



# 1С-Битрикс: Управление сайтом

Быстрый старт разработчика

Глоссарий





## Содержание

Введение .....	3
Глава 1. Глоссарий.....	4



## Введение

---

Учебный курс для начинающего разработчика. Вы первый раз обращаетесь к популярной платформе Bitrix Framework и не знаете откуда начать? Начните с этого курса.

**⚠ Внимание!** Курс *Быстрый старт разработчика* - "выжимка" из огромного объема справочной информации по Bitrix Framework. Он создан с целью облегчить начинающему разработчику изучение системы и не является ни в коей мере заменой документации и базовых учебных курсов. В нем даются только основы с указанием более детальных источников информации.

При условии качественного изучения материалов курса, по его окончании специалист получит базовые понятия о платформе и ссылки на все, что может вам понадобиться при углубленном изучении Bitrix Framework.

Минимальный уровень знаний, необходимый при изучении:

- PHP
- HTML, CSS

**⚠ Примечание.** Для успешной разработки структуры информационных блоков сложных по структуре проектов, полезно будет получить хотя бы базовые знания по [реляционным базам данных](#).

**Список ссылок по теме:**

- [PHP](#)
- [HTML, CSS](#)



## Глава 1. Глоссарий

Каждая платформа имеет свои термины, используемые в работе. Чтобы без проблем ориентироваться в документации, учебных курсах, при общении с разработчиками желательно объяснять ваши трудности не «на пальцах», а на языке понятном специалистам. В главе приведен глоссарий терминов в помощь вам.

<ul style="list-style-type: none"> <li>• <a href="#">Портал</a></li> <li>• <a href="#">Сайт</a></li> <li>• <a href="#">Шаблон сайта</a></li> <li>• <a href="#">Раздел сайта</a></li> <li>• <a href="#">Компонент 1.0</a></li> <li>• <a href="#">Компонент 2.0</a></li> <li>• <a href="#">Включаемые области</a></li> <li>• <a href="#">Язык</a></li> <li>• <a href="#">Языковой файл</a></li> <li>• <a href="#">Шаблон страницы</a></li> <li>• <a href="#">Пролог</a></li> <li>• <a href="#">Эпилог</a></li> <li>• <a href="#">Тело страницы</a></li> <li>• <a href="#">Навигационная цепочка</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">Административная часть</a></li> <li>• <a href="#">Публичная часть</a></li> <li>• <a href="#">Режим Правка</a></li> <li>• <a href="#">Индексная страница (файл)</a></li> <li>• <a href="#">Мета-тэг</a></li> <li>• <a href="#">Локализация</a></li> <li>• <a href="#">Хост</a></li> <li>• <a href="#">Доменное имя (домен)</a></li> <li>• <a href="#">IP адрес</a></li> <li>• <a href="#">Accept-Language</a></li> <li>• <a href="#">Время в Unix-формате</a></li> <li>• <a href="#">Права в Unix системах</a></li> <li>• <a href="#">Путь относительно корня</a></li> <li>• <a href="#">Полный путь</a></li> <li>• <a href="#">Абсолютный путь</a></li> </ul>	<ul style="list-style-type: none"> <li>• <a href="#">DocumentRoot</a></li> <li>• <a href="#">Сабмит</a></li> <li>• <a href="#">Сессия</a></li> <li>• <a href="#">HTML-безопасный вид</a></li> <li>• <a href="#">Связывание переменных</a></li> <li>• <a href="#">Дамп</a></li> <li>• <a href="#">cron (крон)</a></li> <li>• <a href="#">Буферизация</a></li> <li>• <a href="#">Постоянное соединение (persistent)</a></li> <li>• <a href="#">Почтовое событие</a></li> <li>• <a href="#">Тип почтового события</a></li> <li>• <a href="#">Почтовый шаблон</a></li> <li>• <a href="#">Бюджет пользователя</a></li> <li>• <a href="#">Кастомизация</a></li> </ul>
--	--	---

Термин	Описание
Портал	<p>Один набор файлов, хранящихся в каталоге <code>/bitrix/modules/</code>, и одна копия базы. Портал включает в себя один или более <a href="#">сайтов</a>. Синонимом данного термина может служить: «экземпляр продукта», «одна инсталляция системы», «одна копия системы».</p>
Сайт	<p>Совокупность таких понятий как:</p> <ul style="list-style-type: none"> <li>• <b>учетная запись в базе данных</b> - создаётся в административном меню <b>Сайты</b>, включает в себя следующие основные параметры: <ul style="list-style-type: none"> <li>○ <b>идентификатор</b> - набор символов, идентифицирующий сайт;</li> <li>○ <b>доменное имя</b> - одно или более доменное имя сайта;</li> <li>○ <b>папка сайта</b> - путь к каталогу в котором будет храниться <a href="#">публичная часть</a> сайта</li> <li>○ <b>язык сайта</b></li> <li>○ <b>формат даты</b></li> <li>○ <b>формат времени</b></li> <li>○ <b>URL</b> - протокол и доменное имя по умолчанию (например, <code>http://www.site.ru</code>)</li> <li>○ <b>DocumentRoot</b> - если многосайтовость организована на разных доменах, то в данном параметре должен</li> </ul> </li> </ul>



	<p>храниться путь в файловой системе сервера к корню сайта.</p> <ul style="list-style-type: none"> <li>○ <b>условия подключения шаблонов</b> - каждый сайт может иметь один или более <a href="#">шаблонов</a> для отображения скриптов своей <a href="#">публичной части</a>, каждый такой шаблон может быть подключен по тому или иному условию</li> <li>● <a href="#">публичная часть</a> - совокупность скриптов (страниц), лежащих в «папке сайта» и принадлежащих этому сайту</li> <li>● <b>настройки</b> - каждый модуль может иметь ряд настроек, связанных с сайтом, например у модуля «Информационные блоки» эти настройки представляют из себя привязку информационного блока к тому или иному сайту, у модуля «Техподдержка» - привязку статуса, категории и т.п. к сайту.</li> </ul>
--	---

Синонимом можно считать - **дизайн сайта, скин сайта**. Для показа одного [сайта](#) можно использовать несколько различных шаблонов. В шаблон сайта входят:

#### Шаблон сайта

- набор файлов в каталоге **/bitrix/templates/ID\_шаблона/**, где **ID\_шаблона** - поле **ID** в форме редактирования шаблона сайта. Структура данного каталога:
  - **/components/** - каталог с компонентами, принадлежащими тому или иному модулю;
  - **/lang/** - [языковые файлы](#), принадлежащие как данному шаблону в целом, так и отдельным компонентам;
  - **/images/** - каталог с изображениями данного шаблона;
  - **/page\_templates/** - каталог с [шаблонами страниц](#) и их описанием, хранящимся в файле **.content.php**;
  - **/include\_areas/** - каталог с файлами - содержимым включаемых областей;
  - **header.php** - [пролог](#) данного шаблона;
  - **footer.php** - [эпилог](#) данного шаблона;
  - **styles.css** - CSS стили, используемые на страницах сайта, когда используется данный шаблон;
  - **template\_styles.css** - CSS стили, используемые в самом шаблоне;
  - **.тип меню.menu\_template.php** - шаблон вывода меню соответствующего типа;
  - **chain\_template.php** - шаблон по умолчанию для вывода навигационной цепочки;
  - а также ряд других вспомогательных произвольных файлов, входящих в данный шаблон.

<b>Раздел сайта</b>	Каталог в файловой системе сервера. В <b>Bitrix Framework</b> структура сайта - это файловая структура сервера, поэтому страницы сайта - это файлы, а разделы сайта - соответственно каталоги.
---------------------	--

#### Компонент

Является частью того или иного модуля, представляет из себя логически заверченный код, хранящийся в одном файле, принимающий ряд параметров, выполняющий ряд действий и выводящий какой-либо результат (например. в виде HTML кода).



Использование компонента является наиболее предпочтительным способом организации вывода информации как в [публичной](#), так и в [административной](#) частях. В понятие **компонент** входят следующие элементы:

- **основной файл** - подключается с помощью функции `CMain::IncludeFile`, представляет из себя скрипт на языке PHP, реализующий логику компонента;
- **языковые файлы** - используются в компоненте для представления его результата на различных языках;
- **описание** - это PHP код, используемый визуальным редактором для управления компонентом и его параметрами;

<b>Компонент 2.0</b>	<p>Компоненты являются блоками, с помощью которых строится <a href="#">публичная</a> часть сайта.</p> <p>Основные особенности компонентов версии 2.0:</p> <ul style="list-style-type: none"><li>• В компонентах 2.0 разделены логика и представление. Для одной логики может быть несколько представлений, в том числе зависящих от шаблона текущего сайта. Логика - это сам компонент, представление - это шаблон вывода компонента.</li><li>• Компоненты централизованно хранятся в одной папке. Это обеспечивает большую целостность и понятность структуры сайта. Папка доступна для обращений, а значит, компонент и его шаблоны могут легко подключать свои дополнительные ресурсы.</li><li>• Компонент хранит все, что ему нужно для работы, в своей папке. Поэтому компоненты можно легко переносить между проектами. Компонент неделим.</li><li>• Компоненты бывают простые и комплексные. Простые служат для создания различных областей на одной странице (список новостей, топ элементов каталога, случайное фото и т.д.). А комплексные используются для создания полнофункциональных разделов сайта (полный каталог товаров, новостной раздел, форум и т.д.).</li></ul>
<b>Включаемые области</b>	<p>Это специально выделенная область на странице сайта, которую можно редактировать отдельно от основного содержания страницы. Реализуется с помощью специального программного компонента.</p>
<b>Язык</b>	<p>Это учетная запись в базе данных доступная для редактирования в административном меню на странице <b>Языки интерфейса</b> со следующими полями:</p> <ul style="list-style-type: none"><li>• Идентификатор;</li><li>• Название;</li><li>• Формат даты;</li><li>• Формат времени;</li><li>• Кодировка.</li></ul>



	<p>Как в <a href="#">публичной</a>, так и <a href="#">административной</a> частях, язык в первую очередь используется в работе с <a href="#">языковыми файлами</a>. В административной части язык определяет формат времени, даты, кодировку страниц (в публичной - данные параметры определяются настройками сайта).</p>
<b>Языковой файл</b>	<p>Представляет из себя PHP скрипт, хранящий переводы языковых фраз на тот или иной язык. Данный скрипт состоит из массива <code>\$MESS</code>, ключи которого - идентификаторы языковых фраз, а значения - переводы на соответствующий язык. Для каждого <a href="#">языка</a> существует свой набор языковых файлов, хранящихся как правило в каталогах <code>/lang/</code>. Языковые файлы, как правило, используются в административных скриптах модулей и в <a href="#">компонентах</a>. Для работы с языковыми файлами предназначен модуль <b>Перевод</b>.</p>
<b>Шаблон страницы</b>	<p>Это файл, хранящийся в одном из каталогов:</p> <ul style="list-style-type: none"><li>• <code>/bitrix/templates/ID шаблона сайта/page_templates/</code></li><li>• <code>/bitrix/templates/.default/page_templates/</code></li></ul> <p>Данный файл представляет из себя заготовку публичной страницы. Как правило, используется при создании новой страницы в модуле <b>Управление структурой</b>. Для задания порядка сортировки шаблонов страниц используется файл <code>.content.php</code> находящийся в одном из вышеуказанных каталогов.</p>
<b>Пролог</b>	<p>В общем случае, под данным термином понимается верхняя левая часть страницы.</p> <p>Для <a href="#">публичной части</a>, пролог соответствующего <a href="#">шаблона сайта</a> хранится в файле <code>/bitrix/templates/ID шаблона сайта/header.php</code>.</p> <p>Для <a href="#">административной части</a> - пролог храниться в файле <code>/bitrix/modules/main/interface/prolog_admin.php</code>.</p> <p>В свою очередь пролог может быть разделен на служебную и визуальную части. В служебной части подключаются все необходимые классы, создаётся соединения с базой, создаются ряд служебных экземпляров объектов, таких как <code>\$USER</code>, <code>\$APPLICATION</code> и т.д. В визуальной части выводится верхняя левая часть страницы.</p> <p>Если в публичной части необходимо подключить неразделенный пролог, то используем следующий код:</p> <pre>require(\$_SERVER["DOCUMENT_ROOT"]."/bitrix/header.php");</pre> <p>Если по тем или иным причинам необходимо разделить пролог на</p>



служебную (**prolog\_before.php**) и визуальную (**prolog\_after.php**) части, то используем следующий код:

```
require($_SERVER["DOCUMENT_ROOT"].
"/bitrix/modules/main/include/prolog_before.php");
...
require($_SERVER["DOCUMENT_ROOT"].
"/bitrix/modules/main/include/prolog_after.php");
```

## Эпилог

В общем случае под данным термином понимается нижняя правая часть страницы.

Для [публичной части](#), эпилог соответствующего [шаблона сайта](#) хранится в файле **/bitrix/templates/ID шаблона сайта/footer.php**.

Для [административной части](#) - эпилог хранится в файле **/bitrix/modules/main/interface/epilog\_admin.php**.

В свою очередь эпилог может быть разделен на служебную и визуальную части. В служебной части производится ряд таких действий, как: отсылка почтовых сообщений, обработка обработчиков события `OnAfterEpilog` и др. В визуальной части выводится нижняя правая часть страницы.

Если в публичной части необходимо подключить неразделенный эпилог, то используем следующий код:

```
require($_SERVER["DOCUMENT_ROOT"]."/bitrix/footer.
php");
```

Если по тем или иным причинам необходимо разделить эпилог на визуальную (**epilog\_before.php**) и служебную (**epilog\_after.php**) части, то используем следующие коды:

```
require($_SERVER["DOCUMENT_ROOT"].
"/bitrix/modules/main/include/epilog_before.php")
;
...
require($_SERVER["DOCUMENT_ROOT"].
"/bitrix/modules/main/include/epilog_after.php");
```

## Тело страницы

Тело страницы это часть PHP/HTML кода, расположенного в скрипте между подключениями [пролога](#) и [эпилога](#). Тело страницы



не является частью [шаблона сайта](#) и представляет из себя индивидуальное содержимое [публичной](#) либо [административной](#) страницы.

<b>Навигационная цепочка</b>	<p>Это элемент дизайна, предназначенный в первую очередь для улучшения навигации по сайту. Навигационная цепочка выводится в визуальной части <a href="#">пролога</a> и состоит из заголовков разделов сайта с соответствующими ссылками на них. Помимо заголовков разделов, добавляемых автоматически, вы также можете добавлять произвольные пункты в навигационную цепочку.</p>
<b>Административная часть</b>	<p>Это множество скриптов, в которых подключен административный <a href="#">пролог</a> и <a href="#">эпилог</a>. Их подключение подразумевает:</p> <ul style="list-style-type: none"><li>данная страница не принадлежит ни одному <a href="#">сайту</a>;</li><li>в тоже время она принадлежит какому-либо модулю;</li><li>имеет строго определённый административный интерфейс;</li><li>все параметры <a href="#">локализации</a> данной страницы зависят от выбранного текущего языка;</li><li>на этой странице используется дополнительная проверка прав, задаваемых в настройках соответствующего модуля.</li></ul>
<b>Публичная часть</b>	<p>В общем случае, под данным термином понимается множество скриптов, у которых подключен <a href="#">пролог</a> и <a href="#">эпилог</a> одного из <a href="#">шаблонов сайта</a>. Их подключение подразумевает:</p> <ul style="list-style-type: none"><li>данная страница принадлежит какому либо <a href="#">сайту</a>;</li><li>имеет интерфейс текущего шаблона сайта;</li><li>все параметры <a href="#">локализации</a> данной страницы зависят от текущего сайта.</li></ul>
<b>Режим Правка</b>	<p>Режим, при котором текущая страница публичного раздела отображается в особом виде: выделены используемые компоненты, включаемые области, редактируемые области и т.д. Каждая такая область имеет набор кнопок для быстрого перехода к редактированию данного элемента страницы.</p>
<b>Индексная страница (файл)</b>	<p>Это имя файла, который будет использован веб-сервером в случае, если запрашиваемый URL заканчивается на слэш и не содержит в себе имени файла. Порядок, в котором будут искаться индексные страницы для различных веб-серверов, задается по разному, например:</p> <ul style="list-style-type: none"><li><b>Apache</b> - в файле <code>httpd.conf</code>, параметр <code>DirectoryIndex</code>;</li><li><b>IIS</b> - в свойствах сайта, закладка Documents &gt; Enable default content page;</li></ul> <p>К сожалению, в PHP значение данного параметра недоступно, поэтому для определения индексной страницы в коде необходимо</p>



	пользоваться функцией <code>GetDirIndex</code> .
	Синонимами данного термина можно считать «регистрация пользователя», «логин пользователя». Это запись в базе данных, содержащая параметры зарегистрированного пользователя с такими обязательными полями как:
<b>Бюджет (аккаунт) пользователя</b>	<ul style="list-style-type: none"><li>• логин;</li><li>• пароль;</li><li>• E-Mail.</li></ul>
	А также ряд дополнительных полей, содержащих личную информацию о пользователе, информацию по его работе, административные заметки.
<b>Мета-тэг</b>	Мета-тэг это один из элементов HTML, позволяющий задать информацию о странице. Указать, например, кодировку страницы, ее ключевые слова, автора, задать краткое описание страницы. Как правило, содержимое мета-тэгов используется в служебных целях, например, роботами поисковых систем, индексирующих сайт. Мета-тэг задается внутри тэга <b>&lt;head&gt;</b> .
<b>Локализация</b>	Подразумевает представление информации в переводе на соответствующий язык, в соответствующей кодировке данного языка и с использованием соответствующих форматов представления данных, характерных для этого языка (например, дата, время, денежных единицы, числа и т.д.).
<b>Хост</b>	В применении к функциям главного модуля, под данным термином понимается <a href="#">доменное имя</a> или <a href="#">IP адрес</a> , посредством которого можно обратиться к тому или иному <a href="#">сайту</a> .
<b>Доменное имя (домен)</b>	Строго говоря, это одно из полей DNS таблицы ( <b>domain name service</b> ), содержащее в себе строго структурированное имя интернет сайта, заданное по определенным правилам. Основная задача DNS таблицы - это ассоциация доменных имен с <a href="#">IP адресами</a> сайтов. Пример доменного имени: 1c-bitrix.ru
<b>IP адрес</b>	Это "имя" компьютера в сети, заданное по правилам протоколов TCP/IP. IP адрес состоит из четырех октетов, часть из которых идентифицирует подсеть, в которой находится компьютер, а часть - непосредственно этот компьютер в рамках соответствующей подсети. Пример IP адреса: 198.63.210.79
<b>Accept-Language</b>	Набор языков, установленных в браузере посетителя сайта. К примеру, для MS Internet Explorer их можно выставить в меню <i>Tools &gt; Internet Options &gt; General &gt; Languages</i> . Для Mozilla Firefox:



## Инструменты &gt; Настройки &gt; Основное &gt; Языки.

<b>Время в Unix-формате</b>	Количество секунд, прошедшее с 1 января 1970 года, с точностью до микросекунды. На сегодняшний день, время в Unix-формате может фиксироваться только до 2038 года.
<b>Права в Unix системах</b>	<p>В Unix-подобных операционных системах (являющихся доминирующими на сегодняшний день для интернет серверов) поддерживаются три вида прав - чтение, запись и выполнение, которые присваиваются каждому файлу или директории. Права эти повторяются три раза: для владельца файла, для группы пользователей, к которой владелец принадлежит, и для всех остальных пользователей. Как правило, права указываются в числовом формате:</p> <ul style="list-style-type: none"> <li>• 4 - чтение;</li> <li>• 2 - запись;</li> <li>• 1 - выполнение.</li> </ul> <p>Сумма этих чисел дает окончательный набор прав, например, 6 - это чтение и запись, но без выполнения, 7 - все права, 5 - чтение и выполнение.</p> <p>Таким образом, например, право 764 будет означать: 7 - все права для владельца файла, 6 - чтение и запись для группы пользователей, к которой принадлежит владелец файла и 4 - чтение для всех остальных пользователей.</p> <p>В PHP все права задаются в виде восьмиричных чисел, поэтому их надо задавать с обязательным указанием префикса - 0. Пример: 0755.</p>
<b>Путь относительно корня</b>	Путь к файлу, начинающийся от каталога, указанного в параметре <a href="#">DocumentRoot</a> в настройках веб-сервера, заданный по правилам формирования URL-адресов. Пример: <code>/ru/about/index.php</code>
<b>Полный путь</b>	Включает в себя протокол, домен, порт и <a href="#">путь относительно корня</a> к странице (каталогу). Пример: <code>http://www.bitrixsoft.ru/ru/about/index.php</code>
<b>Абсолютный путь</b>	Абсолютный путь к файлу включает в себя <a href="#">DocumentRoot</a> и <a href="#">путь относительно корня</a> .
<b>DocumentRoot</b>	<p>Путь к корню сайта в файловой системе сервера. Задается в настройках веб-сервера, например:</p> <ul style="list-style-type: none"> <li>• для <b>Apache</b> - в файле <code>httpd.conf</code>, параметр <b>DocumentRoot</b>;</li> <li>• для <b>IIS</b> - в свойствах сайта, закладка Home Directory &gt; Local</li> </ul>



Path.

<b>Сабмит</b>	Отправка данных HTML формы на сервер.										
<b>Сессия</b>	Под термином понимается сессия PHP. Сессия может открываться в момент захода на сайт и закрывается при закрытии окна браузера. Также новая сессия открывается при авторизации пользователя, если закончить сеанс авторизации (разлогиниться) - сессия закрывается. Синонимом термина сессия можно считать один "заход на сайт".										
<b>HTML-безопасный вид</b>	<p>Как правило, данный термин применяют к тексту, в котором произведены следующие замены:</p> <table border="1"><thead><tr><th>Исходное</th><th>Результат</th></tr></thead><tbody><tr><td>&lt;</td><td>&amp;lt;</td></tr><tr><td>&gt;</td><td>&amp;gt;</td></tr><tr><td>"</td><td>&amp;quot;</td></tr><tr><td>&amp;</td><td>&amp;amp;</td></tr></tbody></table> <p>Подобные замены позволяют выводить текст внутри HTML кода, не опасаясь, что он будет интерпретирован браузером как часть этого HTML кода.</p>	Исходное	Результат	<	&lt;	>	&gt;	"	&quot;	&	&amp;
Исходное	Результат										
<	&lt;										
>	&gt;										
"	&quot;										
&	&amp;										
<b>Связывание переменных</b>	В применении к SQL-запросам для Oracle версии, под данным термином подразумевается связывание имен переменных (либо полей таблицы) с их значениями. Как правило, подобная технология используется для полей типа <b>BLOB</b> , <b>CLOB</b> , <b>LONG</b> и т.п. предназначенных для хранения больших объемов данных.										
<b>Дамп</b>	В применении к базе данных, под термином подразумевается выгрузка содержимого и, возможно, структуры таблиц в файл в определённом формате, для их возможной дальнейшей загрузки обратно в данную, либо любую другую базу. Для каждой базы данных существуют свои утилиты, позволяющие сделать дамп, например для MySQL утилита <b>mysqldump</b> позволяет выгрузить в формате обычных SQL запросов, для Oracle утилита <b>exp</b> позволяет выгрузить в своем внутреннем формате. В применении к переменным, дамп подразумевает отображение структуры и содержимого переменной в текстовом виде.										
<b>cron (крон)</b>	В Unix-подобных операционных системах утилита <b>cron</b> позволяет										



организовать запуск скриптов по четко указанному расписанию.

<b>Буферизация</b>	<p>Под этим термином в PHP понимается такой режим, при котором весь исходящий поток данных из PHP скрипта (например, HTML код) запоминается предварительно в памяти и не отдается браузеру пользователя. Буферизацию в PHP можно включить с помощью функции <code>ob_start</code>. В дальнейшем буферизацию можно отключить, например с использованием функции <code>ob_end_flush</code>, при этом все накопленные данные будут отосланы браузеру. Буферизация позволяет произвольно манипулировать исходящим потоком данных, на этом принципе основана <a href="#">технология отложенных функций</a>.</p>
<b>Постоянное соединение (persistent)</b>	<p>При создании соединения с базой, в памяти создаётся дескриптор данного соединения. Если соединение обычное, то после отработки скрипта этот дескриптор удаляется, если соединение <b>постоянное</b>, он остается и может быть использован другими процессами при необходимости. Достоинством постоянного соединения является то, что, как правило, времени на него работу требуется меньше. Но есть и недостаток - количество открытых постоянных соединений ограничивается в настройках базы данных и при превышении этого лимита посетитель не сможет зайти на сайт пока не освободятся новые соединения.</p>
<b>Почтовое событие</b>	<p>Это почтовое сообщение, имеющее свой <a href="#">тип</a> и отправляемое по соответствующему <a href="#">почтовому шаблону</a>. Почтовое событие инициализирует поля <a href="#">типа почтового события</a> конкретными значениями. Порядок расположения этих полей в письме, а также текст письма, определяется <a href="#">почтовым шаблоном</a>. Для создания почтового события предназначен класс <code>CEvent</code>. Используется в <b>Почтовой системе</b>.</p>
<b>Почтовый шаблон</b>	<p>Определяет текст почтового сообщения, а также порядок расположения полей (placeholder'ов), заданных в <a href="#">типе почтового события</a>.</p> <p>Почтовые шаблоны доступны в административном разделе на странице <b>Почтовые шаблоны</b> (Настройки системы &gt; Почтовые шаблоны).</p> <p>Для манипуляции почтовыми шаблонами предназначен класс <code>CEventMessage</code>. Используется в <b>Почтовой системе</b>.</p>
<b>Тип почтового события</b>	<p>Определяет набор специальных полей (placeholder'ов), которые могут быть использованы в <a href="#">почтовом шаблоне</a>. В момент создания <a href="#">почтового события</a> эти поля будут инициализированы конкретными значениями.</p> <p>Для манипуляции типами почтовых событий предназначен класс <code>CEventType</code>. Используется в <b>Почтовой системе</b>.</p>



**Кастомизация**

Изменение логики работы компонента или шаблона компонента под частные задачи.