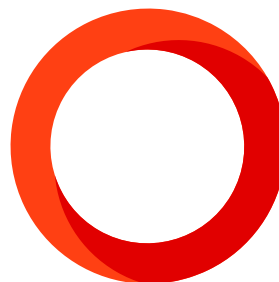
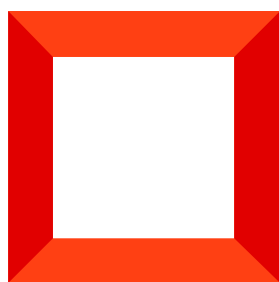
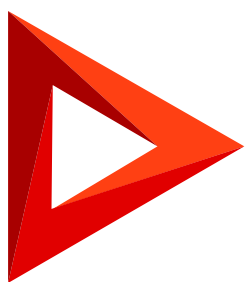


# Creatio IDE

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>Принципы разработки в Creatio IDE</b>	<b>4</b>
Настроить доступ к разделу Конфигурация	4
Открыть раздел Конфигурация	4
Разработка конфигурации	5
Закрыть раздел Конфигурация	18
<b>Разработка конфигурационных элементов</b>	<b>18</b>
Клиентский модуль	20
Объект	26
Исходный код	35

# Принципы разработки в Creatio IDE



**Creatio IDE** — встроенная среда разработки для управления конфигурацией. Управление конфигурацией подразумевает реализацию сложной бизнес-логики, интеграции и настройки. Управление конфигурацией выполняется с помощью Creatio IDE, которая реализована в виде раздела [ Конфигурация ] ([ Configuration ]). **Назначение** раздела [ Конфигурация ] — управление конфигурационными элементами, при помощи которых реализована функциональность системы.

## Настроить доступ к разделу [ Конфигурация ]

Доступ к разделу [ Конфигурация ] настраивается на уровне системных операций. По умолчанию доступ к основным системным операциям есть только у администраторов системы. Но его можно [настроить](#) для пользователей или групп пользователей. **Настройка доступа** к разделу [ Конфигурация ]:

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" ] (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" ] (код [ CanManageSolution ]).

Если у пользователя нет доступа к разделу [ Конфигурация ], то выдается стандартное сообщение с указанием операции и недостающих прав.

## Открыть раздел [ Конфигурация ]

Способы перехода в раздел [ Конфигурация ] для фреймворка **.NET Framework**:

- По кнопке через дизайнер системы. 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` .

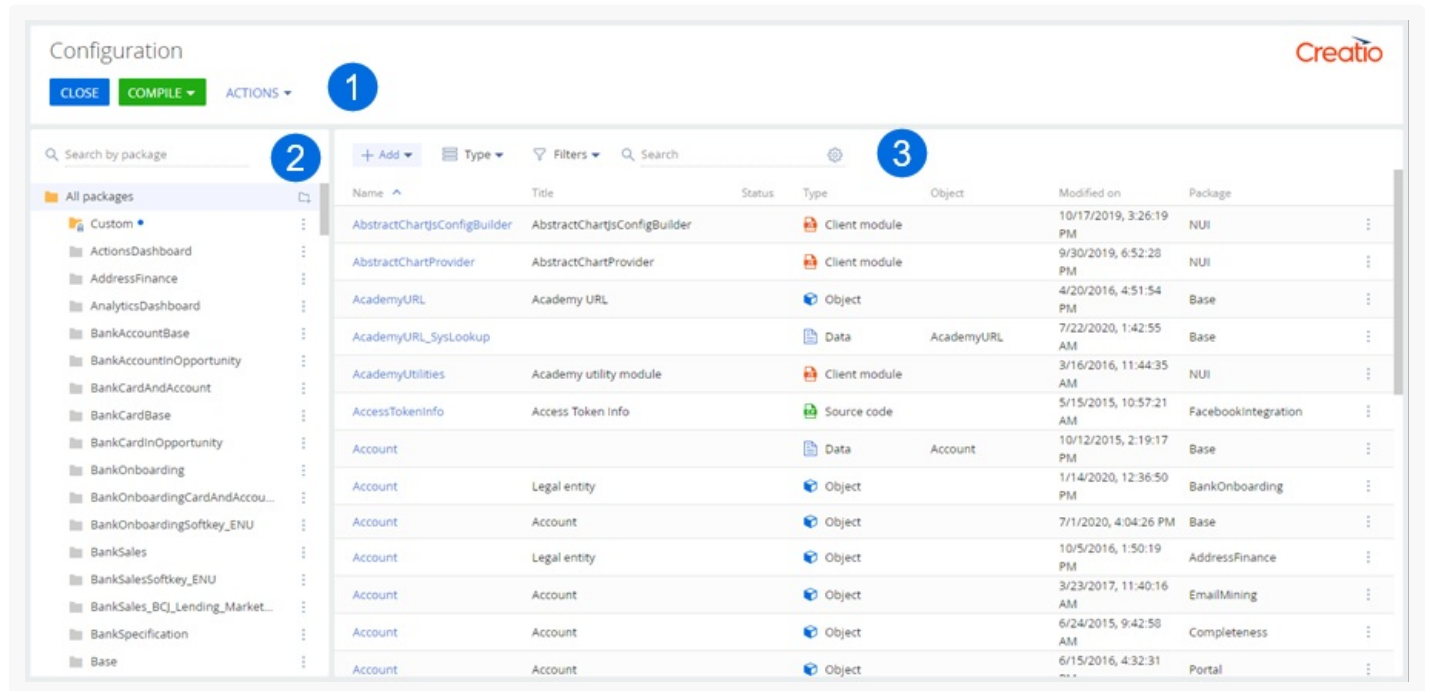
Для фреймворка **ASP.NET Core** способы перехода в раздел аналогичны. Различие — не нужно использовать приставку `/0` .

Раздел [ Конфигурация ] будет открыт в новой вкладке.

# Разработка конфигурации

**Функциональные области** интерфейса раздела [ Конфигурация ]:

- Панель инструментов (1).
- Область работы с пакетами (2).
- Рабочая область (3).

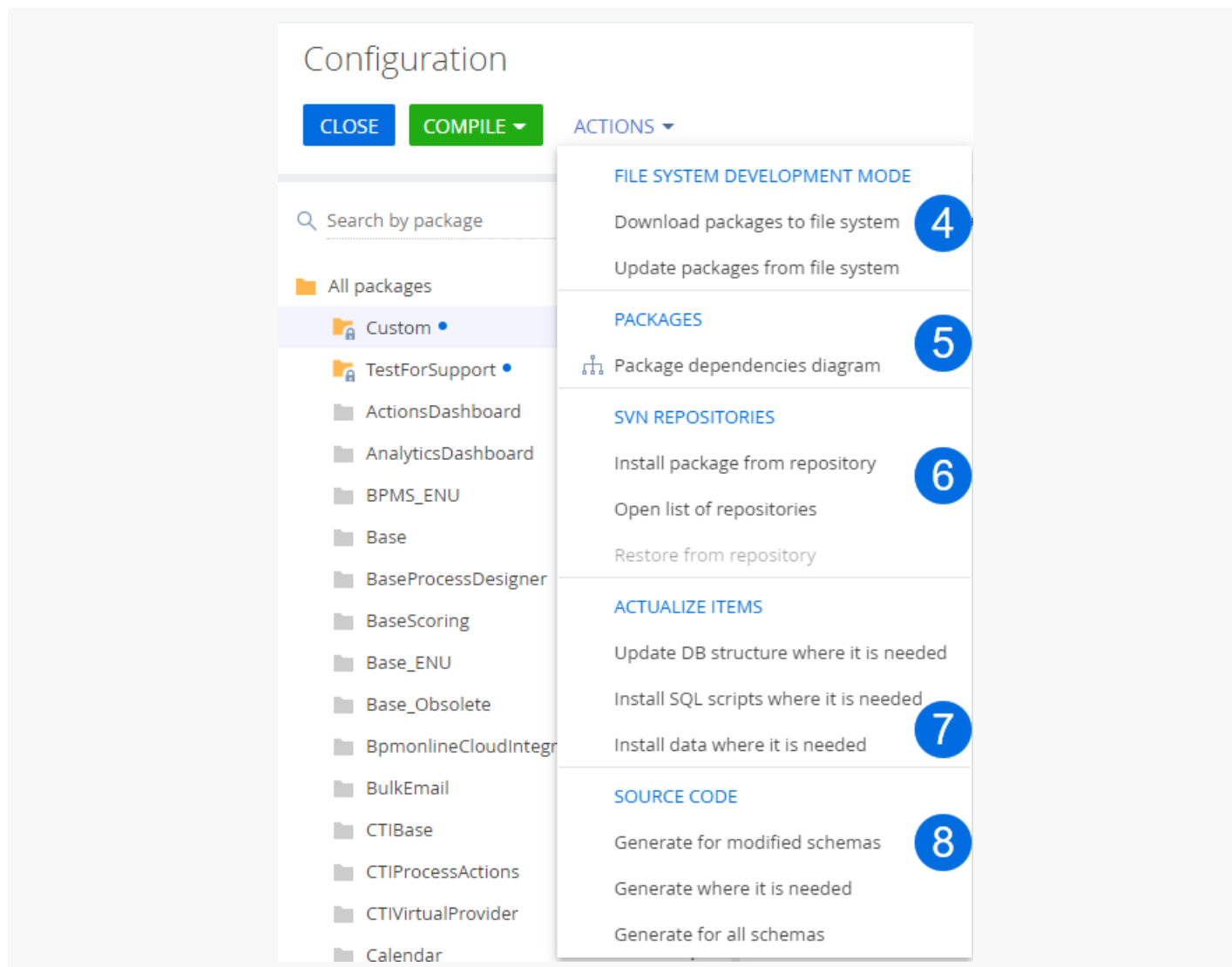


**Инструменты** раздела [ Конфигурация ] позволяют управлять:

- Разработкой в файловой системе.
- Пакетами.
- Конфигурационными элементами пакетов.
- Хранилищами системы контроля версий.
- Компиляцией конфигурации.

## Разработка в файловой системе

Действия, связанные с разработкой в файловой системе, реализованы в группе [ Разработка в файловой системе ] ([ File system development mode ]) (4) выпадающего списка [ Действия ] ([ Actions ]) панели инструментов (1).



**Группа действий** [ *Разработка в файловой системе* ] ([ *File system development mode* ]) (4) позволяет:

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

Пункты группы действий [ *Разработка в файловой системе* ] ([ *File system development mode* ]) (4) доступны только при включенном режиме разработки в [файловой системе](#). Подсказка по включению режима отображается при наведении курсора на название любого пункта текущей группы действий. Включение режима разработки в файловой системе описано в статье [Настроить Creatio для работы в файловой системе](#).

## Пакеты

**Инструменты управления пакетами:**



- Группа [ *Пакеты* ] ([ *Packages* ]) (5) выпадающего списка [ *Действия* ] ([ *Actions* ]) панели инструментов

(1).

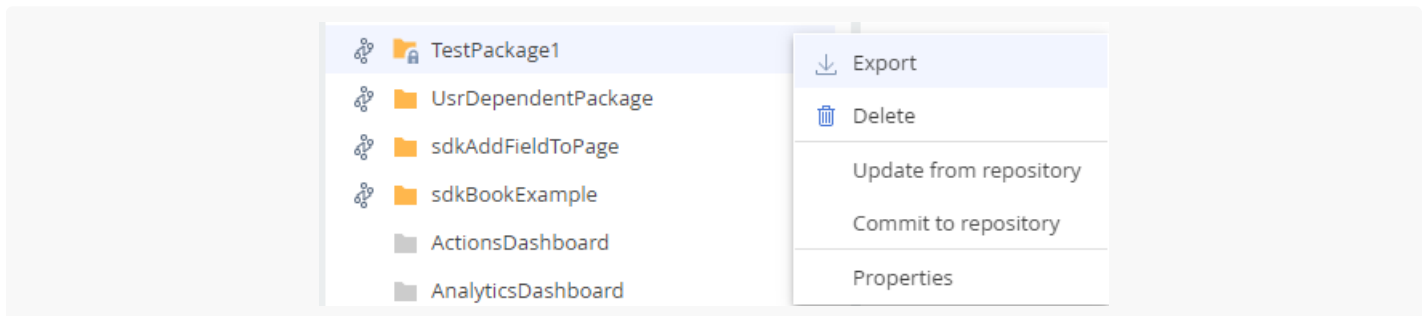
- Область работы с пакетами (2).
- Меню пакета.

**Группа действий [ Пакеты ]** ([ *Packages* ]) (5) позволяет открыть [диаграмму зависимостей](#) пакетов ([ *Диаграмма зависимостей пакетов* ] ([ *Package dependencies diagram* ])).

**Область работы с пакетами** (2) позволяет:

- Выполнить поиск пакета по имени (строка поиска [ *Поиск по пакетам* ] ([ *Search by package* ])).
- Создать пакет (кнопка ). При нажатии на кнопку будет отображено окно создания нового пакета, в котором можно задать название и описание пакета, добавить зависимости, а также указать хранилище системы контроля версий. Создание пакета описано в статье [Создать пользовательский пакет](#).
- Посмотреть перечень пакетов приложения (группа [ *Все пакеты* ] ([ *All packages* ])). Пакеты отображаются в алфавитном порядке. Вверху перечня отображаются измененные пакеты и пакеты, доступные для редактирования. При выборе текущей группы конфигурационные элементы всех пакетов приложения будут отображены в алфавитном порядке в рабочей области раздела [ *Конфигурация* ] (3). При выборе пакета группы в алфавитном порядке будут отображены конфигурационные элементы текущего пакета. Измененные пакеты находятся вверху перечня пакетов группы [ *Все пакеты* ] ([ *All packages* ]) и содержат символ  возле имени пакета.

**Меню пакета** можно вызвать по нажатию на  в строке с названием пакета.



Меню пакета позволяет:

- Удалить пакет (пункт [ *Удалить* ] ([ *Delete* ])). Пункт неактивен для предустановленных пакетов. Можно удалить пустые пакеты и пакеты с конфигурационными элементами, которые не являются родительскими. При попытке удалить пакет с родительскими конфигурационными элементами будет отображен перечень зависимых пакетов и элементов, зависящих от конфигурационных элементов удаляемого пакета, которые препятствуют удалению.
- Посмотреть свойства пакета (пункт [ *Свойства* ] ([ *Properties* ])). Вкладка [ *Свойства пакета* ] ([ *Package properties* ]) позволяет настроить зависимости текущего пакета (если пакет доступен для редактирования). Также можно просмотреть системную информацию: кто создал и отредактировал пакет, даты создания и изменения, уникальный идентификатор, первичный ключ пакета в таблице базы данных. Открыть свойства пакета можно двойным кликом по имени пакета.

## Конфигурационные элементы пакетов

**Инструменты управления** конфигурационными элементами пакетов:

- Группа [ *Актуализировать элементы* ] ([ *Actualize items* ]) (7) выпадающего списка [ *Действия* ] ([ *Actions* ]) панели инструментов (1).
- Группа [ *Исходный код* ] ([ *Source code* ]) (8) выпадающего списка [ *Действия* ] ([ *Actions* ]) панели инструментов (1).
- Панель инструментов рабочей области (3) раздела [ *Конфигурация* ].
- Реестр рабочей области (3) раздела [ *Конфигурация* ].

**Группа действий** [ *Актуализировать элементы* ] ([ *Actualize items* ]) (7) позволяет:

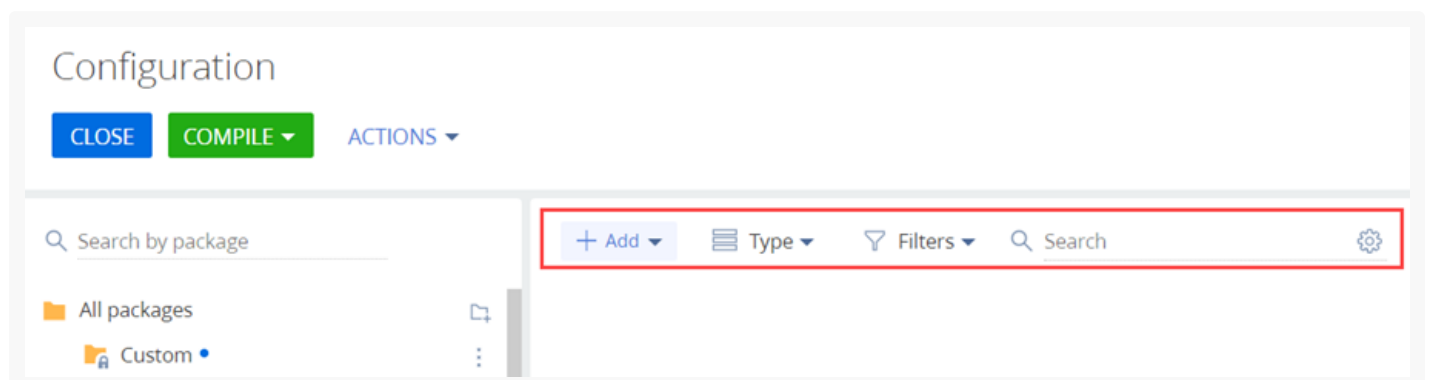
- Обновить структуру базы данных для конфигурационных элементов, которые требуют обновления (пункт [ *Обновить структуру БД для требующих обновления* ] ([ *Update DB structure where it is needed* ])).
- Установить SQL-сценарии, которые требуют установки (пункт [ *Установить SQL сценарии для требующих установки* ] ([ *Install SQL scripts where it is needed* ])).
- Установить данные, которые требуют установки (пункт [ *Установить данные для требующих установки* ] ([ *Install data where it is needed* ])).

После завершения актуализации конфигурационных элементов вы получите уведомление.

**Группа действий** [ *Исходный код* ] ([ *Source code* ]) (8) позволяет:

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

**Панель инструментов** рабочей области (3) раздела [ *Конфигурация* ] представлена на рисунке ниже.

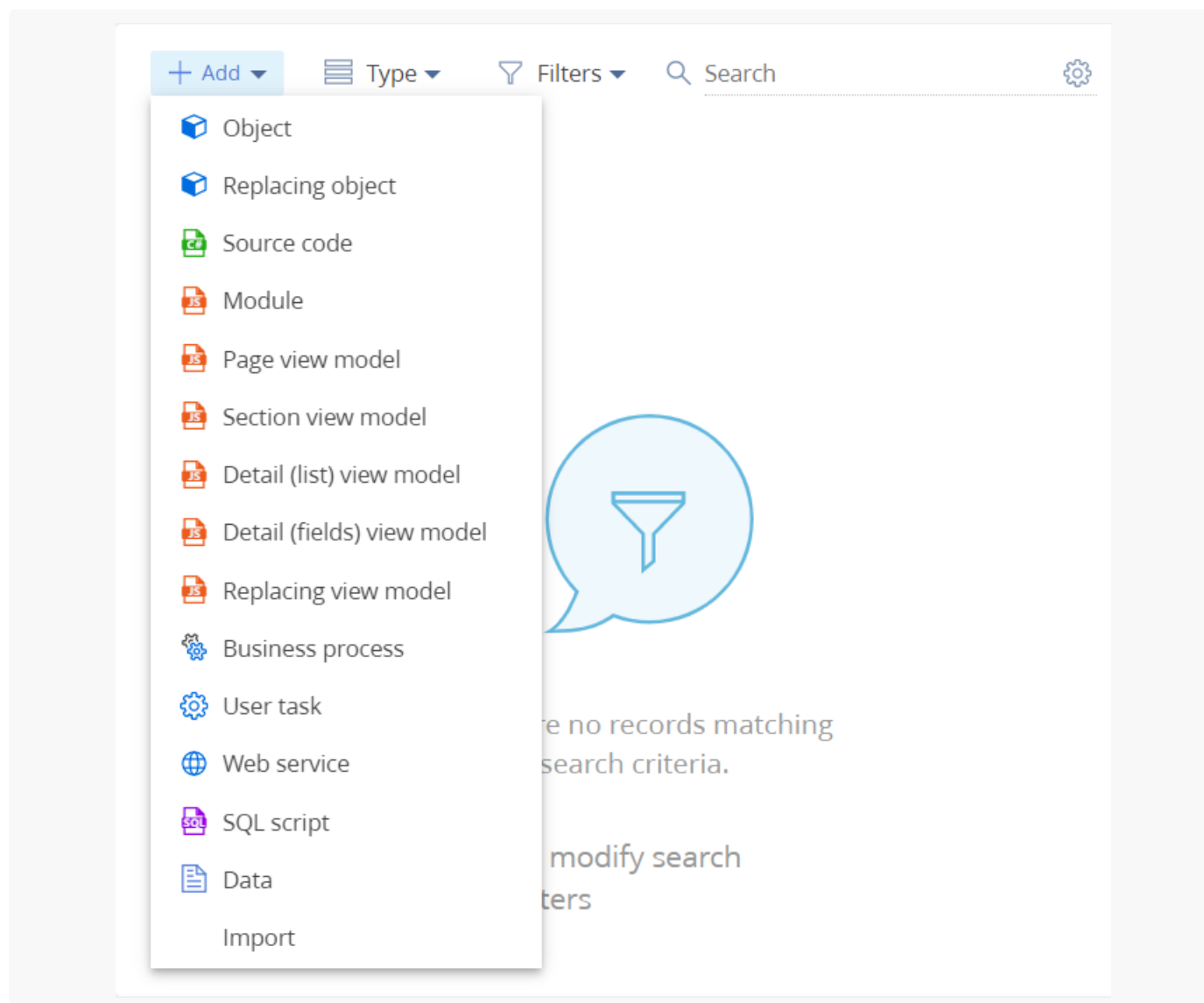


Панель инструментов позволяет:

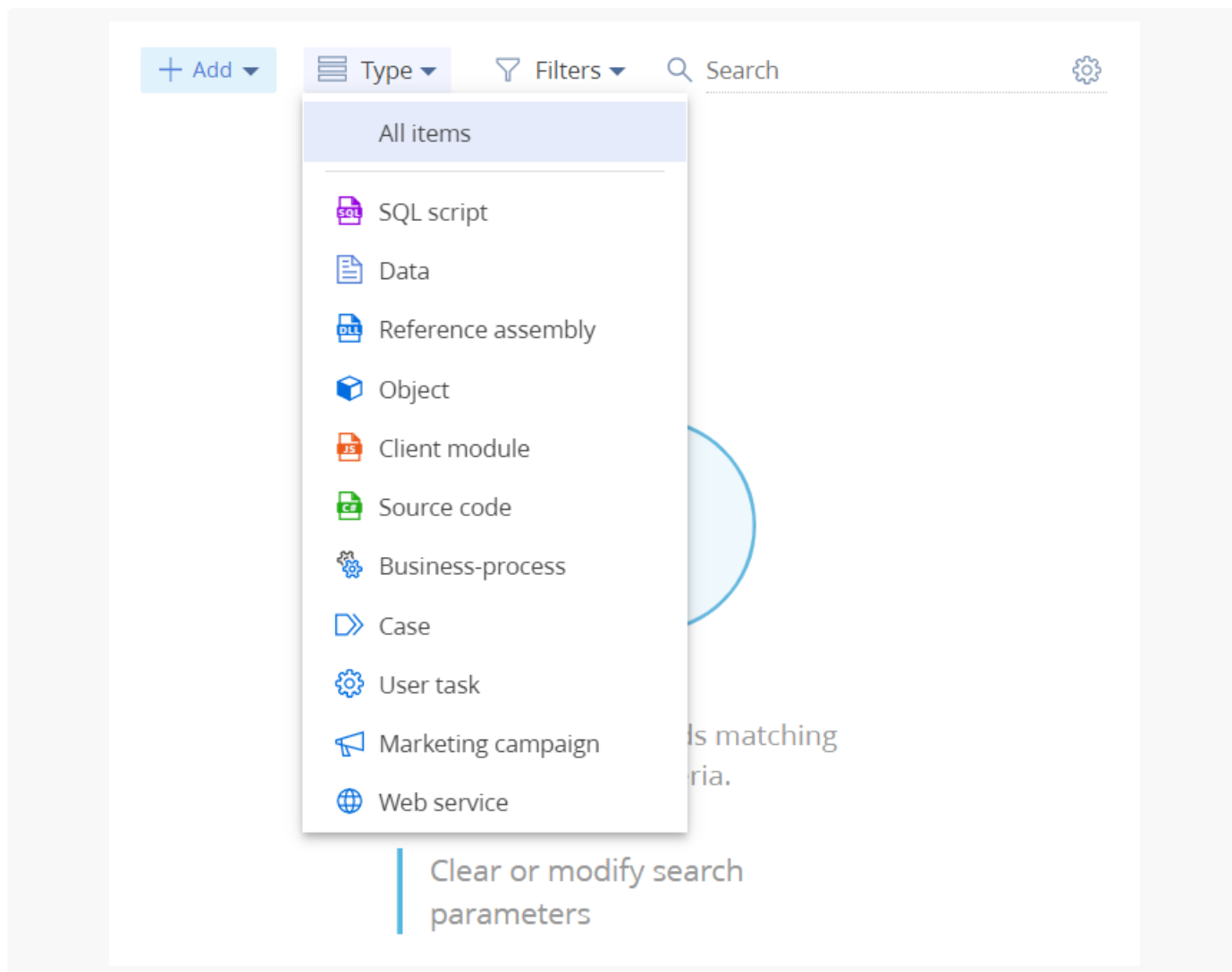
- Выбрать вид конфигурационного элемента для добавления (выпадающий список [ *Добавить* ] ([ *Add* ])). Предварительно необходимо выбрать пакет. Без выбора пакета пункты выпадающего списка [ *Добавить* ] ([ *Add* ]) остаются неактивными. В предустановленные пакеты конфигурационные



элементы добавить невозможно. Возможные **виды** конфигурационных элементов для добавления представлены на рисунке ниже.



- Загрузить в пользовательский пакет схему (\*.md) или внешнюю сборку (\*.dll) (пункт [ *Импортировать* ] ([ *Import* ]) выпадающего списка [ *Добавить* ] ([ *Add* ])).
- Выбрать тип конфигурационных элементов для отображения в реестре раздела (выпадающий список [ *Тип* ] ([ *Type* ])). Возможные **типы** конфигурационных элементов для отображения представлены на рисунке ниже.

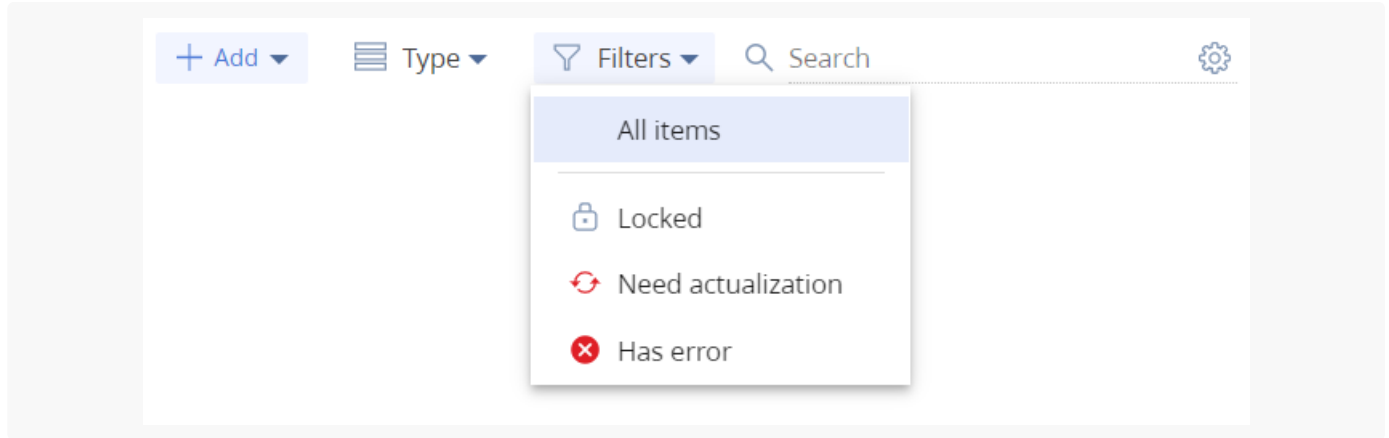


Конфигурационные элементы с типом [ *Клиентский модуль* ] ([ *Client module* ]):


- [ *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* ]).

Конфигурационные элементы с типом [ *Объект* ] ([ *Object* ]):

- [ *Объект* ] ([ *Object* ]).
- [ *Замещающий объект* ] ([ *Replacing object* ]).
- Выбрать состояния конфигурационных элементов для отображения в реестре раздела (выпадающий список [ *Фильтры* ] ([ *Filters* ])). Настройки выпадающего списка [ *Фильтры* ] ([ *Filters* ]) сохраняются в профиле пользователя и отображаются при входе в раздел [ *Конфигурация* ]. Возможные **состояния** конфигурационных элементов для отображения представлены на рисунке ниже.



Конфигурационные элементы со значением [ *Требующие актуализации* ] ([ *Need actualization* ]) выпадающего списка [ *Фильтры* ] ([ *Filters* ]):







- [ *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* ]).
- Выполнить поиск конфигурационного элемента по имени в реестре раздела (строка поиска [ *Поиск* ] ([ *Search* ])). Поиск выполняется в текущем пакете. Чтобы выполнять поиск по всем конфигурационным элементам пакетов, необходимо перейти в группу [ *Все пакеты* ] ([ *All packages* ]) области работы с пакетами (2). По нажатию  настраиваются **параметры поиска**:
  - Поиск по заголовку (опция [ *Поиск по полю "Заголовок"* ] ([ *Search by column "Title"* ]) — по умолчанию включена).
  - Поиск по уникальному идентификатору (опция [ *Поиск по полю "UID"* ] ([ *Search by column "UID"* ]) — по умолчанию выключена).
  - Режим поиска. Возможные значения:
    - Пункт [ *Начинается* ] ([ *Starts with* ]) — по умолчанию включен. Название конфигурационного элемента начинается с текста, который введен в строке поиска [ *Поиск* ] ([ *Search* ]).
    - Пункт [ *Содержит* ] ([ *Contains* ]) — по умолчанию выключен. Название конфигурационного элемента содержит текст, который введен в строке поиска [ *Поиск* ] ([ *Search* ]).
    - Пункт [ *Равно* ] ([ *Equals* ]) — по умолчанию выключен. Название конфигурационного элемента 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 входе в раздел [ *Конфигурация* ].


**Реестр раздела** рабочей области (3) раздела [ *Конфигурация* ] содержит перечень конфигурационных элементов. **Свойства** конфигурационных элементов реестра раздела представлены в таблице.

## Свойства конфигурационных элементов

Колонка	Описание	Дополнительные сведения
<b>[ Название ]</b> ( [ Name ] )	Имя конфигурационного элемента, которое было задано при создании	Все конфигурационные элементы отсортированы в алфавитном порядке. Колонка позволяет посмотреть перечень измененных конфигурационных элементов (символ  возле имени элемента). Измененные конфигурационные элементы находятся вверху перечня конфигурационных элементов реестра раздела.
<b>[ Заголовок ]</b> ( [ Title ] )	Заголовок конфигурационного элемента	
<b>[ Статус ]</b> ( [ Status ] )	Состояние конфигурационного элемента	<p>Содержит значения выпадающего списка [ Фильтры ] ( [ Filters ] ).</p> <p>Колонка [ Статус ] ( [ Status ] ) будет содержать , если необходима актуализация конфигурационного элемента. При наведении на  можно увидеть <b>всплывающие подсказки:</b></p> <ul style="list-style-type: none"> <li>• [ Требуется установки в БД ] ( [ Needs to be installed in database ] ) — для конфигурационных элементов [ SQL сценарий ] ( [ SQL script ] ) и [ Данные ] ( [ Data ] ).</li> <li>• [ Требуется обновление в БД ] ( [ Needs to be updated in database ] ) — для конфигурационного элемента [ Объект ] ( [ Object ] ).</li> <li>• [ 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 ] ).</li> </ul> <p>Колонка [ Статус ] ( [ Status ] ) будет содержать , если конфигурационный элемент содержит ошибку. При наведении на  можно увидеть текстовое описание ошибки.</p> <p>Если необходима актуализация конфигурационного элемента и при этом он содержит ошибку, то колонка [ Статус ] ( [ Status ] ) будет содержать . При наведении курсора можно увидеть информацию о необходимости актуализации конфигурационного элемента и текстовое описание ошибки.</p>

Колонка [ Тип ] ([ Type ])	Описание Тип	Дополнительные сведения
	конфигурационного элемента	
[ Объект ] ([ Object ])	Объект, с которым связаны привязываемые к пакету данные	Колонка заполняется только для конфигурационного элемента [ Данные ] ([ Data ]).
[ Дата изменения ] ([ Modified on ])	Дата изменения конфигурационного элемента	
[ Пакет ] ([ Package ])	Название пакета, который содержит 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 ]). Описание действий с конфигурационными элементами, которые можно выполнить с помощью меню конфигурационного элемента, приведено в таблице.


Действия над конфигурационными элементами

Пункт меню	Действие	Дополнительные сведения	Конфигурационный элемент
[ Удалить ] ([ Delete ])	Удалить конфигурационный элемент	Неактивно для конфигурационных элементов предустановленных пакетов.	[ SQL сценарий ] ([ SQL script ]) [ Данные ] ([ Data ]) [ Внешняя сборка ] ([ Reference assembly ]) [ Объект ] ([ Object ]) [ Клиентский модуль ] ([ Client module ]) [ Исходный код ] ([ Source code ]) [ Кейс ] ([ Case ]) [ Маркетинговая кампания ] ([ Marketing ])

Пункт меню	Действие	Дополнительные сведения	Конфигурационный элемент
			campaign 1) [ Web service ] ([ Web service ])
<b>[ Установить ]</b> ([ <i>Install</i> ])	Установить конфигурационный элемент	<p>Для элемента [ <i>SQL сценарий</i> ] ([ <i>SQL script</i> ]) выполняется установка в базу данных, для элемента [ <i>Данные</i> ] ([ <i>Data</i> ]) выполняется установка данных для объекта колонки [ <i>Объект</i> ] ([ <i>Object</i> ]).</p> <p>При некорректной установке данных для объекта колонки [ <i>Объект</i> ] ([ <i>Object</i> ]) можно посмотреть описание ошибки в колонке [ <i>Статус</i> ] ([ <i>Status</i> ]).</p> <p>Описание ошибки также можно посмотреть в свойствах конфигурационных элементов.</p>	<p>[ <i>SQL сценарий</i> ] ([ <i>SQL script</i> ])</p> <p>[ <i>Данные</i> ] ([ <i>Data</i> ])</p>
<b>[ Отменить изменения ]</b> ([ <i>Discard changes</i> ])	Отменить внесенные изменения	Пункт доступен, если к пакету с конфигурационным элементом подключено хранилище системы контроля версий.	<p>[ <i>SQL сценарий</i> ] ([ <i>SQL script</i> ])</p> <p>[ <i>Данные</i> ] ([ <i>Data</i> ])</p> <p>[ <i>Внешняя сборка</i> ] ([ <i>Reference assembly</i> ])</p> <p>[ <i>Объект</i> ] ([ <i>Object</i> ])</p> <p>[ <i>Клиентский модуль</i> ] ([ <i>Client module</i> ])</p> <p>[ <i>Исходный код</i> ] ([ <i>Source code</i> ])</p> <p>[ <i>Кейс</i> ] ([ <i>Case</i> ])</p> <p>[ <i>Маркетинговая кампания</i> ] ([ <i>Marketing campaign</i> ])</p>

Пункт меню	Действие	Дополнительные сведения	Конфигурационный элемент
			[ <i>Действие процесса</i> ] ([ <i>User task</i> ]) [ <i>Веб-сервис</i> ] ([ <i>Web service</i> ])
[ <b>Свойства</b> ] ([ <b>Properties</b> ])	Открыть окно свойств конфигурационного элемента		[ <i>SQL сценарий</i> ] ([ <i>SQL script</i> ]) [ <i>Данные</i> ] ([ <i>Data</i> ]) [ <i>Внешняя сборка</i> ] ([ <i>Reference assembly</i> ]) [ <i>Объект</i> ] ([ <i>Object</i> ]) [ <i>Клиентский модуль</i> ] ([ <i>Client module</i> ]) [ <i>Исходный код</i> ] ([ <i>Source code</i> ]) [ <i>Кейс</i> ] ([ <i>Case</i> ]) [ <i>Маркетинговая кампания</i> ] ([ <i>Marketing campaign</i> ]) [ <i>Бизнес-процесс</i> ] ([ <i>Business-process</i> ]) [ <i>Действие процесса</i> ] ([ <i>User task</i> ]) [ <i>Веб-сервис</i> ] ([ <i>Web service</i> ])
[ <b>Изменить процесс</b> ] ([ <b>Edit process</b> ])	Открыть объект для редактирования	Если объект создан третьими лицами, будет отображено соответствующее уведомление.	[ <i>Объект</i> ] ([ <i>Object</i> ])
[ <b>Экспортировать</b> ] ([ <b>Export</b> ])	Скачать конфигурационный элемент	Конфигурационный элемент будет скачан в формате *.md.	[ <i>Объект</i> ] ([ <i>Object</i> ]) [ <i>Клиентский модуль</i> ] ([ <i>Client module</i> ]) [ <i>Исходный код</i> ] ([ <i>Source code</i> ]) [ <i>Кейс</i> ] ([ <i>Case</i> ]) [ <i>Маркетинговая кампания</i> ] ([ <i>Marketing campaign</i> ])

Пункт меню	Действие	Дополнительные сведения	Конфигурационный элемент
			[ Действие процесса ] ([ User task ])  [ Веб-сервис ] ([ Web service ])
<b>[ Обновить структуру БД ]</b> ([ Update database structure ])	Обновить структуру базы данных для объекта		[ Объект ] ([ Object ])
<b>[ Сгенерировать исходный код ]</b> ([ Generate source code ])	Сгенерировать исходный код конфигурационного элемента	Будет выполнено, если процесс содержит компилируемые элементы.	[ Объект ] ([ Object ]) [ Бизнес-процесс ] ([ Business-process ]) [ Действие процесса ] ([ User task ]) [ Веб-сервис ] ([ Web service ])
<b>[ Открыть метаданные ]</b> ([ Open metadata ])	Открыть окно метаданных конфигурационного элемента.		[ Объект ] ([ Object ]) [ Клиентский модуль ] ([ Client module ]) [ Исходный код ] ([ Source code ]) [ Кейс ] ([ Case ]) [ Маркетинговая кампания ] ([ Marketing campaign ]) [ Бизнес-процесс ] ([ Business-process ]) [ Действие процесса ] ([ User task ]) [ Веб-сервис ] ([ Web service ])

Реестр позволяет удалить конфигурационный элемент (кнопка ). Данная кнопка появляется при наведении курсора на запись реестра раздела [ Конфигурация ] только для конфигурационных элементов пользовательских пакетов.

## Хранилища системы контроля версий



**Инструменты работы** с хранилищами системы контроля версий:

- Группа [ *Хранилища SVN* ] ([ *SVN repositories* ]) (6) выпадающего списка [ *Действия* ] ([ *Actions* ]) панели инструментов (1).
- Меню пакета.

**Группа действий** [ *Хранилища SVN* ] ([ *SVN repositories* ]) (6) позволяет:

- Установить пакет из хранилища системы контроля версий (пункт [ *Установить пакет из хранилища* ] ([ *Install package from repository* ])).
- Открыть вкладку [ *Список хранилищ* ] ([ *List of repositories* ]) (пункт [ *Открыть список хранилищ* ] ([ *Open list of repositories* ])), которая позволяет создавать, настраивать и удалять ссылки на доступные хранилища системы контроля версий.
- Синхронизировать конфигурацию с хранилищем до последней ревизии (пункт [ *Восстановить из хранилища* ] ([ *Restore from repository* ])). Это приведет к потере изменений, не зафиксированных в хранилище системы контроля версий.

Использование системы контроля версий SVN доступно только для приложения Creatio на платформе .NET Framework. Описание работы с хранилищами системы контроля версий содержится в статье [Контроль версий в Creatio IDE](#).

**Меню пакета** позволяет:

- Выгрузить пакет в zip-архив (пункт [ *Экспортировать* ] ([ *Export* ])). Экспорт пакетов описан в статье [Экспорт и импорт пакетов](#).
- Обновить пакет из подключенного хранилища системы контроля версий (пункт [ *Обновить из хранилища* ] ([ *Update from repository* ])). Обновление пакета описано в статье [Обновить пакет из системы контроля версий](#).
- Зафиксировать пакет в подключенном хранилище системы контроля версий (пункт [ *Зафиксировать в хранилище* ] ([ *Commit to repository* ])). Фиксация пакета в системе контроля версий описана в статье [Фиксировать пакет в системе контроля версий](#).



## Компиляция конфигурации

Запуск компиляции изменений в конфигурации выполняется нажатием [ *Компилировать* ] ([ *Compile* ]) на панели инструментов (1).

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 и изменения вступят в силу для пользователей, работающих в текущей конфигурации.

При компиляции могут возникнуть **ошибки**, которые отображаются в диалоговом окне. **Свойства** ошибок компиляции:

- Иконка типа ошибки (ошибка  или предупреждение .
- Имя файла с ошибкой.
- Описание ошибки.

- Код ошибки.
- Номер строки с ошибкой.

## Заккрыть раздел [ Конфигурация ]

Чтобы закрыть вкладку с разделом [ Конфигурация ], нажмите [ Заккрыть ] ([ Close ]) на панели инструментов (1).

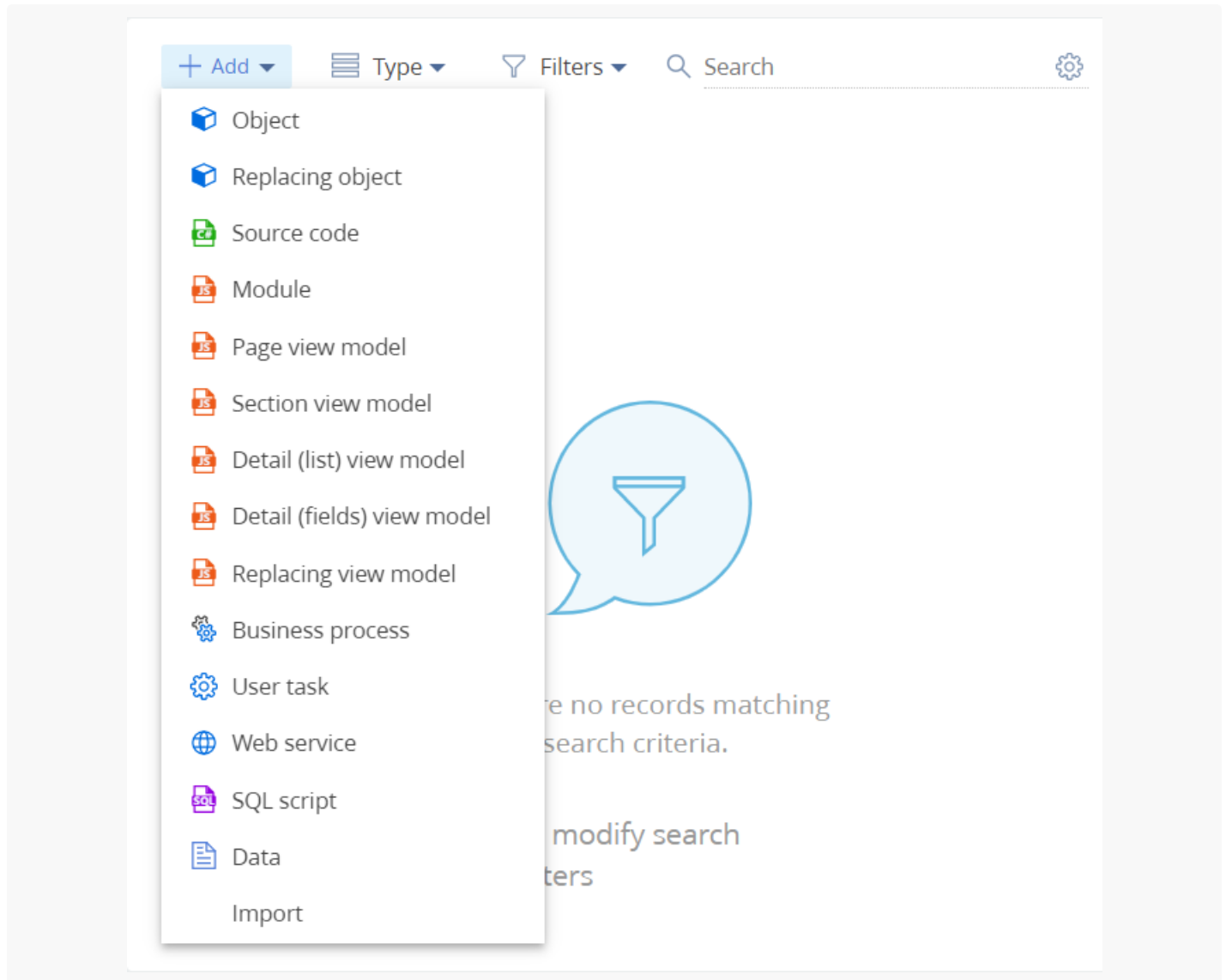
# Разработка конфигурационных элементов



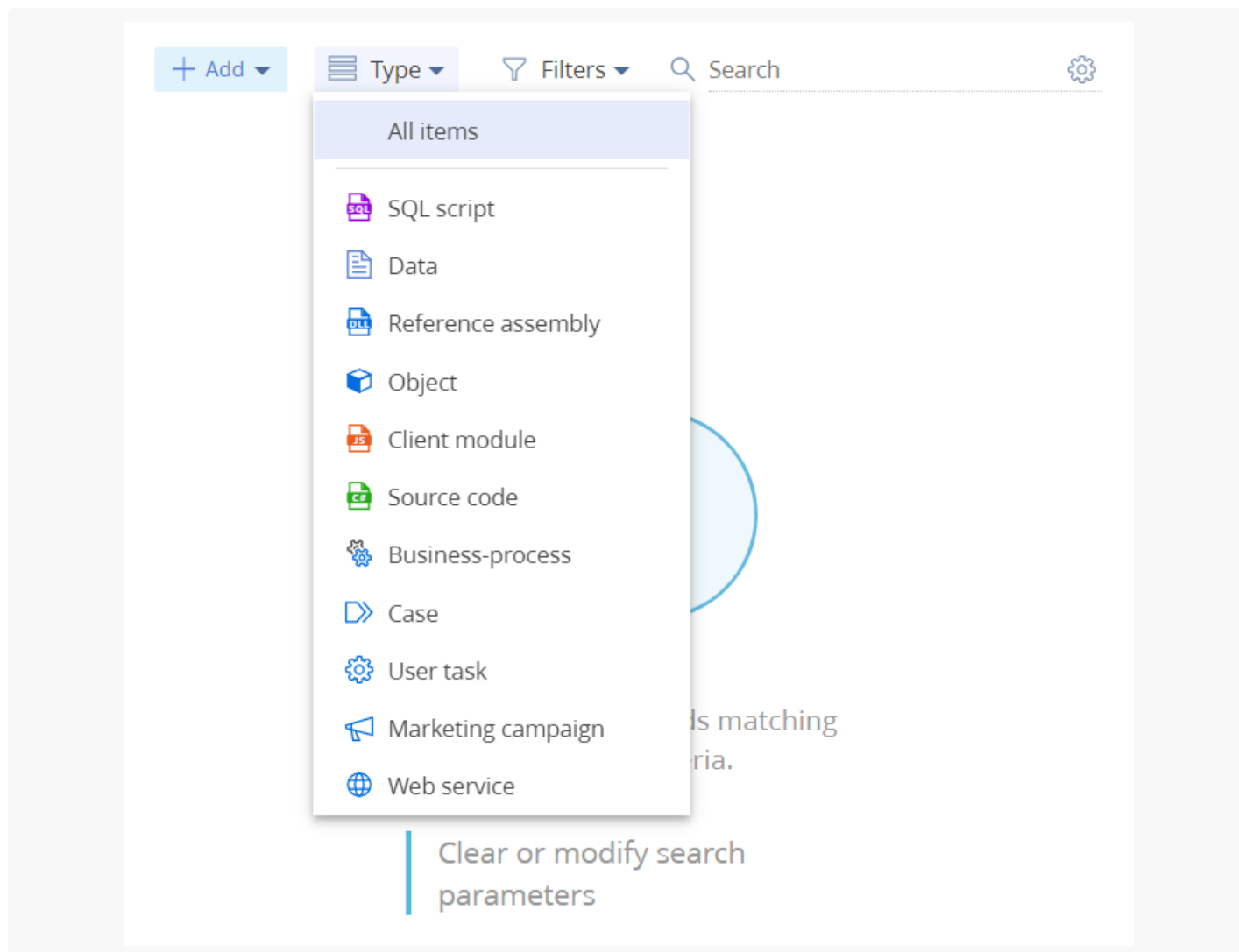
Сложный

**Схема** — основа конфигурации Creatio. С точки зрения программной реализации схема — это класс ядра, который наследуется от базового класса `Schema`. Конфигурационные элементы (значения выпадающего списка [ Добавить ] ([ Add ]) панели инструментов реестра раздела [ Конфигурация ] ([ Configuration ])), представлен схемой соответствующего типа (выпадающий список [ Тип ] ([ Type ])).

Виды конфигурационных элементов



Типы схем



## Клиентский модуль

**Клиентский модуль** — это отдельный блок функциональности, который загружается и запускается по требованию. В соответствии с подходом [AMD](#) и несмотря на некоторые функциональные различия, клиентские модули Creatio имеют одинаковую структуру описания.

Клиентские модули используются для front-end разработки (на языке JavaScript) в приложении Creatio.

**Виды** схем клиентских модулей:

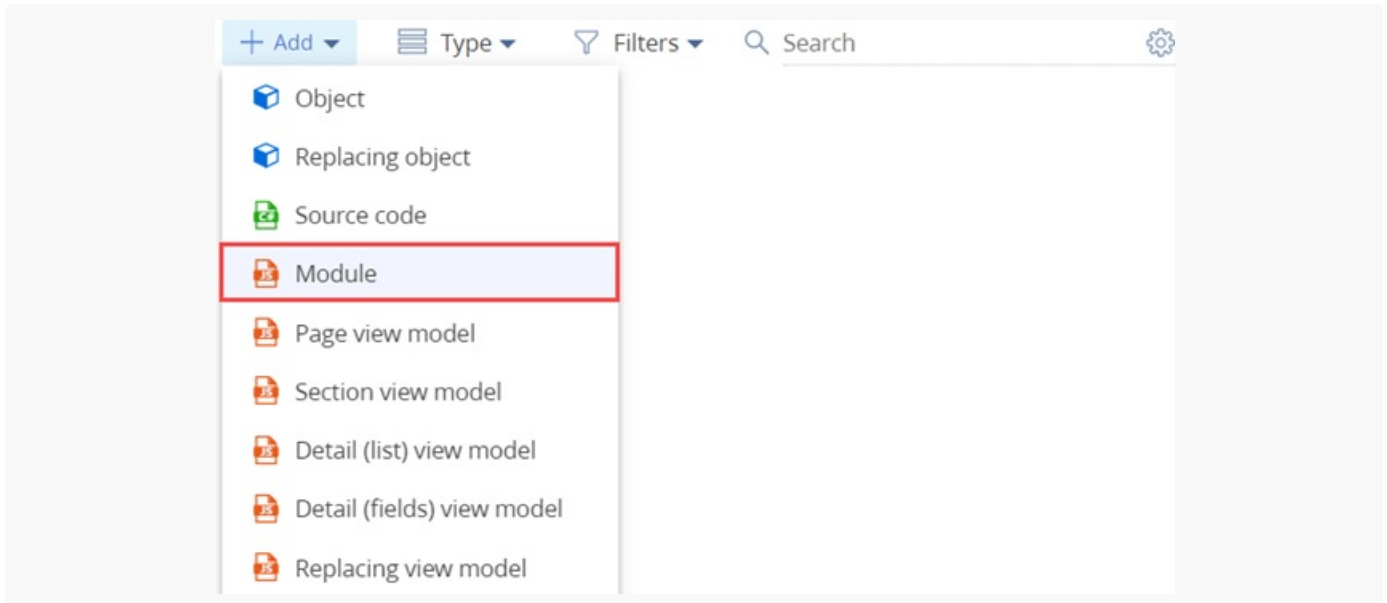
- Невизуальный модуль (схема модуля).
- Визуальный модуль (схема модели представления).
- Модуль расширения и замещающий клиентский модуль (схема замещающей модели представления).

Модули описаны в статье [Виды модулей](#).

## Схема модуля

**Алгоритм** разработки схемы:

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Модуль* ] ([ *Add* ] —> [ *Module* ]).





3. В дизайнера модуля заполните свойства схемы.



Основные **свойства** схемы:

- [ *Код* ] ([ *Code* ]) — название схемы (обязательное свойство). Должно содержать префикс (по умолчанию `usr`), указанный в системной настройке [ *Префикс названия объекта* ] (код [ *SchemaNamePrefix* ]), символы латинского алфавита и цифры.
- [ *Заголовок* ] ([ *Title* ]) — локализуемый заголовок схемы (обязательное свойство).
- [ *Пакет* ] ([ *Package* ]) — пользовательский пакет, в котором создается схема. Заполняется автоматически и недоступно для редактирования.
- [ *Описание* ] ([ *Description* ]) — локализуемое описание схемы.

Для применения заданных свойств нажмите [ Применить ] ([ Apply ]).

**Панель свойств** позволяет изменить основные свойства схемы (кнопка ) и задать дополнительные (кнопка ). Дополнительными свойствами являются [ Локализуемые строки ] ([ Localizable strings ]), [ Сообщения ] ([ Messages ]), [ Изображения ] ([ Images ]).

- В дизайнере модуля добавьте исходный код. Название модуля в функции `define()` должно совпадать с названием схемы (свойство [ Код ] ([ Code ])).

Если при написании кода допущена ошибка, то слева возле номера строки отображается тип ошибки (ошибка  или предупреждение ). При наведении курсора на тип ошибки отображается всплывающая подсказка с текстовым описанием.

- На панели инструментов дизайнера модуля нажмите [ Сохранить ] ([ Save ]).

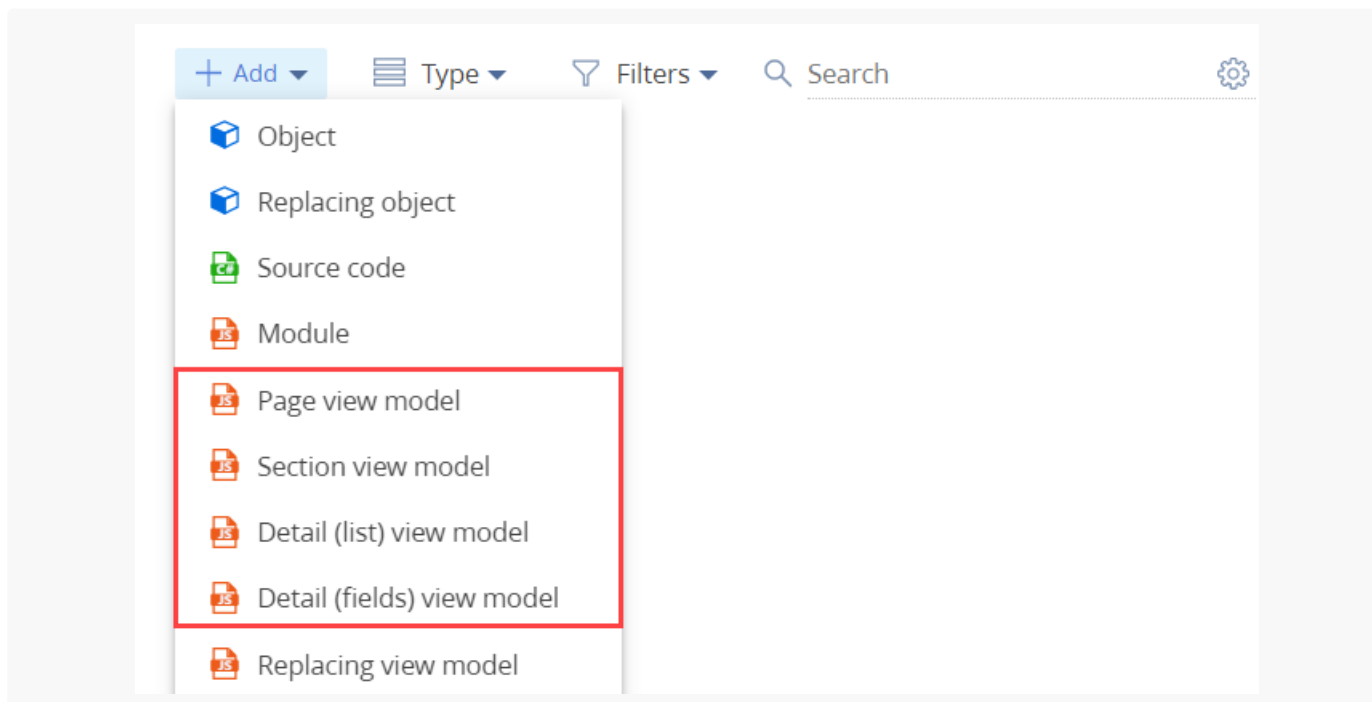
## Схема модели представления

**Виды** схем модели представления:

- Схема страницы записи раздела (пункт [ Модель представления страницы ] ([ Page view model ])).
- Схема страницы раздела с реестром и итогами (пункт [ Модель представления раздела ] ([ Section view model ])).
- Схема страницы детали с реестром (пункт [ Модель представления детали с реестром ] ([ Detail (list) view model ])).
- Схема страницы детали с полями ([ Модель представления детали с полями ] ([ Detail (fields) view model ])).

**Алгоритм** разработки схемы:

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] ([ *Add* ]) и выберите вид схемы модели представления.



3. В дизайнера модуля заполните свойства схемы.

Основные **свойства** схемы:



- [ *Код* ] ([ *Code* ]) — название схемы (обязательное свойство). Должно содержать префикс (по умолчанию `usr`), указанный в системной настройке [ *Префикс названия объекта* ] (код [ `SchemaNamePrefix` ]), символы латинского алфавита и цифры.
- [ *Заголовок* ] ([ *Title* ]) — локализуемый заголовок схемы (обязательное свойство).
- [ *Пакет* ] ([ *Package* ]) — пользовательский пакет, в котором создается схема. Заполняется автоматически и недоступно для редактирования.
- [ *Родительский объект* ] ([ *Parent object* ]) — родительский объект для текущего объекта. В выпадающем списке выберите родительский объект, свойства которого необходимо наследовать.
- [ *Описание* ] ([ *Description* ]) — локализуемое описание схемы.

The image shows a 'Module' configuration window with the following fields and values:



- Code \***: UsrContactSection
- Title \***: Contacts section
- Parent object \***: Contacts section
- Package**: Custom
- Description**: (empty)

At the bottom right, there are two buttons: 'CANCEL' and 'APPLY'.

Для применения заданных свойств нажмите [ Применить ] ([ Apply ]).

**Панель свойств** позволяет изменить основные свойства схемы (кнопка ) и задать дополнительные (кнопка ). Дополнительными свойствами являются [ Локализуемые строки ] ([ Localizable strings ]) и [ Изображения ] ([ Images ]).

- В дизайнере модуля добавьте исходный код. Название модуля в функции `define()` должно совпадать с названием схемы (свойство [ Код ] ([ Code ])). Схема модели представления обязательно должна быть наследником базовой схемы `BaseModulePageV2`.

Если при написании кода допущена ошибка, то слева возле номера строки отображается тип ошибки (ошибка  или предупреждение ). При наведении курсора на тип ошибки отображается всплывающая подсказка с текстовым описанием.

- На панели инструментов дизайнера модуля нажмите [ Сохранить ] ([ Save ]).

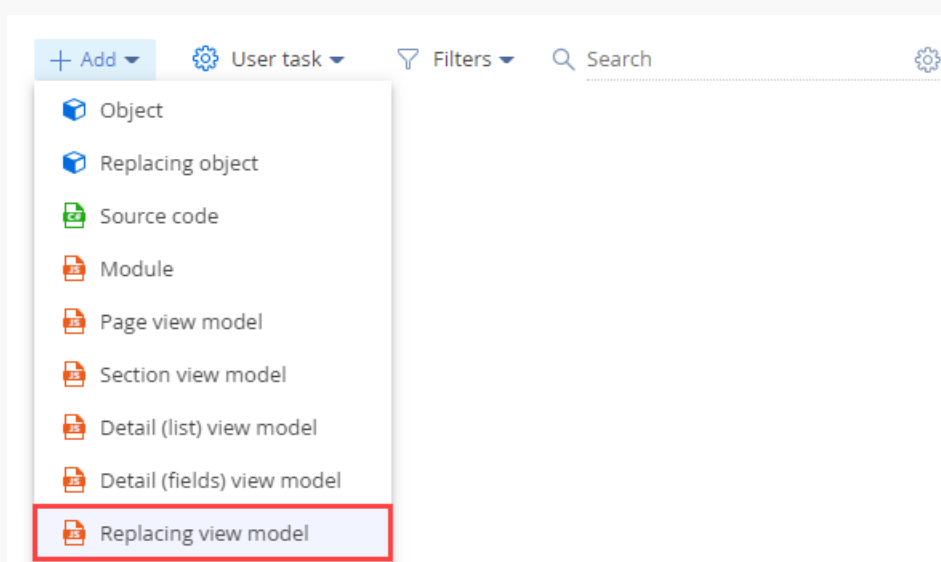
## Схема замещающей модели представления

Схемы **замещающих моделей представления** предназначены для расширения функциональности существующих схем. При этом существующие схемы также могут быть замещающими и принадлежать разным пакетам.

**Алгоритм** разработки схемы:

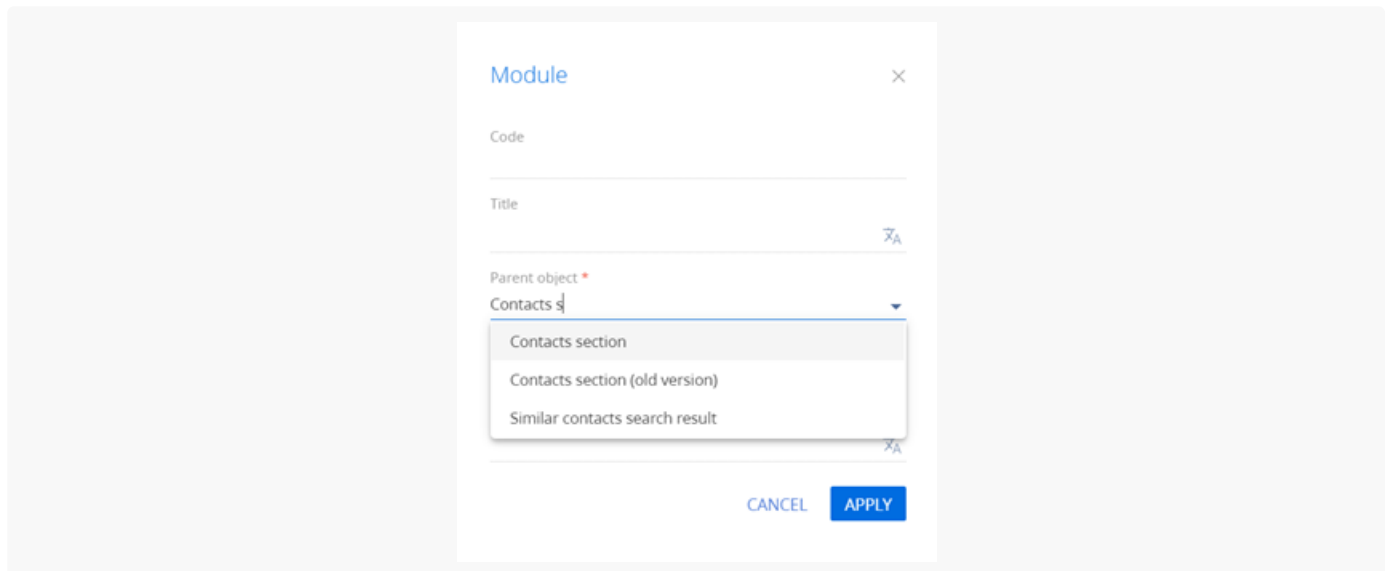
- [Перейдите в раздел \[ Конфигурация \]](#) ([ Configuration ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
- На панели инструментов реестра раздела нажмите [ Добавить ] —> [ Замещающая модель представления ] ([ Add ] —> [ Replacing view model ]).





### 3. В дизайнере модуля выберите родительский объект.

Чтобы модуль замещал раздел или страницу, в обязательном свойстве [ *Родительский объект* ] ([ *Parent object* ]) схемы укажите заголовок той базовой схемы модели представления, которую необходимо заместить. Например, для создания замещающей схемы раздела [ *Контакты* ] ([ *Contacts* ]) в качестве родительского объекта необходимо указать схему `ContactSectionV2`. Для этого в поле [ *Родительский объект* ] ([ *Parent object* ]) свойств замещающей схемы необходимо начать вводить заголовок "Раздел контакты" ("Contacts section") и выбрать нужное значение из выпадающего списка.





После подтверждения выбранного родительского объекта остальные свойства будут заполнены автоматически.

The screenshot shows a 'Module' configuration window with the following fields:



- Code \***: UsrContactSection
- Title \***: Contacts section
- Parent object \***: Contacts section
- Package**: Custom
- Description**: (empty)

At the bottom right, there are two buttons: 'CANCEL' and 'APPLY'.

Для применения заданных свойств нажмите [ Применить ] ([ Apply ]).

**Панель свойств** позволяет изменить основные свойства схемы (кнопка ) и задать дополнительные (кнопка ). Дополнительными свойствами являются [ Локализуемые строки ] ([ Localizable strings ]) и [ Изображения ] ([ Images ]).

- В дизайнере модуля добавьте исходный код. Название модуля в функции `define()` должно совпадать с названием схемы (свойство [ Код ] ([ Code ])).

Если при написании кода допущена ошибка, то слева возле номера строки отображается тип ошибки (ошибка  или предупреждение ). При наведении курсора на тип ошибки отображается всплывающая подсказка с текстовым описанием.

- На панели инструментов дизайнера модуля нажмите [ Сохранить ] ([ Save ]).

## Объект

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

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

**Виды** схем объектов:

- Базовые. Недоступны для редактирования, находятся в предустановленных [пакетах](#). Базовые схемы могут замещаться пользовательскими.
- Пользовательские. Создаются при кастомизации, находятся в пользовательских пакетах.

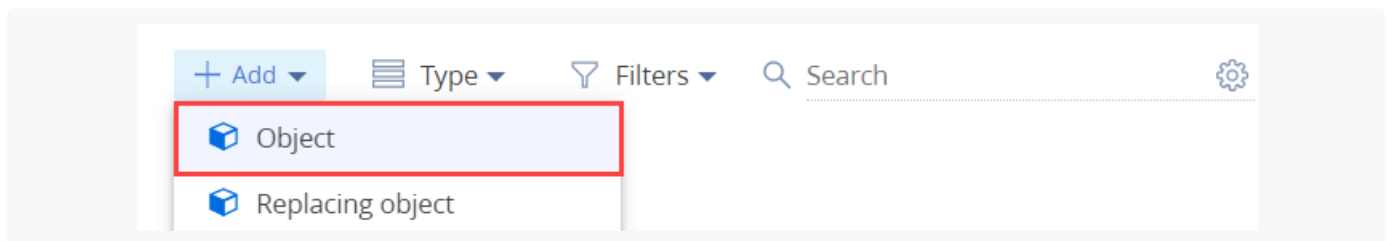
Платформа Creatio не ограничивает количество колонок объекта. Количество колонок в объекте ограничивается максимально допустимым количеством столбцов в таблицах базы данных, которую использует клиент.

Объекты используются для back-end разработки (на языке C#) в приложении Creatio.

## Схема объекта

**Алгоритм** разработки схемы:

1. [Перейдите в раздел \[ Конфигурация \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Объект* ] ([ *Add* ] —> [ *Object* ]).



3. В дизайнере объекта заполните свойства схемы.

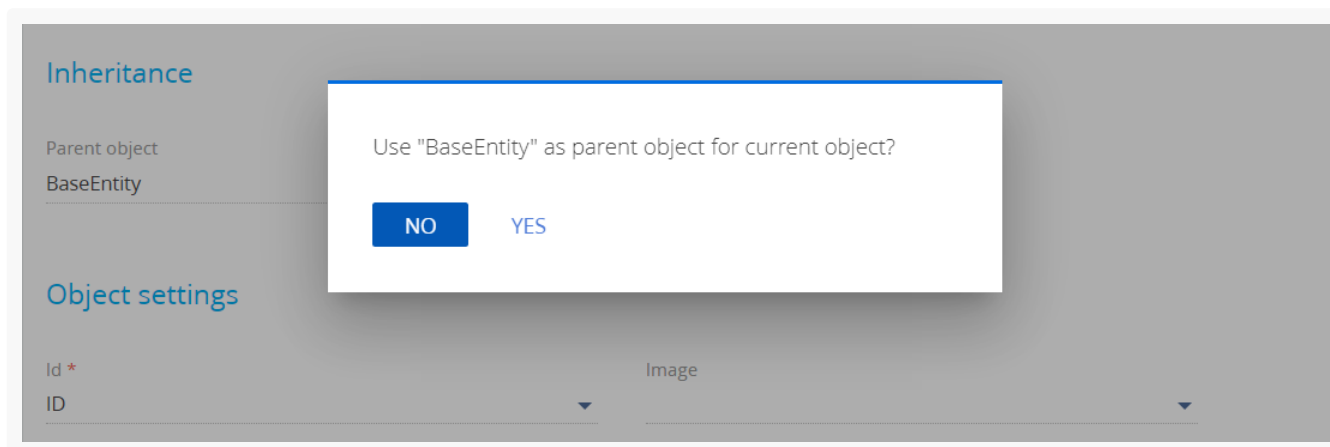
Основные **свойства** схемы:

- [ *Код* ] ([ *Code* ]) — название схемы (обязательное свойство). Должно содержать префикс (по умолчанию `Usr`), указанный в системной настройке [ *Префикс названия объекта* ] (код [ *SchemaNamePrefix* ]), символы латинского алфавита и цифры. Допустимая длина имени объекта — 128 символов. На базах Oracle ниже версии 12.2 не допускаются к использованию объекты с длиной имени более 30 символов.
- [ *Заголовок* ] ([ *Title* ]) — локализуемый заголовок схемы (обязательное свойство).

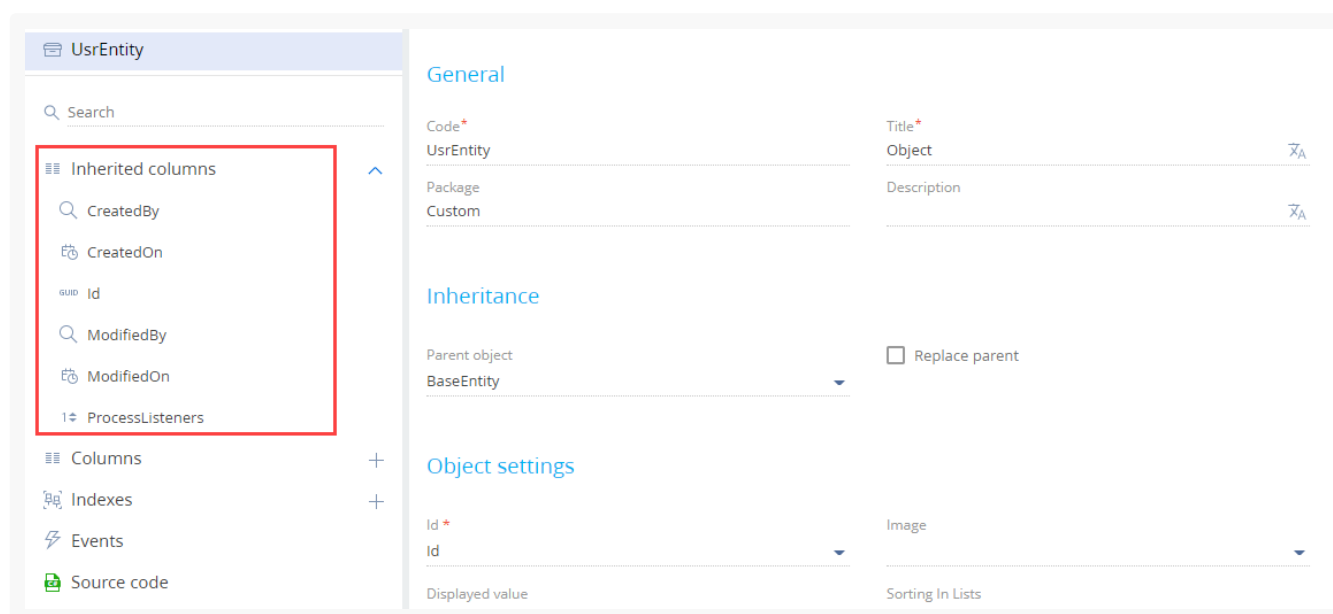
 A screenshot of the 'General' tab in the Creatio object designer. It shows two main sections. The left section has 'Code\*' with a value of 'UsrEntity' and 'Package' with a value of 'Custom'. The right section has 'Title\*' with a value of 'Object' and 'Description' which is empty. There are icons for localization (a square with 'x' and 'A') next to the 'Title' and 'Description' fields.

- [ *Родительский объект* ] ([ *Parent object* ]) — родительский объект для текущего объекта. Чтобы объект наследовал функциональность базового объекта, в свойстве [ *Родительский объект* ] ([ *Parent object* ]) схемы укажите код той базовой схемы объекта, функциональность которой необходимо наследовать. Например, для наследования функциональности базовой схемы `BaseEntity` в поле [ *Родительский объект* ] ([ *Parent object* ]) свойств схемы необходимо начать вводить код `BaseEntity` и выбрать нужное значение из выпадающего списка. После подтверждения выбранного родительского объекта к структуре объекта будут добавлены колонки, унаследованные от базового объекта.

Подтверждение использования родительского объекта



Унаследованные колонки в структуре объекта



- [ Идентификатор ] ([ Id ]) — системная колонка, используемая в качестве первичного ключа в таблице базы данных (обязательное свойство). Заполняется автоматически после установки свойства [ Родительский объект ] ([ Parent object ]).

Поскольку объект в системе является представлением таблицы в базе данных, то он обязательно должен содержать колонку-идентификатор. Для установки значения свойства [ Идентификатор ] ([ Id ]) в качестве родительского объекта укажите один из базовых объектов системы или в выпадающем списке выберите пользовательскую колонку типа [ Уникальный идентификатор ] ([ Unique identifier ]). Добавление пользовательской колонки рассмотрено ниже. Если попытаться сохранить схему объекта без идентификатора, то система выдаст предупреждение.

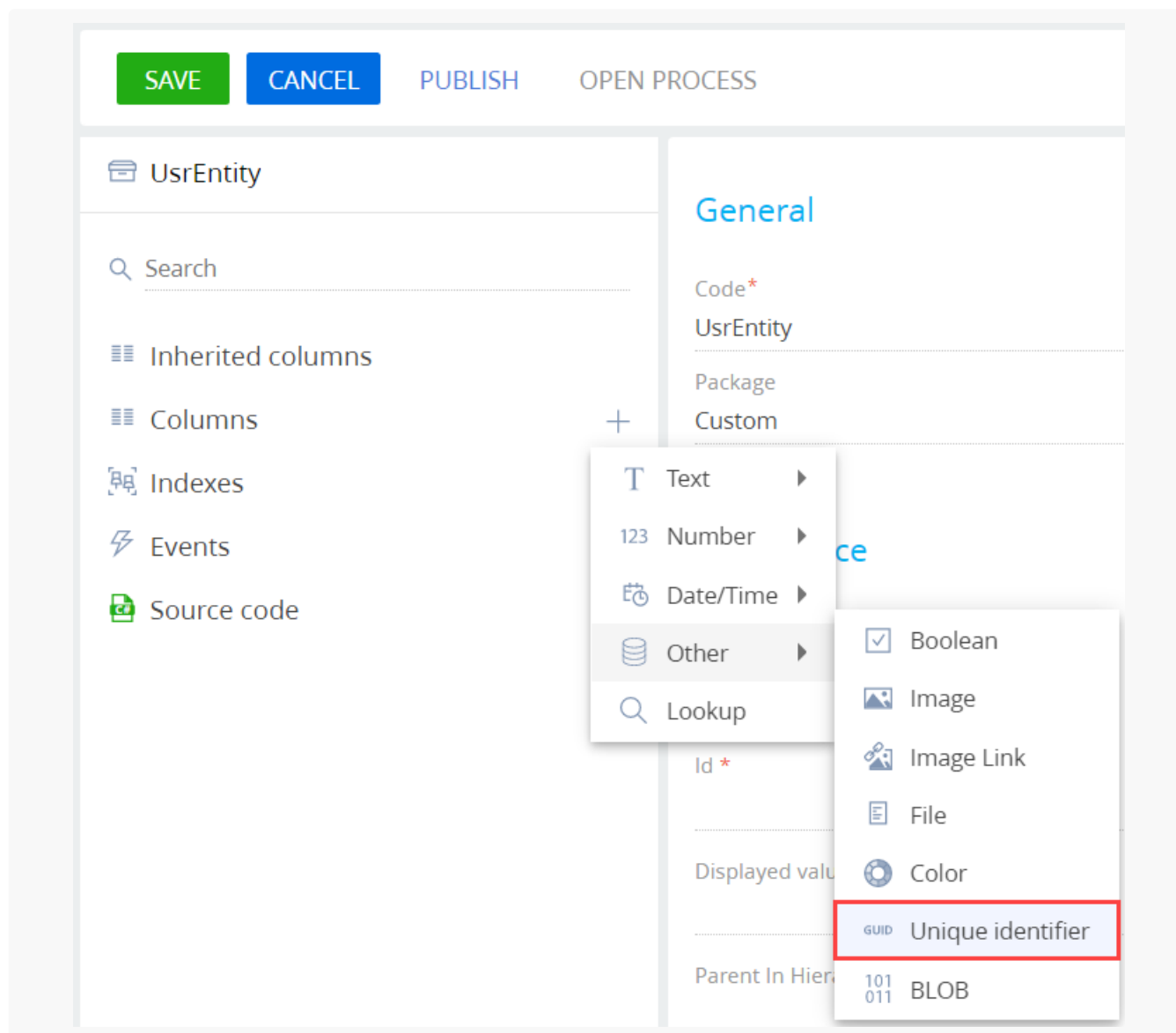
#### 4. Добавьте пользовательскую колонку в объект.

**Алгоритм** добавления в объект пользовательской колонки:

- В контекстном меню узла [ Колонки ] ([ Columns ]) структуры объекта нажмите +.
- В выпадающем меню выберите тип колонки и задайте ее свойства.

Для добавления колонки-идентификатора нажмите [ Другие ] —> [ Уникальный идентификатор ]


([ Other ] —> [ Unique identifier ]).



с. В дизайнера объекта заполните свойства добавляемой колонки.

Основные **свойства** добавляемой колонки:

- [ Код ] ([ Code ]) — название колонки (обязательное свойство). Значение по умолчанию устанавливается дизайнером объекта и может быть изменено.
- [ Заголовок ] ([ Title ]) — локализуемый заголовок колонки (обязательное свойство).
- [ Тип данных ] ([ Data type ]) — тип данных, содержащихся в колонке. Значение по умолчанию устанавливается дизайнером объекта в зависимости от выбранной команды добавления колонки.
- [ Обязательное ] ([ Required ]) — обязательность колонки. Выберите "На уровне приложения" ("Application Level"), поскольку колонка должна обязательно содержать значение.
- [ Значение по умолчанию ] ([ Default value ]) — значение по умолчанию.

Для установки значения по умолчанию нажмите  и заполните **поля**:

- [ Тип значения ] ([ *Default value type* ]) — выберите "Системная переменная" ("System variable").
- [ Системная переменная ] ([ *System variable* ]) — выберите "Новый идентификатор" ("New Id"), поскольку идентификаторы должны быть уникальными.

- [ Режим использования ] ([ *Usage mode* ]) — выберите "Расширенный" ("Advanced").

**Режимы использования** колонок, реализованные в Creatio IDE:

- [ Общие ] ([ *General* ]) — стандартный режим колонок в приложении.
- [ Расширенный ] ([ *Advanced* ]) — колонка отображается в конфигурации и доступна для

использования в приложении.

- [ *Никогда* ] ([ *None* ]) — колонка отображается в конфигурации как системная и недоступна для использования в приложении.

l. На панели инструментов дизайнера объекта нажмите [ *Сохранить* ] ([ *Save* ]) для временного сохранения изменений в метаданных.

m. Добавьте индексы в объект.

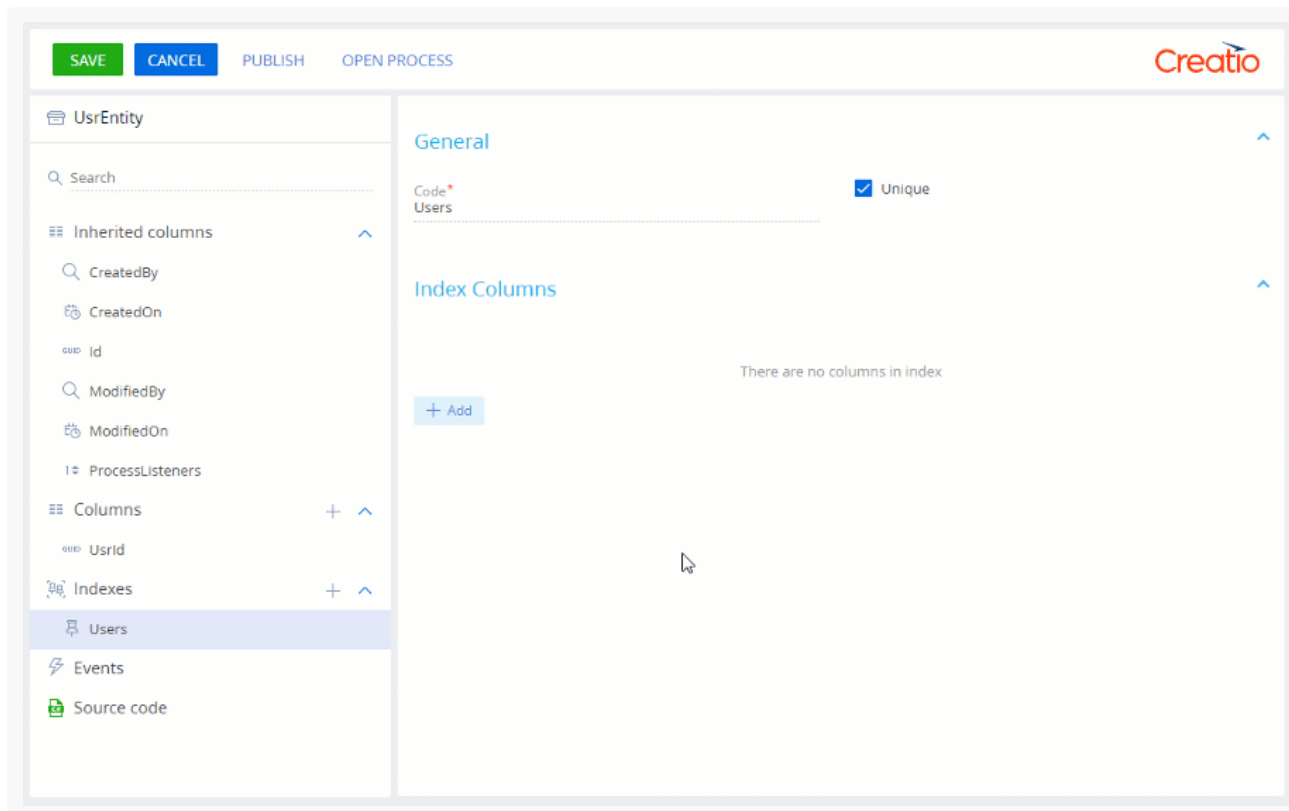
Кроме колонок, в объект могут быть добавлены индексы, которые при публикации объекта будут автоматически созданы в таблице базы данных.

В блоке свойств [ *Поведение* ] ([ *Behavior* ]) установите признак [ *Индексируемая* ] ([ *Indexed* ]), если необходимо создать индекс по одной колонке. В системе по умолчанию справочные колонки являются индексируемыми.



**Алгоритм** добавления составного индекса:

- a. Задайте название индекса. Для этого в контекстном меню элемента [ *Индексы* ] ([ *Indexes* ]) нажмите **+** и в поле [ *Код* ] ([ *Code* ]) укажите пользовательское название.
- b. Установите признак [ *Уникальный* ] ([ *Unique* ]) если для колонок индекса необходимо реализовать ограничение целостности (исключить возможность вставки повторяющихся комбинаций значений).
- c. Добавьте необходимые колонки в индекс. Для этого в блоке [ *Колонки индекса* ] ([ *Index Columns* ]) нажмите [ *Добавить* ] ([ *Add* ]), выберите колонку объекта и укажите направление сортировки.

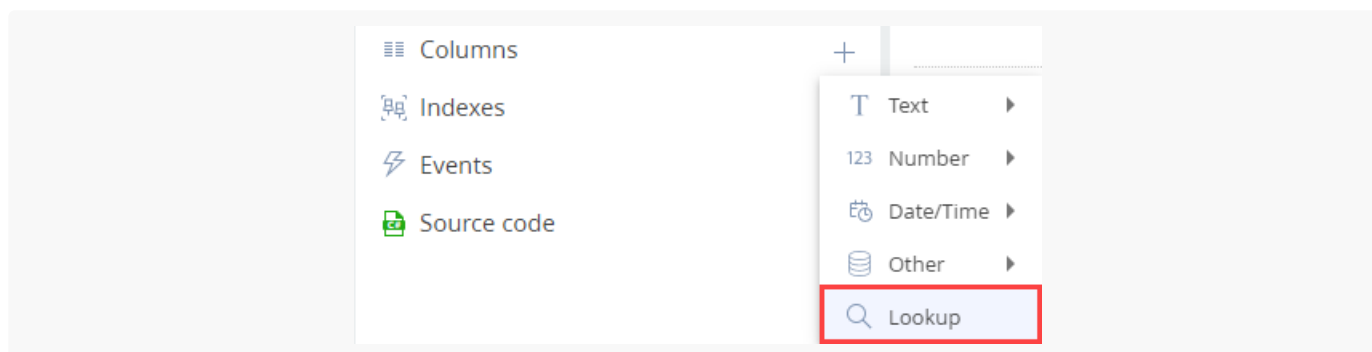


- п. На панели инструментов дизайнера объекта нажмите [ Сохранить ] ([ Save ]) для временного сохранения изменений в метаданных схемы.
- о. На панели инструментов дизайнера объекта нажмите [ Опубликовать ] ([ Publish ]) для окончательного сохранения схемы и создания соответствующей таблицы в базе данных.

Для объекта можно установить **каскадную связь**. Она настраивается для колонки типа [ Справочник ] ([ Lookup ]).

**Алгоритм** добавления в объект колонки типа [ Справочник ] ([ Lookup ]):

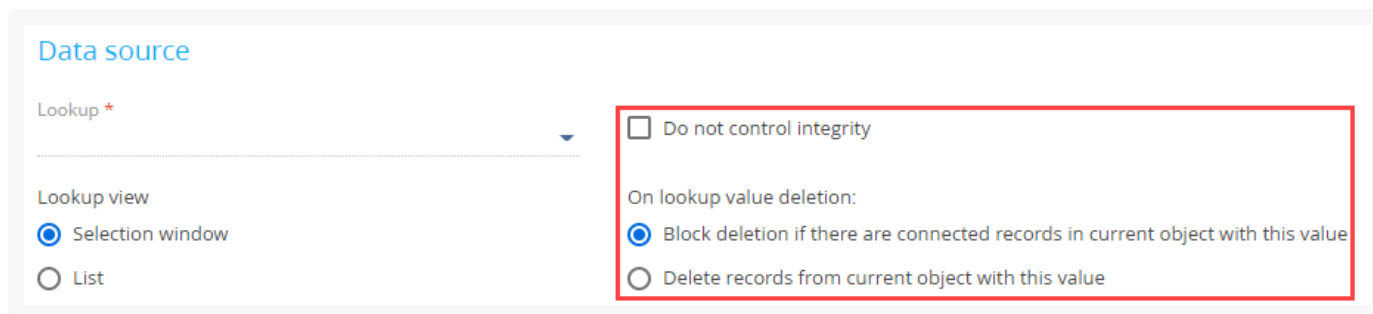
1. В контекстном меню узла [ Колонки ] ([ Columns ]) структуры объекта нажмите +.
2. Для добавления колонки типа [ Справочник ] ([ Lookup ]) нажмите [ Справочник ] ([ Lookup ]).



Каскадная связь настраивается в блоке свойств [ Источник данных ] ([ Data source ]) с помощью:

- признака [ Не контролировать целостность ] ([ Do not control integrity ]).
- опций пункта [ При удалении значения справочника ] ([ On lookup value deletion ]).





**Data source**

Lookup \*

Lookup view

☒ Selection window

☐ List

☐ Do not control integrity

On lookup value deletion:

☒ Block deletion if there are connected records in current object with this value

☐ Delete records from current object with this value

Рассмотрим каскадную связь на примере объекта [ *Контакт* ] ([ *Contact* ]), который связан по справочной колонке [ *AccountId* ] с объектом [ *Контрагент* ] ([ *Account* ]). Для этого в поле [ *Выбор объекта* ] ([ *Lookup* ]) выберите [ *Account* ].

**Варианты настройки** каскадной связи:

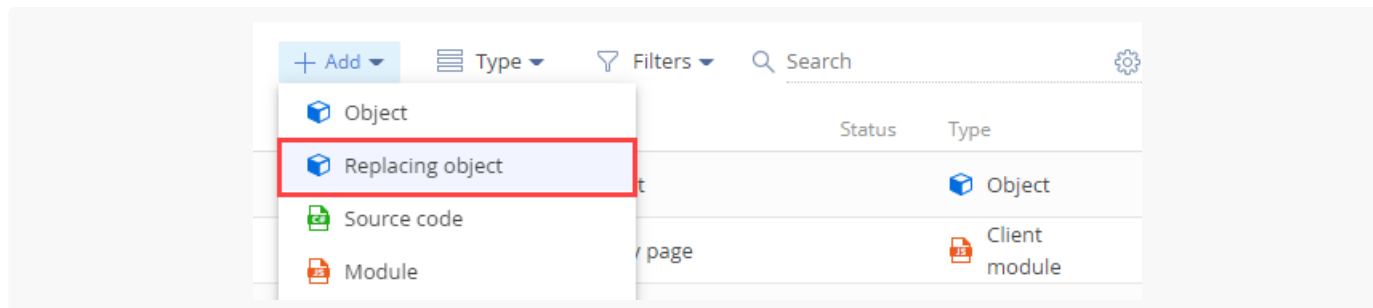
- Если **установлен признак** [ *Не контролировать целостность* ] ([ *Do not control integrity* ]), то удаление контрагента будет выполнено. При этом не будут удалены контакты, связанные с текущим контрагентом.
- Если **не установлен признак** [ *Не контролировать целостность* ] ([ *Do not control integrity* ]) и **выбрана опция** [ *Блокировать удаление, если есть связанные записи в текущем объекте с этим значением* ] ([ *Block deletion if there are connected records in current object with this value* ]), то удаление контрагента будет заблокировано, если присутствуют контакты, связанные с текущим контрагентом. В этом случае приложение выдаст предупреждающее сообщение. После подтверждения удаление контрагента будет выполнено. При этом не будут удалены контакты, связанные с текущим контрагентом.
- Если **не установлен признак** [ *Не контролировать целостность* ] ([ *Do not control integrity* ]) и **выбрана опция** [ *Удалять записи из текущего объекта с этим значением* ] ([ *Delete records from current object with this value* ]), то удаление контрагента будет выполнено вместе с удалением контактов, связанных с текущим контрагентом.

## Схема замещающего объекта

Схемы **замещающих объектов** предназначены для расширения функциональности существующих схем. При этом существующие схемы также могут быть замещающими и принадлежать разным пакетам.

**Алгоритм** разработки схемы:

1. [Перейдите в раздел \[ \*Конфигурация\* \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Замещающий объект* ] ([ *Add* ] —> [ *Replacing object* ]).



### 3. В дизайнере объекта выберите родительский объект.

Чтобы объект замещал функциональность базового объекта, в обязательном свойстве [ *Родительский объект* ] ([ *Parent object* ]) схемы укажите название той базовой схемы объекта, функциональность которой необходимо заместить. Например, для замещения функциональности базовой схемы `BaseEntity` в поле [ *Родительский объект* ] ([ *Parent object* ]) свойств схемы необходимо начать вводить код `BaseEntity` и выбрать нужное значение из выпадающего списка. После подтверждения выбранного родительского объекта остальные свойства будут заполнены автоматически.

### General

Code \*

BaseEntity

Package

Custom

Title \*

Base object

Description

### Inheritance

Parent object \*

BaseEntity

☒ Replace parent

### Object settings

Id \*

Id

Displayed value

Parent In Hierarchy

Change Log Object Name

Localization Object Name

Image

Sorting In Lists

Owner

Permission Object Name

### 4. На панели инструментов дизайнера объекта нажмите [ *Сохранить* ] ([ *Save* ]) для временного

сохранения изменений в метаданных схемы.

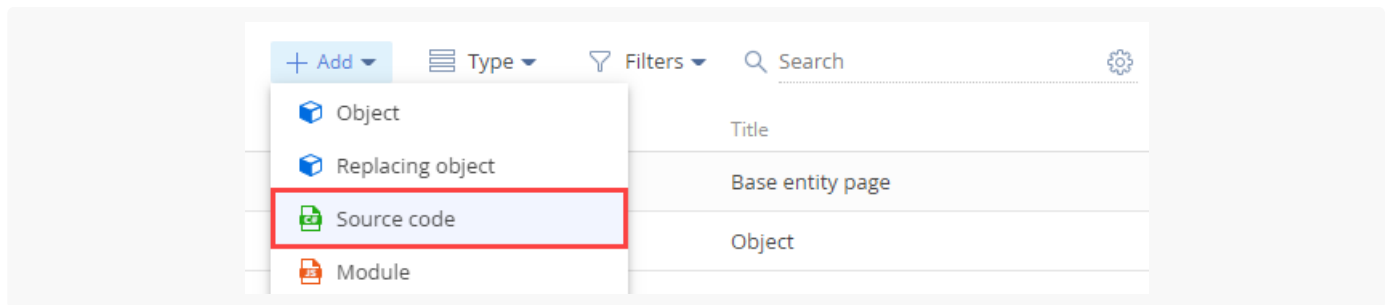
5. На панели инструментов дизайнера объекта нажмите [ *Опубликовать* ] ([ *Publish* ]) для выполнения изменений на уровне базы данных. Результатом успешной публикации объекта являются созданные (или измененные) физические объекты в базе данных — таблица, столбцы, индексы.

## Исходный код

Схема [ *Исходный код* ] ([ *Source code* ]) используется для back-end разработки (на языке C#) в приложении Creatio.

**Алгоритм** разработки схемы:



1. [Перейдите в раздел \[ \*Конфигурация\* \]](#) ([ *Configuration* ]) и выберите пользовательский [пакет](#), в который будет добавлена схема.
2. На панели инструментов реестра раздела нажмите [ *Добавить* ] —> [ *Исходный код* ] ([ *Add* ] —> [ *Source code* ]).



3. В дизайнере исходного кода заполните свойства схемы.

Основные **свойства** схемы:

- [ *Код* ] ([ *Code* ]) — название схемы (обязательное свойство). Должно содержать префикс (по умолчанию `usr`), указанный в системной настройке [ *Префикс названия объекта* ] (код [ *SchemaNamePrefix* ]), символы латинского алфавита и цифры.
- [ *Заголовок* ] ([ *Title* ]) — локализуемый заголовок схемы (обязательное свойство).
- [ *Пакет* ] ([ *Package* ]) — пользовательский пакет, в котором создается схема. Заполняется автоматически и недоступно для редактирования.
- [ *Описание* ] ([ *Description* ]) — локализуемое описание схемы.

**Панель свойств** позволяет изменить основные свойства схемы (кнопка ) и задать дополнительные (кнопка ). Дополнительным свойством является [ *Локализуемые строки* ] ([ *Localizable strings* ]).

Source code

Code \*

UsrcSourceCode

Title \*

Source code

Package

Custom

Description

CANCEL APPLY

Для применения заданных свойств нажмите [ Применить ] ([ Apply ]).

4. В дизайнера исходного кода добавьте исходный код. Название класса, объявленного в исходном коде, должно совпадать с названием схемы (свойство [ Код ] ([ Code ])).
5. На панели инструментов дизайнера исходного кода нажмите [ Сохранить ] ([ Save ]) для временного сохранения изменений в метаданных схемы.
6. На панели инструментов дизайнера исходного кода нажмите [ Опубликовать ] ([ Publish ]) для выполнения изменений на уровне базы данных.