Важно.** Если у пользователя нет доступа в раздел [ Конфигурация ] ([ Configuration ]), то ему выдается стандартное сообщение с указанием операции и недостающих прав.

## Описание интерфейса раздела [ Конфигурация ] ([ Configuration ])

Интерфейс раздела [ Конфигурация ] ([ Configuration ]), который состоит из нескольких функциональных областей.



## Панель инструментов

На панели инструментов расположены следующие элементы:

- кнопка [ Закрыть ] ([ Close ]) — закрыть вкладку с разделом [ Конфигурация ] ([ Configuration ]).
- кнопка [ Компилировать ] ([ Compile ]) — запуск компиляции конфигурации. Нажатие на кнопку запускает компиляцию изменений в конфигурации. Выбор пункта [ Перекомпилировать все ] ([ Compile all ]) в выпадающем списке кнопки [ Компилировать ] ([ Compile ]) запускает компиляцию конфигурации всех без исключения конфигурационных элементов. В результате будут обновлены исполняемые файлы и [статический контент](#) будет выгружен в каталог `...\Terrasoft.WebApp\conf`. После завершения компиляции пользователь получит уведомление о ее завершении и изменения вступят в силу для пользователей, работающих в текущей конфигурации.

**На заметку.** Если в процессе компиляции возникли ошибки, то они будут отображены в диалоговом окне, которое содержит следующую информацию:

- иконку типа ошибки (ошибка или предупреждение ).

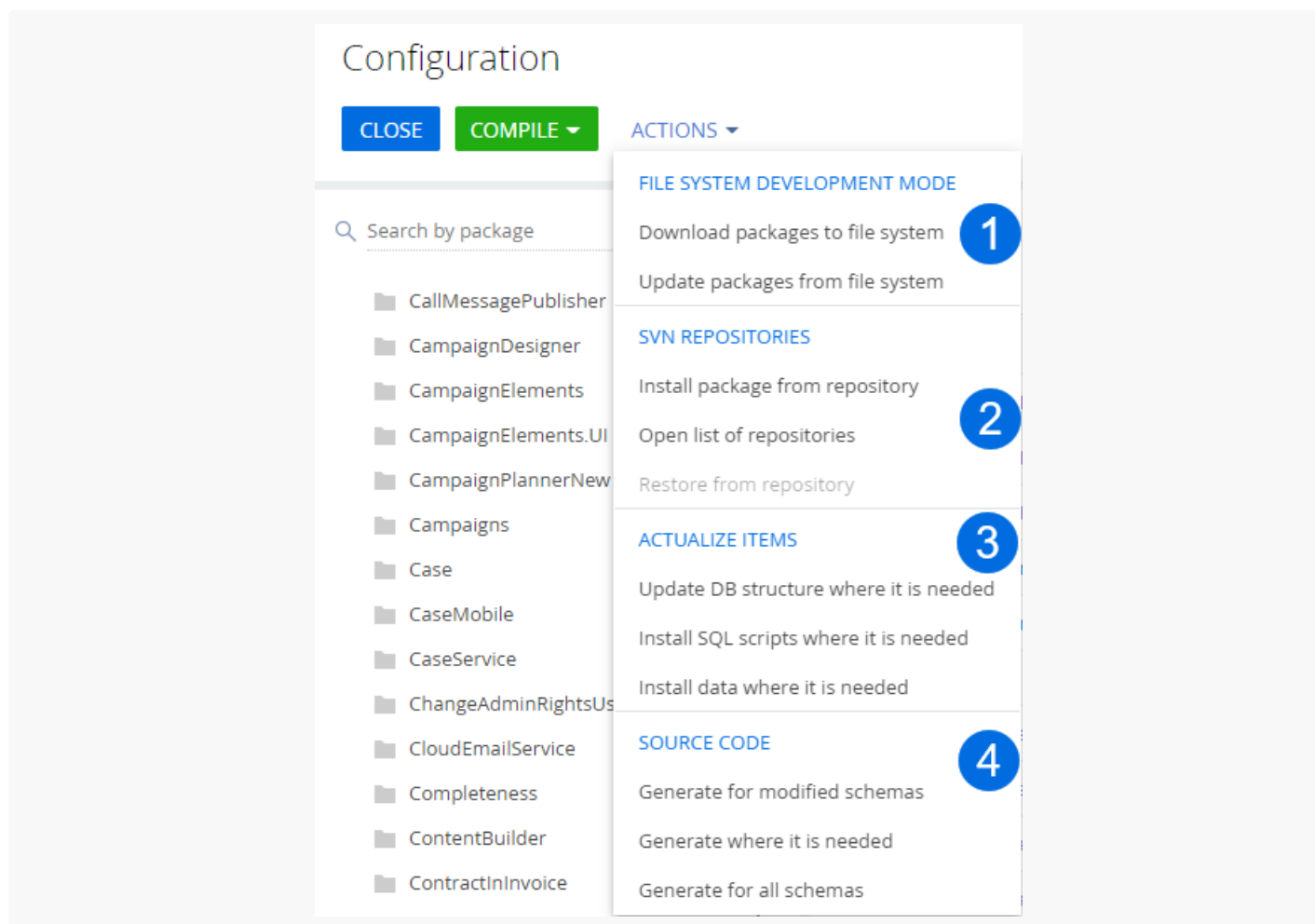
- имя файла с ошибкой.
- описание ошибки.
- код ошибки.
- номер строки с ошибкой.

- выпадающий список [ Действия ] ([ Actions ]) — содержит перечень действий раздела [ Конфигурация ] ([ Configuration ]). Действия разделены на несколько групп.

Действия раздела [ Конфигурация ] ([ Configuration ]) включают в себя следующие группы:

- [ Разработка в файловой системе ] ([ File system development mode ]) — действия, связанные с режимом разработки в файловой системе.
- [ Хранилища SVN ] ([ SVN repositories ]) — действия по работе с хранилищами системы контроля версий.
- [ Актуализировать элементы ] ([ Actualize items ]) — действия по актуализации конфигурационных элементов.
- [ Исходный код ] ([ Source code ]) — действия по работе с исходным кодом.

При наведении курсора на название группы, отображается подсказка по включению режима.



Группа действий [ *Разработка в файловой системе* ] ([ *File system development mode* ]) (1) содержит следующие пункты:

- [ *Выгрузить все пакеты в файловую систему* ] ([ *Download packages to file system* ]) — загружает пакеты из базы данных приложения в каталог `...\Terrasoft.WebApp\Terrasoft.Configuration\Pkg`.
- [ *Обновить пакеты из файловой системы* ] ([ *Update packages from file system* ]) — загружает пакеты из каталога `...\Terrasoft.WebApp\Terrasoft.Configuration\Pkg` в базу данных.

**Важно.** Пункты группы действий [ *Разработка в файловой системе* ] ([ *File system development mode* ]) (1) доступны только в [режиме разработки в файловой системе](#).

Группа действий [ *Хранилища SVN* ] ([ *SVN repositories* ]) (2) содержит следующие пункты:

- [ *Открыть список хранилищ* ] ([ *Open list of repositories* ]) — открывает новую вкладку [ *Список хранилищ* ] ([ *List of repositories* ]), при помощи которой можно создавать, настраивать и удалять ссылки на доступные хранилища системы контроля версий.

Группа действий [ *Актуализировать элементы* ] ([ *Actualize items* ]) (3) содержит следующие пункты:


- [ *Обновить структуру БД для требующих обновления* ] ([ *Update DB structure where it is needed* ]) — обновляет структуру базы данных для конфигурационных элементов, которые требуют обновления. После завершения обновления структуры БД пользователь в диалоговом окне получит уведомление.
- [ *Установить SQL сценарии для требующих установки* ] ([ *Install SQL scripts where it is needed* ]) — установить SQL сценарии, которые требуют установки. После завершения установки SQL сценариев пользователь в диалоговом окне получит уведомление.
- [ *Установить данные для требующих установки* ] ([ *Install data where it is needed* ]) — установить данные, которые требуют установки. После завершения установки данных пользователь в диалоговом окне получит уведомление.

Группа действий [ *Исходный код* ] ([ *Source code* ]) (4) содержит следующие пункты:

- [ *Сгенерировать для измененных* ] ([ *Generate for modified schemas* ]) — генерирует исходный код для схем, которые были изменены в текущей конфигурации.
- [ *Сгенерировать для требующих генерации* ] ([ *Generate where it is needed* ]) — генерирует исходный код для всех схем, которые требуют генерации исходного кода.
- [ *Сгенерировать для всех схем* ] ([ *Generate for all schemas* ]) — генерирует исходный код для всех без исключения схем текущей конфигурации. Эта операция может занять длительное время (больше 10 минут).

## Область работы с пакетами




Область работы с пакетами состоит из следующих элементов:


- строка поиска [ *Поиск по пакетам* ] ([ *Search by package* ]) — выполнить поиск пакета по его имени.
- кнопка  — [создать новый пакет](#). При нажатии на кнопку будет отображено окно для создания нового пакета, в котором можно задать название и описание пакета, добавить [зависимости](#), а также указать репозиторий.




- группа [ *Все пакеты* ] ([ *All packages* ]) — содержит перечень всех пакетов приложения в алфавитном порядке. Вверху перечня отображаются измененные пакеты и пакеты, доступные для редактирования. При выборе группы [ *Все пакеты* ] ([ *All packages* ]) конфигурационные элементы всех пакетов приложения в алфавитном порядке будут отображены в рабочей области раздела [ *Конфигурация* ] ([ *Configuration* ]). При выборе одного пакета группы [ *Все пакеты* ] ([ *All packages* ]) в алфавитном порядке будут отображены конфигурационные элементы текущего пакета.

**На заметку.** Группа [ *Все пакеты* ] ([ *All packages* ]) может включать в себя следующие типы пакетов:

-  — предустановленные пакеты (недоступны для изменения).
-  — пользовательские пакеты, недоступные для изменения по причине создания другими пользователями системы.
-  — пользовательские пакеты, созданные текущим пользователем и доступны к изменению.

**На заметку.** Измененные пакеты содержат символ  возле имени пакета и находятся вверху перечня пакетов группы [ *Все пакеты* ] ([ *All packages* ]).

При нажатии на  в строке с названием пакета будет отображено меню, которое состоит из следующих пунктов:

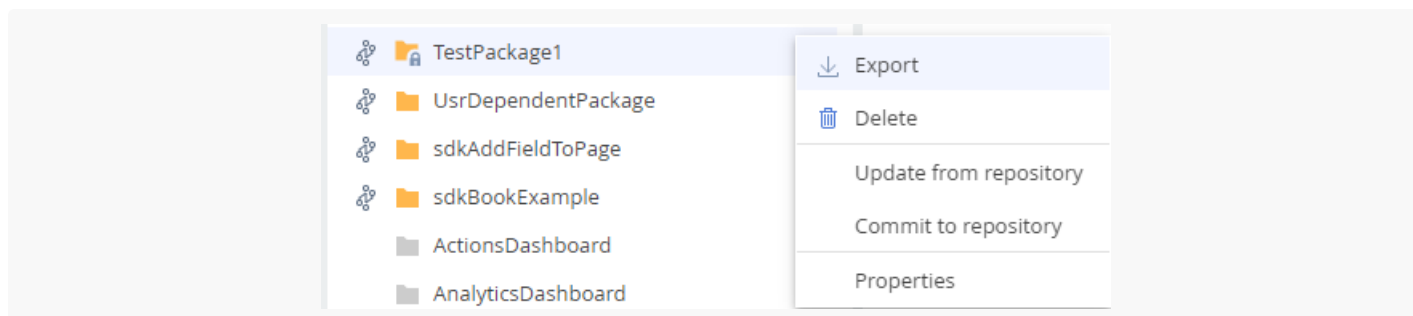
- [ *Экспортировать* ] ([ *Export* ]) — [выгрузить пакет в zip-архив](#).
- [ *Удалить* ] ([ *Delete* ]) — удалить пакет. Неактивна для предустановленных пакетов.

**Важно.** Можно удалить следующие пакеты:

- пустые пакеты.
- пакеты с конфигурационными элементами, которые не являются родительскими.

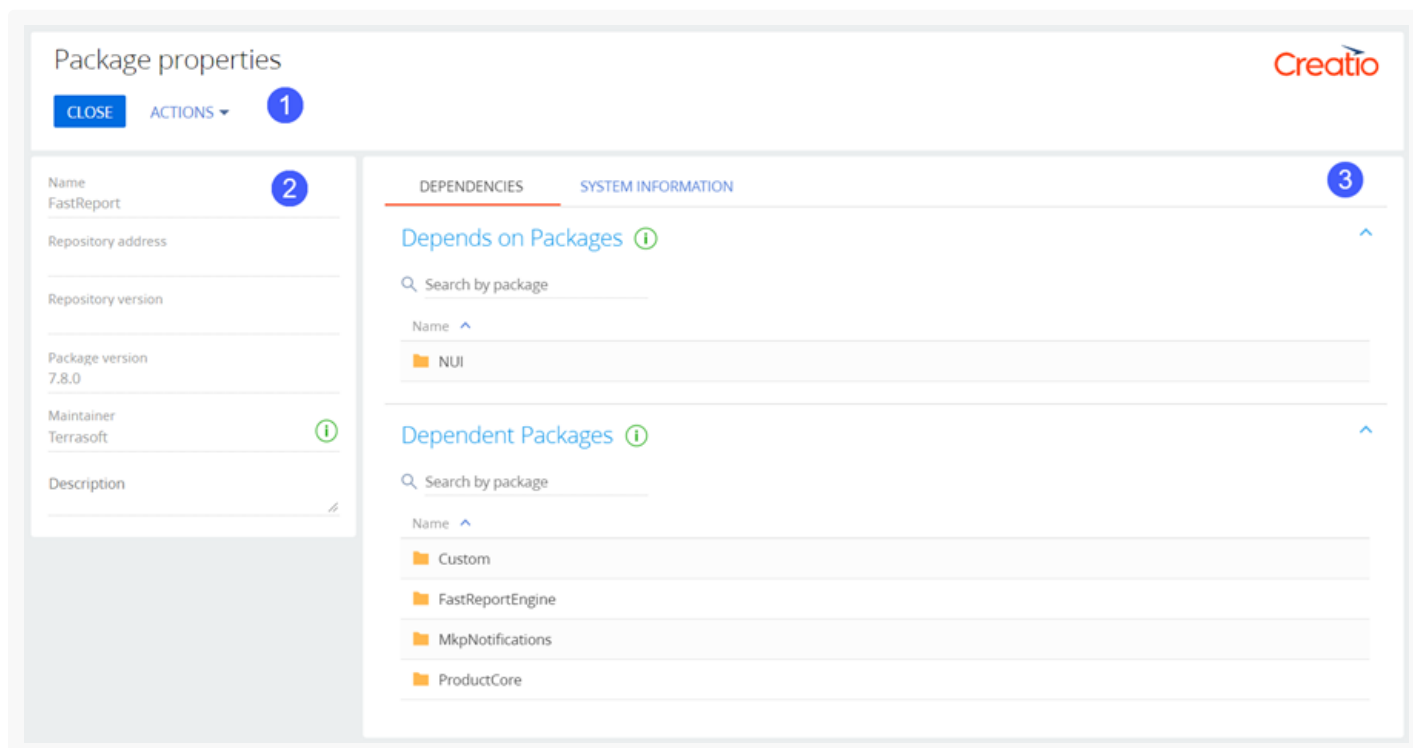
При попытке удалить пакет с родительскими конфигурационными элементами будет отображено диалоговое окно с перечнем зависимых пакетов и зависимых от конфигурационных элементов удаляемого пакета элементами, которые препятствуют удалению пакета.

- [ *Обновить из хранилища* ] ([ *Update from repository* ]) — [обновление пакета](#) из подключенного хранилища системы контроля версий.
- [ *Зафиксировать в хранилище* ] ([ *Commit to repository* ]) — [фиксация пакета](#) в подключенном хранилище системы контроля версий.
- [ *Свойства* ] ([ *Properties* ]) — открыть страницу [ *Свойства пакета* ] ([ *Package properties* ]). Страница будет открыта в новой вкладке. На странице можно настроить зависимости текущего пакета, просмотреть системную информацию (кто создал и отредактировал пакет, даты создания и изменения, уникальный идентификатор, первичный ключ пакета в таблице базы данных). Также открыть свойства пакета можно двойным кликом по имени пакета.



Интерфейс страницы [ *Свойства пакета* ] ([ *Package properties* ]) состоит из нескольких функциональных областей.

Свойства пакета



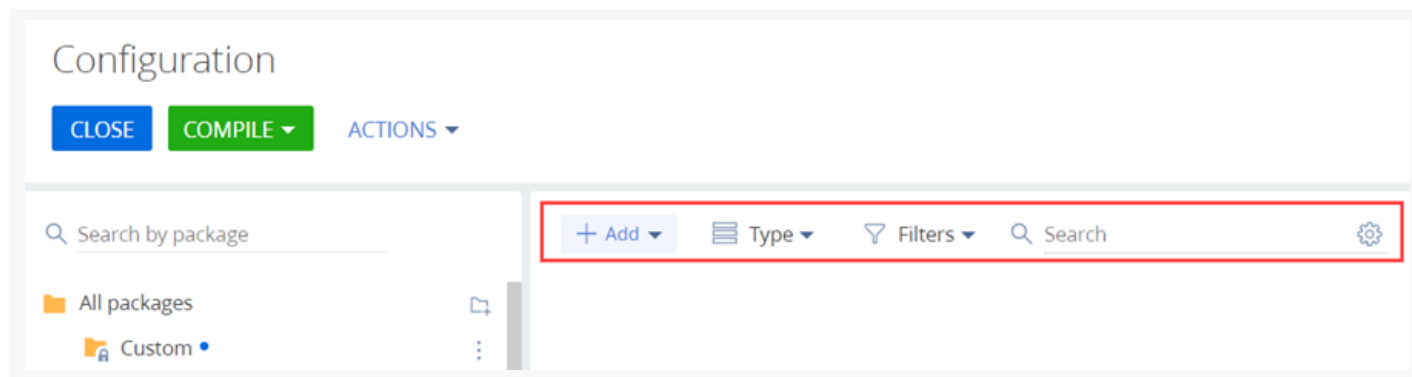
## Рабочая область

В рабочей области раздела осуществляется основная работа с конфигурационными элементами. Рабочая область содержит следующие элементы:

- панель инструментов — позволяет добавлять и настраивать отображение конфигурационных элементов в реестре раздела, а также выполнить поиск.
- реестр раздела, который содержит перечень конфигурационных элементов.

## Панель инструментов

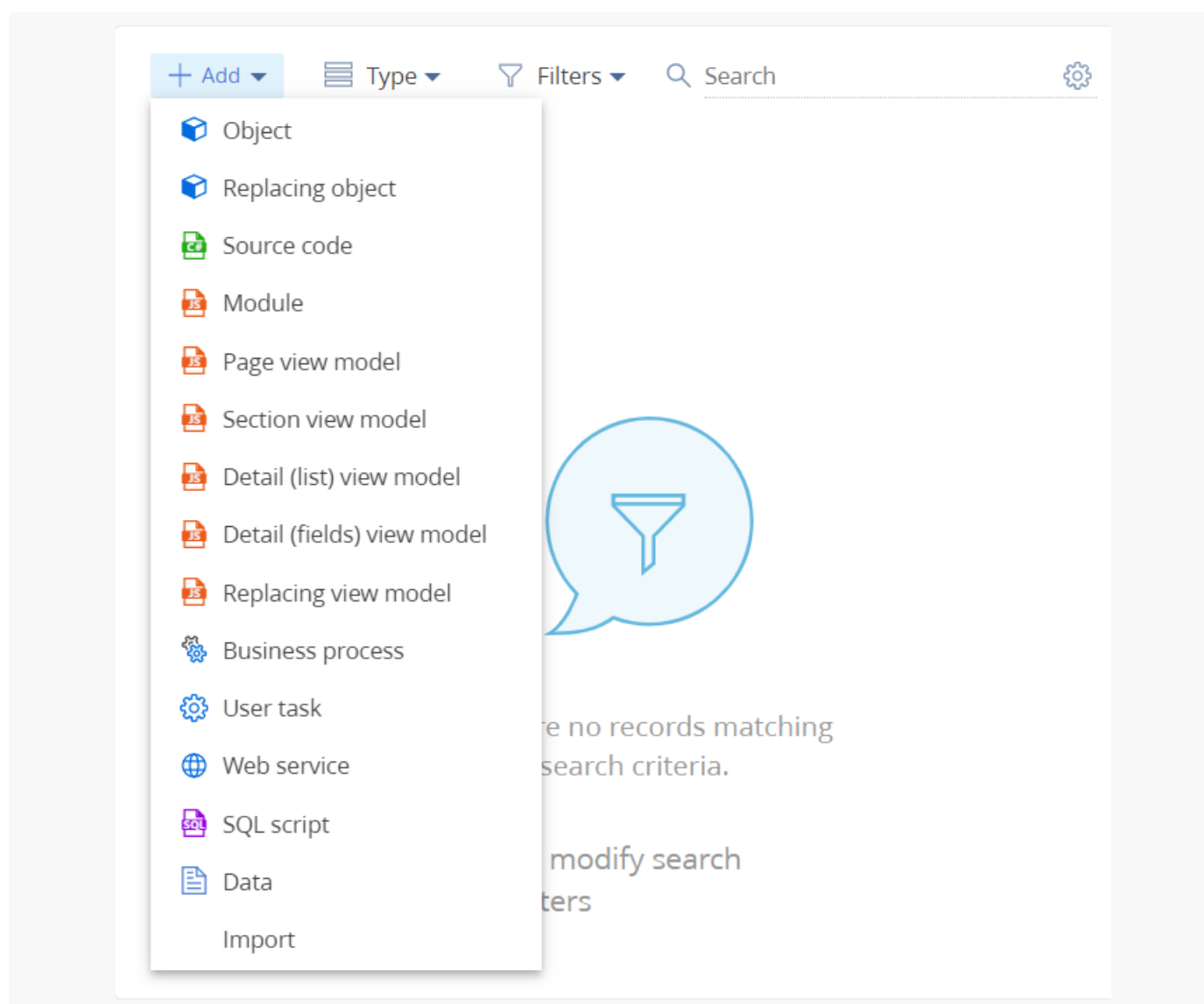
Панель инструментов предназначена для добавления и настройки отображения конфигурационных элементов в реестре раздела, а также выполнения поиска.



Панель инструментов содержит следующие элементы:

- выпадающий список [ Добавить ] ([ Add ]) — выбор конфигурационного элемента, который необходимо добавить.

Конфигурационные элементы для добавления



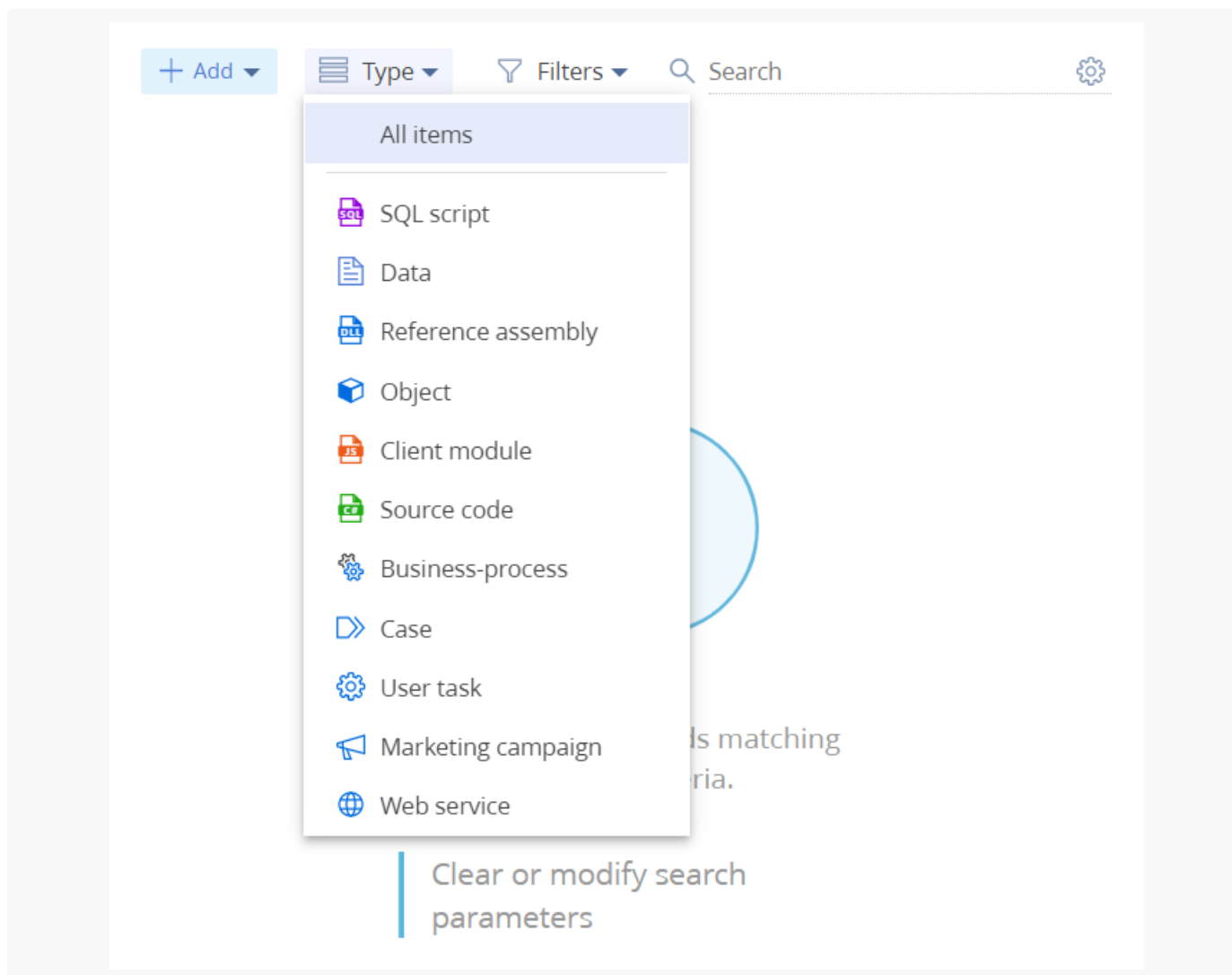
**Важно.** Пункты выпадающего списка [ *Добавить* ] ([ *Add* ]) неактивны для предустановленных пакетов.

Чтобы добавить конфигурационный элемент необходимо предварительно выбрать пакет. Без выбора пакета пункты выпадающего списка [ *Добавить* ] ([ *Add* ]) остаются неактивными.

Выбор пункта [ *Импортировать* ] ([ *Import* ]) в выпадающем списке [ *Добавить* ] ([ *Add* ]) позволяет загрузить в пользовательский пакет любой конфигурационный элемент с расширением \*.md или \*.d11 кроме [ *SQL сценарий* ] ([ *SQL script* ]) и [ *Данные* ] ([ *Data* ]).

- выпадающий список [ *Тип* ] ([ *Type* ]) — выбор типа конфигурационных элементов, которые необходимо отобразить в реестре раздела.

Типы конфигурационных элементов для отображения



**На заметку.** Значение [ *Клиентский модуль* ] ([ *Client module* ]) колонки [ *Тип* ] ([ *Type* ]) содержат следующие конфигурационные элементы:

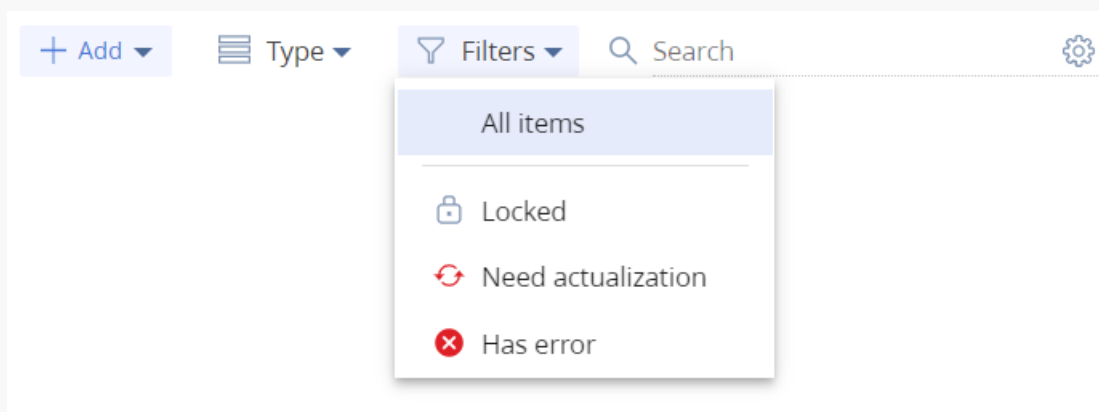
- [ *Модуль* ] ([ *Module* ]).
- [ *Модель представления страницы* ] ([ *Page view model* ]).

- [ Модель представления раздела ] ([ Section view model ]).
- [ Модель представления детали с реестром ] ([ Detail (list) view model ]).
- [ Модель представления детали с полями ] ([ Detail (fields) view model ]).
- [ Замещающая модель представления ] ([ Replacing view model ]).

Значение [ Объект ] ([ Object ]) колонки [ Тип ] ([ Type ]) содержат конфигурационные элементы [ Объект ] ([ Object ]) и [ Замещающий объект ] ([ Replacing object ]).

- выпадающий список [ Фильтры ] ([ Filters ]) — выбор состояния конфигурационных элементов, которые необходимо отобразить в реестре раздела.


Состояния конфигурационных элементов для отображения

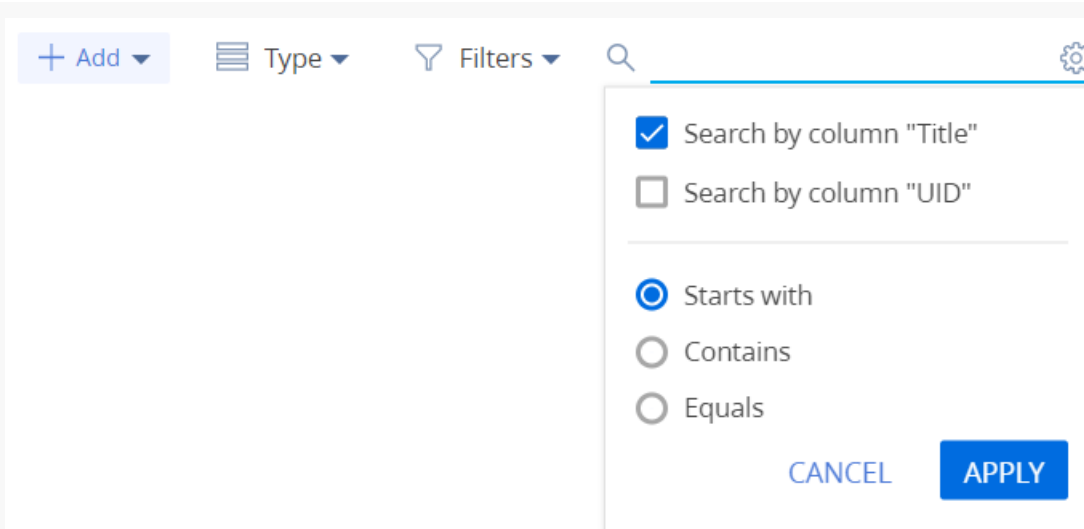



**На заметку.** Значение [ Требующие актуализации ] ([ Need actualization ]) выпадающего списка [ Фильтры ] ([ Filters ]) может использоваться для следующих конфигурационных элементов:

- [ SQL сценарий ] ([ SQL script ]) и [ Данные ] ([ Data ]) — при установленном свойстве [ Требуется установки в БД ] ([ Needs to be installed in database ]).
- [ Объект ] ([ Object ]) — при установленном свойстве [ Требуется обновление в БД ] ([ Needs to be updated in database ]).
- [ Клиентский модуль ] ([ Client module ]), [ Исходный код ] ([ Source code ]), [ Кейс ] ([ Case ]), [ Маркетинговая кампания ] ([ Marketing campaign ]), [ Бизнес-процесс ] ([ Business-process ]), [ Действие процесса ] ([ User task ]) — при установленном свойстве [ Требуется генерации исходного кода ] ([ Needs generate source code ]).

Настройки выпадающего списка [ Фильтры ] ([ Filters ]) сохраняются в профиле пользователя и отображаются при последующем входе в раздел [ Конфигурация ] ([ Configuration ]).

- строка поиска [ Поиск ] ([ Search ]) — выполнить поиск конфигурационного элемента по его имени в реестре раздела. По нажатию на  можно задать дополнительные параметры поиска. Поиск выполняется в рамках выделенного пакета. Чтобы выполнять поиск по всем конфигурационным элементам, необходимо перейти в группу [ Все пакеты ] ([ All packages ]).



**На заметку.** По нажатию на  можно задать следующие параметры поиска:

- опция [ Поиск по полю "Заголовок" ] ([ Search by column "Title" ]) — поиск по заголовку (по умолчанию включен).
- опция [ Поиск по полю "UID" ] ([ Search by column "UID" ]) — поиск по уникальному идентификатору (по умолчанию выключен).
- пункт [ Начинается ] ([ Starts with ]) — режим поиска (по умолчанию включен). Название конфигурационного элемента начинается с текста, который введен в строке поиска [ Поиск ] ([ Search ]).
- пункт [ Содержит ] ([ Contains ]) — режим поиска (по умолчанию выключен). Название конфигурационного элемента содержит текст, который введен в строке поиска [ Поиск ] ([ Search ]).
- пункт [ Равно ] ([ Equals ]) — режим поиска (по умолчанию выключен). Название конфигурационного элемента полностью соответствует тексту, который введен в строке поиска [ Поиск ] ([ Search ]).



Для применения внесенных изменений нажмите [ Применить ] ([ Apply ]).

Настройки поиска сохраняются в профиле пользователя и отображаются при последующем входе в раздел [ Конфигурация ] ([ Configuration ]).



## Реестр раздела


Реестр раздела содержит перечень конфигурационных элементов. Данные представлены в виде таблицы, которая состоит из следующих колонок:

- [ Название ] ([ Name ]) — имя конфигурационного элемента, которое было задано при его создании. Все конфигурационные элементы отсортированы в алфавитном порядке.
- [ Заголовок ] ([ Title ]) — заголовок конфигурационного элемента.
- [ Статус ] ([ Status ]) — состояние конфигурационного элемента. Содержит данные с выпадающего списка [ Фильтры ] ([ Filters ]).

**На заметку.** Колонка [ Статус ] ([ Status ]) будет содержать  если необходима актуализации конфигурационного элемента. При наведении курсора мыши на , можно увидеть следующие всплывающие подсказки:


- [ Требуется установки в БД ] ([ Needs to be installed in database ]) — для конфигурационных элементов [ SQL сценарий ] ([ SQL script ]) и [ Данные ] ([ Data ]).
- [ Требуется обновление в БД ] ([ Needs to be updated in database ]) — для конфигурационного элемента [ Объект ] ([ Object ]).
- [ Требуется генерации исходного кода ] ([ Needs generate source code ]) — для конфигурационных элементов [ Клиентский модуль ] ([ Client module ]), [ Исходный код ] ([ Source code ]), [ Кейс ] ([ Case ]), [ Маркетинговая кампания ] ([ Marketing campaign ]), [ Бизнес-процесс ] ([ Business-process ]), [ Действие процесса ] ([ User task ]).

Колонка [ Статус ] ([ Status ]) будет содержать  если конфигурационный элемент содержит ошибку. При наведении курсора мыши на , можно увидеть текстовое описание ошибки.

Если необходима актуализация конфигурационного элемента и при этом он содержит ошибку, то колонка [ Статус ] ([ Status ]) будет содержать , а при наведении курсора мыши можно увидеть информацию о необходимости актуализации конфигурационного элемента и текстовое описание ошибки.

- [ Тип ] ([ Type ]) — тип конфигурационного элемента, который был выбран при создании конфигурационного элемента в выпадающем списке [ Добавить ] ([ Add ]).
- [ Объект ] ([ Object ]) — объект, с которым связаны привязываемые к пакету данные. Колонка заполняется только для конфигурационного элемента [ Данные ] ([ Data ]).
- [ Дата изменения ] ([ Modified on ]) — дата изменения текущего конфигурационного элемента.
- [ Пакет ] ([ Package ]) — название пакета, который содержит текущий конфигурационный элемент.

**На заметку.** Отсортировать данные в колонке по возрастанию или по убыванию можно нажатием по имени колонки.

При нажатии на  в строке конфигурационного элемента будет отображено меню, которое зависит от значения колонки [ Тип ] ([ Type ]).

Меню конфигурационных элементов [ SQL сценарий ] ([ SQL script ]) и [ Данные ] ([ Data ]) содержит следующие пункты:

- [ Удалить ] ([ Delete ]) — удалить конфигурационный элемент. Неактивно для конфигурационных элементов предустановленных пакетов.
- [ Установить ] ([ Install ]) — установить конфигурационный элемент (для элемента [ SQL сценарий ] ([ SQL script ]) выполнится установка в базу данных, для элемента [ Данные ] ([ Data ]) выполнится установка данных для объекта колонки [ Объект ] ([ Object ])).

**На заметку.** При некорректной установке данных для объекта колонки [ Объект ] ([ Object ]) можно посмотреть описание ошибки в колонке [ Статус ] ([ Status ]). Описание ошибки также можно посмотреть в свойствах конфигурационных элементов.

- [ Отменить изменения ] ([ *Discard changes* ]) — отменить внесенные изменения. Пункт доступен если пакет с конфигурационным элементом подключен к репозиторию.
- [ Свойства ] ([ *Properties* ]) — открыть окно свойств конфигурационного элемента.

Меню конфигурационного элемента [ *Внешняя сборка* ] ([ *Reference assembly* ]) содержит следующие пункты:

- [ Удалить ] ([ *Delete* ]) — удалить конфигурационный элемент. Неактивно для конфигурационных элементов предустановленных пакетов.
- [ Отменить изменения ] ([ *Discard changes* ]) — отменить внесенные изменения. Пункт доступен если пакет с конфигурационным элементом подключен к репозиторию.
- [ Свойства ] ([ *Properties* ]) — открыть окно свойств конфигурационного элемента.

Меню конфигурационного элемента [ *Объект* ] ([ *Object* ]) содержит следующие пункты:

- [ Изменить процесс ] ([ *Edit process* ]) — открыть объект для редактирования. Если объект был создан третьими лицами, будет отображено соответствующее уведомление.
- [ Экспортировать ] ([ *Export* ]) — скачать объект. Объект будет скачан в формате \*.md .
- [ Удалить ] ([ *Delete* ]) — удалить конфигурационный элемент. Неактивно для конфигурационных элементов предустановленных пакетов.
- [ Обновить структуру БД ] ([ *Update database structure* ]) — обновить структуру базы данных для текущего объекта.
- [ Сгенерировать исходный код ] ([ *Generate source code* ]) — сгенерировать исходный код текущего конфигурационного элемента. Будет выполнено если процесс содержит компилируемые элементы.
- [ Отменить изменения ] ([ *Discard changes* ]) — отменить внесенные изменения. Пункт доступен если пакет с конфигурационным элементом подключен к репозиторию.
- [ Свойства ] ([ *Properties* ]) — открыть окно свойств конфигурационного элемента.

Меню конфигурационных элементов [ *Клиентский модуль* ] ([ *Client module* ]), [ *Исходный код* ] ([ *Source code* ]), [ *Кейс* ] ([ *Case* ]) и [ *Маркетинговая кампания* ] ([ *Marketing campaign* ]) содержит следующие пункты:

- [ Экспортировать ] ([ *Export* ]) — скачать конфигурационный элемент. Конфигурационный элемент будет скачан в формате \*.md .
- [ Удалить ] ([ *Delete* ]) — удалить конфигурационный элемент. Неактивно для конфигурационных элементов предустановленных пакетов.
- [ Отменить изменения ] ([ *Discard changes* ]) — отменить внесенные изменения. Пункт доступен если пакет с конфигурационным элементом подключен к репозиторию.
- [ Свойства ] ([ *Properties* ]) — открыть окно свойств конфигурационного элемента.


Меню конфигурационных элементов [ *Бизнес-процесс* ] ([ *Business-process* ]) и [ *Действие процесса* ] ([ *User task* ]) содержит следующие пункты:


- [ Экспортировать ] ([ *Export* ]) — скачать конфигурационный элемент. Конфигурационный элемент будет скачан в формате \*.md .
- [ Удалить ] ([ *Delete* ]) — удалить конфигурационный элемент. Неактивно для конфигурационных



элементов предустановленных пакетов.

- [ *Сгенерировать исходный код* ] ([ *Generate source code* ]) — сгенерировать исходный код текущего конфигурационного элемента. Пункт доступен если элементы бизнес-процесс содержат исходный код.
- [ *Отменить изменения* ] ([ *Discard changes* ]) — отменить внесенные изменения. Пункт доступен если пакет с конфигурационным элементом подключен к репозиторию.
- [ *Свойства* ] ([ *Properties* ]) — открыть окно свойств конфигурационного элемента.

Выполнить удаление конфигурационного элемента можно нажатием на . Данная кнопка появляется при наведении курсора мыши на запись реестра раздела [ *Конфигурация* ] ([ *Configuration* ]) только для конфигурационных элементов, которые были созданы текущим пользователем.

**На заметку.** Измененные конфигурационные элементы содержат символ  возле имени элемента и находятся вверху перечня конфигурационных элементов реестра раздела.

## Создать клиентскую схему

 **Легкий**

**Клиентские модули** — это отдельные блоки функциональности, которые загружаются и запускаются по требованию, в соответствии с технологией AMD. Реализация всей пользовательской функциональности выполняется в клиентских модулях. Несмотря на некоторые функциональные различия, все клиентские модули Creatio имеют одинаковую структуру описания, которая соответствует формату описания модулей AMD. Подробнее о клиентских модулях можно узнать из статьи [Front-end \(JS\)](#).

В Creatio можно выделить несколько разновидностей клиентских модулей:

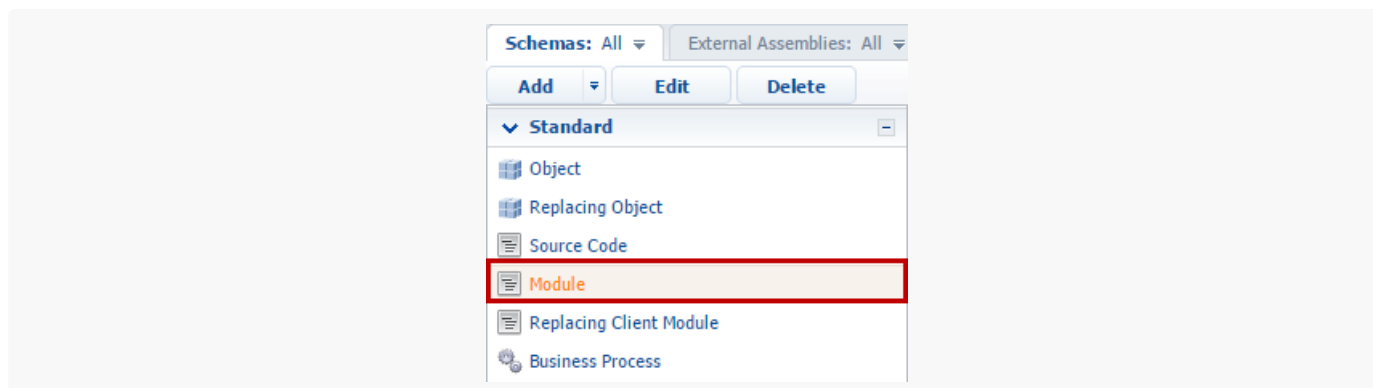
- невизуальный модуль (схема модуля);
- визуальный модуль (схема модели представления);
- модуль расширения и замещающий клиентский модуль (замещающая схема клиентского модуля).

Последовательность создания клиентской схемы несколько отличается для разных типов схем.

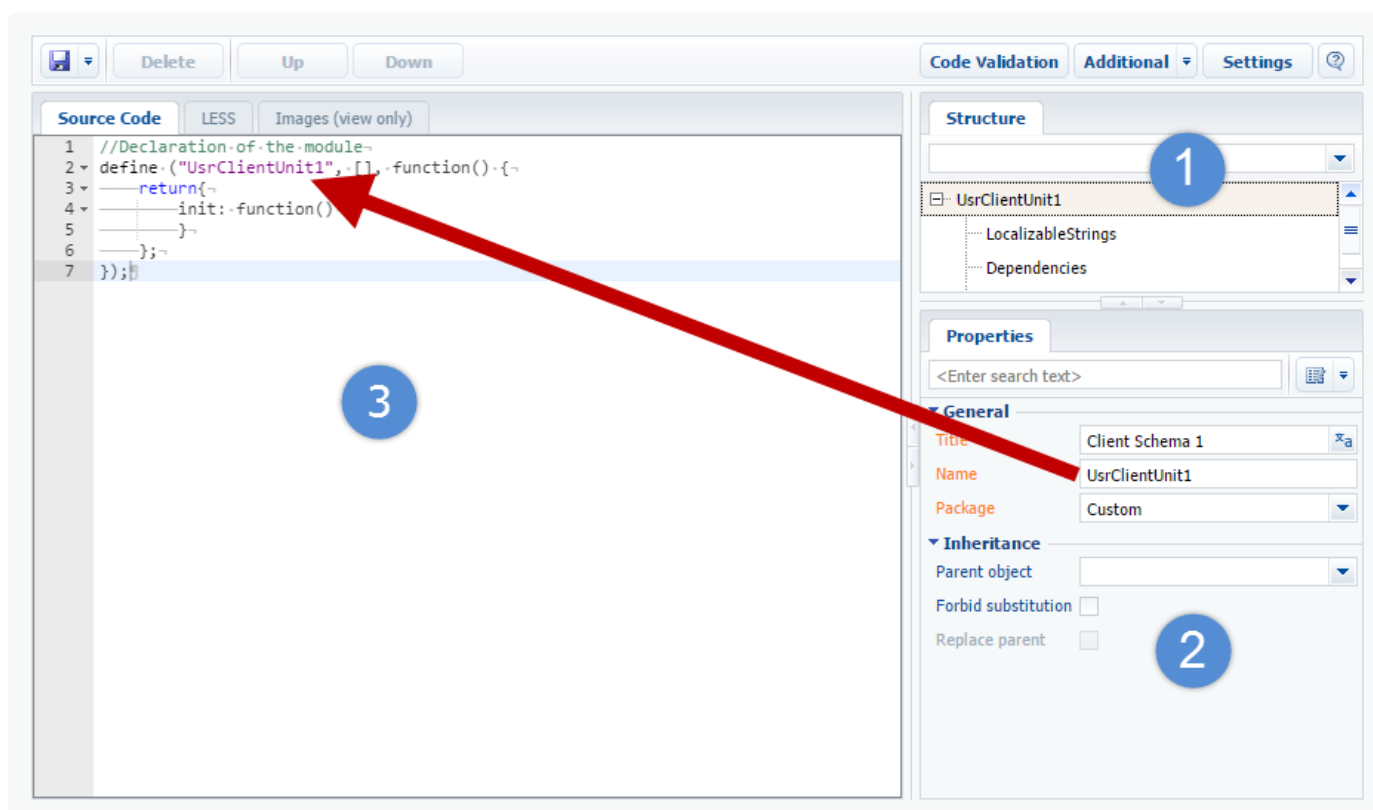
### Создание новой схемы невизуального модуля

Для создания схемы невизуального модуля необходимо выполнить следующую последовательность действий.

1. Перейти в раздел [ *Конфигурация* ] и выбрать пользовательский пакет, в который будет добавлена новая схема.
2. На вкладке [ *Схемы* ] выполнить команду [ *Добавить* ] —> [ *Модуль* ].

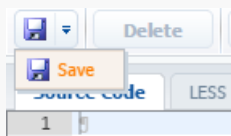


3. В появившемся дизайнера клиентского модуля необходимо выбрать корневой элемент структуры и заполнить свойства создаваемой схемы модуля:



К основным свойствам схемы модуля относятся:

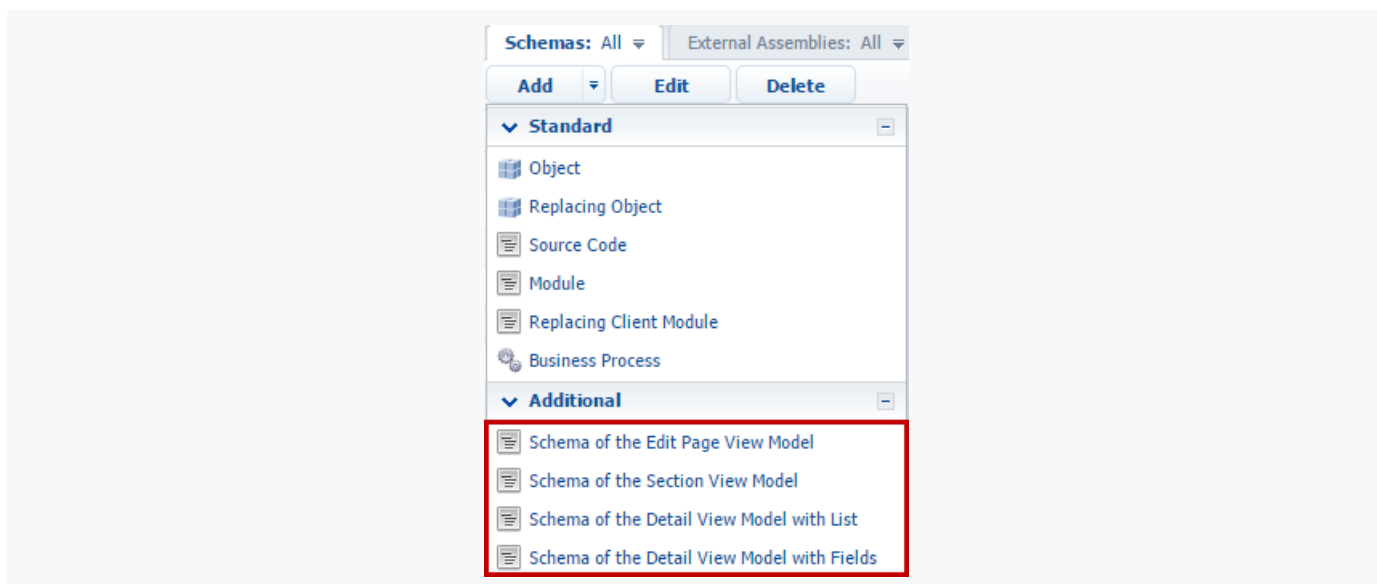
- [ *Название* ] — название схемы. Может содержать только символы латинского алфавита и цифры. Содержит префикс, указанный в системной настройке [ *Префикс названия объекта* ] ( `SchemaNamePrefix` ).
  - [ *Заголовок* ] — заголовок схемы. Может быть локализован.
  - [ *Пакет* ] — пользовательский пакет, в котором создается схема.
4. На вкладке [ *Исходный код* ] дизайнера модуля добавить исходный код модуля. При этом необходимо удостовериться, что название модуля в функции `define()` совпадает с названием схемы модуля.
5. После внесения всех изменений схему модуля нужно сохранить:



## Создание новой схемы модели представления

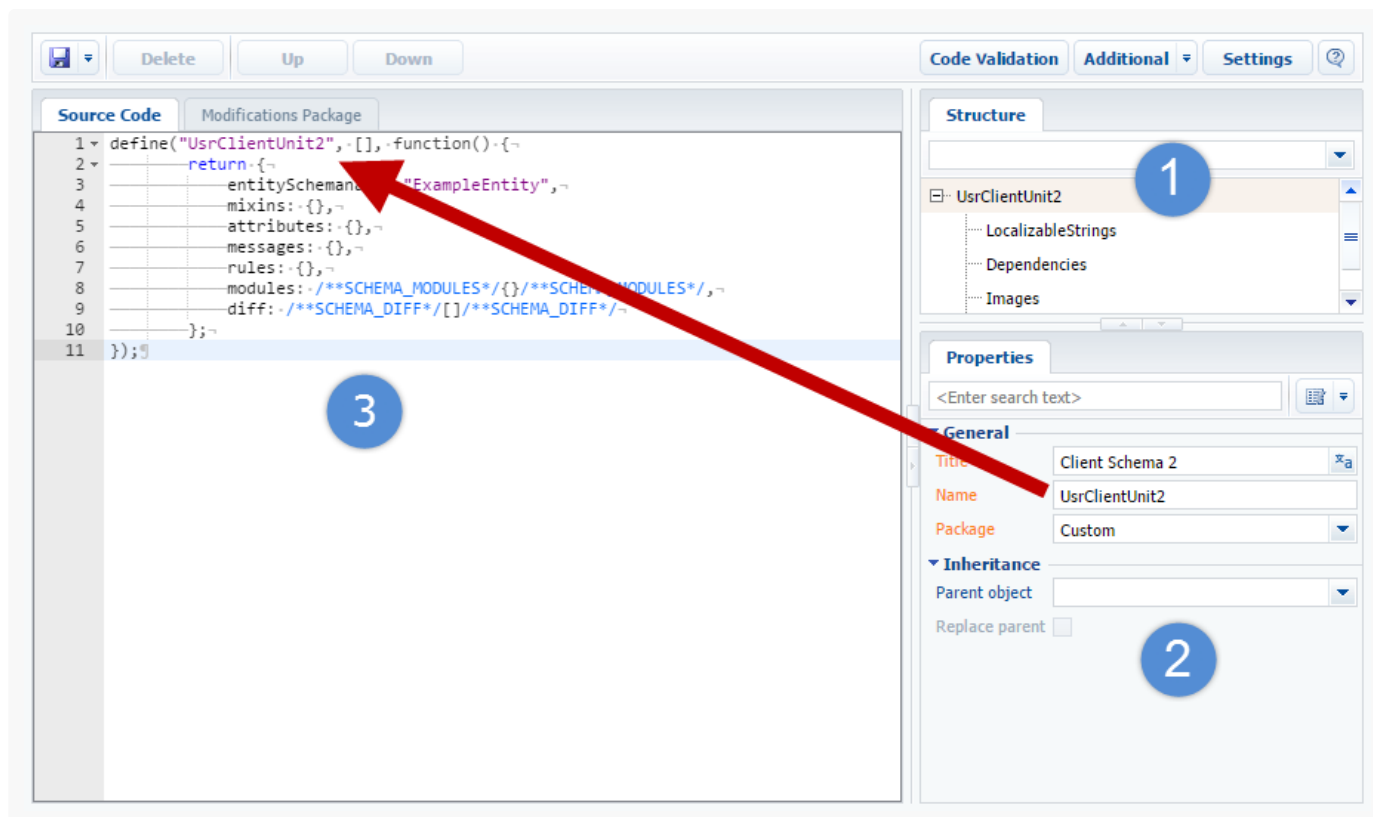
Для создания схемы визуального модуля необходимо выполнить следующую последовательность действий.

1. Перейти в раздел [ *Конфигурация* ] и выбрать пользовательский пакет, в который будет добавлена новая схема.
2. На вкладке [ *Схемы* ] выполнить одну из команд добавления схемы модели представления, доступных в расширенном меню команд:



С помощью этих команд можно добавить следующие типы схем визуальных модулей:

- [ *Схема модели представления карточки* ] — схема страницы редактирования записи раздела.
  - [ *Схема модели представления раздела* ] — схема страницы раздела с реестром и итогами.
  - [ *Схема модели представления детали с реестром* ] — схема страницы редактирования детали с реестром.
  - [ *Схема модели представления детали с полями* ] — схема страницы редактирования детали с полями.
3. В открывшемся дизайнере клиентской схемы модели представления выбрать корневой элемент структуры заполнить свойства создаваемой схемы.



Основные свойства схемы модели представления совпадают с основными свойствами схемы не визуального модуля, приведенными выше.

4. На вкладке [ *Исходный код* ] дизайнера модуля нужно добавить исходный код схемы модели представления. При этом необходимо удостовериться, что название визуального модуля в функции `define()` совпадает с названием схемы модели представления.

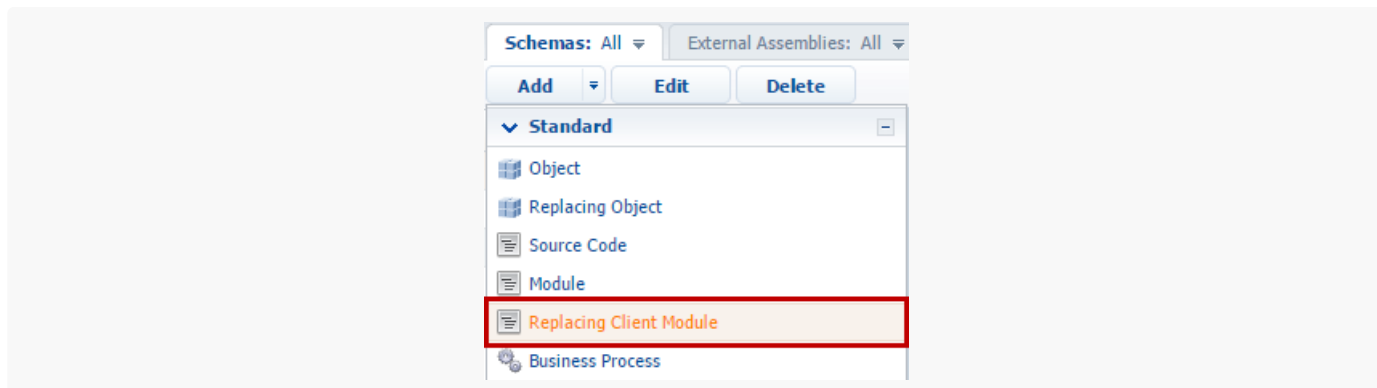
После внесения всех изменений схему модуля нужно сохранить.

## Создание замещающей схемы

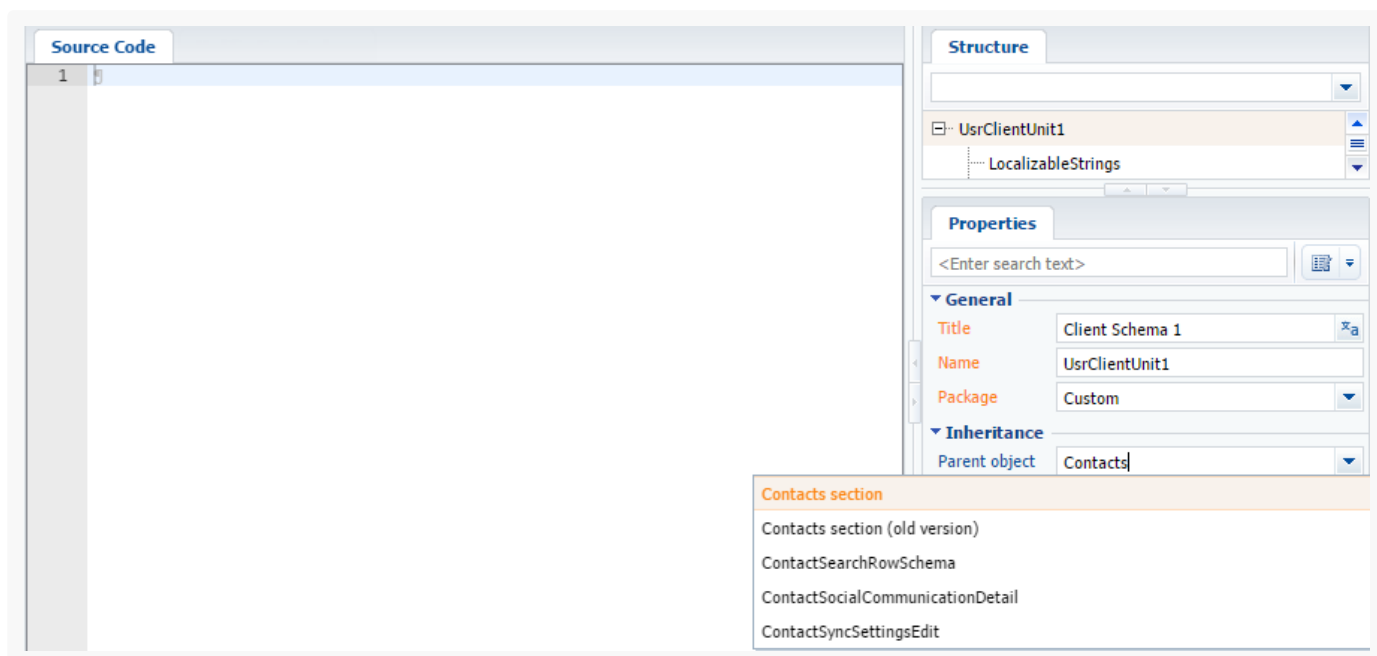
Замещающие схемы предназначены для расширения функциональности уже существующих схем. Причем существующие схемы могут также быть замещающими и принадлежать разным пакетам.

Для создания замещающей схемы не визуального или визуального модуля необходимо выполнить следующие шаги.

1. Перейти в раздел [ *Конфигурация* ] и выбрать пользовательский пакет, в который будет добавлена новая схема модуля.
2. На вкладке [ *Схемы* ] выполнить команду [ *Добавить* ] —> [ *Замещающий клиентский модуль* ].

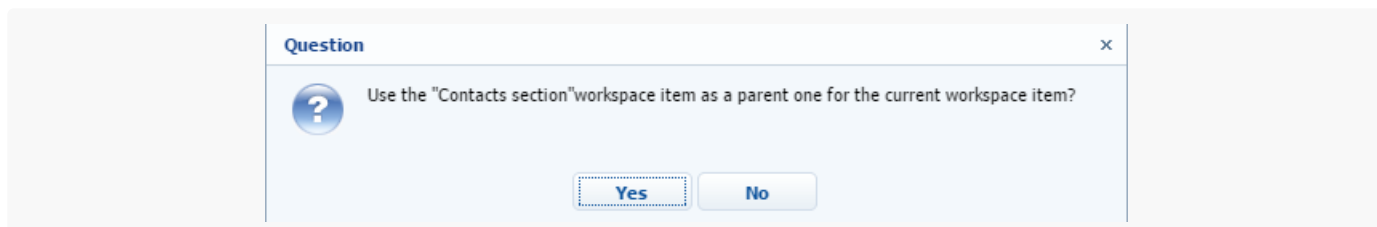


3. В открывшемся дизайнера клиентского модуля нужно выбрать корневой элемент структуры.

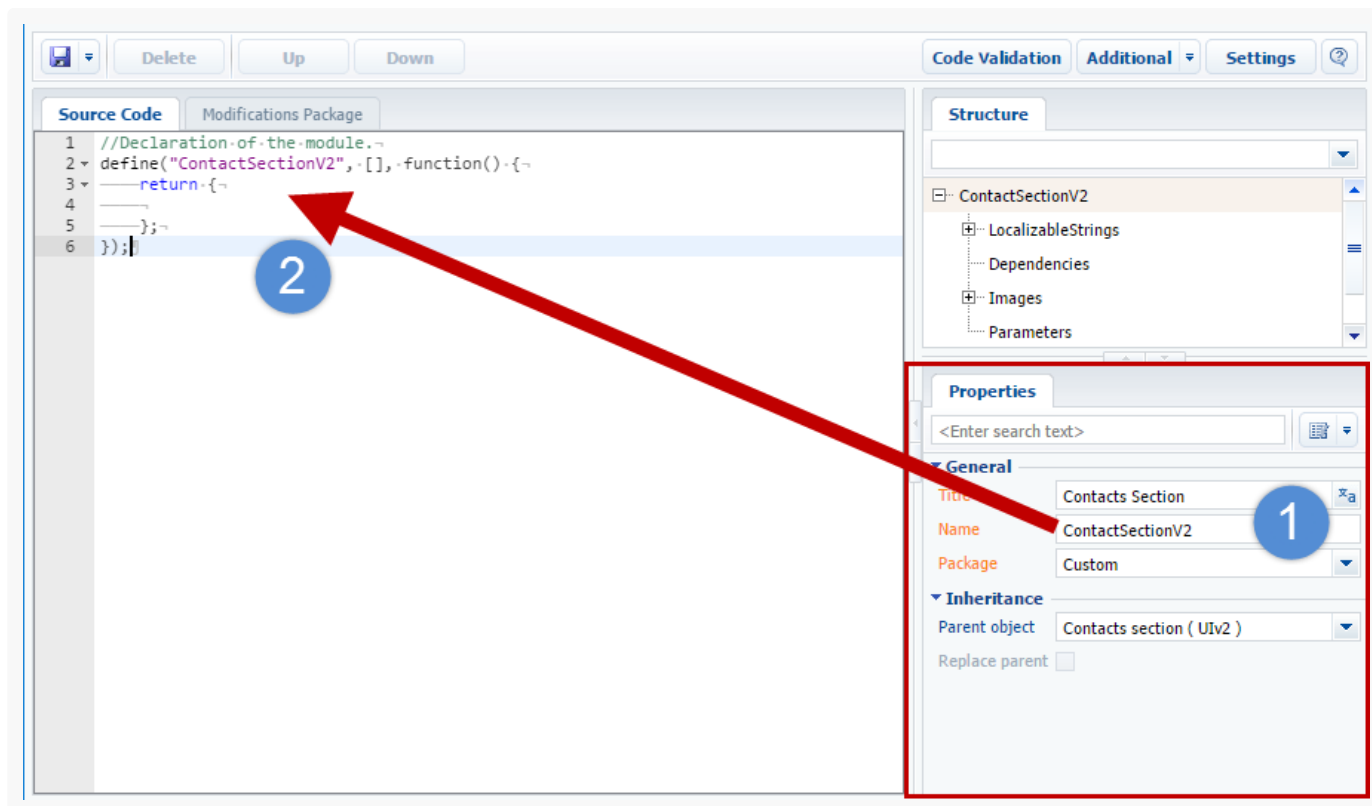


4. Для того чтобы модуль замещал определенный раздел или страницу, необходимо в поле [ Родительский объект ] свойств схемы указать заголовок той базовой схемы модели представления, которую требуется заместить. Например, для создания замещающей схемы раздела [ Контакты ] необходимо указать в качестве родительского объекта схему `ContactSectionV2`. Для этого в поле [ Родительский объект ] свойств замещающей схемы необходимо начать вводить заголовок схемы "Раздел контакты" и выбрать нужное значение из выпадающего списка. После подтверждения выбранного родительского объекта остальные поля свойств заполнятся автоматически.

Диалог подтверждения использования родительской схемы



Замещающая клиентская схема в дизайнера клиентских схем



5. На вкладке [ *Исходный код* ] дизайнера модуля нужно добавить исходный код схемы модели представления. При этом необходимо удостовериться, что название замещающего модуля в функции `define()` совпадает с названием схемы модели представления.
6. После внесения всех изменений схему модуля нужно сохранить.

## Создать схему объекта

 Легкий

Объектный слой ORM ([Object-relational mapping](#)) в Creatio основан на объектах (`Entity`). Объект — это бизнес-сущность, которая позволяет объявить на уровне серверного ядра новый класс ORM-модели. На уровне базы создание объекта означает создание записи таблицы с таким же именем, как у созданного объекта, и с таким же составом колонок. То есть в большинстве случаев каждый объект в системе является системным представлением одной физической таблицы в базе данных.

Основой конфигурации Creatio является схема. Каждый тип элемента конфигурации представляется схемой соответствующего типа. С точки зрения программной реализации схема любого типа — это класс ядра, который наследуется от базового класса `Schema`.

Объект, как элемент конфигурации, представлен схемой объекта, которая реализована соответствующим классом `EntitySchema`. Именно в схеме объекта описывается состав колонок, индексов и методов объекта.

Различают базовые и пользовательские схемы объектов.

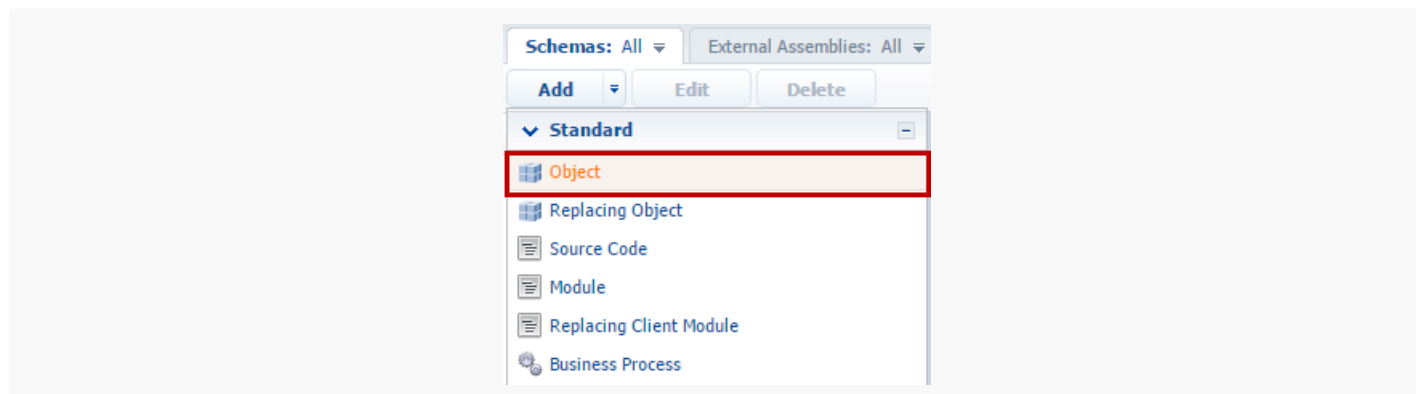
Базовые схемы объектов недоступны для редактирования и располагаются в предустановленных пакетах.

Пользовательские схемы объектов — это схемы, которые можно создавать при конфигурировании и

помещать в свои пользовательские пакеты. Базовые схемы могут замещаться пользовательскими.

## Создать пользовательскую схему объекта

Для создания пользовательской схемы объекта необходимо в разделе [ *Конфигурация* ] выбрать пользовательский пакет и на вкладке [ *Схемы* ] выполнить команду меню [ *Добавить* ] —> [ *Объект* ].



В результате откроется окно дизайнера объектов, в котором выполняется настройка созданного объекта.

Для созданной схемы объекта необходимо присвоить значения следующим обязательным свойствам:

- [ *Заголовок* ] — локализуемый заголовок схемы. Значение по умолчанию устанавливается дизайнером объектов и может быть изменено.
- [ *Название* ] — название схемы. Значение по умолчанию устанавливается дизайнером объектов и может быть изменено. Должно содержать префикс, установленный в системной настройке [ *Префикс названия объекта* ] ( `SchemaNamePrefix` ). По умолчанию это префикс `Usr`.

**Важно.** Максимально допустимая длина имени объекта — 128 символов.

Объекты с длиной имени более 30 символов нельзя использовать на базах Oracle ниже версии 12.2.

- [ *Пакет* ] — пакет, в котором схема объекта будет размещена после публикации. По умолчанию содержит название пакета, выбранного перед созданием схемы. Значение может быть изменено на любое из выпадающего списка.
- [ *Идентификатор* ] — системная колонка, используемая в качестве первичного ключа в таблице базы данных. Показывается в расширенном режиме отображения свойств объекта.

The screenshot shows the 'EntitySchemaDesigner' interface. On the left, the 'Structure' pane shows a tree view with 'UsrEntity1' expanded, containing 'Columns' and 'Indexes'. The main area is the 'Properties' tab, which has a search bar and several sections: 'General', 'Inheritance', 'Behavior', 'Access rights', 'System Columns', and 'History Columns'. The 'General' section includes fields for 'Name' (UsrEntity1), 'Title' (Object 1), 'Change Log Object Name', 'Permission Object Name', 'Localization Object Name', 'Description', and 'Package' (sdkCreateEntitySchema). The 'System Columns' section includes a dropdown for 'Id' (highlighted in orange), 'Displayed value', 'Image', 'Sorting in Lists', 'Parent in Hierarchy', and 'Owner'. The 'History Columns' section is currently collapsed.

**На заметку.** Для отображения всех свойств схемы в дизайнере объектов необходимо выбрать опцию [ Все ] в меню отображения свойств объекта.

Поскольку объект в системе является представлением записи таблицы в базе данных, то он обязательно должен содержать колонку-идентификатор, используемую в качестве первичного ключа таблицы. Если попытаться сохранить схему объекта без идентификатора, то система выдаст предупреждение.

The screenshot shows a warning message box with a yellow background and an orange warning icon. The text reads: "Warning Required property 'Id' is not specified for item 'UsrEntity1'". Above the warning box, the top toolbar of the 'EntitySchemaDesigner' is visible, showing buttons for 'Add', 'Delete', 'Up', and 'Down'.

Установить значение для свойства [ Идентификатор ] можно выбрав колонку определенного типа из выпадающего списка, либо указав один из базовых объектов системы в качестве родительского объекта.



The screenshot shows a software interface for creating an object schema. It is divided into two main sections: 'Structure' on the left and 'Properties' on the right.

**Structure:** A tree view showing the hierarchy of the object 'UsrEntityWithId'. It includes sub-items: 'Columns', 'IsrId', and 'Indexes'.

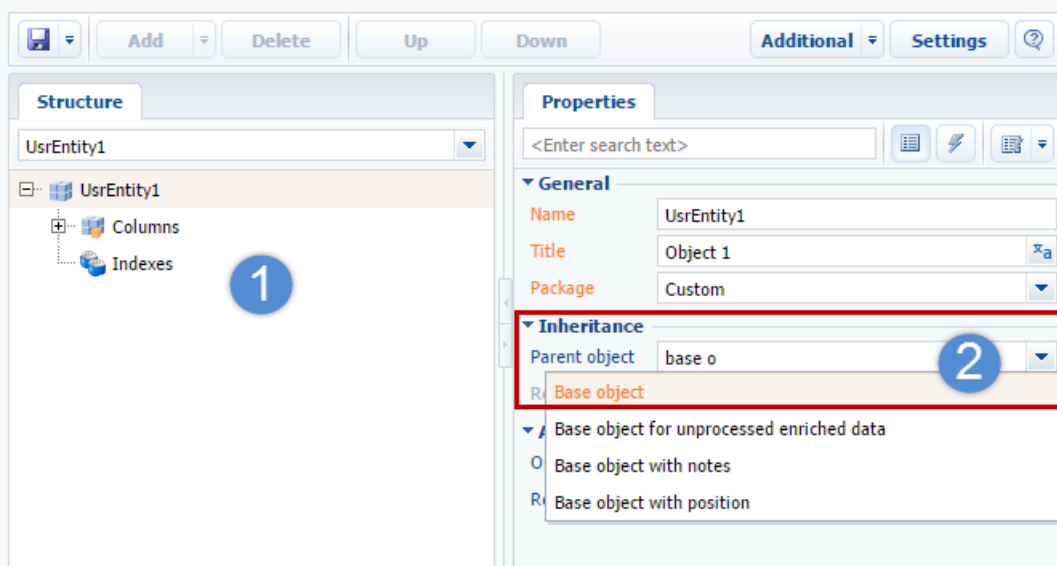
**Properties:** A form for configuring the object's properties. It includes a search bar and several sections:

- General:**
  - Name: UsrEntityWithId
  - Title: Object With Id
  - Change Log Object Name: (empty)
  - Permission Object Name: (empty)
  - Localization Object Name: (empty)
  - Description: (empty)
  - Package: Custom
- Inheritance:** (empty)
- Behavior:** (empty)
- Access rights:** (empty)
- System Columns:**
  - Id: (dropdown menu)
  - Displayed value: ID
  - Image: (dropdown menu)

## Указать родительский объект

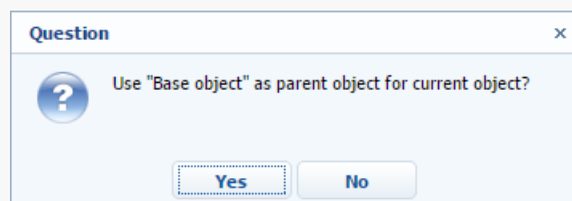
Для объектов Creatio реализован механизм наследования. Он может быть полезным в случае, если создаваемая схема объекта должна содержать функциональность, уже реализованную в существующих объектах. В большинстве случаев в качестве родительских используются базовые объекты системы, например, [ *Базовый объект* ] или [ *Базовый справочник* ].

Чтобы реализовать наследование создаваемой схемы объекта от существующей, необходимо выбрать корневой элемент структуры схемы объекта. Затем в поле [ *Родительский объект* ] свойств схемы указать заголовок той базовой схемы объекта, функциональность которой требуется унаследовать. Например, чтобы наследовать функциональность схемы [ *Базовый объект* ] необходимо в поле [ *Родительский объект* ] свойств схемы начать вводить заголовок схемы [ *Базовый объект* ] и выбрать нужное значение из выпадающего списка.

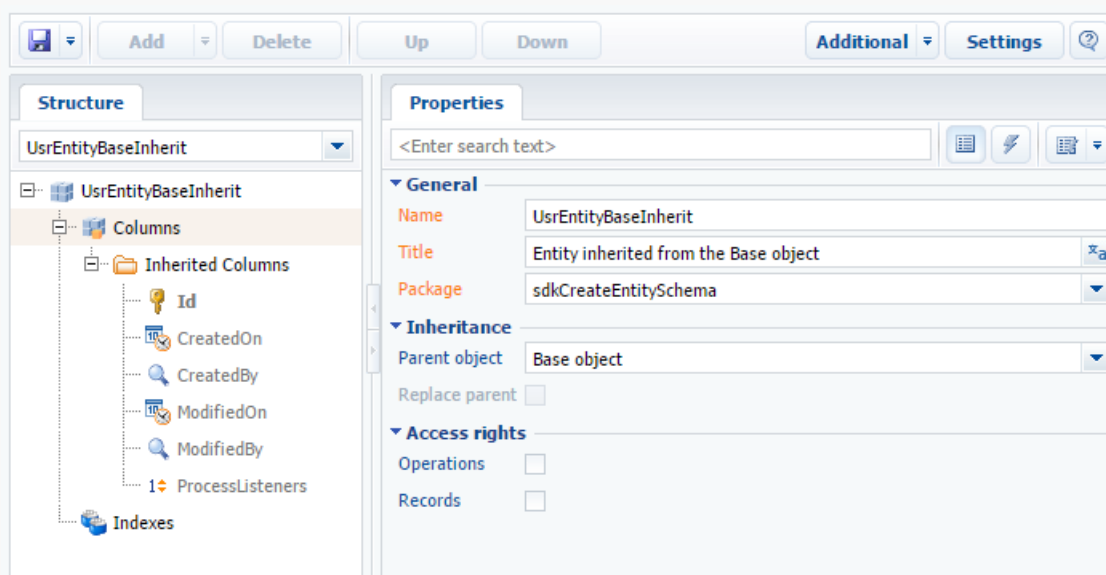


После подтверждения выбранного родительского объекта к структуре объекта будут добавлены колонки, унаследованные от базового объекта.

Диалог подтверждения использования родительского объекта



Унаследованные колонки в структуре объекта



**На заметку.** Для отображения унаследованных колонок необходимо включить опцию

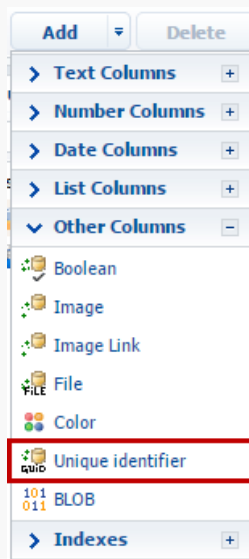
[ Показывать системные колонки ] в окне настроек дизайнера объектов.

Для временного сохранения всех изменений в метаданных схему объекта необходимо сохранить. Для окончательного сохранения схемы и создания соответствующей таблицы в базе данных схему необходимо опубликовать.

## Добавить пользовательскую колонку в объект

Добавление пользовательской колонки в схему объекта будет рассмотрено на примере добавления колонки-идентификатора.

Для добавления пользовательской колонки можно использовать кнопку [ Добавить ] либо команду [ Добавить ] контекстного меню узла [ Колонки ] ( Columns ) в структуре объекта. В выпадающем меню выбирается тип колонки и далее задаются свойства колонки. Для добавления колонки-идентификатора следует выполнить команду [ Добавить ] —> [ Уникальный идентификатор ].



**На заметку.** Для отображения в меню добавления всех типов добавляемых колонок необходимо включить опцию [ Отображать расширенный список типов колонок ] в окне настроек дизайнера объектов.

Затем следует задать основные свойства колонки-идентификатора. Основные свойства, касающиеся добавляемой колонки:

- [ Заголовок ] — локализуемый заголовок колонки. Значение по умолчанию устанавливается дизайнером объектов и может быть изменено.
- [ Название ] — название колонки. Значение по умолчанию устанавливается дизайнером объектов и может быть изменено.
- [ Тип данных ] — тип данных, содержащихся в колонке. Значение по умолчанию устанавливается дизайнером объектов в зависимости от выполненной команды добавления колонки и может быть изменено.

- [ *Обязательная для заполнения* ] — устанавливает обязательность колонки. Поскольку колонка должна обязательно содержать значение, то для этого свойства необходимо выбрать "На уровне приложения".
- [ *Значение по умолчанию* ] — устанавливает значение по умолчанию. Для добавляемой колонки необходимо в диалоговом окне выбора значения по умолчанию выбрать вариант "Установить из системной переменной". Затем в поле [ *Название* ] следует выбрать название системной переменной. Поскольку все идентификаторы должны быть уникальными, следует выбрать значение "Новый идентификатор".
- [ *Режим использования* ] — режим использования "Расширенный".

Свойства колонки-идентификатора

Установка значения по умолчанию

После присвоения значений всем необходимым свойствам добавленной колонки схему следует сохранить.

## Добавить индексы в объект

Кроме колонок в объект могут быть добавлены индексы, которые при публикации объекта будут автоматически созданы в таблице базы данных.

Если необходимо создать индекс по одной колонке, то для этой колонки достаточно установить признак [ *Индексируемая* ] в блоке свойств [ *Поведение* ]. По умолчанию в системе все справочные колонки являются индексируемыми.

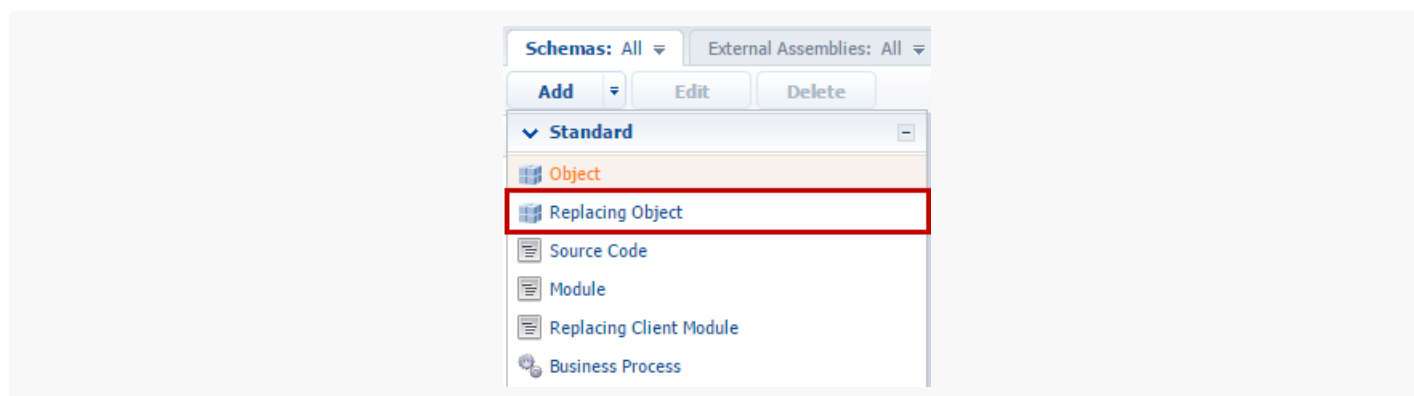
Если нужно создать составной индекс по нескольким колонкам, то сделать это можно следующим образом:

1. В контекстном меню элемента [ *Индексы* ] выбрать [ *Добавить* ] —> [ *Индекс* ]. Для индекса можно указать пользовательское название либо поставить опцию [ *Автогенерируемое название* ]. После этого система сгенерирует уникальное название индекса.
2. Если для колонок индекса необходимо реализовать ограничение целостности, то есть исключить возможность вставки повторяющихся комбинаций значений, то для индекса следует установить признак [ *Уникальный* ].
3. Далее в индекс нужно добавить необходимые колонки. В контекстном меню узла индекса следует выполнить команду [ *Добавить* ] —> [ *Колонка индекса* ]. Затем для добавленной колонки нужно установить свойства — выбрать колонку объекта и указать направление сортировки.

## Создать замещающую схему объекта

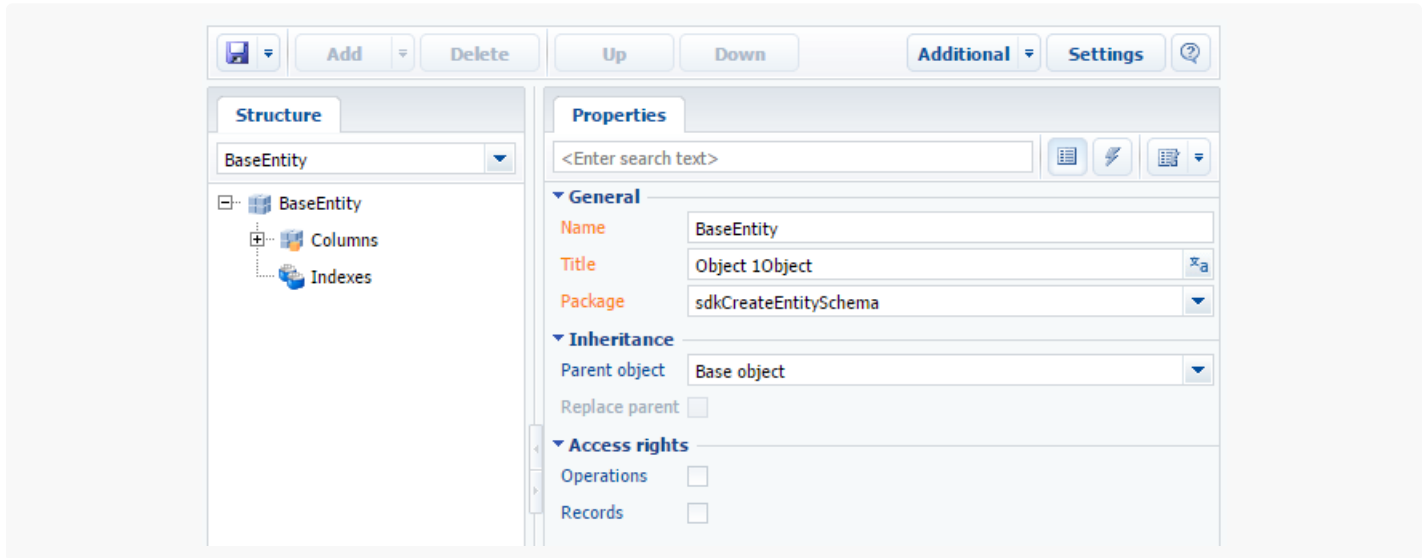
Замещающие схемы объектов предназначены для расширения функциональности уже существующих схем. При этом существующие схемы в свою очередь могут быть замещающими и принадлежать разным пакетам.

Для создания замещающей схемы объекта необходимо перейти в раздел [ *Конфигурация* ] и выбрать пользовательский пакет, в который будет добавлена новая схема модуля. Затем необходимо на вкладке [ *Схемы* ] выполнить команду [ *Добавить* ] —> [ *Замещающий объект* ].



Чтобы реализовать замещение создаваемой схемы объекта, необходимо выбрать корневой элемент структуры схемы объекта. Затем в поле [ *Родительский объект* ] свойств схемы указать заголовок той базовой схемы объекта, функциональность которой требуется заместить. Например, чтобы заместить функциональность схемы [ *Базовый объект* ] необходимо в поле [ *Родительский объект* ] свойств схемы начать вводить заголовок схемы [ *Базовый объект* ] и выбрать нужное значение из выпадающего списка. После подтверждения выбранного родительского объекта остальные поля свойств заполнятся

автоматически.



После внесения всех изменений замещающую схему объекта нужно опубликовать.

## Сохранить и опубликовать объект

Все изменения структуры бизнес-объекта, выполняемые в дизайнера, осуществляются в оперативной памяти.

Чтобы изменения были сохранены на уровне метаданных схемы, схему следует сохранить. Для этого нужно в дизайнере объектов выполнить команду [ *Сохранить* ].

Для того чтобы изменения произошли на уровне базы данных, объект необходимо опубликовать. Результатом успешной публикации объекта являются созданные (или измененные) физические объекты в базе данных — таблица, столбцы, индексы.

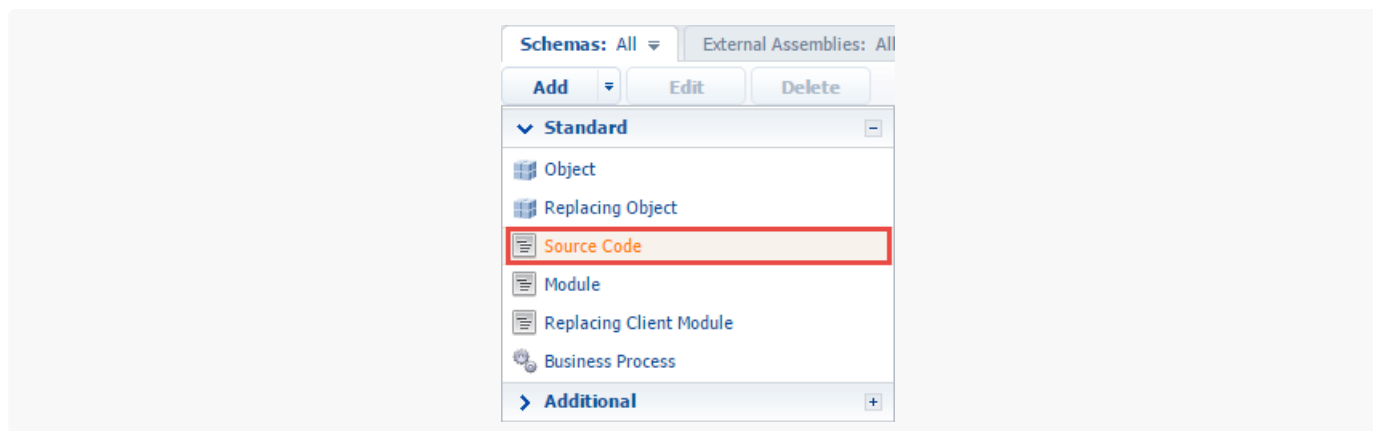
## Создать схему [Исходный код]



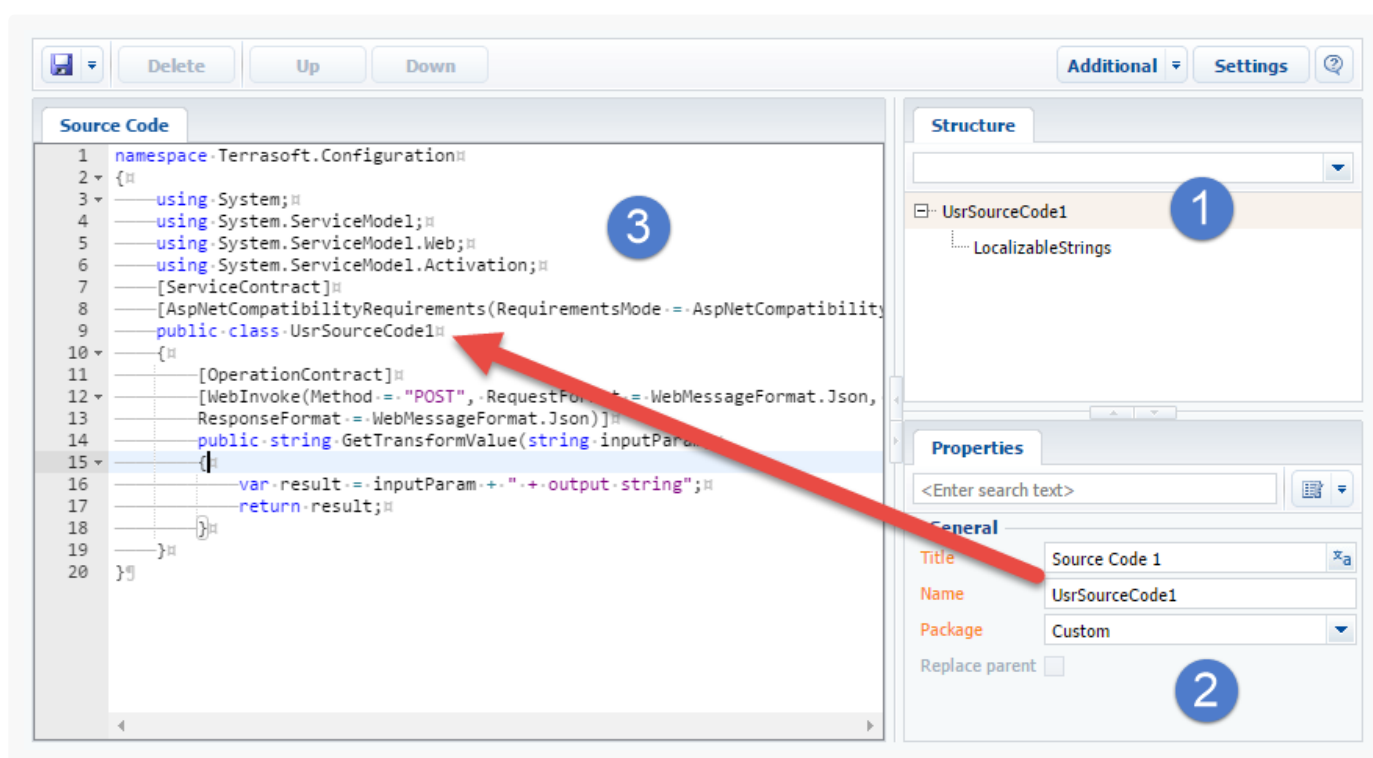
Легкий

Для создания схемы [ *Исходный код* ] необходимо выполнить следующую последовательность действий:

1. Перейти в раздел [ *Конфигурация* ] и выбрать пользовательский пакет, в который будет добавлена новая схема.
2. На вкладке [ *Схемы* ] ([ *Schemas* ]) выполнить команду [ *Добавить* ] —> [ *Исходный код* ] ([ *Add* ] —> [ *Source Code* ]).

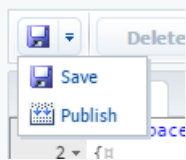


- В появившемся дизайнере схемы выбрать корневой элемент структуры и заполнить свойства создаваемой схемы.



К основным свойствам схемы [ *Исходный код* ] относятся:

- [ *Название* ] ([ *Name* ]) — название схемы. Может содержать только символы латинского алфавита и цифры. Содержит префикс, указанный в системной настройке [ *Префикс названия объекта* ] ( `SchemaNamePrefix` ).
  - [ *Заголовок* ] ([ *Title* ]) — заголовок схемы. Может быть локализован.
  - [ *Пакет* ] ([ *Package* ]) — пользовательский пакет, в котором создается схема.
- На вкладке [ *Исходный код* ] ([ *Source Code* ]) дизайнера схемы добавить исходный код. При этом необходимо удостовериться, что в исходном коде объявлен класс с названием, совпадающим с названием схемы.
  - После внесения всех изменений схему нужно опубликовать:



**Важно.** Все изменения схемы, выполняемые в дизайнера, осуществляются в оперативной памяти. Чтобы изменения были сохранены на уровне метаданных схемы, схему следует сохранить. Для этого нужно в дизайнера объектов выполнить команду [ *Сохранить* ]. Для того чтобы изменения произошли на уровне базы данных, схему необходимо опубликовать.