

# Портал

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>Общие принципы работы портала</b>	<b>4</b>
Работа с порталом	4
Схема архитектуры портала	6
Масштабируемость портала	6
Совместимость портала с продуктами Creatio	7
Варианты развертывания портала	8
<b>Портал самообслуживания</b>	<b>8</b>
Устройство портала	8
Работа с мастером страниц на портале	9
Настройка портала и порталых пользователей	9
Ограничение доступа к веб-сервисам для пользователей портала	9
<b>Пример использования миксина PortalMessagePublisherExtensions</b>	<b>12</b>
Реализация примера	12
<b>Изменить доступ к сервису для пользователей портала</b>	<b>14</b>
Реализация примера	15
<b>Миксин PortalMessagePublisherExtensions</b>	<b>16</b>
Методы	16

# Общие принципы работы портала

## Основы

**Портал Creatio** — это компонент low-code платформы, который представляет собой рабочее место пользователя и предоставляет ограниченный доступ к данным и функциональности основного приложения Creatio для внешних и внутренних пользователей, клиентов и партнеров компании. Интерфейс и [инструменты настройки портала](#) такие же, как в основном приложении Creatio, а разработка портала выполняется аналогично разработке основного приложения.

## Работа с порталом

Портал Creatio предоставляет возможности для решения широкого спектра бизнес-задач. Портал Creatio подходит для решения большого спектра задач, самыми популярными из которых являются:

- **Самообслуживание клиентов, например, в области технической поддержки.** Добавление канала самообслуживания для клиентов и переключение внимания сотрудников службы поддержки на более важные задачи, чем регистрация обращений. Возможность самостоятельного создания запросов клиентами и отслеживания прогресса по их разрешению непосредственно на портале. Предоставление доступа к статьям базы знаний помогает клиентам быстрее находить ответы на свои вопросы. Возможность одновременного обслуживания большого количества клиентов без очередей и потерь продуктивности.
- **Взаимодействие с внутренними и внешними клиентами, например, HR-портал.** Настройка возможности обслуживания внешних сотрудников и подрядчиков, которые не используют приложение Creatio для повседневной работы: создание заявки, передача их на рассмотрение и отслеживание изменения их состояний. Портал может стать источником всех важных документов и регламентов компании, которые находятся в свободном доступе.
- **Взаимодействие с внешними пользователями (клиентами, дилерами, партнерами) на всех этапах процесса продаж компании.** Создание партнерских программ, а затем, совместно с партнерами, обработка лидов и продаж, используя процессы управления лидами и корпоративными продажами. Фиксация проведения обучающих мероприятий и сертифицированных специалистов партнера.

На портале Creatio пользователи могут получить доступ к разделам, доступным в рамках [лицензий](#) пользователя приложения и связанным с ними данным. Управлять доступом внешних пользователей к бизнес-данным на портале и контролировать сохранность и конфиденциальность важной информации возможно с помощью стандартного механизма [настройки прав доступа](#).

Пользователи портала могут:

- Создавать и изменять записи.
- Добавлять примечания.
- Прикреплять файлы.
- Оставлять сообщения для поддержки.
- Работать по процессам.

Управление доступом пользователей портала к данным системы осуществляется не только через права доступа, но и через справочник [ *Список объектов, доступных пользователям портала* ] ([ *List of objects available for portal users* ]). Только те объекты, которые перечислены в справочнике, будут доступны из интерфейса портала.

Наиболее частым сценарием является совместная работа пользователей портала и пользователей системы в одном разделе, но с разным набором полей на страницах. Для порталовых пользователей введено ограничение на колонки, которые можно настроить в реестре раздела и колонки для фильтрации (в быстрых фильтрах или настройке динамических групп). При добавлении колонок на страницу записи с помощью мастера эти колонки по умолчанию добавляются в справочник [ *Список доступных полей объектов на портале* ] ([ *List of schema fields for portal access* ]) и становятся доступными. Если поле не добавлено на страницу, но есть необходимость вывести его в реестр портала, то необходимо вручную добавить его в справочник. Внешний вид реестра и набор отображаемых колонок можно [настроить](#) индивидуально.

Все [пользователи, которые имеют доступ на портал](#), входят в группу организационных ролей "

Все пользователи портала" ("All portal users"). В системе можно создавать индивидуальных пользователей портала или группировать их в организацию, привязав к определенному контрагенту.

Ограничения доступа к веб-сервисам для пользователя портала описаны в [статье](#).

Портал Creatio доступен в следующих конфигурациях:

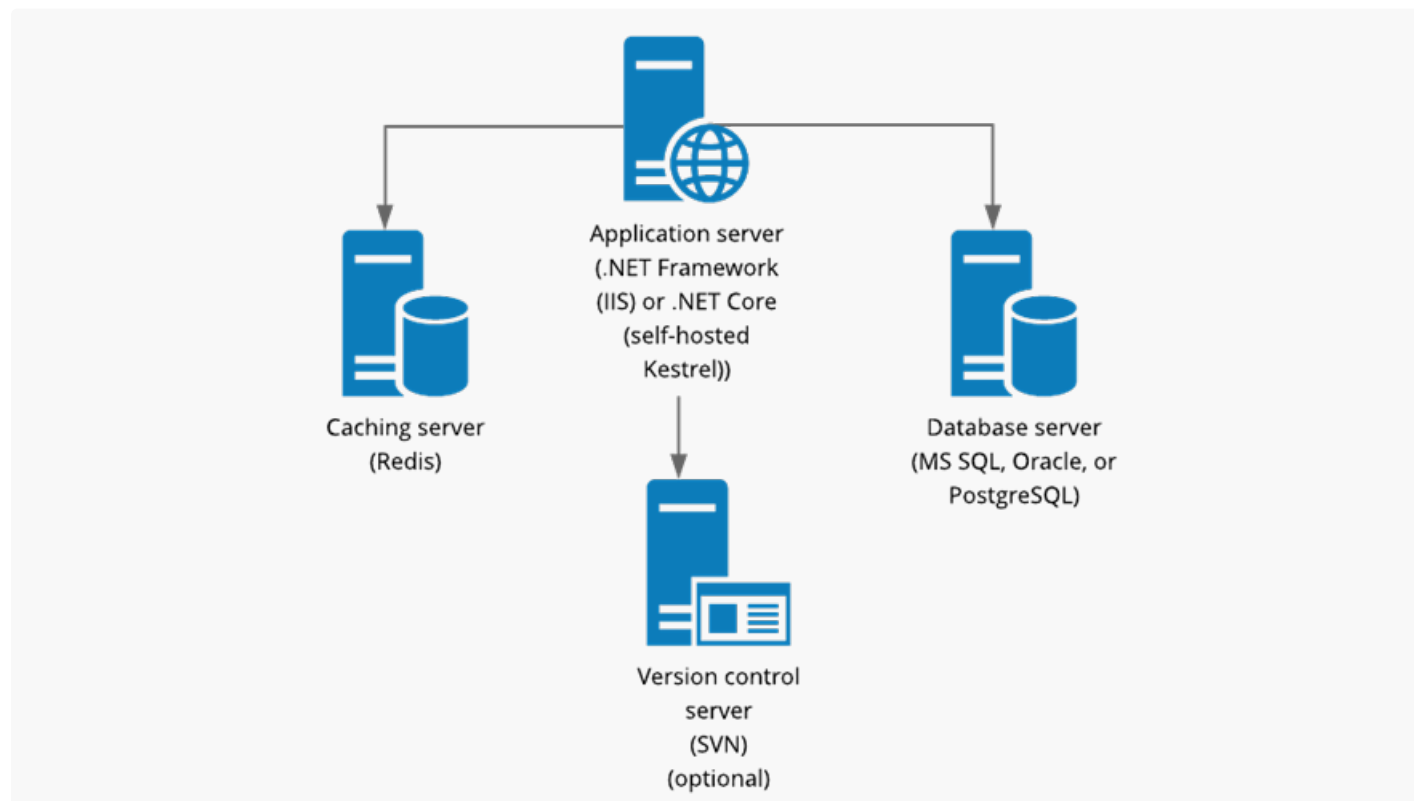
- **[Портал самообслуживания](#)**. Портал самообслуживания может использоваться как основной канал регистрации обращений, так и в качестве дополнения к уже существующим каналам. Пользователь портала может найти информацию в базе знаний или создать запрос в службу поддержки. Данная конфигурация позволяет работать только с разделами [ *Обращения* ] ([ *Case* ]) и [ *База знаний* ] ([ *Knowledge base* ]), которые можно кастомизировать с помощью дизайнера системы. Добавление дополнительных разделов на портал самообслуживания не поддерживается.
- **[Клиентский портал](#)**. Клиентский портал предназначен для автоматизации процессов, например, предоставления услуг, обработки заявок и запросов на обслуживание. Пользователь портала может инициировать процессы (например, создание заказов, запросов) или участвовать в них (например, утверждать заявки). Такой тип конфигурации портала позволяет настраивать и использовать до трех пользовательских разделов на портале. Пользовательский раздел сначала нужно создать в приложении и затем на его основе создать порталную версию этого раздела. Разделы [ *Обращения* ] ([ *Case* ]) и [ *База знаний* ] ([ *Knowledge base* ]), также доступны пользователям клиентского портала, а для пользователей продукта Creatio Bank Customer Journey будет доступен раздел [ *Заявки* ] ([ *Applications* ]). Дополнительно можно вывести на портал раздел [ *Документы* ] ([ *Documents* ]), если он доступен в основном приложении. Создание раздела на основе лицензируемого объекта из состава базового продукта не считается пользовательским разделом.
- **[Партнерский портал](#)**. Партнерский портал предназначен для компаний, работающих с клиентами через сеть партнеров. Партнерский портал является единой платформой для коммуникаций и совместной с партнерами работы над лидами и продажами. В данной конфигурации пользователям портала доступны разделы [ *Партнерская программа* ] ([ *Partner program* ]), а также [ *Лиды* ] ([ *Leads* ]), [ *Продажи* ] ([ *Opportunities* ]), [ *Маркетинговые активности* ] ([ *Marketing activities* ]) и [ *Заказы* ] ([ *Orders* ]).

Конфигурации портала могут использоваться одновременно. Пользователям каждой из конфигураций доступна базовая функциональность портала (главная страница, настройки в профиле пользователя, база знаний, сегментация данных), а перечень разделов, доступных на портале, различается в зависимости от конфигурации портала.

Описание работы с порталом содержится в [статье](#).

## Схема архитектуры портала

Схема архитектуры портала Creatio не отличается от схемы архитектуры [основного приложения](#).

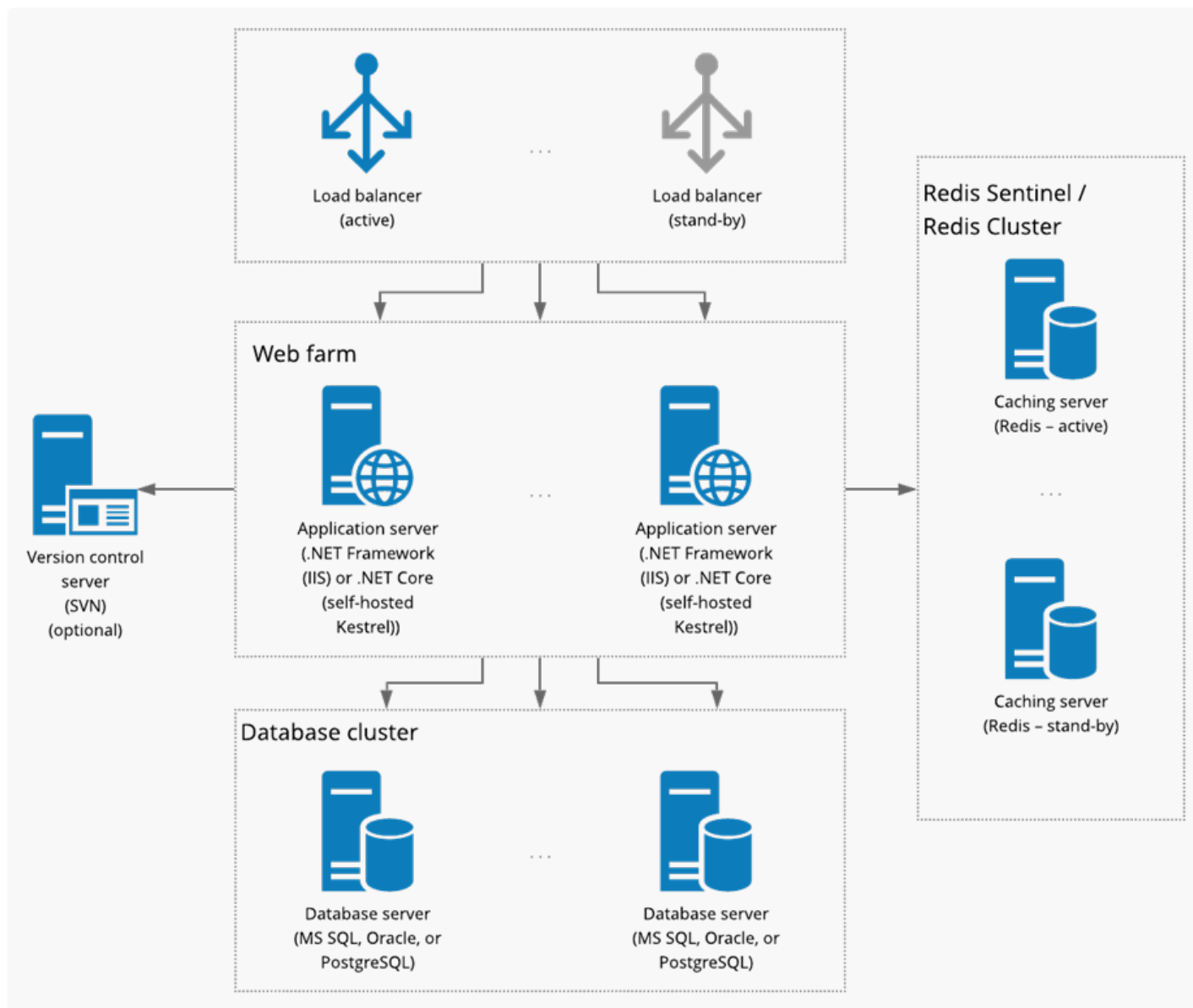


## Масштабируемость портала

В Creatio существует возможность повысить производительность крупных проектов с помощью [горизонтального масштабирования](#). При использовании горизонтального масштабирования схема архитектуры портала включает в себя следующие компоненты:

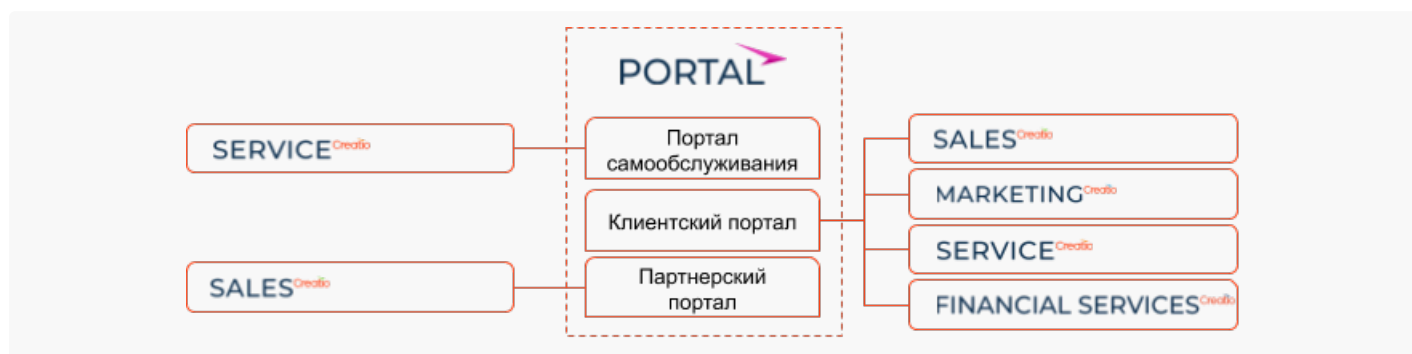
- [Балансировщики нагрузки](#). Балансировщик нагрузки может быть аппаратным или программным. Для работы в отказоустойчивом режиме используется балансировщик HTTP/HTTPS-трафика с поддержкой протокола WebSocket. Описание установки и настройки балансировщика HAProxy содержится в [статье](#).
- [Веб-ферма](#) (несколько серверов приложений).
- [Сервер кэширования Redis или Redis Sentinel / Кластер Redis \(несколько серверов кэширования Redis\)](#).
- Сервер баз данных или кластер баз данных (несколько серверов баз данных).
- Сервер системы контроля версий (опционально). Описание настройки сервера системы контроля версий содержится в [статье](#).

Схема архитектуры портала Creatio при использовании веб-фермы, кластера Redis и кластера баз данных



Настройка безопасного доступа к portalу, а также типовая схема установки системы с доступом к portalу из внешней сети описана в [статье](#).

## Совместимость portalа с продуктами Creatio



## Варианты развертывания портала

Портал является частью платформы Creatio и доступен в базовой конфигурации как в приложениях, развернутых как cloud, так и on-site.

Для обеспечения безопасности данных при установке портала on-site приложение должно быть развернуто с использованием веб-фермы. Настройка веб-фермы описана в [статье](#). Установка приложения с доступом к portalу из внешней сети описана в [статье](#).

## Портал самообслуживания



Портал самообслуживания (Self Service Portal, SSP) является неотъемлемой функциональной частью продуктов [Service Creatio, enterprise edition](#), [Service Creatio, customer center edition](#), а также всех производных пакетов (bundle) продуктов, в которые они входят.

Портал представляет собой отдельное рабочее место (Workplace), в которое автоматически попадает пользователь портала при входе через стандартное окно авторизации приложения.

По умолчанию в поставляемом продукте порталному пользователю доступно два раздела — [ *Обращения портала* ] и [ *База знаний* ], а также [ *Главная страница портала* ], которая содержит общую сводную информацию и выполняет роль единого рабочего места пользователя портала.

Раздел [ *Обращения портала* ] служит для самостоятельной регистрации обращений пользователем, просмотра статуса своих обращений, внесения дополнительной информации по обращению, а также для получения информации по решению своего обращения. По умолчанию порталный пользователь имеет доступ только к тем обращениям, в которых он указан в качестве контакта. На странице обращения пользователь может внести дополнительную информацию, публиковать сообщения, вести переписку по своему обращению с сотрудниками службы поддержки. История переписки по обращению отображается на странице обращения на вкладке [ *Обработка* ].

Раздел [ *База знаний* ] служит для самостоятельного поиска пользователем справочной информации или решения по своему вопросу. Данный раздел по умолчанию может наполняться только сотрудниками поддержки из основного интерфейса системы.

## Устройство портала

С точки зрения разработки, портал является преднастроенным отдельным рабочим местом. Это рабочее место по умолчанию недоступно обычным пользователям приложения. Пользователь системы с типом "Пользователь портала" при авторизации автоматически попадает в данное рабочее место, а именно на Главную страницу портала.

Разделы и страницы портала — это обыкновенные [разделы](#) и [страницы записи](#), и работают с теми же сущностями ( `Entity` ), что и страницы, доступные в основном интерфейсе системы. В частности, страница обращения на портале имеет более простой вид, и не содержит большинства полей. В конфигурации это две разные страницы записи — `CasePage` и `PortalCasePage`.

Система прав доступа на портале имеет только одно небольшое отличие. Чтобы дать порталным пользователям доступ к определенным сущностям ( `EntitySchema` ), необходимо дополнительно указать эти сущности в справочнике [ *Список объектов доступных порталному пользователю* ]. Лицензии портала самообслуживания ограничивают количество записей, которое может быть добавлено в данный



справочник. По умолчанию это 70 записей.

## Работа с мастером страниц на портале

Портальному пользователю недоступна функциональность мастеров страниц, деталей и разделов. Эта функциональность может быть вызвана пользователем с правами администратора из основного интерфейса системы следующим образом:

1. Перейти в раздел [ *Настройка рабочих мест* ] дизайнера системы.
2. Выбрать рабочее место [ *Портал* ] и нажать [ *Открыть* ].
3. Выбрать требуемый раздел и нажать кнопку [ *Настройка раздела* ].

После этого откроется стандартный мастер разделов.

## Настройка портала и порталных пользователей

Для того чтобы начать пользоваться порталом, необходимо выполнить следующие действия:

1. Убедиться, что в файле `web.config` загрузчика приложения (т. н. "внешний" `web.config`) опция `/configuration/terrasoft/auth` имеет следующий вид.

**Опция** `/configuration/terrasoft/auth` **файла** `web.config`

```
<terrasoft>
  <auth providerNames="InternalUserPassword,SSPUserPassword" ...>
  ...
</terrasoft>
```

Данная настройка отвечает за возможность авторизации в системе порталных пользователей.

2. Создать контакт для пользователя.
3. Создать пользователя с типом "Пользователь портала". Заполнить необходимые поля.
4. Раздать необходимые лицензии пользователю.

У порталного пользователя в профиле должен быть указан корректный часовой пояс. По умолчанию у нового пользователя он не указан. Настройка выполняется самим пользователем портала в его профиле. Отталкиваясь от этого значения, в системе будут отображаться все даты и время.

## Ограничение доступа к веб-сервисам для пользователей портала

Основной сценарий использования портала — предоставить доступ большому количеству внешних по отношению к организации пользователей к ограниченному количеству данных из системы. Для решения таких задач возникает потребность управлять тем, какие пользователи (портальные или пользователи компании) имеют доступ к [веб-сервисам приложения](#).

**Важно.** Данная статья актуальна для версии системы 7.13.2 и новее.

## Префиксы маршрутов конфигурационных веб-сервисов

Для управления доступом к веб-сервисам приложения в Creatio реализован механизм префиксов маршрута. Используя определенные `ServiceRoute` атрибуты класса сервиса, можно задать необходимый префикс маршрута.

Префиксы маршрутов конфигурационных веб-сервисов

Доступ	Атрибут	Префикс маршрута	Пример кода
Только для пользователей Портала самообслуживания	<code>SspServiceRoute</code>	<code>/ssp</code>	<div><pre>[ServiceContract] [SspServiceRoute] public class SspOnlyService { }</pre></div> <div>Пример вызова</div> <div><pre>~/ssp/rest/SspOnlyService ~/ssp/soap/SspOnlyService</pre></div>
Только для внутренних пользователей	<code>DefaultServiceRoute</code> или не указывать ни один <code>ServiceRoute</code> атрибут	отсутствует	<div><pre>[ServiceContract] public class InternalService { }</pre></div> <div>или</div> <div><pre>[ServiceContract] [DefaultServiceRoute] public class InternalService { }</pre></div> <div>Пример вызова</div>

~/rest/InternalService  
~/soap/InternalService

И для внутренних  
пользователей, и  
для Портала

одновременно  
атрибуты

DefaultServiceRoute

и

SspServiceRoute

/ssp

или

отсутствует

```
[ServiceContract]
[DefaultServiceRoute]
[SspServiceRoute]
public class CommonService
{
}
```

**Пример вызова**

~/rest/CommonService  
~/soap/CommonService  
~/ssp/rest/CommonService  
~/ssp/soap/CommonService

атрибут ServiceRoute  
с указанием  
префикса (например,  
"custom ")

ServiceRoute("custom")

произвольный  
префикс  
маршрута  
сервиса

```
[ServiceContract]
[ServiceRoute("custom")]
public class CustomPrefixS
{
}
```

**Пример вызова**

~/custom/rest/CustomPrefix  
~/custom/soap/CustomPrefix

**На заметку.** Допустимо использовать одновременно несколько атрибутов `ServiceRoute` , `SspServiceRoute` и `DefaultServiceRoute` . В результате будут созданы маршруты для сервиса со всеми вариантами префиксов.

Ограничение доступа порталых пользователей к внутреннему API

Если пользователь портала ( `SspUserConnection` ) обратится к сервису по маршруту без префикса “ `/ssp` ”, сервис вернет в ответ страницу с кодом 403 ( `Forbidden` ).

## Ограничение доступа внутренних пользователей к порталному API

Если внутренний пользователь приложения ( `UserConnection` ) обратится к сервису по маршруту с префиксом “ `/ssp` ”, сервис вернет в ответ страницу с кодом ● 403 Forbidden.

## ServiceStack сервисы ядра

Методы сервисов `ServiceStack` ядра ( `DataService`, `ManagerService` и другие) по умолчанию доступны только для внутренних пользователей.

Для того, чтобы какой-то из методов был доступен также и на Портале, необходимо пометить такой метод атрибутом `SspServiceAccess` - в этом случае метод будет иметь дополнительный маршрут с префиксом вида `~/DataService/ssp/...`

Если нужно, чтобы логика метода отличалась для Портала, необходимо создать новый сервис, указав для него атрибут `SspServiceAccess`, в конструктор которого передать имя оригинального метода. Например:

```
[SspServiceAccess(nameof(SelectQuery))]
public class SspSelectQuery : SelectQuery
{
}
```

создает контракт, метод которого будет зарегистрирован по пути `~/DataService/ssp/SelectQuery`.

Доступ к методам `ServiceStack` с префиксом “ `ssp` ” запрещен внутренним пользователям, а к методам `ServiceStack` без префикса “ `ssp` ” - пользователям портала.

# Пример использования миксина PortalMessagePublisherExtensions

 Сложный

**Пример.** Использовать МИКСИН `PortalMessagePublisherExtensions`.

## Реализация примера

Реализовать пример использования миксина `PortalMessagePublisherExtensions`.

`PortalMessagePublisherExtensions`

```

define("CaseSectionActionsDashboard", ["PortalMessagePublisherExtensions"], function() {
    return {
        mixins: {
            /**
             * @class PortalMessagePublisherExtensions extends tabs and publishers configs.
             */
            PortalMessagePublisherExtensions: "Terrasoft.PortalMessagePublisherExtensions"
        },
        methods: {
            /**
             * @inheritdoc Terrasoft.SectionActionsDashboard#getExtendedConfig
             * @inheritdoc
             */
            getExtendedConfig: function() {
                // Получение объекта конфигурации вкладок из родительского метода.
                var config = this.callParent(arguments);
                // Вызов метода миксина, добавление конфигурации портальной вкладки.
                this.mixins.PortalMessagePublisherExtensions.extendTabsConfig.call(this, config)
                // Возвращение расширенного объекта конфигурации.
                return config;
            },

            /**
             * @inheritdoc Terrasoft.SectionActionsDashboard#getSectionPublishers
             * @inheritdoc
             */
            getSectionPublishers: function() {
                // Получение коллекции издателей сообщений из родительского метода.
                var publishers = this.callParent(arguments);
                // Вызов метода миксина, добавление портального канала.
                this.mixins.PortalMessagePublisherExtensions.extendSectionPublishers.call(this,
                // Возвращение расширенной коллекции издателей сообщений.
                return publishers;
            }
        },
        diff: /**SCHEMA_DIFF*/[
            {
                "operation": "insert",
                "name": "PortalMessageTab",
                "parentName": "Tabs",
                "propertyName": "tabs",
                "values": {
                    "items": []
                }
            },
            {
                "operation": "insert",
                "name": "PortalMessageTabContainer",

```

```

        "parentName": "PortalMessageTab",
        "propertyName": "items",
        "values": {
            "itemType": this.Terrasoft.ViewItemType.CONTAINER,
            "classes": {
                "wrapClassName": ["portal-message-content"]
            },
            "items": []
        }
    },
    {
        "operation": "insert",
        "name": "PortalMessageModule",
        "parentName": "PortalMessageTab",
        "propertyName": "items",
        "values": {
            "classes": {
                "wrapClassName": ["portal-message-module", "message-module"]
            },
            "itemType": this.Terrasoft.ViewItemType.MODULE,
            "moduleName": "PortalMessagePublisherModule",
            "afterrender": {
                "bindTo": "onMessageModuleRendered"
            },
            "afterrerender": {
                "bindTo": "onMessageModuleRendered"
            }
        }
    }
}
]/**SCHEMA_DIFF*/
});
});

```

## Изменить доступ к сервису для пользователей портала



Сложный

В приложении существует набор базовых сервисов, доступ к которым есть только у внутренних пользователей.

Для изменения доступа к базовому сервису необходимо:

1. В пользовательском пакете создать сервис с настройкой доступа для порталных пользователей.
2. Добавить для него те методы базового сервиса, которые должны быть доступны для порталных пользователей.

3. Изменить пользовательские или расширить базовые клиентские схемы, изменив вызов базового сервиса на вызов созданного сервиса (см. шаг 1).
4. Скомпилировать конфигурацию.

**Пример.** Изменить доступ к сервису для пользователей портала.

## Реализация примера

Пример исходного кода сервиса, который расширяет доступ к базовому сервису `ActivityUtilService` представлен ниже.

### PartnerActivityUtilService

```
namespace Terrasoft.Configuration
{
    using System;
    using System.IO;
    using System.Runtime.Serialization;
    using System.ServiceModel;
    using System.ServiceModel.Activation;
    using System.ServiceModel.Web;
    using Terrasoft.Configuration.FileUpload;
    using Terrasoft.Core.Factories;
    using Terrasoft.Web.Common;
    using Terrasoft.Web.Common.ServiceRouting;

    [ServiceContract]
    // Сервис доступен и для внутренних пользователей и для Портала.
    [DefaultServiceRoute]
    [SspServiceRoute]
    [AspNetCompatibilityRequirements(RequirementsMode = AspNetCompatibilityRequirementsMode.RequirementsModes.All)]
    public class PartnerActivityUtilService: BaseService {
        // Базовый сервис, доступ к которому необходимо расширить.
        private static readonly ActivityUtilService _baseService = new ActivityUtilService();

        [OperationContract]
        [WebInvoke(Method = "POST", BodyStyle = WebMessageBodyStyle.Wrapped, RequestFormat = WebMessageFormat.Json)]
        public Guid CreateActivityFileEntity(JsonActivityFile jsonActivityFile) {
            return _baseService.CreateActivityFileEntity(jsonActivityFile);
        }
        [OperationContract]
        [WebInvoke(Method = "POST", BodyStyle = WebMessageBodyStyle.Wrapped, RequestFormat = WebMessageFormat.Json)]
        public Guid CreateFileEntity(JsonEntityFile jsonEntityFile) {
            return _baseService.CreateFileEntity(jsonEntityFile);
        }
    }
}
```

}

# Миксин PortalMessagePublisherExtensions

JS

 Сложный

**Миксин** — это класс-примесь, предназначенный для расширения функциональности других классов. Это отдельно созданные классы с дополнительной функциональностью.

Миксин `PortalMessagePublisherExtensions` предназначен для расширения схемы `SectionActionsDashboard` и ее наследников. Он позволяет расширить конфигурацию вкладок `SectionActionsDashboard` вкладкой порталных сообщений `PortalMessageTab` и добавить соответствующий канал сообщений `Portal`. Миксин реализован в пакете `PortalMessagePublisher` и доступен в продукте `ServiceEnterprise`, а также в бандлах, в которые входит этот продукт.

## Методы

`extendTabsConfig(config) : Object`

Расширяет конфигурацию вкладок `SectionActionsDashboard` вкладкой порталных сообщений `PortalMessageTab`.

Возвращает дополненный объект (`Object`) конфигурации вкладок `SectionActionsDashboard`.

### Параметры

`config (Object)`

Объект конфигурации вкладок `SectionActionsDashboard`.

`extendSectionPublishers(publishers) : Array`

Добавляет порталный канал (`Portal`) в коллекцию издателей сообщений.

Возвращает дополненную коллекцию издателей сообщений (`Array`).

### Параметры

`publishers (Array)`

Коллекция издателей сообщений.