Kentico CMS 7.0 On-line Marketing Guide

Kentico
unlimited website possibilities

www.kentico.com

Table of Contents

Part I

Introduction

1 Introduction

1.1 About this guide

This guide contains information about on-line marketing features of Kentico Enterprise Marketing Solution, the full-featured on-line marketing tool based on Kentico CMS. It will guide you through the basics of all the on-line marketing features and explain how your website can benefit from using them. It will also navigate you through the related sections of the user interface and provide step-by-step examples that can be tried out in order to get hands-on experience and see how they behave in real-world scenarios.

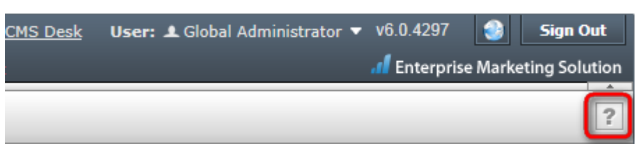
To be able to understand the content of this guide, it is highly recommended to be already familiar with the basics of Kentico CMS. If you are completely new to Kentico CMS and have no previous experience with it, we recommend you to go through Kentico CMS Tutorial before proceeding further in this guide.

In the On-line marketing features topic, individual features are listed with links leading to their dedicated chapters. Before going through the related step-by-step examples and trying out their functionality on your Kentico CMS instance, it is recommended to install the sample Corporate Site and adjust on-line marketing settings to make the features are enabled. It is also recommended to see the First sight topic in order to get familiar with the sections of the user interface where the on-line marketing features are located.

Built-in Context Help

Basic documentation of particular sections of the system's user interface can be found in Kentico CMS Context Help. It is useful in case that you need a quick explanation of the currently used functionality.

If you click the ? icon in the top right corner of each UI screen, you get redirected to the appropriate Context Help page which describes the current UI screen. The Context Help is embedded in the web project, so you don't even need to have Internet connection to view it.



1.2 On-line marketing features

Kentico Enterprise Marketing Solution provides the following features related to on-line marketing:

- Web analytics - measure and analyze key metrics of your website such as visits, page views, file downloads, traffic sources, etc.
  - Conversions - track actions performed by your website's visitors (e.g. user registration, product order, etc.) and record them as conversions.
  - Campaigns - accurately monitor traffic generated by individual on-line marketing campaigns (e.g. banners, marketing e-mails, etc.).
  - Reporting - view analytics data presented in highly flexible reports that contain various kinds of charts and tables.

- Contact management - centralized overview of all users who visit your website, used to gather information about them and their activities on the website.
  - Contacts - individuals who visit the website and about whom information is gathered.
  - Accounts - companies or other commercial or non-commercial bodies where contacts operate.
  - Contact groups - organizational units that can be used to group contacts or accounts based on various criteria.
  - Merging - merge contacts or accounts into one in order to avoid duplicities.
  - Activities - analyze behavior and actions that contacts perform on the website.
  - Salesforce - replicate Kentico contacts into Salesforce as leads.
  - Data.com - update your contacts' and accounts' information using the Data.com business directory.

- Marketing automation - visually design and automate your marketing campaigns and tasks.

- Scoring - numerically evaluate individual contacts on multiple scales based on their properties and activities.

- Website optimization - create different versions of pages and evaluate them according to the behavior of the website visitors.
  - A/B testing - create one or more modified versions of a given page, diversify traffic to them and track how individual versions affect behavior of website visitors.
  - Multivariate testing - create different versions of elements on a page and track which versions produce the best results when displayed to website visitors.

- Content personalization - create pages that display different content depending on the circumstances in which they are viewed.

- E-mail marketing - attract new customers to your website or keep in touch with existing clients by sending marketing e-mails.

- Security - allow access to user interfaces of on-line marketing features only to certain users.
- Internals and API - modify or enhance the out-of-the-box functionality by your custom code using Kentico CMS API.

• Analytics engine
• Conversions
• Campaings
• Reporting
Web analytics
• Contacts
• Activities
• Accounts
• Automatic segmentation
• Salesforce
• Data.com
• Marketing automation
Contact management
• Score rules
• Dynamic evaluation
• Potential clients
Scoring
• A/B testing
• Multivariate testing
• Content personalization
Web site optimization
• Portal engine integration
• Users
• Newsletter subscribers
• E-commerce customers
• On-line forms
• Search
Integration with other modules
• Newsletters
• Opened e-mail tracking
• Clicked link tracking
• Bounced e-mails
• A/B testing
E-mail marketing

Part II

Getting started

2 Getting started

2.1 Installing the sample Corporate Site

Examples in this guide are based on the sample **Corporate Site**. It is therefore recommended to have this website installed in order to try out the examples in practice. This topic provides information on how to get the website installed and is divided into two parts, depending on if you already have Kentico CMS installed (but without the Corporate Site), or if you are performing a completely new installation.

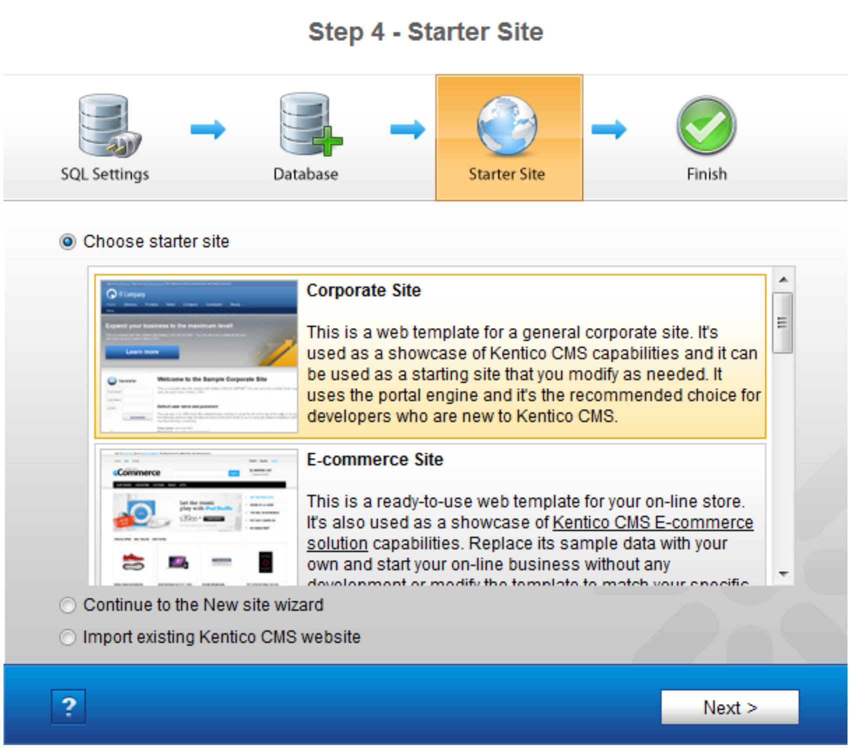
New Kentico CMS installation

Installation of Kentico CMS is fully described in Developer's Guide -> Installation and deployment. The following points sum up which steps need to be taken in order to perform the installation and provide links to the relevant topics in that chapter of the Developer's Guide:

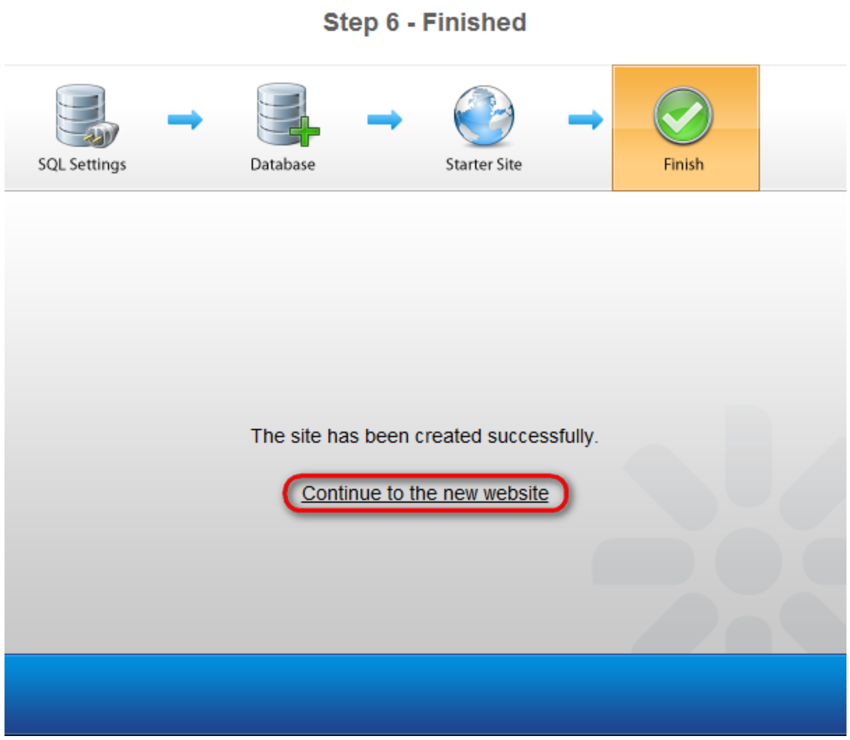
1. Launch *KenticoCMS_<version>.exe* and follow the instruction in Installation procedure -> Setup (KenticoCMS.exe).

2. Once the setup finishes, launch *Kentico CMS Web Installer* and follow the instructions in Installation procedure -> Web installer.

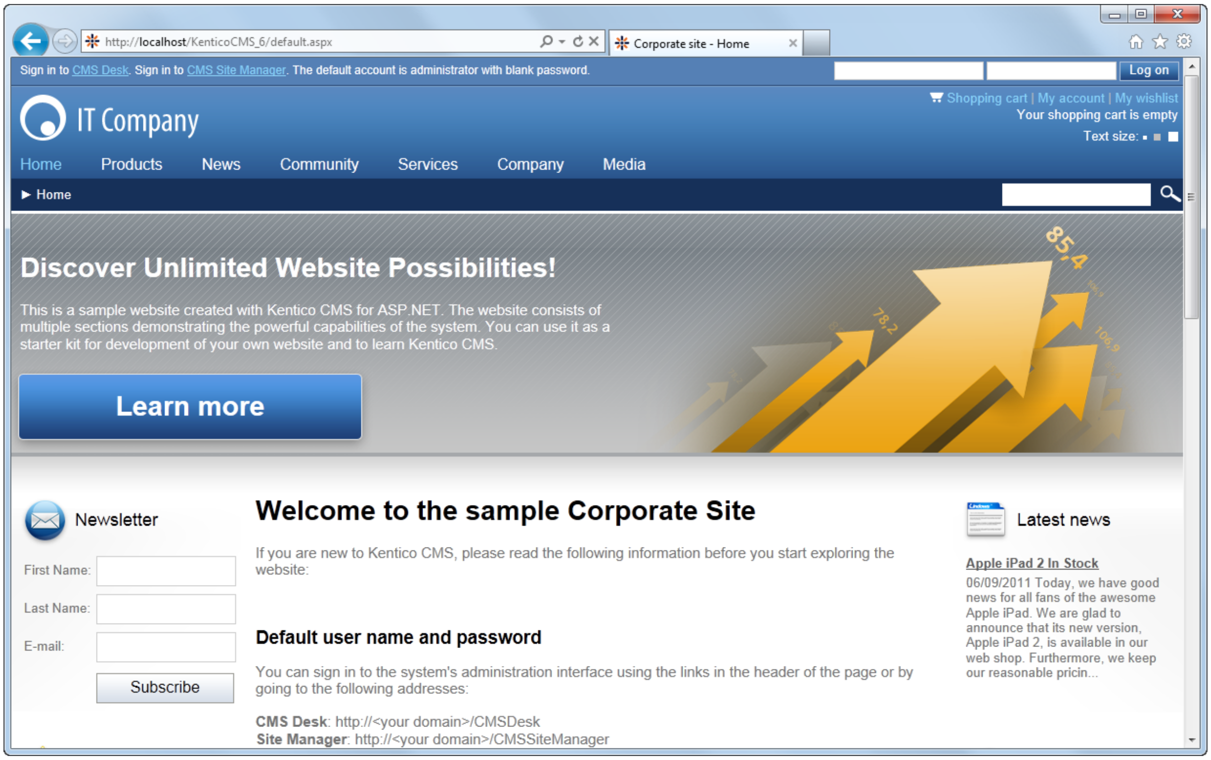
3. When the *Web Installer* is finished, access the URL of the new website. The first step of the **Database Setup** will be displayed. Follow the instructions in Installation procedure -> Database setup, until you reach the **Starter Site** step (the number of the step may vary). In this step, select the **Choose starter site** option and choose the **Corporate Site** sample site. Click **Next** to continue.



4. Wait until the **Database Setup** finishes importing the site. Once finished, the **Finished** step will be displayed. Click the **Continue to the new website** link highlighted in the screenshot below.



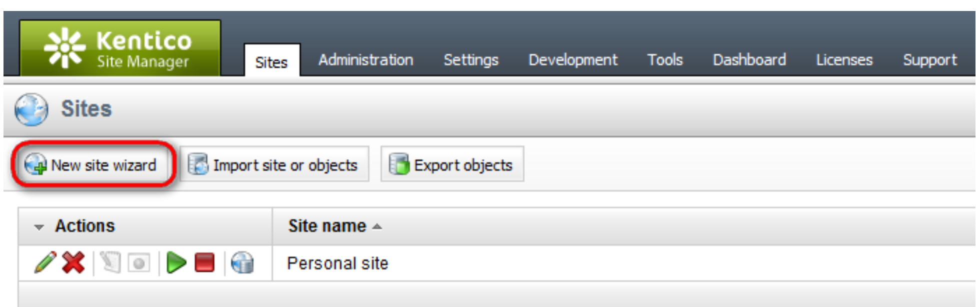
5. You will be redirected to the **Home** page of the website.



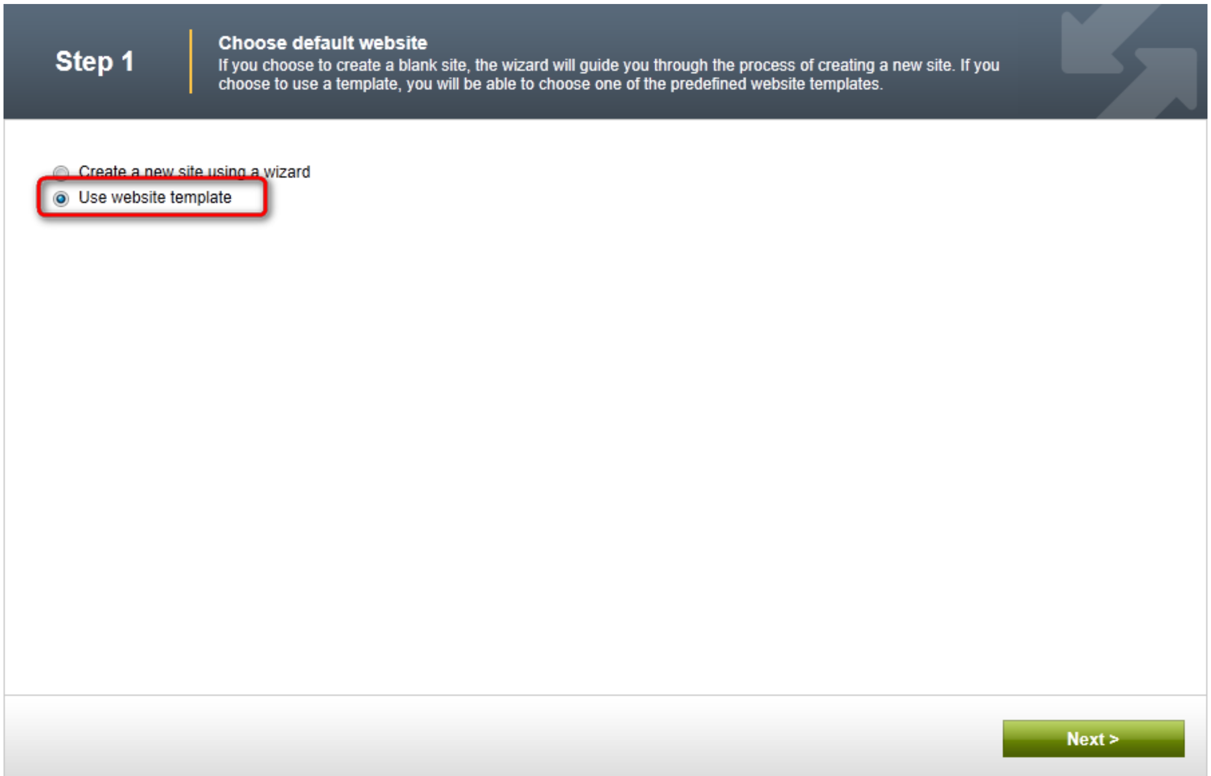
Existing Kentico CMS installation

If you already have Kentico CMS installed, but without the sample Corporate Site, you can add it using the New site wizard.

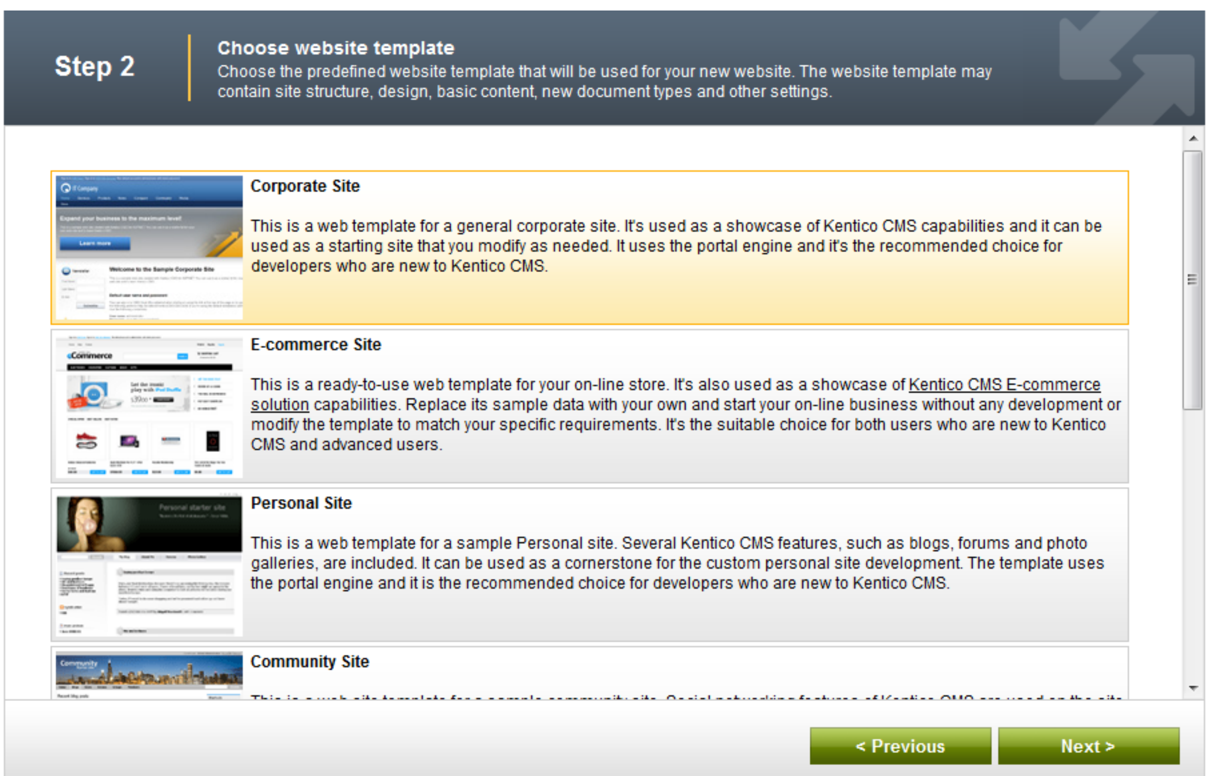
1. Log on to **Site Manager** and on the **Sites** tab, click **New site wizard**.



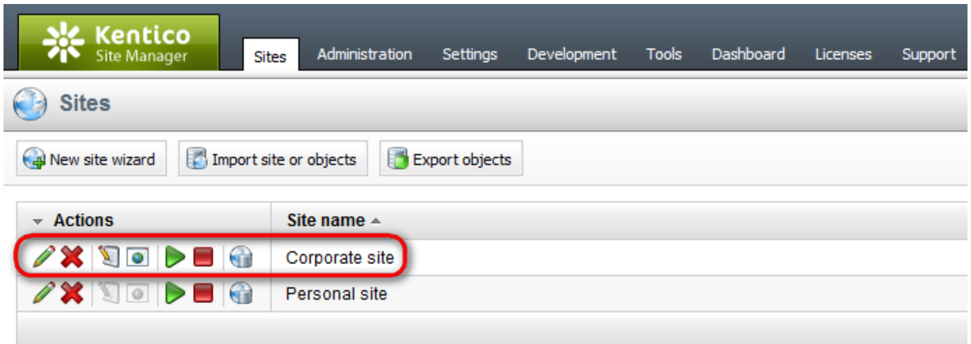
2. In the first step of the wizard, choose the **Use web site template** option and click **Next**.



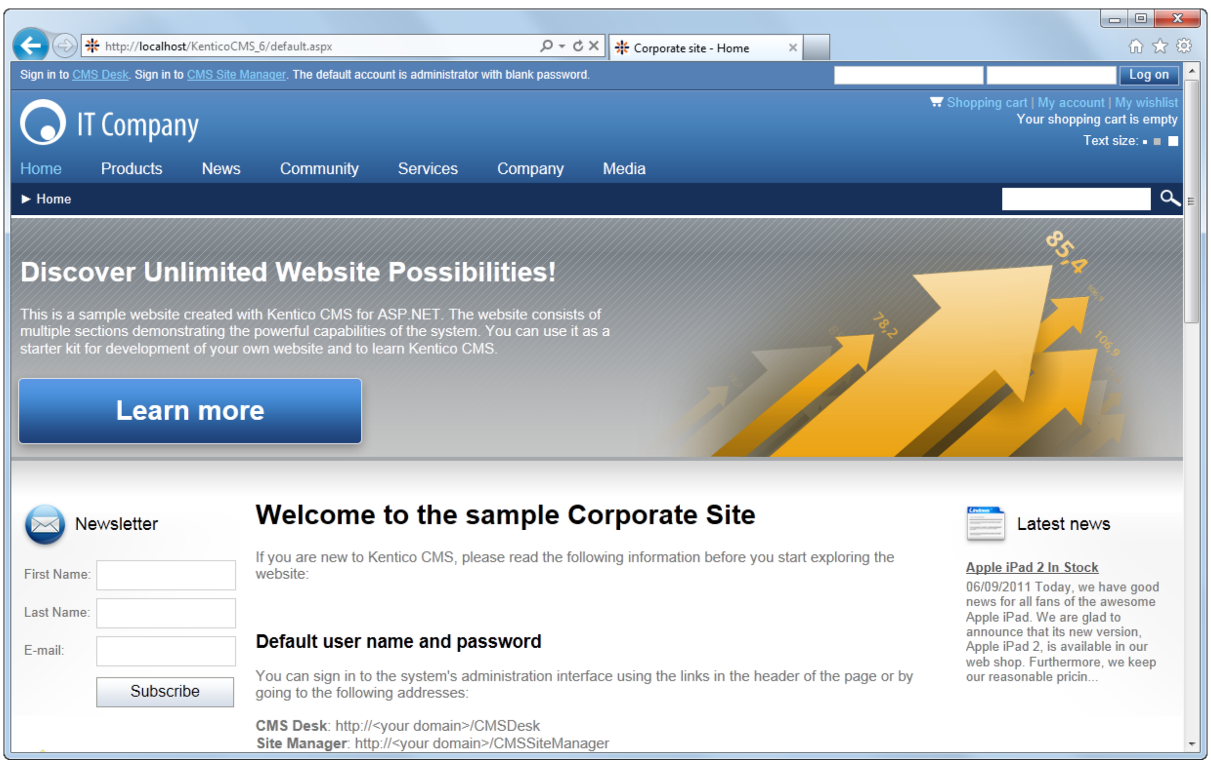
3. In **Step 2**, choose **Corporate Site** from the list of available website templates and click **Next**.



4. Go through the rest of the wizard as described in Developer's Guide -> ... -> New site wizard -> Website template. It is recommended to leave the default settings unless you really need to change them according to your specific requirements. Once the wizard finishes, you should see the website present in the list of websites in **Site Manager -> Sites**.



5. Sign out of the administration interface (use the **Sign out** button at the top right corner of the UI). You will be redirected to the **Home** page of the website.



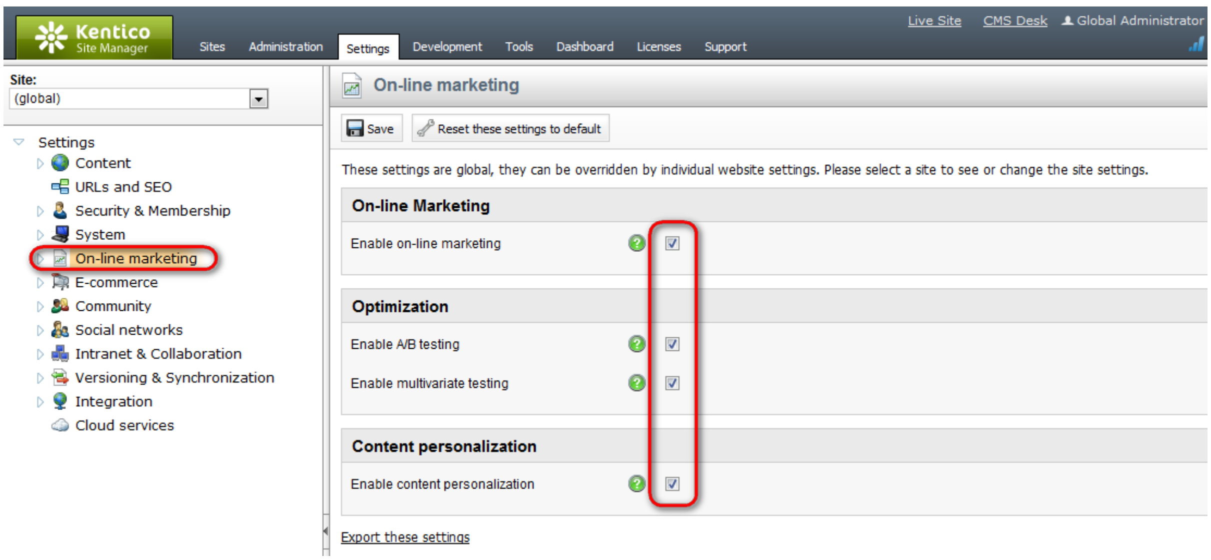
2.2 Enabling On-line marketing

This topic explains the configuration required to enable the on-line marketing features of Kentico CMS. It only mentions the most important settings necessary to get the examples in the following chapters working and produce meaningful results. Detailed information about all the available settings can be found in the respective chapters of this guide.

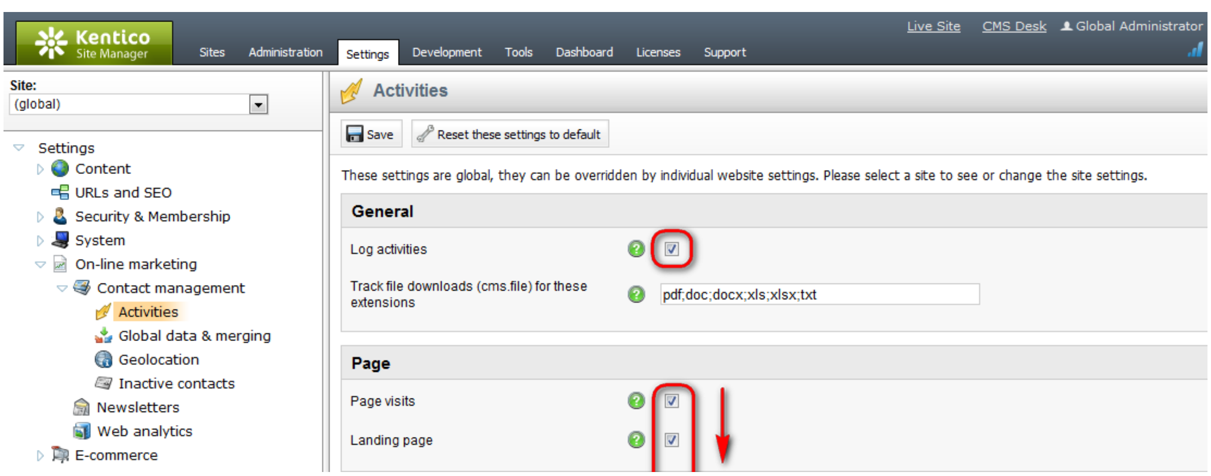
1. Go to **Site Manager -> Settings**. In the Site drop-down list, you can either leave the **(global)** option selected and let the global settings be inherited by the Corporate Site, or you can choose **Corporate Site** and adjust the settings only for it in case that you do not want other websites running in the system to be affected. Then choose the **On-line marketing** category in the tree on the left and make sure to enable the following settings:

- **Enable on-line marketing**
- **Enable A/B testing**
- **Enable multivariate testing**
- **Enable content personalization**

Leave the rest of the settings at their default values and click **Save**.



2. Switch to the **On-line marketing -> Contact management -> Activities** category. Here, make sure that the **Log activities** option is enabled and that logging of all possible types of activities is also enabled by the check-boxes below.



3. Finally, switch to the **On-line marketing -> Newsletters** category and configure the options as described in [E-mail marketing -> Monitoring bounced e-mails -> Configuration](#).

Immediate logging of activities

It is also recommended to add the following key to the *appSettings* section of your projects's *web.config* file:

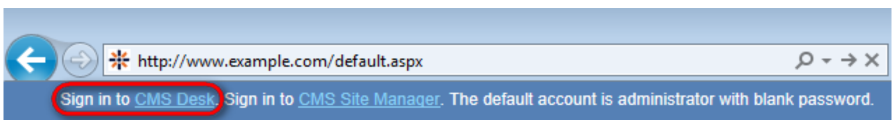
```
<add key="CMSLogActivityImmediatelyToDB" value="true" />
```

This key ensures that activities are logged directly into the system database instead of being logged into a file and periodically batch-processed by a scheduled task. This may generate additional load on the system database, but results in minimal delays in displaying of activity-related data (e.g. activity-based rules in Scoring). It is therefore recommended to use the key when going through the examples in this guide in order to see activity-related results in the user interface immediately.

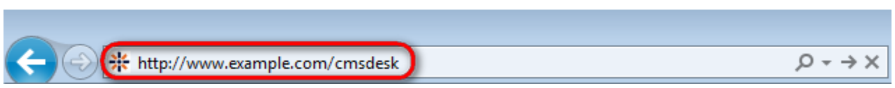
2.3 First sight

Once you have Kentico CMS and the sample Corporate Site installed, it is a good idea to get familiar with the parts of Kentico CMS user interface where the on-line marketing features are located.

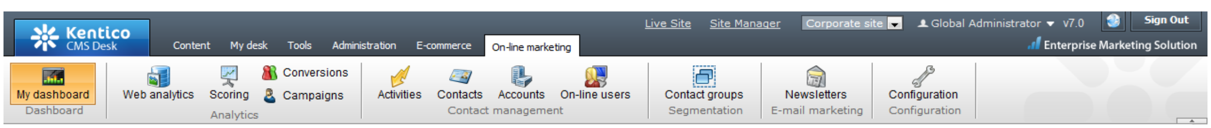
1. To get started, log on to **CMS Desk**. This can be done either by clicking the **CMS Desk** link in the header of the sample Corporate Site ...



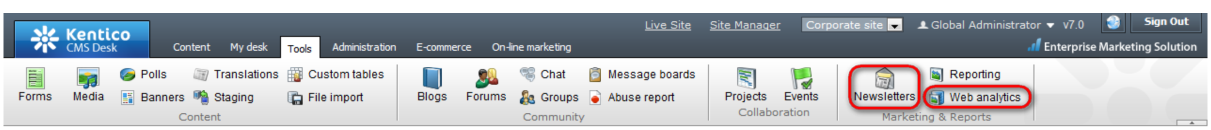
... or by directly entering an address in the following format into your browser's address bar: **http://<website domain>/cmsdesk**



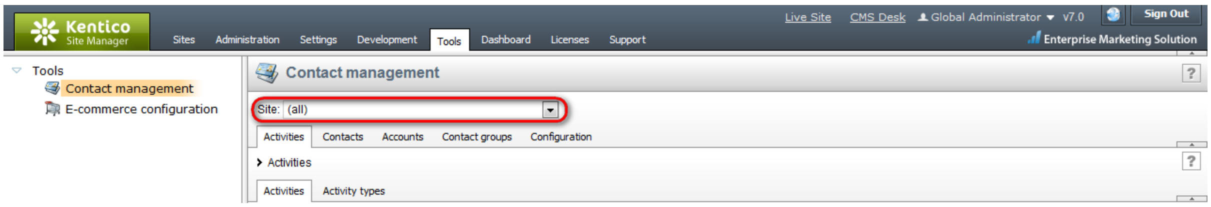
2. Once in **CMS Desk**, navigate to the **On-line marketing** tab. This is the main on-line marketing user interface which encapsulates user interfaces of the most of the on-line marketing features. You can access the individual features using the ribbon toolbar at the top of the screen.



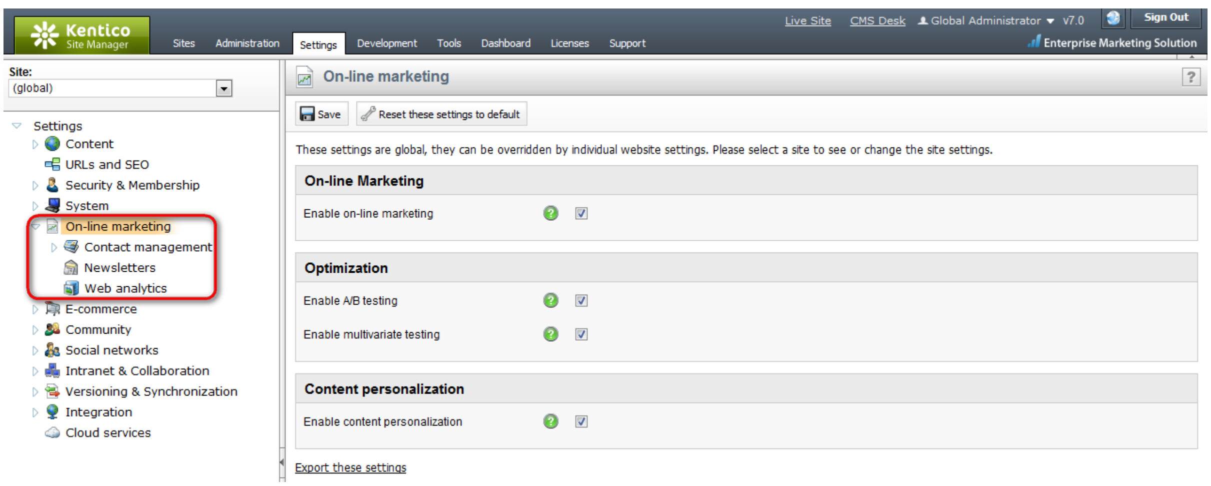
3. The **Web analytics**, **Conversions**, **Campaigns** and **Newsletters** features have duplicate user interfaces also on the **Tools** tab in **CMS Desk**. These user interfaces are identical to the ones on the **On-line marketing** tab, they provide the same options and display the same data. Please note that **Conversions** and **Campaigns** are embedded under **Web analytics** interface on the **Tools** tab.



4. The above mentioned interfaces are site-specific, which means that they provide functionality related only to the website currently edited in CMS Desk. In **Site Manager -> Tools -> Contact management**, there is a global user interface which enables management of [Contact management]() objects and configuration for all websites running in the system. Depending on the selection made in the **Site** drop-down list, you can choose if you want the UI to be related to a specific site, to global objects shared by all websites running in the system, or to all websites running in the system.



5. Finally, there is a number of settings that can modify the behavior of the on-line marketing features of Kentico CMS. These settings can be adjusted in **Site Manager -> Settings** and are located under the **On-line marketing** settings category highlighted in the picture below. You can learn more about them in the [Enabling On-line marketing]() topic.



Part III

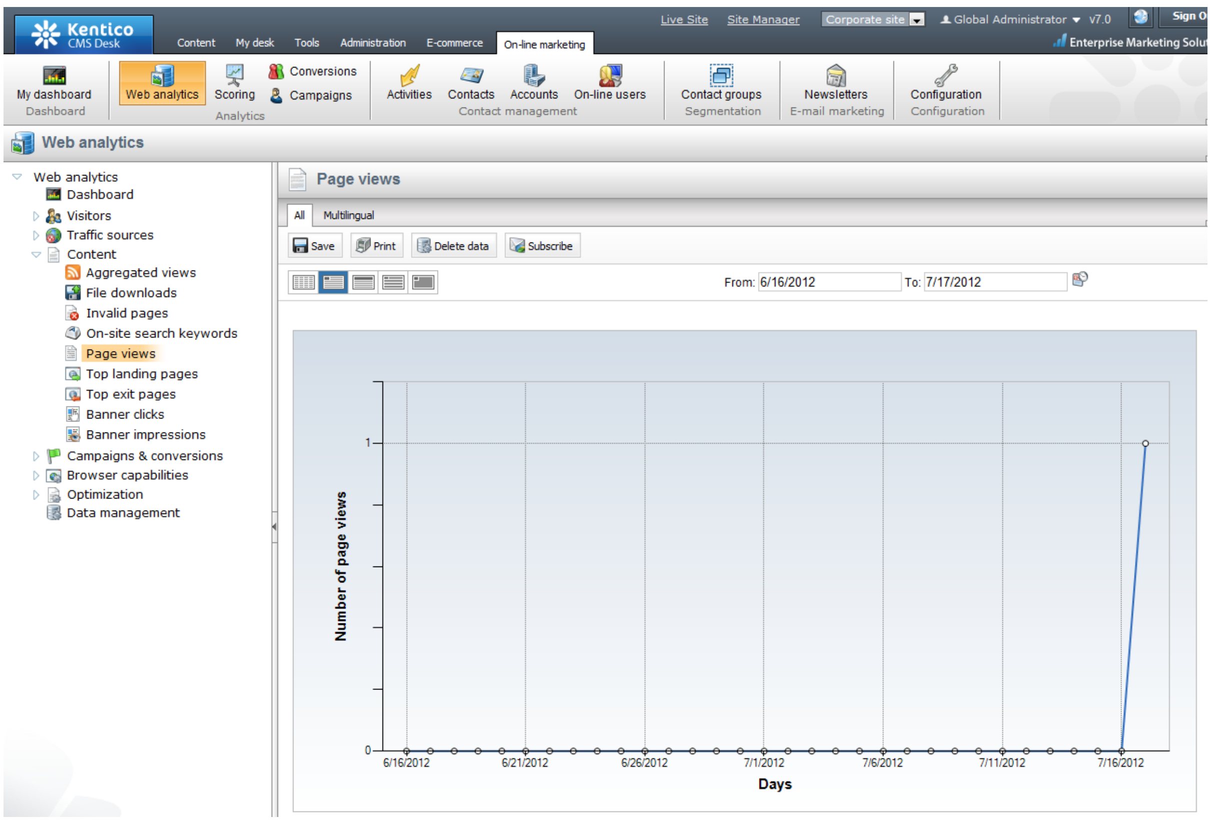
Web analytics

3 Web analytics

3.1 Overview

The web analytics module provides a way to measure and later analyze key metrics of your website such as visits, page views, file downloads, traffic sources and much more. Monitoring is only done on the live site, so actions that take place on the pages belonging to the Kentico CMS administration interface (CMS Desk and Site Manager) are not included in the statistics.

You can access the web analytics interface in **CMS Desk -> On-line marketing -> Web Analytics**. There are various types of statistics available that keep track of events that occur on the website. When you select a statistic, the page displays the corresponding data in a report.



In addition to measuring various statistics for the website and its pages, web analytics are also used by several other on-line marketing features:

- Conversions
- Campaigns
- Website optimization (A/B and multivariate testing)

To learn about all available features of the web analytics module, its internal mechanics and advanced configuration or customization options, please refer to the full documentation in the Modules -> Web analytics chapter of the Kentico CMS Developer's Guide.

3.2 Using web analytics reports

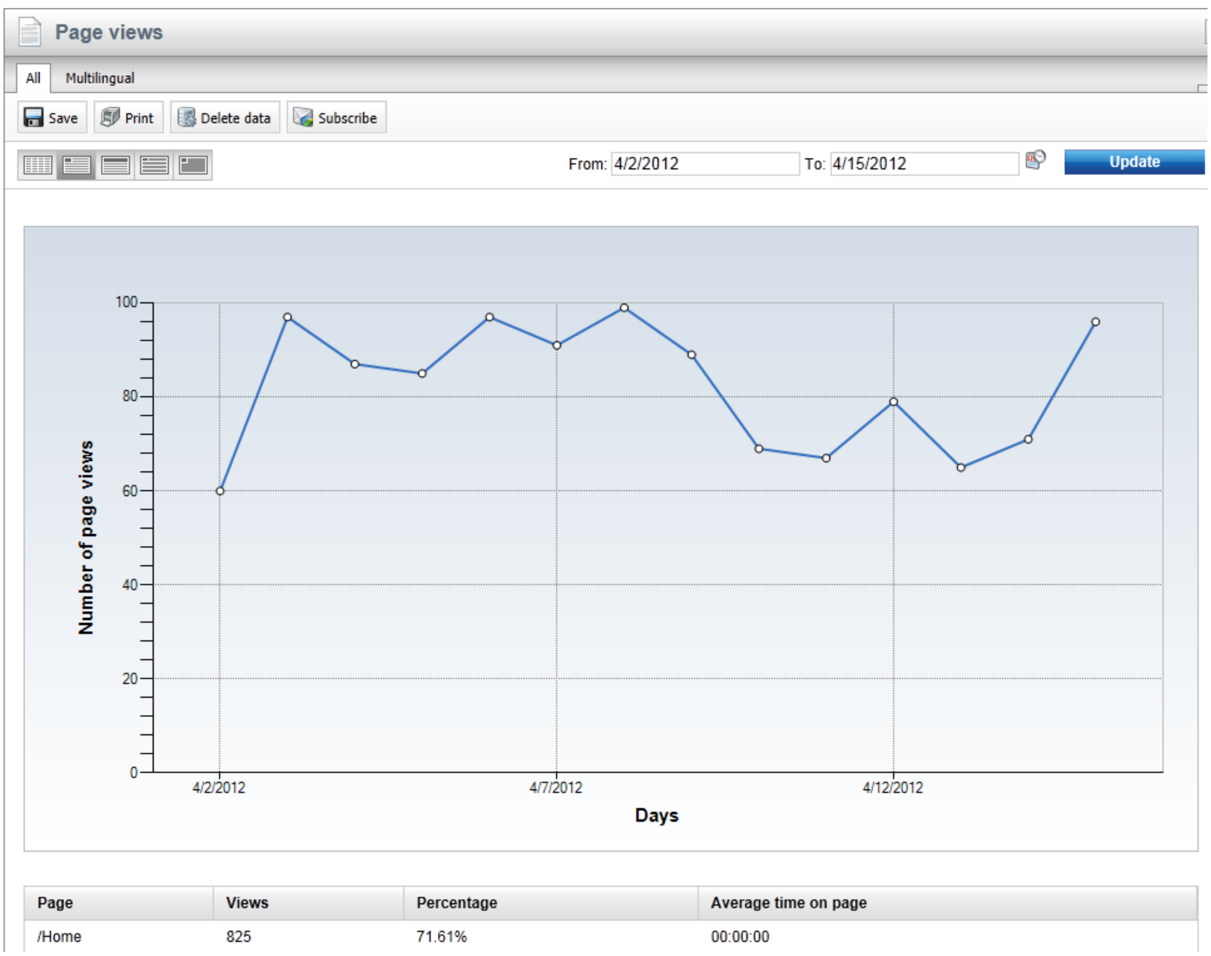
The data logged for web analytics statistics is presented in highly flexible reports that contain various kinds of charts and tables. The same type of reports are also used to display the results of other related on-line marketing features, such as campaigns and website optimization tests.

When viewing a web analytics report, the **From** and **To** fields on the right can be used to enter a time period. Only hits that were logged for the statistic during the specified interval will be included in the displayed data.

The following options allow you to choose which unit of time should be used in the report:

- **Hour**
- **Day**
- **Week**
- **Month**
- **Year**

This selection determines the length of time which is represented by individual units in the report's graphs (if there are any) and the precision that can be specified in the **From** and **To** fields.



The following actions may also be performed for every report:

- **Save** - saves the report in its current state (according to the selected time interval). To view saved reports at a later time, go to *CMS Desk -> Tools -> Reporting*, select the matching report under the appropriate category and switch to the *Saved reports* tab.
- **Print** - allows the report to be printed. The available options depend on the used browser.
- **Delete data** - may be used to clear the data measured for the given statistic. Please note that this permanently removes the data from the database. This action is only available for users who have the *Manage data* permission for the *Web analytics* module.
- **Subscribe** - opens a new dialog where you can subscribe to the report. Subscribing allows you to periodically receive e-mails with the up-to-date content of the given report. It is also possible to subscribe to a specific reporting component (graph or table) by right clicking on it and selecting the *Subscribe to* option in the displayed context menu.

The data displayed in the reports may also be exported into external files using various formats. This can be done by right clicking on a graph or table in the given report, which will open a context menu offering the following export options:

- **Export to Excel** - exports the data displayed by the given object to an XLSX spreadsheet.
- **Export to CSV** - exports data to a CSV file.
- **Export to XML** - exports data to an XML file.

After you select an action from the menu, your browser's standard file download dialog will pop up, letting you open or save the file with the exported data just like when downloading any other type of file.

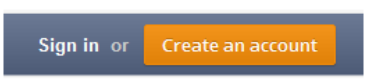
3.3 Google analytics

Kentico CMS allows you to use Google analytics as an alternative way of measuring and analyzing key metrics of your website. This is done by placing Google analytics web part on your web page. This topic provides you with steps necessary to set up your own Google analytics account and place the web part on your Kentico web page.

Google analytics account setup

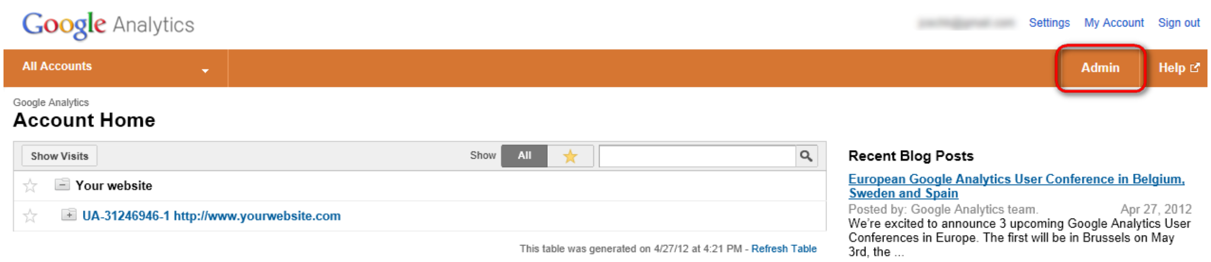
To be able to configure the web part, you need to have an existing Google analytics account set up with the website you want to track added to its accounts.

On the Google analytics website, you can either **Sign in** to your existing account or **Create an account** if you don't have one already.

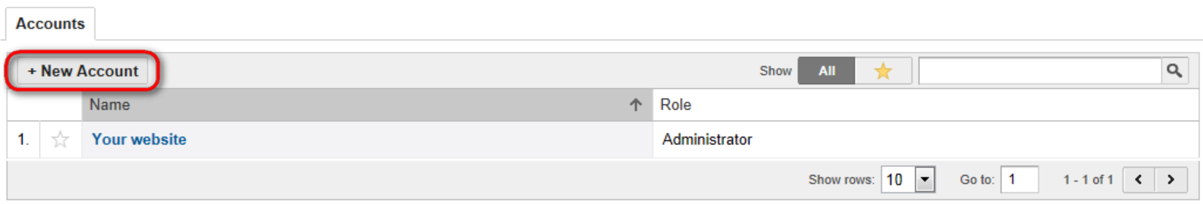


Sign in using an existing account

If you sign in using an existing Google analytics account, you will be redirected to a page listing your currently tracked websites. Click on **Admin** at the top right corner of the page to access **Account administration**.



In **Account administration,** click on **New Account.** Rest of the procedure is described in the following Create an account section**.**

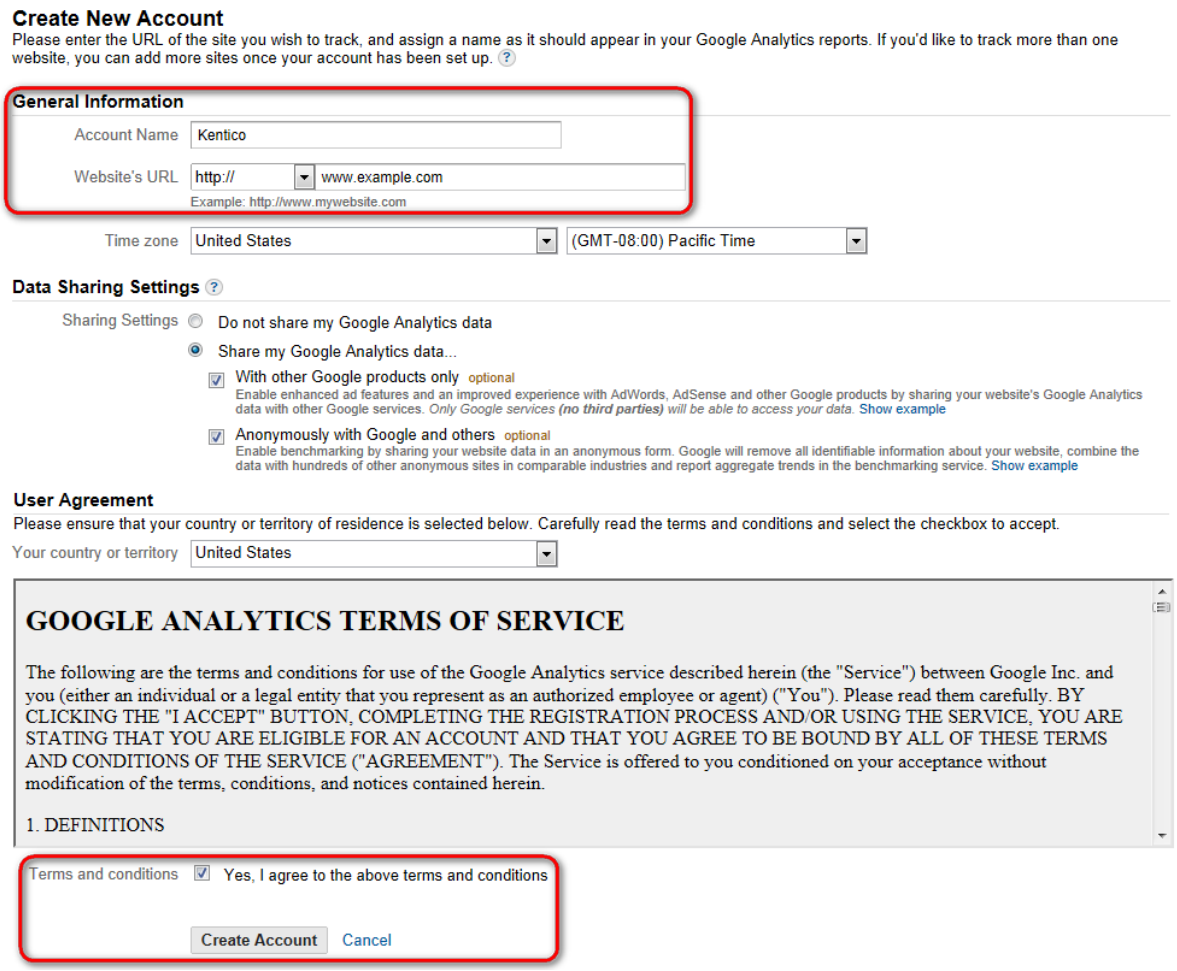


Create an account

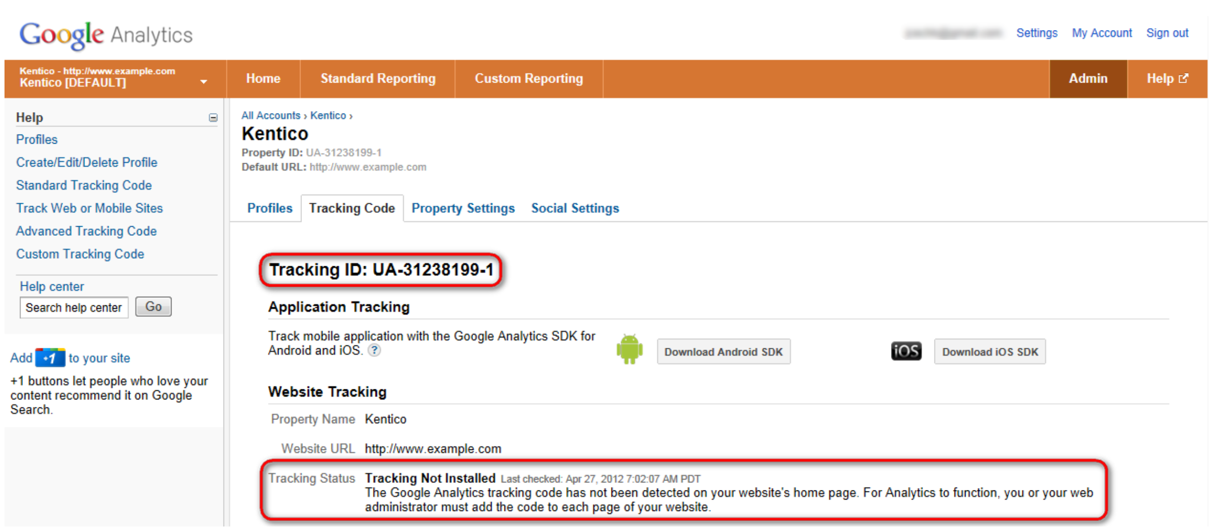
Once you choose to create a new account, you will be asked to fill in information about the website you want Google analytics to track:

- **Account Name -** the name under which you want your website to be listed in Google analytics.
- **Website's URL -** the URL used to access the website you want tracked by Google analytics.

Fill in the rest of the options based on your preferences. To finish creating the account, you need to turn the **Terms and conditions** checkbox on and click **Create Account.**

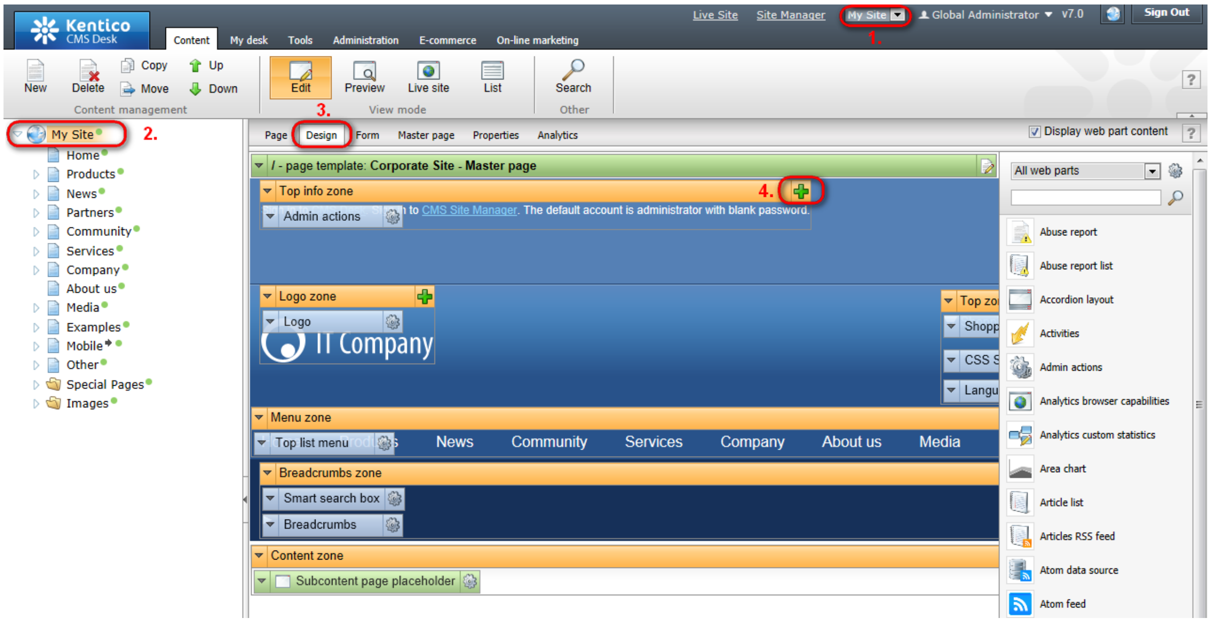


On the next page, you will be provided a **Tracking ID**, which you will use to configure the Google analytics web part in Kentico CMS. Also notice the **Tracking Status** attribute, which now notes that tracking hasn't been installed on your website.



Google analytics web part

Now that your Google analytics account is ready and you have your **Tracking ID**, it is time to put the Google analytics web part on the web page itself. Navigate to **CMS Desk** in Kentico CMS.



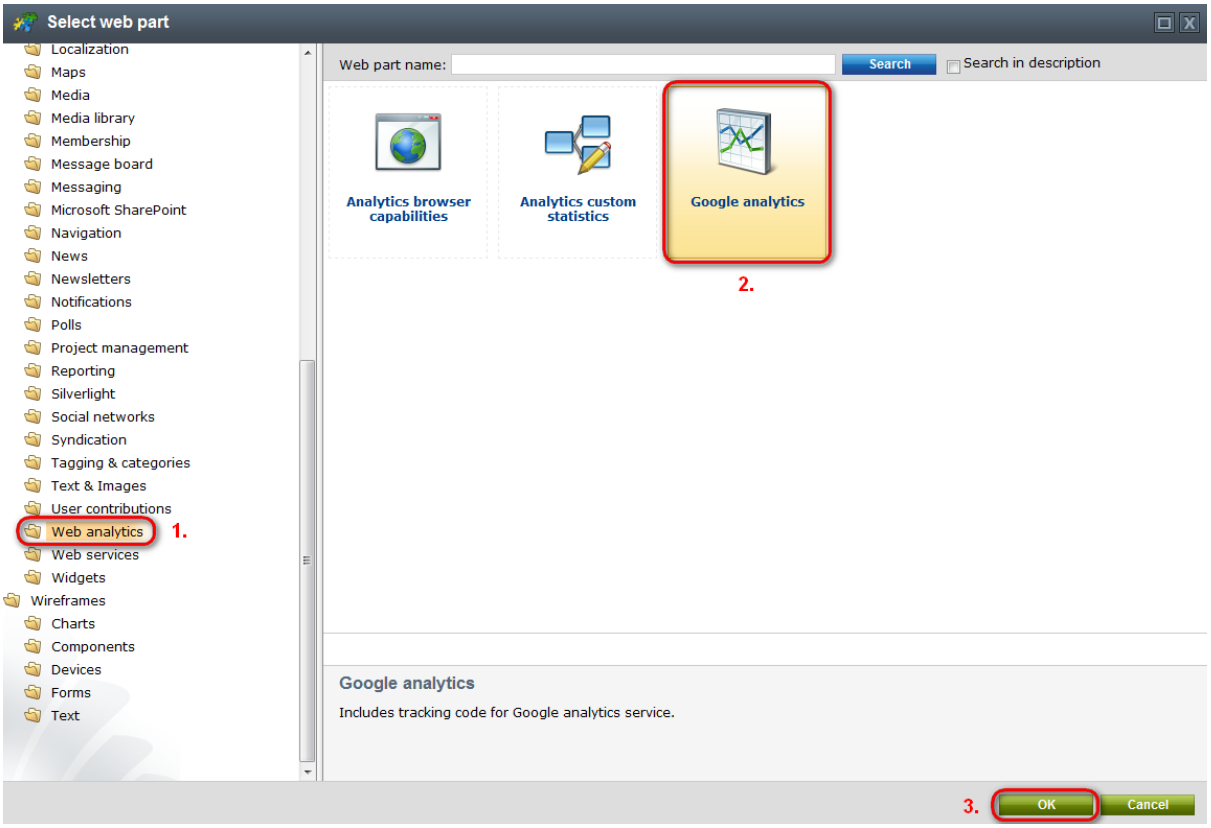
1. Make sure you choose the correct site in the drop-down list at the top of the page.
2. In the left **content tree**, click on the **root** of your site.
3. Switch to the **Design** tab.
4. Click on the **Add web part** (**+**) button in a web part zone of the page.

Please note

We recommend putting the web part to a web part zone that is inherited by the rest of the pages under the current site. This way, you can make sure the web part will be present on all of your pages. You can find more information on inheriting content in Developer's guide -> Development -> Web development overview -> Portal engine development model -> Visual inheritance.

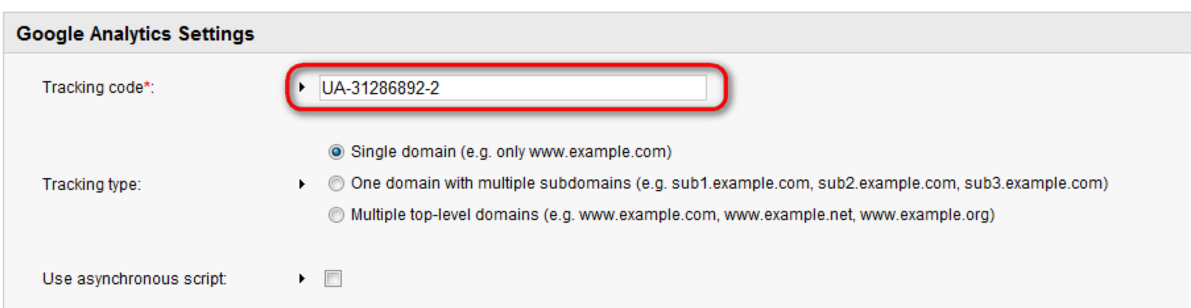
The Google analytics web part has no impact on the page's final layout.

A new dialog listing the web parts available in Kentico CMS opens.



1. In the left **content tree**, choose **Web analytics**.
2. Choose **Google Analytics**.
3. Click **OK**.

The Web part properties dialog of the **Google analytics** web part appears.

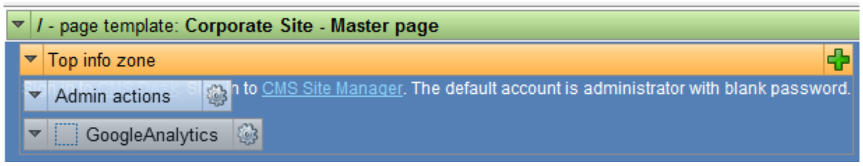


Under **Tracking code**, enter the code that you received on the Google analytics website earlier. Leave the rest of the options unchanged and click **OK** to save the changes. You can find more information on the properties of the Google analytics web part in the Google analytics properties documentation.

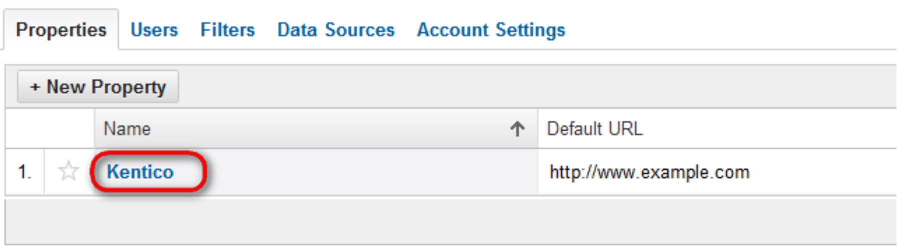
Please note

By turning on the **Use asynchronous script** option, you can have the web part use a different version of the tracking script. The script loads sooner upon a visit to the website. This makes it more likely the visit gets counted into the statistics even when the visitor leaves the page before it is fully loaded.

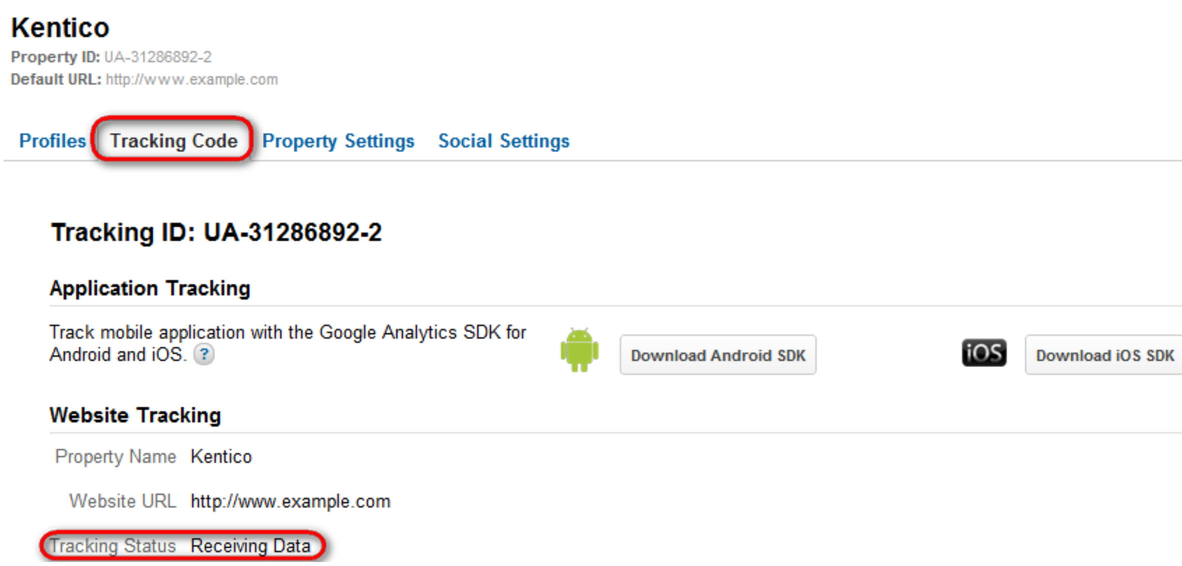
You should now see the Google analytics web part added to the web part zone.



Now on the Google analytics website, navigate to **Admin -> Account administration -> Account name -> Property name.**



Couple of minutes after placing the web part into Kentico CMS, you should see the **Tracking Status** change.



The possible tracking statuses that you may see on the page are:

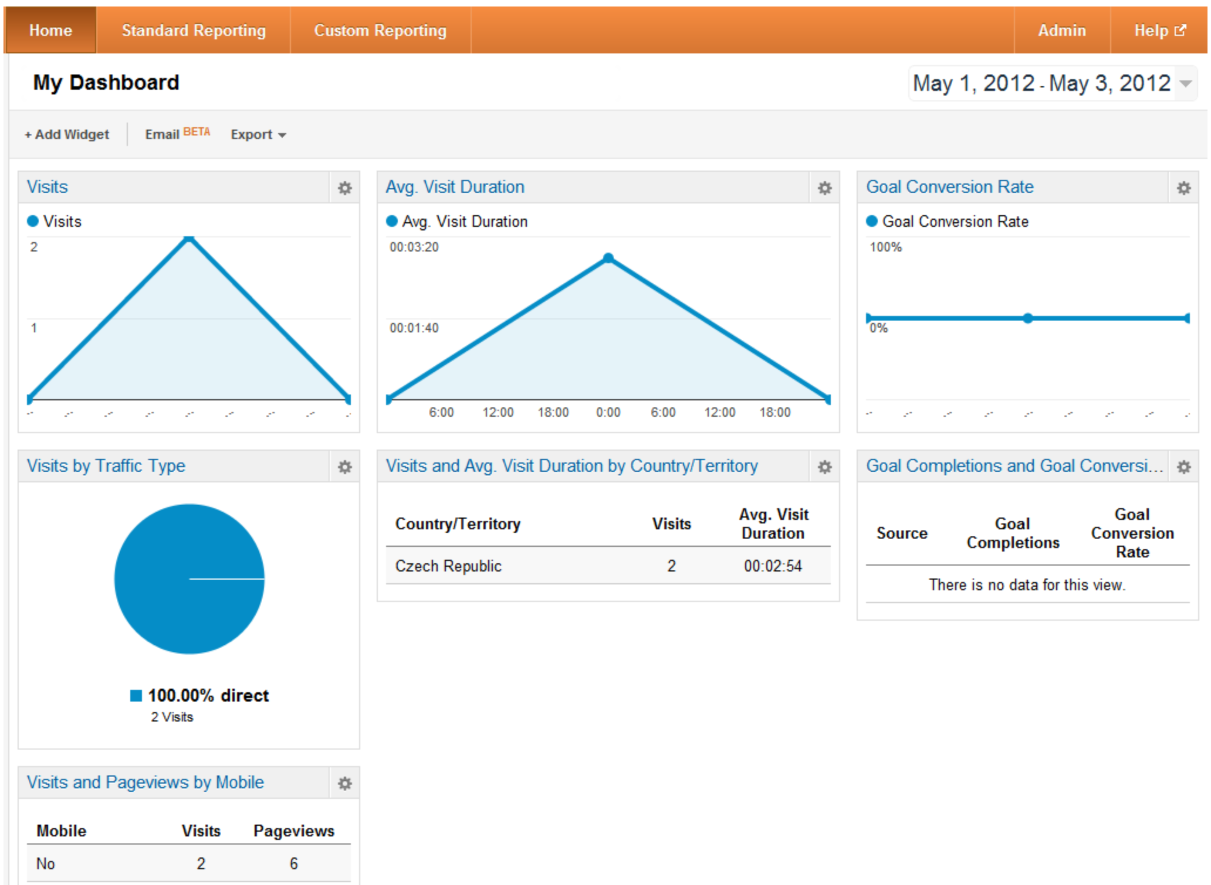
- **Receiving Data** - the tracking code has been detected on the home page of your website, and Google analytics is aggregating the data to populate into your reports.
- **Tracking Not Installed** or **Not Verified** - the tracking code has not been detected on the homepage

of your website.
- **Tracking Installed** - the tracking code is working properly and data is being populated into your reports.

Leave up to 24 hours for Google analytics to aggregate the website's data and for the status to change from **Receiving data** to **Tracking installed**.

Google analytics reports

Once Google analytics tracking is working correctly, you will be able to see reports of your website's visits and other statistics on the Google analytics website. Refer to the Google analytics Help website for more information on the various types of reports and ways to correctly customize them.



Please note

It may take up to 24 hours for any new data, e.g., visits to your website, to be visible in the Google analytics reports.

Part IV

Conversions

4 Conversions

4.1 Overview

The Web analytics module provides a way to track actions performed by your website's visitors and record them as *conversions*. This is typically done for desired events that somehow benefit the website, such as the registration of a new user, a product order, subscription to a newsletter or similar. Conversions are represented in the system by corresponding tracking objects, which are described in more detail in Managing conversions.

Once an action is defined as a conversion, a *conversion hit* is logged whenever it occurs. Additionally, a numerical value may be stored along with each hit to indicate its importance. To learn how you can assign conversions to individual types of actions and ensure that they are logged correctly, please refer to the Logging actions as conversions topic.

Once you start tracking conversions on the live site, you can compare the recorded data with the values of other statistics, such as the total amount of visitors. This allows you to evaluate the website and adjust it as necessary.

While simply tracking all conversions that occur on the website can be useful, in many cases you may also require additional information about the context in which the given actions occurred. For this reason, conversion tracking is integrated with several other on-line marketing features. When used together with Campaign tracking, conversions allow you to record actions only for visitors who arrive on the website in a specific way (e.g. as a result of a marketing campaign).

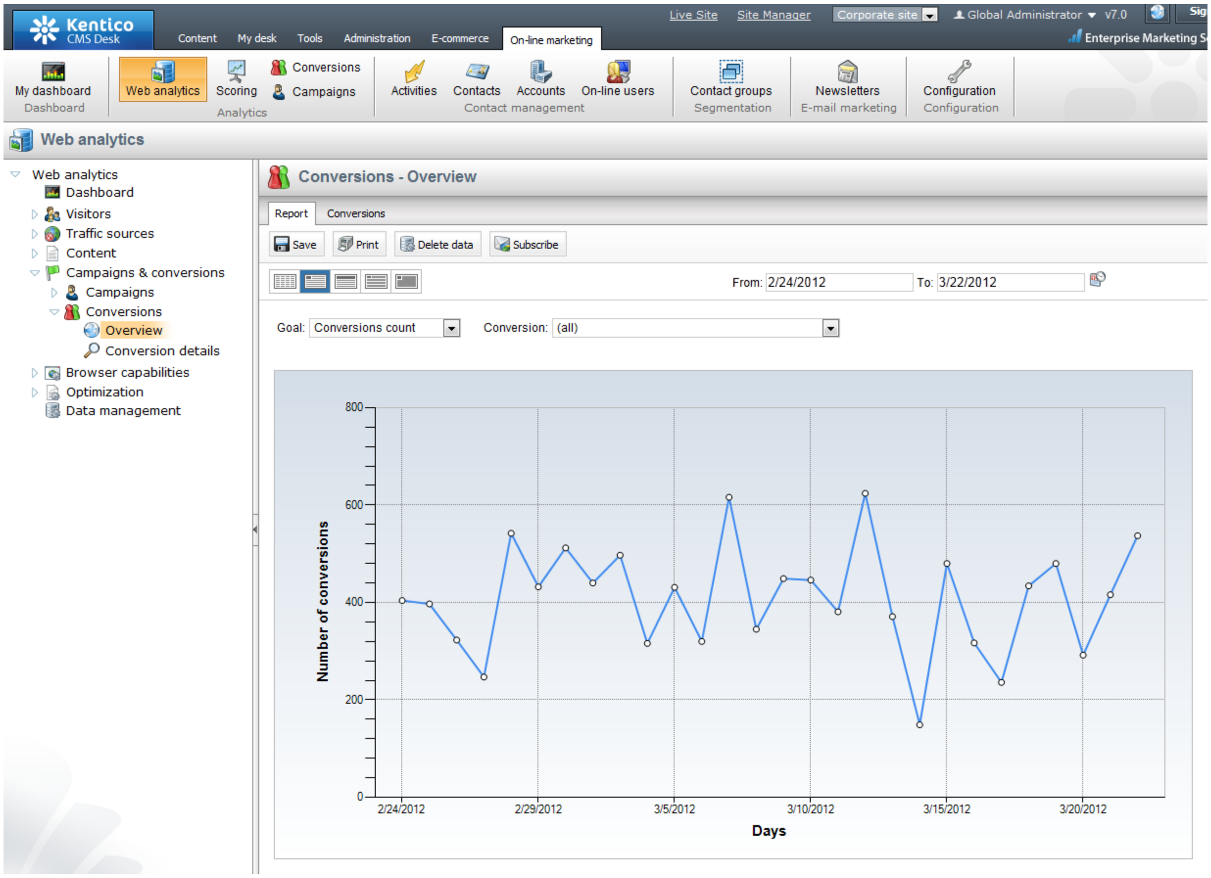
If you wish to optimize your site to increase its conversion rate (i.e. make it more user friendly to get better results), you may utilize A/B or Multivariate testing. These features allow you to accurately measure how changes made to the content or structure of your website's pages affect the behavior of users.

Conversions only record the total statistics of an action and do not store information about the specific users who performed it, or any other details. This advanced type of tracking can however be achieved by using the Contact Management module. It provides a way to log specific actions as Activities, including various information about the context in which the given action occurred.

4.2 Managing conversions

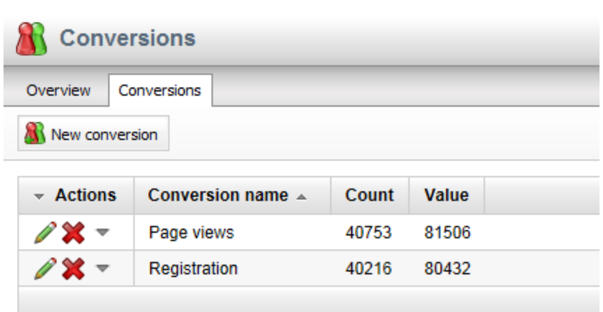
To access the management interface dedicated to conversions, go to **CMS Desk -> On-line marketing -> Web analytics**, expand the **Campaigns & conversions** category in the tree on the left and select **Conversions**. This section of the UI may alternatively be reached by selecting the **Conversions** option directly from the main on-line marketing menu.

The **Overview** contains a web analytics report displaying the conversions that were logged on the website over the specified time period.



This report only provides a general overview of the conversion statistics. Additional data for conversions that were logged under special circumstances is available in the **Conversion details** report and in the specialized reports under the **Campaigns** and **Optimization -> A/B tests** or **MVT tests** categories.

If you switch to the **Conversions** tab, you can view a list of all conversions defined for the current site and manage them as necessary.

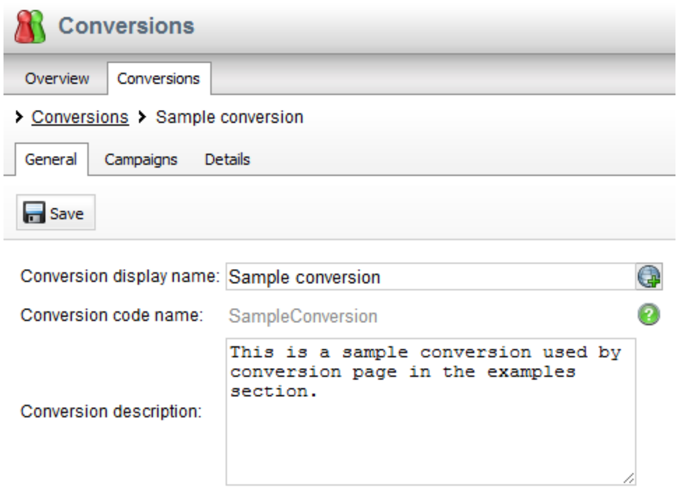


To define a new conversion, click **New conversion** and fill in the following properties in the displayed dialog:

- **Conversion display name** - the name of the conversion displayed in the administration interface and in reports.
- **Conversion code name** - sets a code name that serves as a unique identifier for the conversion.

- **Conversion description** - can be used to enter text describing the conversion's purpose.

The entered data may be modified at any time by editing ( ) the given conversion object on the **General** tab.



The **Campaigns** tab allows you to configure which campaigns should track the currently edited conversion as part of their statistics. By default, campaigns log all possible conversions, so it is only necessary to assign campaigns that are configured to work with a limited set of conversions. This type of configuration is not available for A/B or multivariate tests, which automatically keep track of all conversions performed on the website.

As you can see, the objects representing conversions are very simple and do not require any advanced configuration. It is however necessary to assign the conversions to the appropriate actions to ensure that they are logged correctly. This can be done through various other parts of the Kentico CMS administration interface as described in the Logging actions as conversions topic.

4.3 Logging actions as conversions

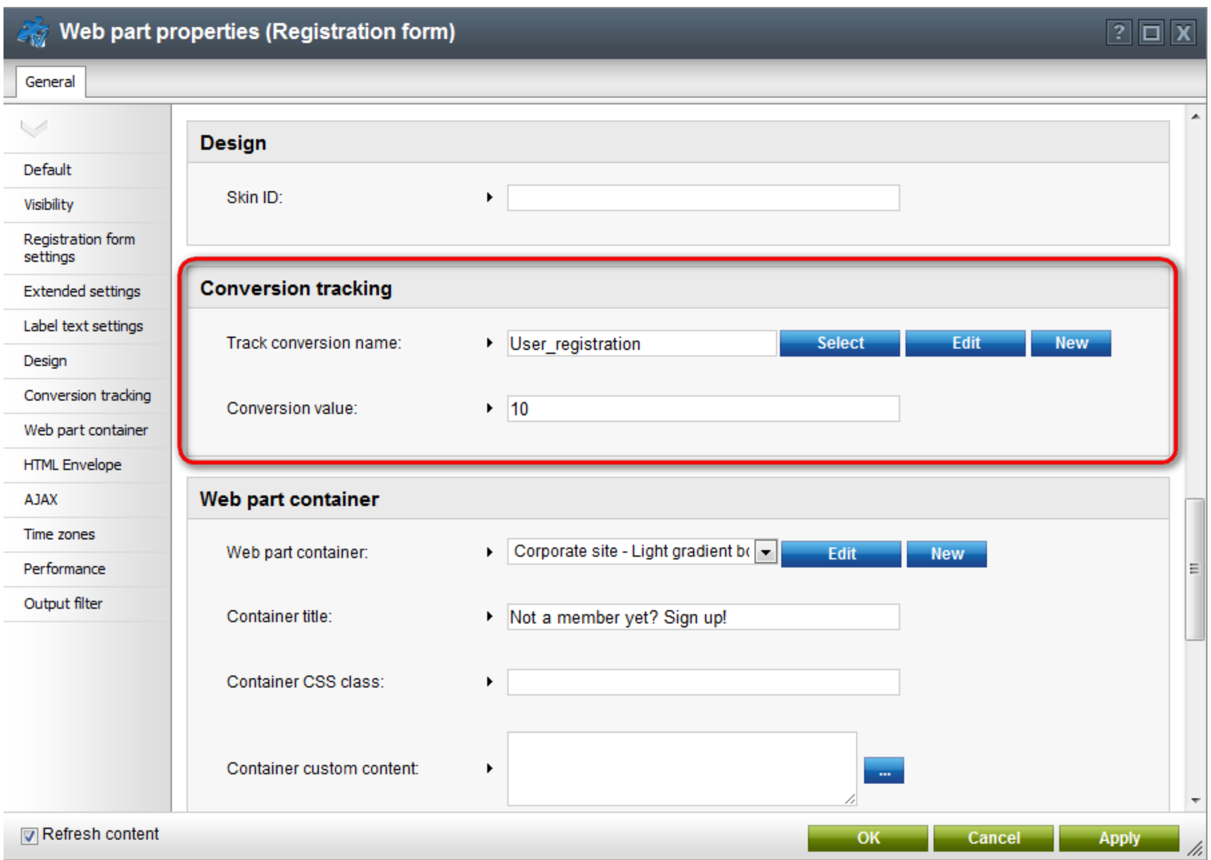
This topic describes the possible options that may be used to ensure that the system logs specific events as conversions. When assigning a conversion through the user interface, two types of fields are provided.

The first is a standard conversion selector. You can either enter the code name of a conversion into the text box or click the **Select** button to choose from a list of conversions defined for the current site. If you enter a name that does not match any conversion in the system, a conversion with this name will automatically be added. The **New** and **Edit** buttons allow you to create a new conversion or modify the properties of the selected one directly from the given part of the user interface.

The second field is optional and provides a way to set a number that will be recorded along with the conversion when the tracked action is performed. This may be used to indicate the relative importance of the conversion, the profit generated by a single conversion hit or similar. The values are cumulative, i.e. when a conversion hit is logged, the specified value will be added to the total sum previously recorded for the conversion. You may insert a Macro expression into this field to dynamically retrieve a value from the current site context. For examples of conversion value macros, please see the sections below dedicated to individual types of actions.

Web part and widget actions

Many of the default Kentico CMS Web parts and Widgets that allow users to perform important actions come with built-in support for conversion tracking. To configure a specific web part or widget instance to log actions as conversions, open its properties dialog and enter the appropriate values into the **Track conversion name** and **Conversion value** properties.



Below you can find a list of all types of actions that can be tracked as conversions through web parts:

Action	Web part(s)
User registration	In this case, the conversion will be logged when a visitor successfully completes their registration using the given web part. There are multiple web parts that allow users to register on the website: • Registration form

	• Custom registration form • Facebook Connect logon and Facebook Connect required data • LinkedIn logon and LinkedIn required data • Windows LiveID and LiveID required data • OpenID logon and OpenID required data
Newsletter subscription	Newsletter subscriptions may be tracked as conversions through the Newsletter subscription or Custom subscription form web part. This can also be done for the widget based on the web part.
Shopping cart actions	The Shopping cart web part may be used to track two types of events: • **Registration** - occurs when a customer registers on the website through the shopping cart. • **Order** - logged when a customer successfully completes an order. You can assign conversions to these actions for specific instances of the shopping cart web part through the corresponding **conversion name** properties. The conversion values of these events can be configured for the entire website via the e-commerce website settings described in a dedicated section below.
Filling in an on-line form	A conversion can be logged when a user submits a form displayed by the On-line form web part.
Voting in a poll	The Poll web part may be used to log a conversion whenever a user votes in the displayed poll.

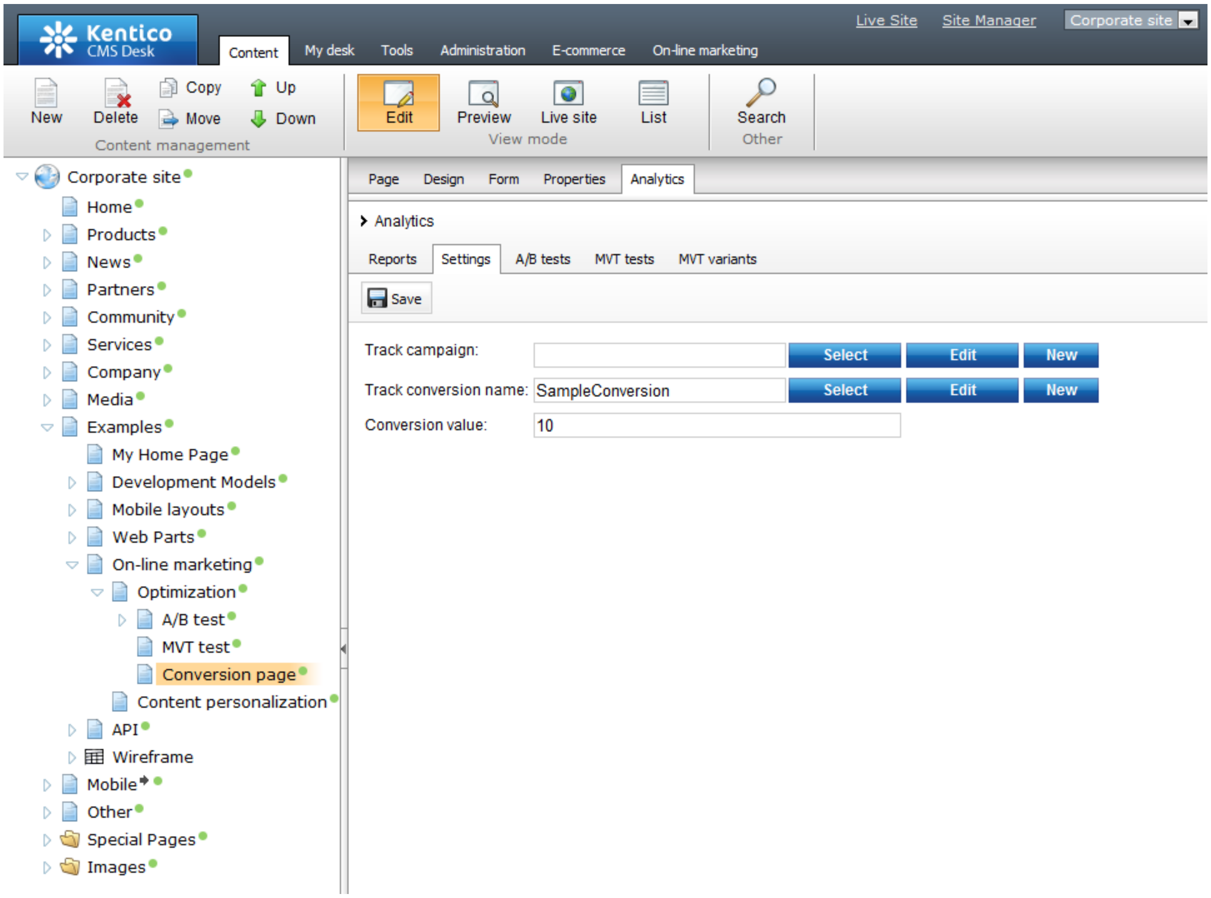
Entering conversion value macros into web part properties

The **Conversion value** properties of web parts only support numeric (decimal) values, so it is not possible to specify a macro expression directly. However, you can enter macros via the **Edit value** dialog that can be opened by clicking the icon next to the given property.

In the case of widgets, macros entered by users into properties are not resolved at all. If necessary, macro expressions can be pre-set as the default values of widget properties by administrators. This can be done by editing the given widget in **Site Manager -> Development -> Widgets** on the **Properties** tab.

Page views of specific documents

You can also use conversions to keep track of the amount of hits received by individual pages. To configure this behavior for a page, go to **CMS Desk -> Content -> Edit**, select the document representing the given page from the content tree and switch to its **Analytics -> Settings** tab. To assign a conversion and associated value, fill in the **Track conversion name** and **Conversion value** fields as described above.



The specified conversion will be logged every time the given page is opened by the website's users. You can find an example of such a page on the sample Corporate Site under the **Examples** sections as shown in the image above.

E-commerce actions

Conversions may be configured for e-commerce actions that occur on the entire website using the settings in **Site Manager -> Settings -> E-commerce**. There are three types of events that can be tracked:

- **Registration** - occurs when a customer registers on the website through the checkout process.
- **Order** - logged when a customer successfully places a product order.
- **Add to shopping cart** - occurs when a user adds a product to a shopping cart on the website.

You can assign a different conversion and value to each of these actions through the appropriate **conversion name** and **conversion value** settings. The registration and order conversion name settings can be overridden for individual instances of the shopping cart web part through their corresponding properties.

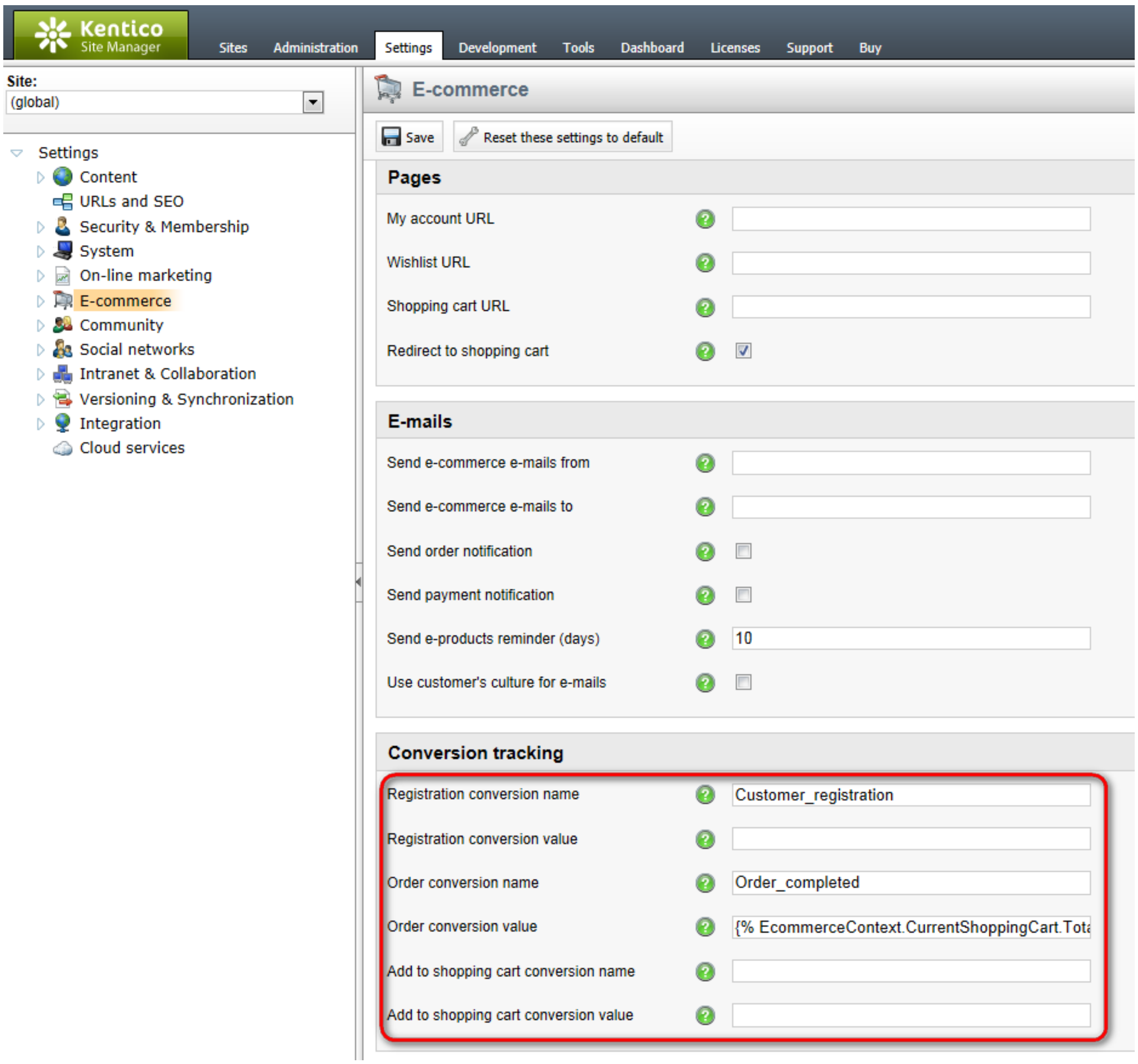
If you wish to log the conversion value dynamically based on item prices, you may use macro expressions, for example:

Order conversion value: *{% EcommerceContext.CurrentShoppingCart.TotalPrice %}*

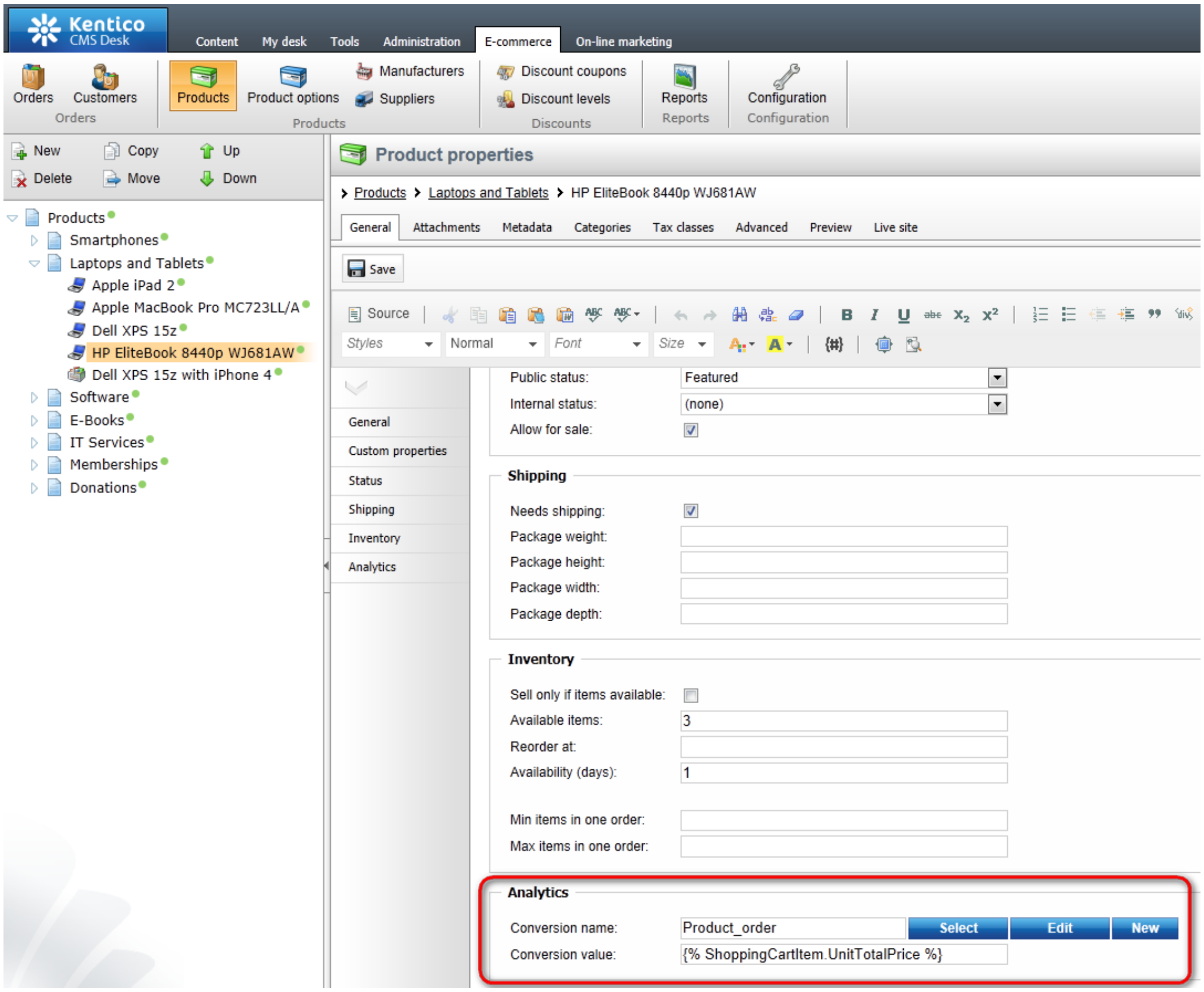
This macro is resolved into the total price of all items contained in the order, including tax and shipping. With this configuration, each *Order* conversion automatically stores the price of the given order as its value.

Add to shopping cart conversion value: *{% ShoppingCartItem.UnitTotalPrice %}*

With this macro as the conversion value, the *Add to shopping cart* conversion logs the price (including tax) of the specific product that was added to the shopping cart.



The settings described above allow you to track entire orders, but in some cases you may wish to log a separate conversion hit every time a product of a specific type is purchased. This can be done by navigating to the product list in **CMS Desk -> E-commerce -> Products** and selecting the particular product (or when editing a product document in the main content tree on the **Form** tab).



Here you can assign a conversion through the **Conversion name** property at the bottom of the product's editing page.

The **Conversion value** field can be used to specify an appropriate value that will be recorded for each conversion hit. In addition to numeric values, you may enter macro expressions here, for example: *{% ShoppingCartItem.UnitTotalPrice %}*. This macro allows the conversion to log the price of the given product as its value. The advantage of a macro is that it retrieves the price dynamically, including tax and any potential discounts applied by the given customer.

Please note that these properties are not available for global products, since each site has its own separate set of conversions.

You can also use the same approach to configure different conversion settings for individual product options (via **E-commerce -> Product options -> edit Category -> Options**).

Here you can assign a conversion through the **Conversion name** property at the bottom of the product's editing page.

Part V

Campaigns

5 Campaigns

5.1 Overview

One of the most common techniques used to bring new traffic to a website are on-line marketing campaigns. When attempting to determine if a campaign was cost-effective or when measuring its exact results, simply tracking referring sites or URLs may not always provide sufficient data.

In these cases, the campaign tracking support provided by the Web analytics module of Kentico CMS may be used, which allows you to accurately monitor traffic generated by individual campaigns. This can include banners, marketing e-mails or any other method used to present links to your website. In addition to logging the amount of visits, this feature lets you keep records of any important actions performed on the website by users who arrive as a result of a campaign.

How the tracking works

There are several ways to identify visitors who come to your site through a campaign link (further details can be found in the Managing campaigns (Example) topic). When this happens, the system updates the visit statistics and stores a cookie named **Campaign** in the user's browser, which saves the name of the corresponding campaign tracking object as its value. Only one campaign may be assigned to a user at a given time and the current value is overwritten if the user returns to the site through a different campaign.

Until the cookie expires (its duration is 24 hours), any actions performed by the visitor that are defined as conversions will be logged as part of the statistics kept for the campaign. Please refer to the Conversions chapter for information about how the tracking of individual actions can be configured.

The currently active campaign will also be stored among the details of any Activities that are logged by the Contact management module.

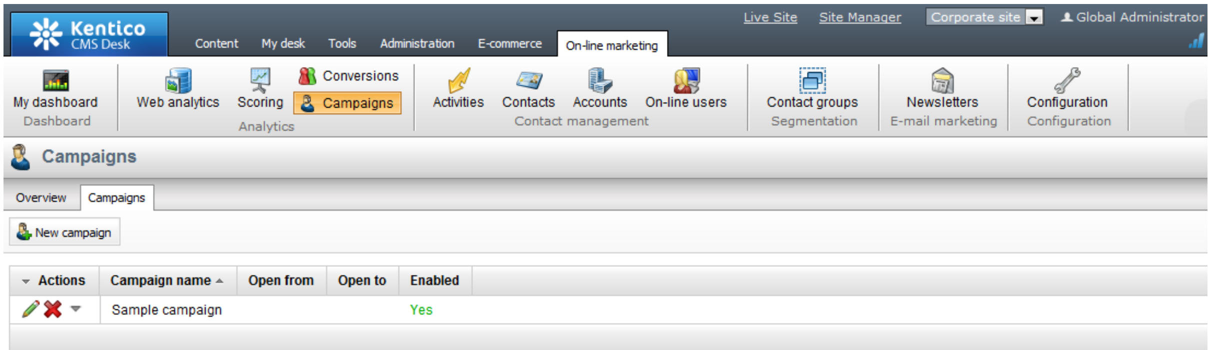
If the visitor registers on the website while the cookie is still valid, the name of the currently active campaign will be permanently stored in their user settings (administrators may view the value in **Administration -> edit () user -> Settings -> Campaign**). This data field only serves to provide information about users and is not used to track activity on the website.

Campaign statistics and results can be viewed using special reports available in the web analytics interface, which are described in the Evaluating campaigns topic.

5.2 Managing campaigns (Example)

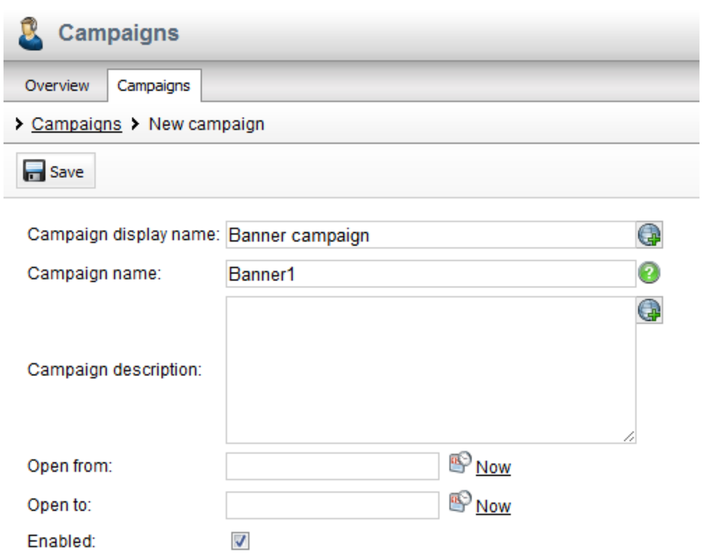
Each marketing campaign is represented in the system by a corresponding object. To configure these objects, go to **CMS Desk -> On-line marketing** and select **Campaigns** from the menu. Then, switch to the **Campaigns** tab, where you can view a list of all campaigns defined for the current website and manage them as needed.

By default, the Corporate Site contains a single campaign called **Sample campaign**.



To better demonstrate the functionality of campaigns, we will need two additional ones. Click **New campaign** and fill in the following properties:

- **Campaign display name**: *Banner campaign*; the name of the campaign displayed in the administration interface (in campaign lists and reports).
- **Campaign name**: *Banner1*; sets a code name that serves as a unique identifier for the campaign. It is also stored as the value of the *Campaign* browser cookie used to identify visitors who came to the website through the given campaign.
- **Campaign description** - can be used to enter a text description for the campaign in order to give information about its purpose, goals etc. You may leave this empty.
- **Open from/to** - sets the time interval during which the campaign should be active. The *Calendar* button ( ) can be used to select an exact date and time. Leave these empty for this example.
- **Enabled** - this property may be used to manually start or stop the campaign. Visitor statistics and actions will not be logged for disabled campaigns. Leave the box checked.



Click **Save** to add the campaign. The **General** tab of the campaign's editing interface will now be displayed, where you can modify the fields listed above at any time. Several additional properties are also available, but these are described later in this chapter.

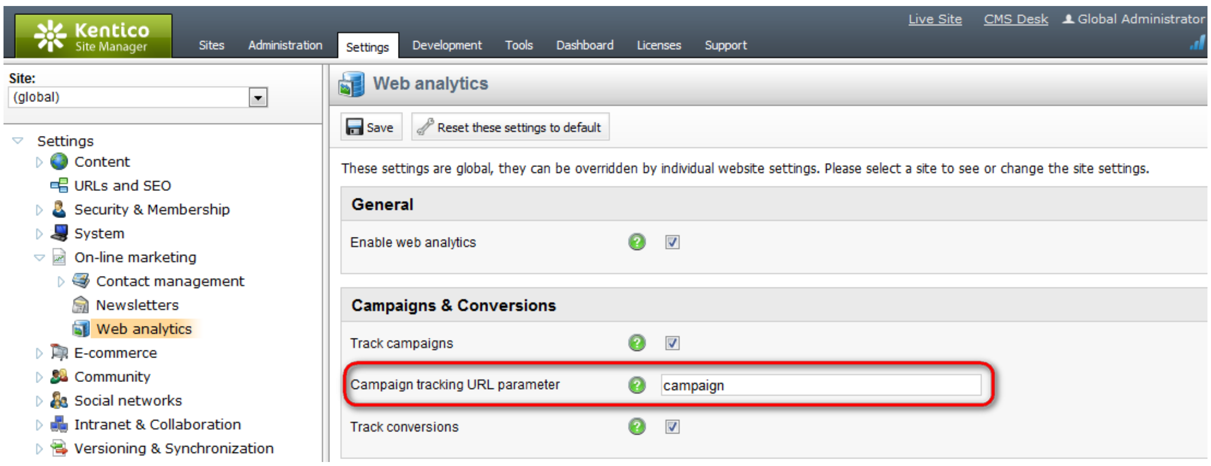
Return to the campaign list and add a third campaign with the following names:

- **Campaign display name**: E-mail marketing
- **Campaign name**: Emails

Now that some campaign tracking objects are prepared, it is necessary to ensure that visitors who access the website through marketing campaigns are correctly identified and assigned to the appropriate object. There are three different ways how this can be done.

URL parameter

This method utilizes a URL query string parameter to indicate that the traffic source is a campaign. All you need to do is enter the name of the parameter that you wish to use into the **Site Manager -> Settings -> On-line marketing -> Web analytics -> Campaign tracking URL parameter** setting.

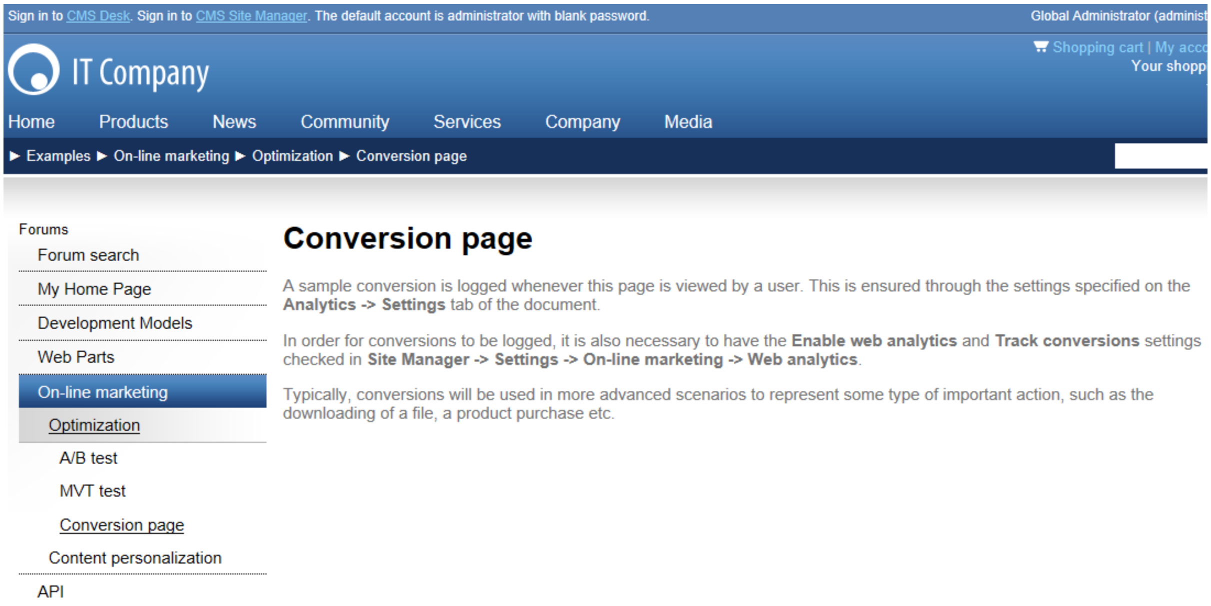


For now, leave the setting with its default value (*campaign*).

Open a new tab in your browser and enter the URL of your site's Home page into the address bar. Include the campaign tracking parameter and set its value to **samplecampaign** (the name of the default campaign):

http://localhost/KenticoCMS/Home.aspx?campaign=samplecampaign

You will have to adjust the URL according to the domain of your website. You may also need to **Close** the preview mode of the page to get to the actual live site. The Home page will be opened as usual, but the system will recognize that you have arrived via a campaign. Scroll down to the bottom of the page and click the **Examples** link under the **Where to learn more?** section. On this page, use the menu on the left to reach **On-line marketing -> Optimization -> Conversion page**, which will cause a sample conversion to be logged.



Because the *Sample campaign* is currently active in your browser, the conversion will also be recorded in the campaign statistics. You can try reloading the page several times to log multiple conversions.

In a real-world scenario, you would need to ensure that all links to your website placed in the given campaign's marketing materials contain the tracking parameter with the name of the appropriate campaign object as the value. This type of campaign tracking is not limited to a specific document and may be used when linking to any page on the website.

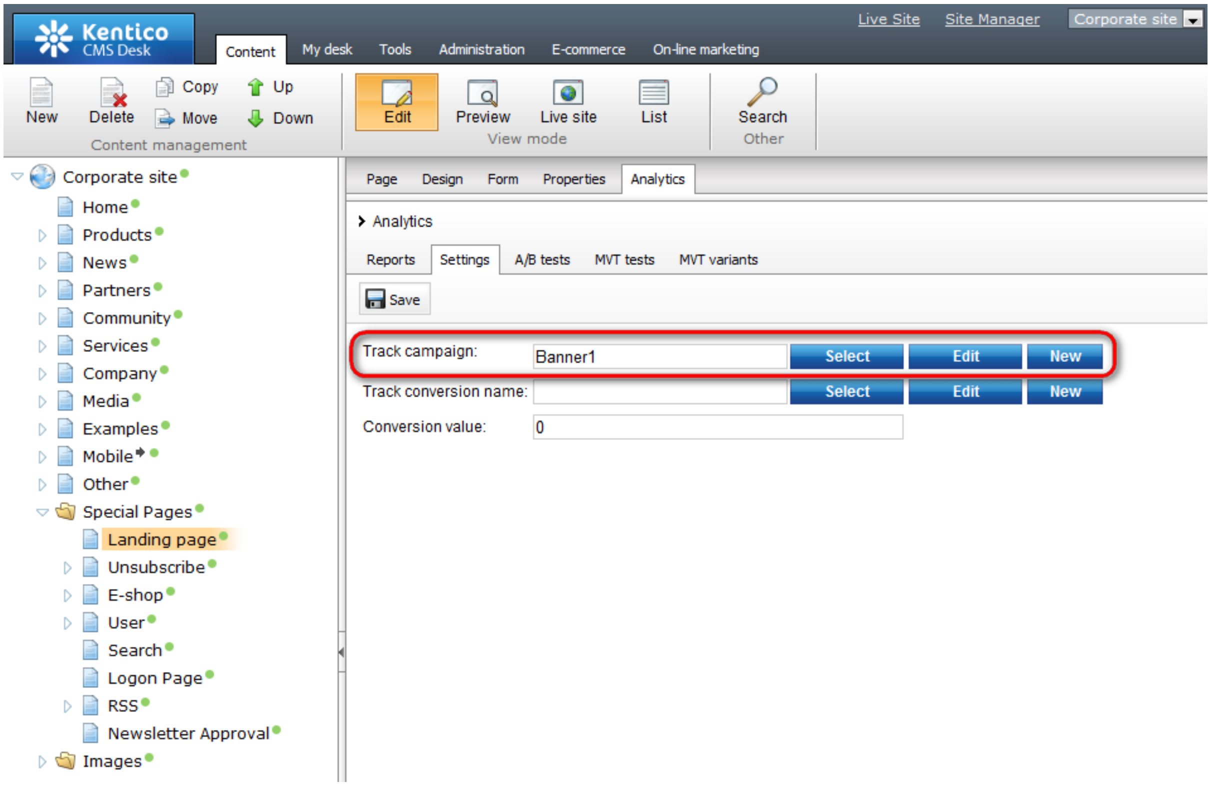
Document-specific campaigns

Alternatively, you can configure a document to behave as a campaign landing page. When this type of page is viewed by a user, a specified campaign will automatically be assigned.

To ensure accurate campaign tracking, it is recommended to create a special page that will not be available in the website's standard navigation, since the campaign will be assigned to all users who view this page (and repeated views by the same user will also be logged as campaign visits). When using this approach, all of the campaign's links need to be directed at this special landing page.

Log back in to **CMS Desk**, select the **Special Pages** folder in the content tree and create a **New** page under it named *Landing page*. You can choose the **Create a blank** page option in the template selection dialog, since the content of the page is not important for the purposes of this example.

Select the new document and switch to its **Analytics -> Settings** tab. Here, click the **Select** button next to the **Track campaign** field and choose **Banner campaign** in the displayed dialog. Then click the **Save** button.



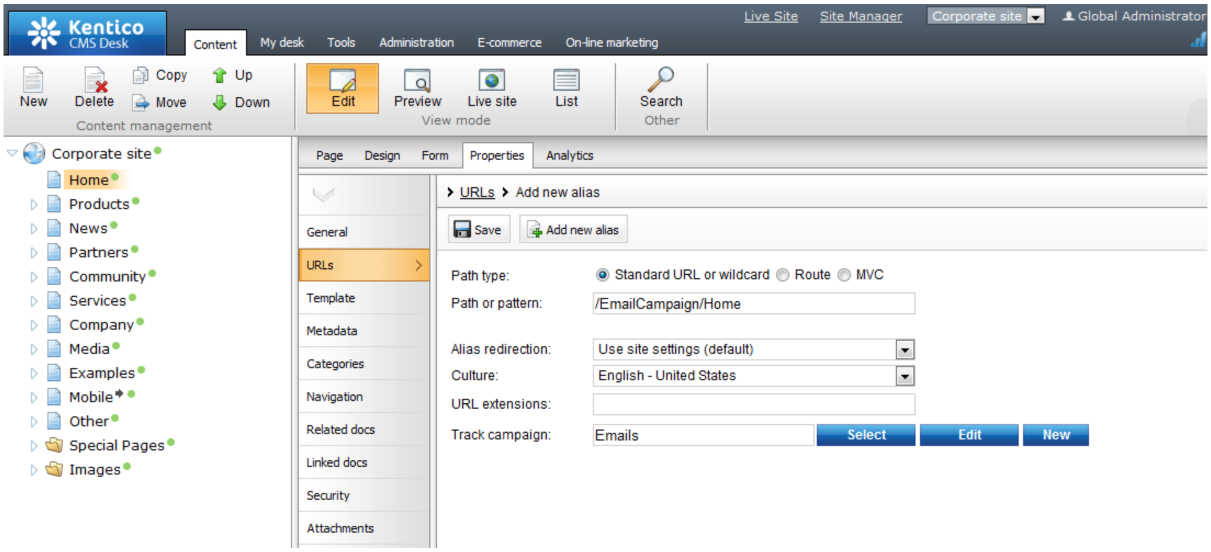
Now open this new page on the live site (you can access it through its URL: *<domain>/SpecialPages/Landing-Page.aspx*). The *Banner campaign* will become active, replacing the previously assigned campaign. Navigate to **Examples -> On-line marketing -> Optimization -> Conversion page** again, so that a conversion is logged for this campaign as well.

Campaigns for document aliases

You can also define campaign tracking for a page's specific Document alias. This approach is useful if you wish for the campaign to link visitors to an easily accessible page, but want to avoid using a query string parameter. The campaign will only be assigned to users who access the document through this alias.

Select the **Home** page from the content tree in **CMS Desk** and go to **Properties -> URLs**. Here, click on **Add new alias** and configure the alias's settings as shown below:

- **Path type**: Standard URL or wildcard
- **Path or pattern**: /EmailCampaign/Home
- **Track campaign**: select the *E-mail marketing* campaign



Click **Save** and the alias will be created.

To activate the campaign, open the Home page of the site through the URL of this alias (*<domain>/EmailCampaign/Home.aspx*). This will override the previous campaign and you can visit the sample **Conversion page** again.

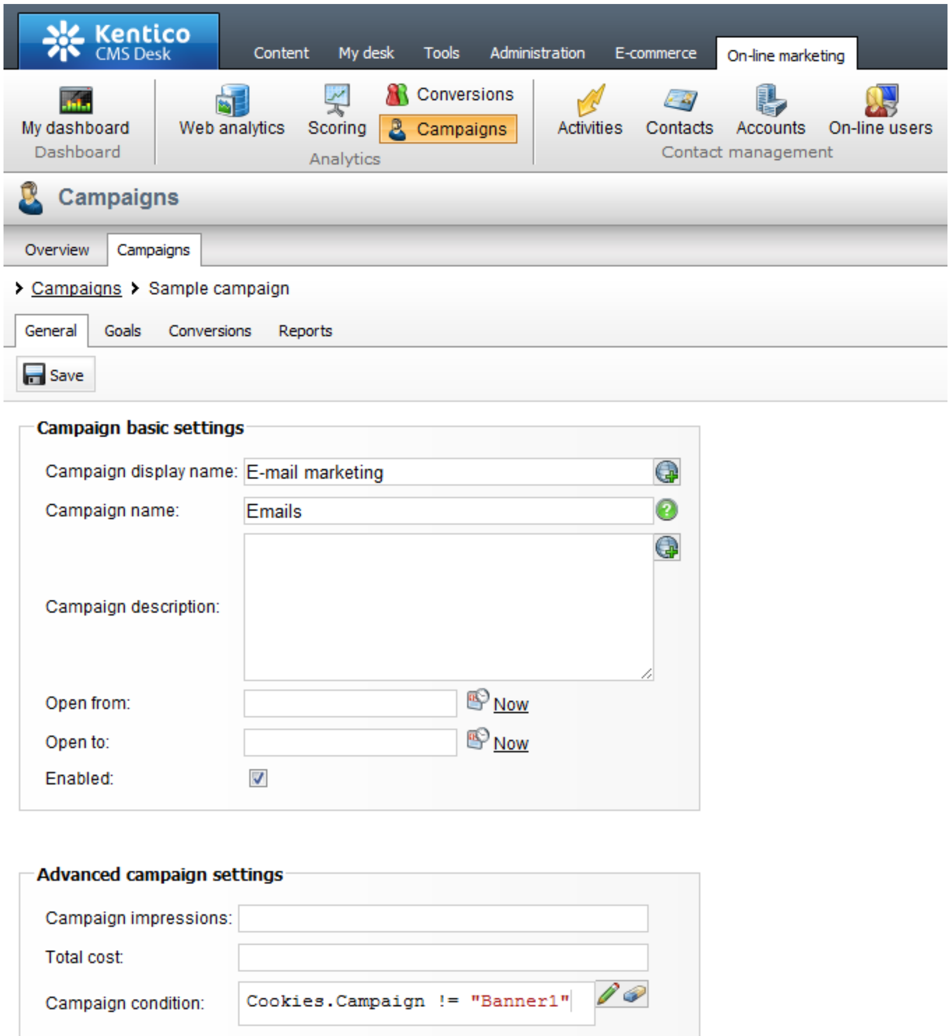
The campaign tracking methods are listed in the order of their priority. So if a campaign is specified for a document via **Settings -> Analytics**, but the page is also accessed via a URL containing a tracking parameter, the campaign specified via the parameter will be assigned to the given visitor.

Continue in the Evaluating campaigns topic to see how you can view the results of these steps. Keep in mind that this sample scenario only serves to quickly demonstrate the functionality of campaigns. In the case of a real campaign, visitors would not need to enter URLs manually, but would simply follow the appropriate links and the system would keep track of more practically defined conversions.

Advanced campaign settings

As you may have noticed, there are some additional configuration options that may be specified when editing campaigns in the **CMS Desk -> On-line marketing** interface.

On the **General** tab, the **Campaign impressions** and **Total cost** values may be used to help evaluate the campaign and calculate its goals. An important property is the **Campaign condition**, which allows you to set a macro condition that must be fulfilled in order for the campaign to be assigned to visitors.



For example: *Cookies.Campaign != "Banner1"*

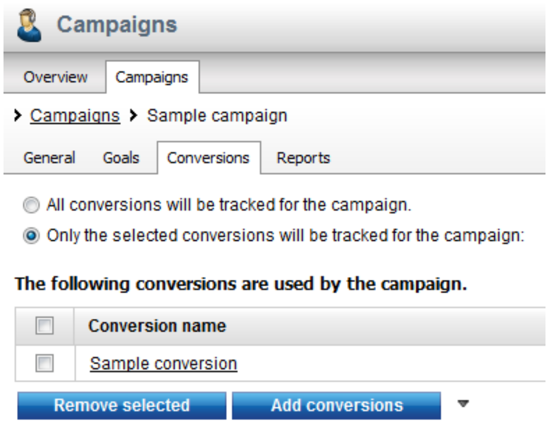
This sample condition checks the value of the *Campaign* cookie and prevents the edited campaign (E-mail marketing) from being assigned to visitors who are already being tracked by the **Banner1** campaign. In practice, this would mean that if a visitor was brought to the site through a banner and then later clicked on a link in a marketing e-mail, the banner campaign would remain active instead of being overwritten.

You can write any other type of condition according to your specific requirements. Through the edit icon (), you can open the Macro condition editor, which allows you to build conditions through a graphical interface. For more information about the available macro options, please refer to Internals and API -> On-line marketing macros.

If you switch to the **Conversions** tab, you can configure which Conversions should be tracked by the campaign. There are two possible options:

- **All conversions will be tracked for the campaign** - if chosen, any conversion hit logged on the entire website will be included in the campaign's statistics.
- **Only the selected conversions will be tracked for the campaign -** this option allows you to limit

which conversions should be tracked. Specific conversions may be assigned to the campaign by clicking the *Add conversions* button and choosing from the list in the displayed dialog.



In both cases, only conversions performed by visitors who arrive on the website as a result of the given campaign will be logged.

Automatic campaign creation

When a visitor comes to the website through an undefined campaign (typically specified via the tracking URL parameter), it will be ignored by default and no campaign tracking will be performed.

If you wish to change this behaviour, edit your application's **web.config** file and add the following key to the **/configuration/appSettings** section:

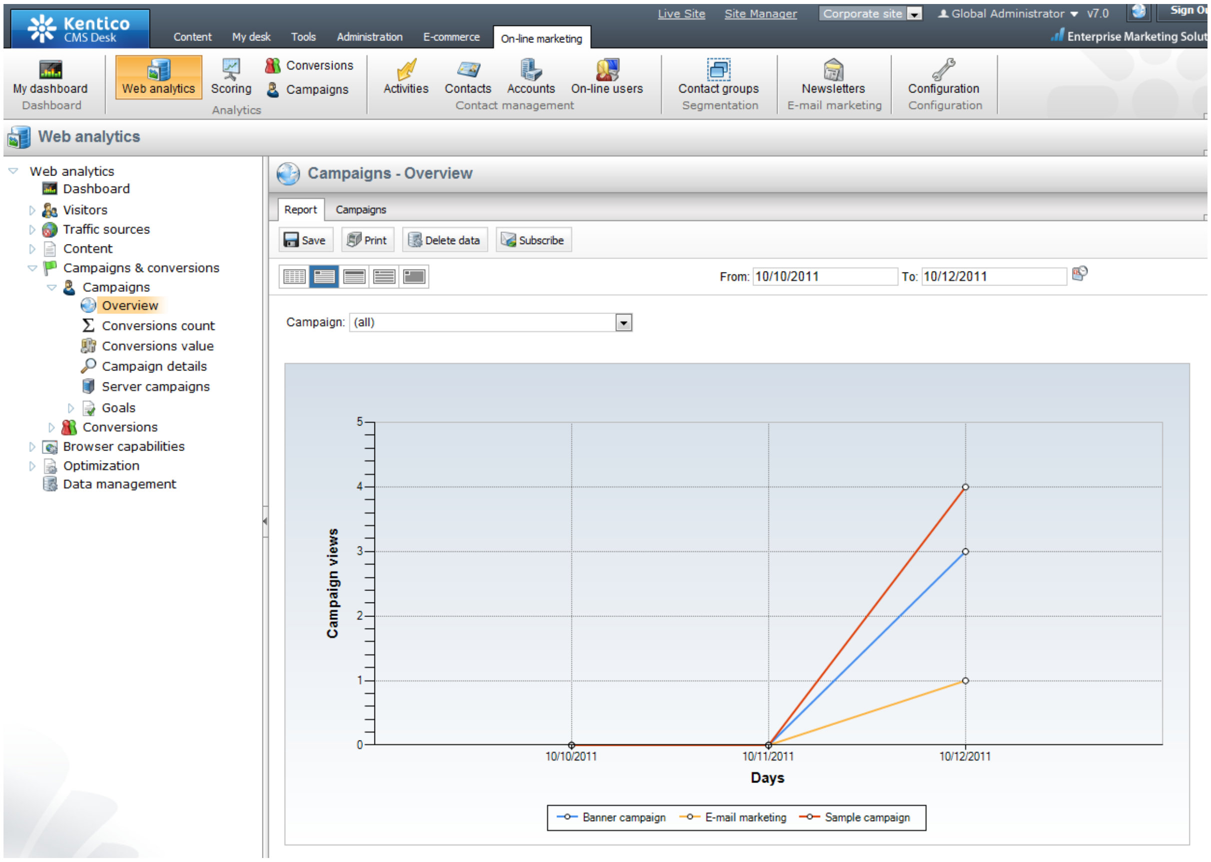
```
<add key="CMSEnableAutomaticCampaignCreate" value="true" />
```

Once this is done, unknown campaigns will automatically be added to the list of campaign objects for the given website.

Campaigns created this way will be enabled by default and active for an unlimited time interval, but they can be configured using the same approach as mentioned previously.

5.3 Evaluating campaigns

The statistics logged for campaigns may be viewed using various types of web analytics reports. To access these reports, go to **CMS Desk -> On-line marketing -> Web analytics** and expand the **Campaigns & conversions -> Campaigns** category.



Here you can check the results of the sample campaigns created according to the instructions in the Managing campaigns (Example) topic. Because of the way the analytics log is processed, you may have to wait approximately one minute before conversion hits or campaign visits are reflected in the reports.

Campaign reports use the same basic format as all other web analytics reports, so please refer to Web analytics -> Using web analytics reports for details about the actions that can be used to filter or otherwise manage the displayed data. The following types of reports are available for campaigns:

Overview	Displays the amount of hits that the website received as a result of the selected campaign(s). It contains a line chart that shows the number of visits recorded over time and a table with the total amount of page views generated by the given campaigns during the entire time period.
Conversions count	Displays the number of conversion hits that were performed by users who arrived on the website through the selected campaign. This report includes two types of charts, one that shows the progress of the amount of conversion hits recorded during the specified time period and another with detailed statistics for individual units of time according to the selected report type (hours, days, months etc.).
Conversions value	Displays the sum of the conversion values generated by users who arrived on the website through the selected campaign.

	This report includes two types of charts, one that shows the progress of the conversion values recorded during the specified time period and another with detailed statistics for individual units of time according to the selected report type.
Campaign details	Displays the values of the following campaign metrics: • **Visits** - the amount of hits that the website received as a result of the given campaign. • **Conversions** - the number of conversions that were performed by users who arrived on the site through the campaign. • **Conversions rate** - the percentage of visitors that performed a conversion. This can be higher than *100%* if the average visitor generated more than one conversion. • **Conversions value** - the total sum of the values recorded for all of the campaign's conversions. • **Conversions value per visit** - the average conversion value contributed by the campaign's visitors. • **Total cost** - shows the total cost that was specified for the given campaign. • **ROI** - the rate of investment, calculated as the sum of the campaign's conversion values divided by the total cost. This statistic is accurate only if your conversion values are set to reflect the income generated by the matching conversions. You can either choose to view all campaigns defined for the website and compare their values, or select a specific campaign and analyze the statistics logged for individual types of conversions.
Server campaigns	May be used to compare the results of campaigns created for different sites in the system. You can choose to display the *Views*, *Conversion count* or *Conversion value* statistics for the campaigns belonging under the selected site.

Campaign goals

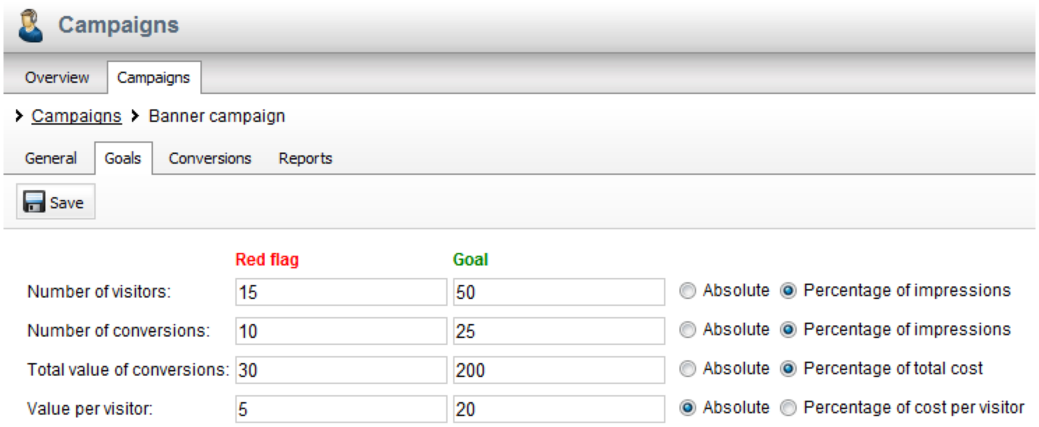
In addition to the data provided by the reports listed above, you can optionally specify goals that should be achieved by each campaign and then compare them with the actual results.

To enter a campaign's goals, select one of the reports and switch to the **Campaigns** tab, which contains the campaign management interface (the same one that is available under the **Campaigns** item in the main on-line marketing menu). Next, edit (✎) a campaign. On the **General** tab, the following two properties are available among the advanced settings:

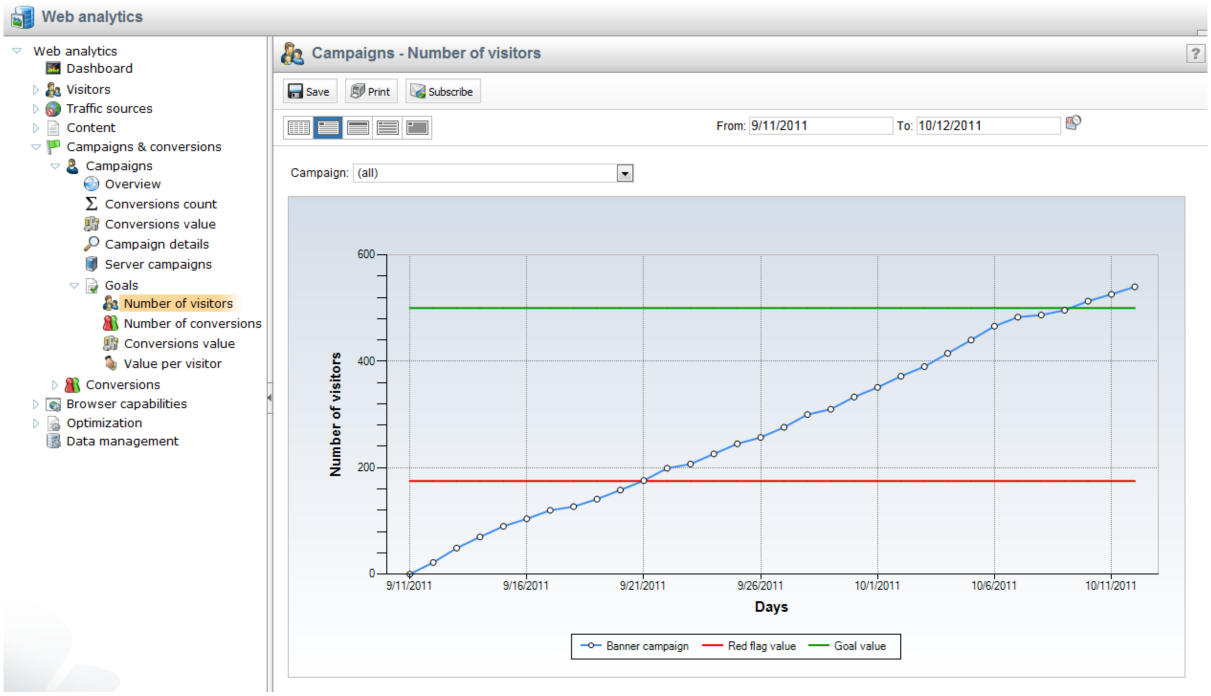
- **Campaign impressions** - this field can be used to specify how many people were targeted by the given marketing campaign. For example, if you sent marketing e-mails containing a link to the website to ten thousand people, the amount of impressions would be *10000*.
- **Total cost** - allows you to manually enter the total cost of the given marketing campaign. This can be used to determine whether the campaign was a success and when calculating the campaign's goals.

Now switch to the **Goals** tab, where you can configure the following types of target values for the campaign:

- **Number of visitors** - sets how many visitors should be brought to the website by the campaign. It can either be specified directly as an *Absolute* number, or as a percentage of the campaign's *Impressions* according to the value set on the *General* tab.
- **Number of conversions** - specifies the expected amount of conversions performed by users who visit the website as a result of the campaign. It can either be specified directly as an *Absolute* number, or as a percentage of the campaign's *Impressions*.
- **Total value of conversions** - sets a target number for the sum of all conversion values logged as a result of the campaign. It can either be specified directly as an *Absolute* number, or as a percentage of the campaign's *Total cost*.
- **Value per visitor** - this indicator allows you to specify the average conversion value that should be generated by a single campaign visitor. It is calculated as the campaign's *Total value of conversions* divided by its *Number of visitors*. The value can either be specified directly as an *Absolute* number, or as a percentage of the campaign's *Cost per visitor* (i.e. the *Total cost* divided by the *Number of Visitors*).



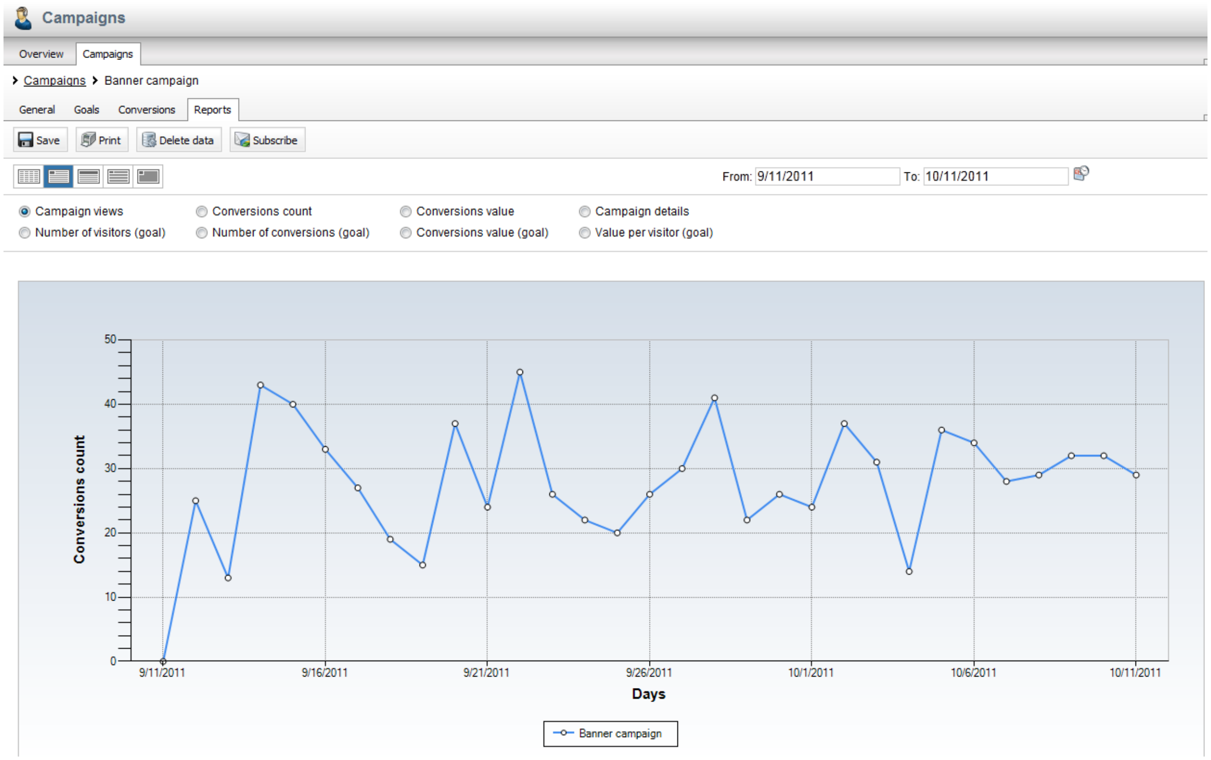
Each goal may have two values. The **Red flag** may be used to set a number that must be reached in order for the campaign to be at least partially successful (campaigns below this value are considered to have failed). The **Goal** value sets a target that should ideally be achieved by the campaign. The progress that individual campaigns make towards their goals can then be followed using the corresponding reports under the **Goals** sub-category in the web analytics tree.



The red flag and goal values configured for the selected statistic are displayed in the reports as red and green lines, so you can easily see the current status of the specified campaign.

Reports for individual campaigns

Any of the reports described above may also be viewed when editing a campaign in **CMS Desk -> On-line marketing -> Campaigns** on the **Reports** tab. You can choose a specific report via the radio buttons at the top of the page.



Only the data logged for the given campaign is displayed here.

Part VI

Contact management

6 Contact management

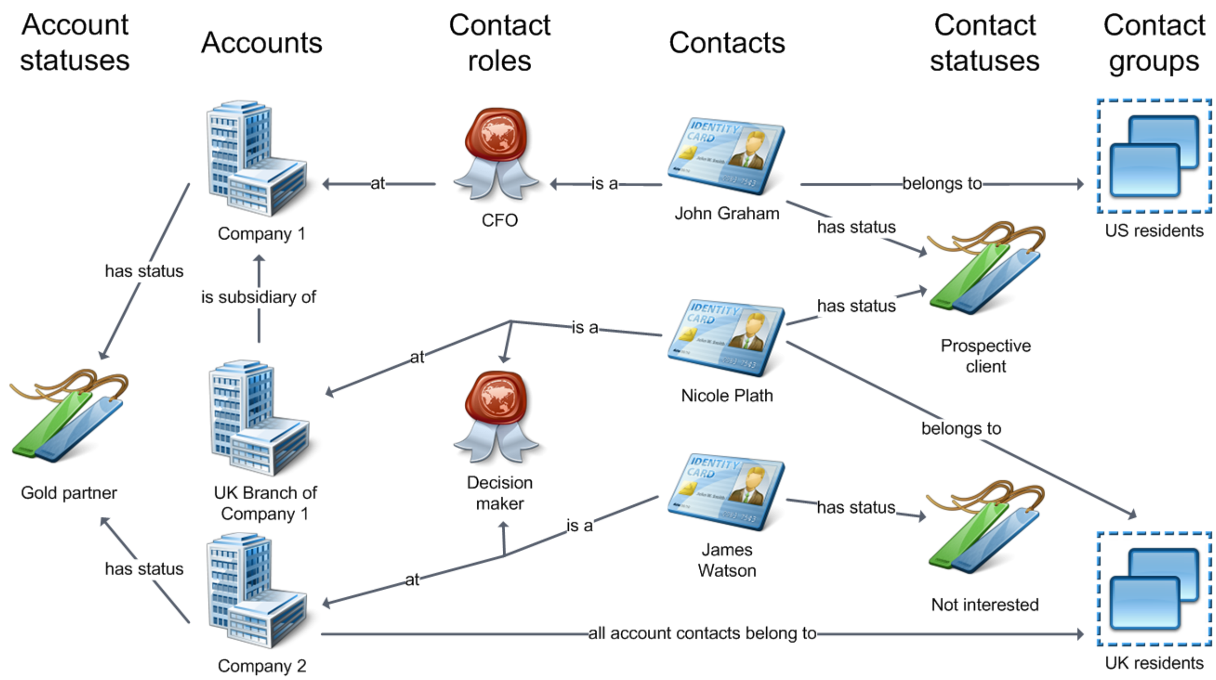
6.1 Overview

The Contact management module is an on-line marketing tool which provides a centralized overview of all users who visit your website. It can be used to gather information about them and about the activities that they perform on the website. The data gathered by the module can be used to track campaign results, analyze client behavior, optimize content, identify potential buyers and perform other marketing-related actions.

To get started with the module, it is important to get familiar with the following basic terms represented by objects of the module. Click the links to learn more about the particular objects, their management, use and configuration.

- Contact - a single person who visits the website and about whom information is gathered by the module.
- Account - a company or other commercial or non-commercial body where a contact operates. Individual contacts can belong to multiple accounts and an account can be a subsidiary of another account.
- Contact role - the role which a contact has in an account. It is typically a job position or a type of competence which the contact has within the account. One contact can have only one role in one account.
- Contact status - individual contacts can be labeled with pre-defined statuses. One contact can be labeled with only one status at a time.
- Account status - individual accounts can be labeled with pre-defined statuses, similarly as contacts, while there are separate sets of statuses for contacts and accounts. One account can be labeled with only one status at a time.
- Contact group - taxonomy units which enable you to sort contacts into groups. Contacts can be added to groups either manually, or automatically based on dynamically evaluated macro conditions. It is also possible to add an account to a contact group, which makes all contacts belonging to the account belong to the contact group as well.

The sample diagram below depicts how relationships between the objects may look in a real world scenario. It shows three contacts who operate as different roles in three accounts. They are grouped in two contact groups and both contacts and accounts are labeled with statuses.



In certain situations, it may happen that you have multiple contacts or accounts in the system and get to the conclusion that they actually represent one object. In the Merging contacts and accounts sub-chapter, you can learn how to get rid of such duplicate contacts or accounts by merging them into a single object.

Another important part of the Contact management module are Activities. Activities are actions that a contact performs on the website, e.g. visit a page, vote in a poll, add a product to their shopping cart or wishlist, etc. These activities can be tracked and logged so that you can view a log of all activities that individual contacts performed. You can deduce conclusions about the contacts from the activities that they performed and the logged activities can also be used for contact evaluation by the Scoring module.

Near the end of this chapter, you can find the Settings sub-chapter. In it, you can learn how to enable or disable various parts of the module and configure its functionality.

Finally, the Database separation sub-chapter describes how to transfer and host all data related to contact management in a different database than the one used for the main application.

6.2 Contacts

6.2.1 Overview

Contacts represent visitors of the website and store marketing-related information about them. Contacts cover both anonymous visitors (identified by IP address or cookie) and regular registered users or customers (identified by username or e-mail address). The system automatically gathers data about contacts based on the actions and input of the associated users.

Kentico EMS allows you to:

- Assign statuses to contacts (for example *VIP* contacts)
- Associate contacts with accounts (companies or other bodies where the contact operates) and

specific roles within accounts
- Group contacts into segments together with other related contacts
- Track and analyze the activities of contacts on the website
- Define automated processes for communicating with contacts or performing other actions in the system

Topics:

- Managing contacts
- Viewing on-line contacts
- Mapping fields to contact attributes
- Example: Creating a sample contact

6.2.2 Managing contacts

In this topic you will learn how to manage contacts. Contacts can be managed in:

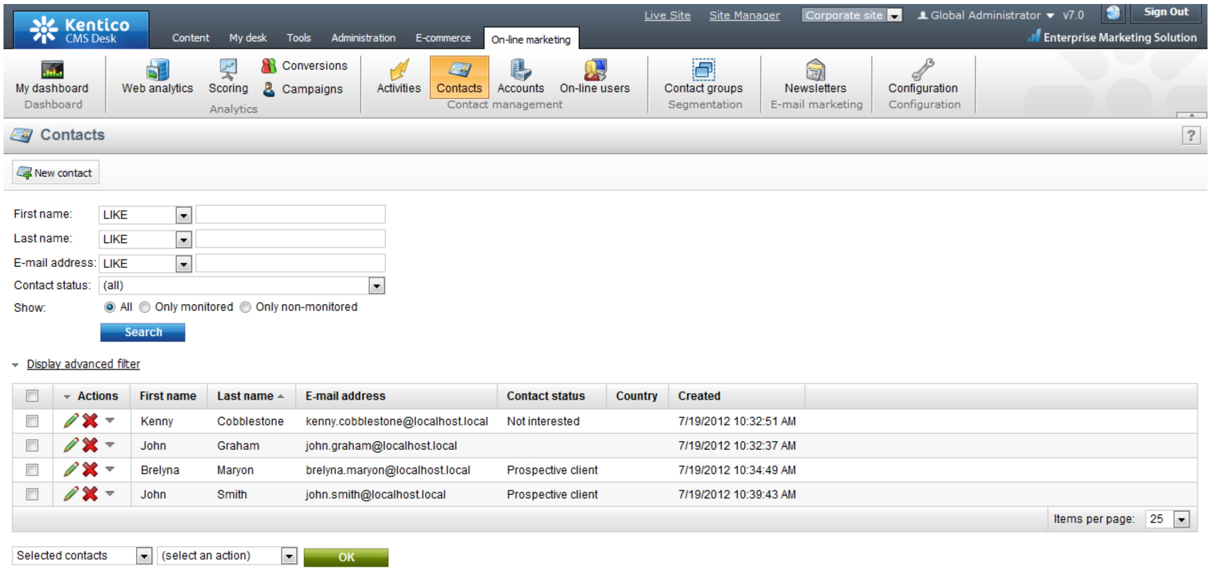
- **CMS Desk -> On-line marketing -> Contacts**
- **Site Manager -> Tools -> Contact management -> Contacts**

The user interface in CMS Desk allows management of contacts that belong to the currently edited website. The one in Site Manager allows management of contacts belonging to all websites running in the system. If global contacts are enabled, they can be managed in both sections.

On the initial page of the user interface, you can see a list of contacts available on the current site. You may need to display only contacts matching certain search criteria (e.g. first name, last name, e-mail etc.). To do so, choose either the simplified or advanced filter, enter the required values and click **Search**.

Individual contacts can be **Edited** () or **Deleted** () and you can also perform **Other actions** such as **Export**, **Backup** or **Restore**. New contacts can be added by clicking the **New contact** button.

Using the pair of drop-down lists below the list, you can perform certain contact management tasks with multiple contacts at once, specifically adding contacts to a contact group, deleting and merging contacts or changing their status. To do this, choose either *Selected contacts* or *All contacts* from the first drop-down list, the required action from the second one and click **OK**.



If Editing ( ) a contact, the following tabs are available:

- General tab
- Custom fields tab
- Accounts tab
- Membership tab
- Activities tab
- IPs tab
- Contact groups tab
- Scoring tab
- Merge tab
- Data.com tab

General tab

On this tab you can view and edit the basic attributes of the selected contact. Once you have modified the values as required, click **Save** to save the contact.

General

- **First name** - the first name of the contact.
- **Middle name** - the middle name of the contact.
- **Title before** - allows to define a contact title used before the name.
- **Last name** - the last name of the contact.
- **Salutation** - allows to define a contact salutation.
- **Title after** - allows to define a contact title used after the name.

Personal info

- **Birthday** - the birthday of the contact. Enter the value manually, use the **Calendar** ( ) or use the **Now** link.
- **Gender** - the gender of the contact.
- **Job title** - the job title of the contact.
- **Created** - indicates when the contact was created.

Contact settings

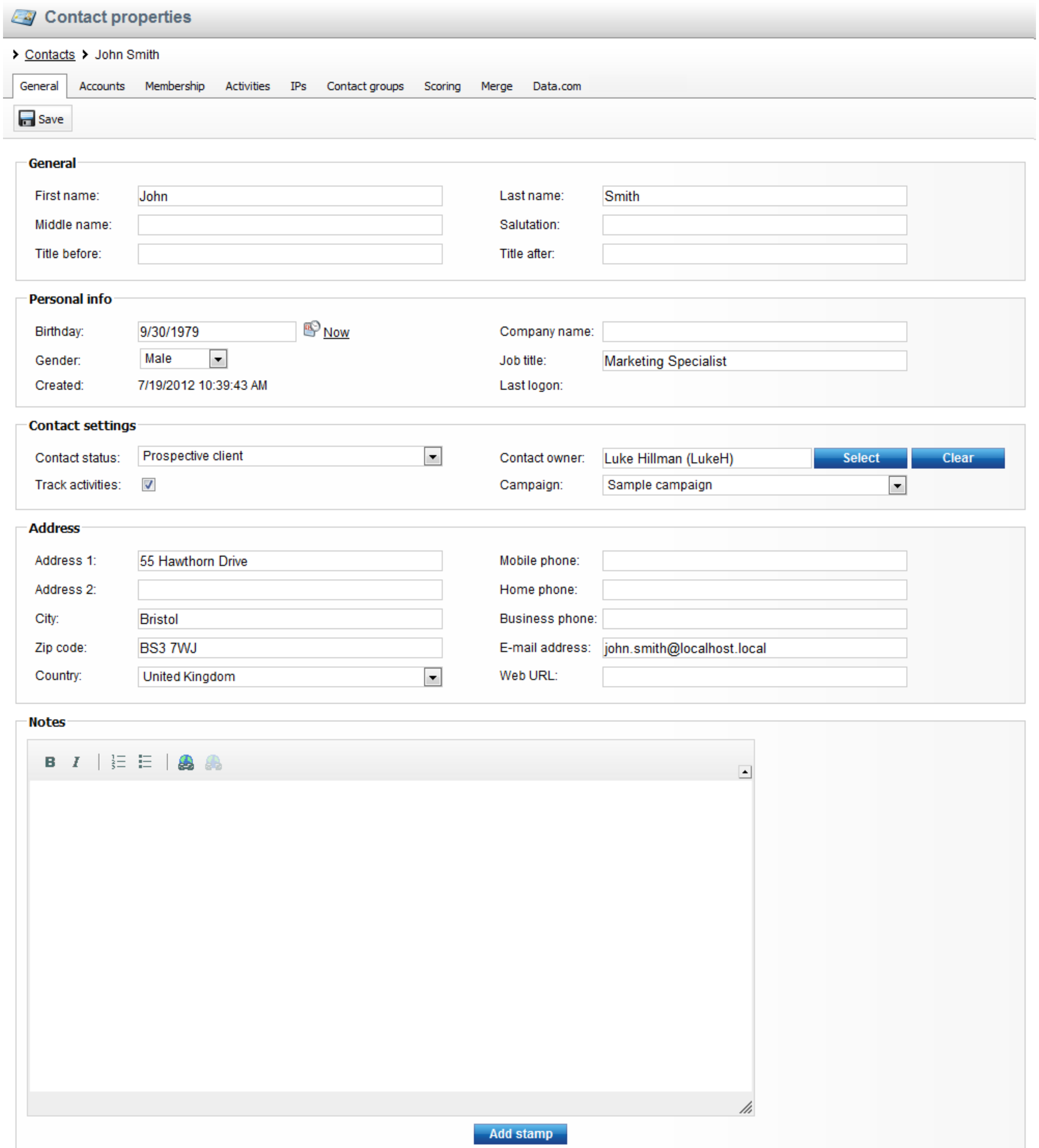
- **Contact status** - allows to select a contact status. If defined, the contact can be labeled with a contact status, e.g. a V.I.P. contact.
- **Track activities** - if enabled, allows to track contact activities.
- **Contact owner** - allows to define a contact owner, i.e. a user responsible for management of the given contact.
- **Campaign** - allows to choose a Campaign to be associated with the current contact.

Address

- **Address 1** - the first line of the contact's address.
- **Address 2** - the second line of the contact's address.
- **City** - allows to define the city where the contact has their address.
- **Zip code** - allows to define the zip code of the area where the contact has their address.
- **Country** - allows to define the country where the contact has their address.
- **Mobile phone** - the mobile phone number of the contact.
- **Home phone** - the home phone number of the contact.
- **Business phone** - the business phone number of the contact.
- **E-mail** - the e-mail address of the contact.
- **Web URL** - the web URL of the contact.

Notes - you can leave contact notes either as plain text or as text with basic formatting, and you can also add your stamp by clicking the **Add stamp** button.

Please note that if you are viewing properties of a merged contact, you can click on **Split from parent** to split the current contact from the parent one.



Contact data values

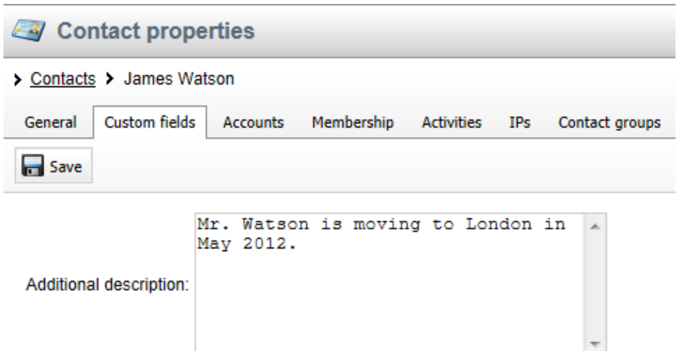
You do not need to enter the values of the attributes on the General tab manually for all contacts.

The system will automatically try to collect and update the data of each contact based on the information provided by the corresponding user on the website. Please see the

Mapping fields to contact attributes topic for more information.

Custom fields tab

On this tab you can enter values into the custom fields defined for contacts in **Site manager -> Development -> System tables -> Edit (** **)** ***Contact management - Contact*** on the **Fields** tab. When you have entered the values, click **Save** to confirm the changes.

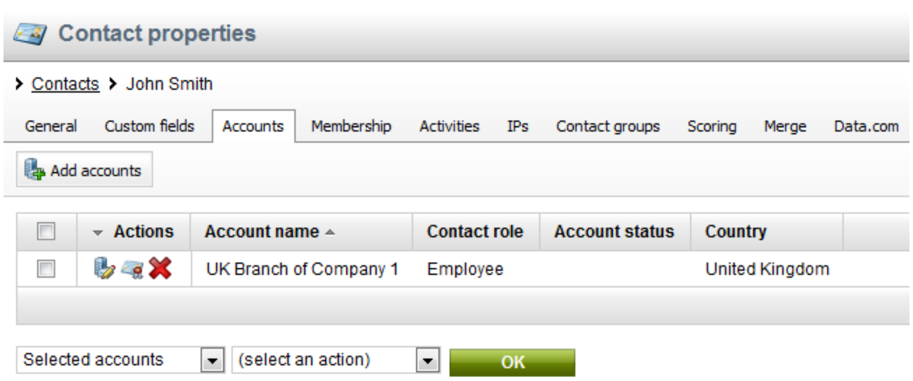


Accounts tab

On this tab you can see a list of accounts assigned to the current contact and you can also create accounts by clicking on **New account**. If you decide to do so, the **Select account** dialog opens, enabling you to assign selected accounts to the current contact; optionally, a contact role may be selected (if defined). Please note that if you edit a global contact, only global accounts are offered. Similarly, if you edit a site contact, only current site accounts are offered.

You can **View account details** ( ) of individual accounts and you can also **Select contact role** ( ) or **Remove** ( ) an account. The displayed list can be sorted by account name, contact role, account status and country.

Using the pair of drop-down lists below the list, you can perform certain management tasks with multiple accounts at once, specifically *Selecting contact role* or *Removing accounts*. To do this, choose either *Selected accounts* or *All accounts* from the first drop-down list, the required action from the second one and click **OK**.



Membership tab

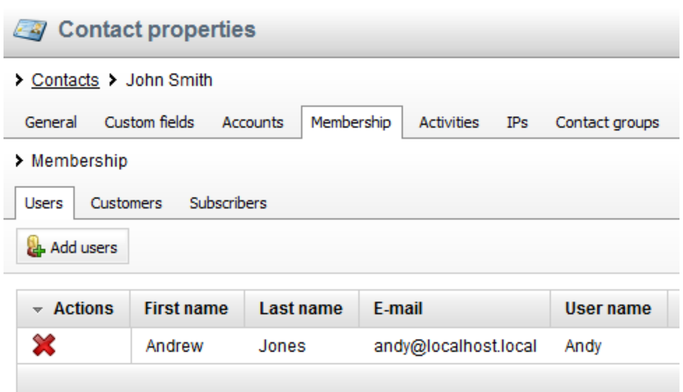
On this tab you can manage the current contact's memberships, i.e. users, e-commerce customers and newsletter subscribers that should be associated with the currently edited contact.

The following tabs are available under the **Membership** tab:

- **Users** - here you can see a list of users under the current contact and you can also manually **Add users**. Please note that the usual scenario is that a contact acts as one registered user. However, more users can be assigned to a contact as well.
- **Customers** - here you can see a list of customers under the current contact and you can also manually **Add customers**.
- **Subscribers** - here you can see a list of subscribers under the current contact and you can also manually **Add subscribers**.

On each tab, the listed items can be **Deleted** () and the list can be sorted e.g. by name and e-mail.

You may need to display only items matching certain search criteria (e.g. first name, last name, e-mail) on the respective tabs. To do so, enter the required values and click **Search**.

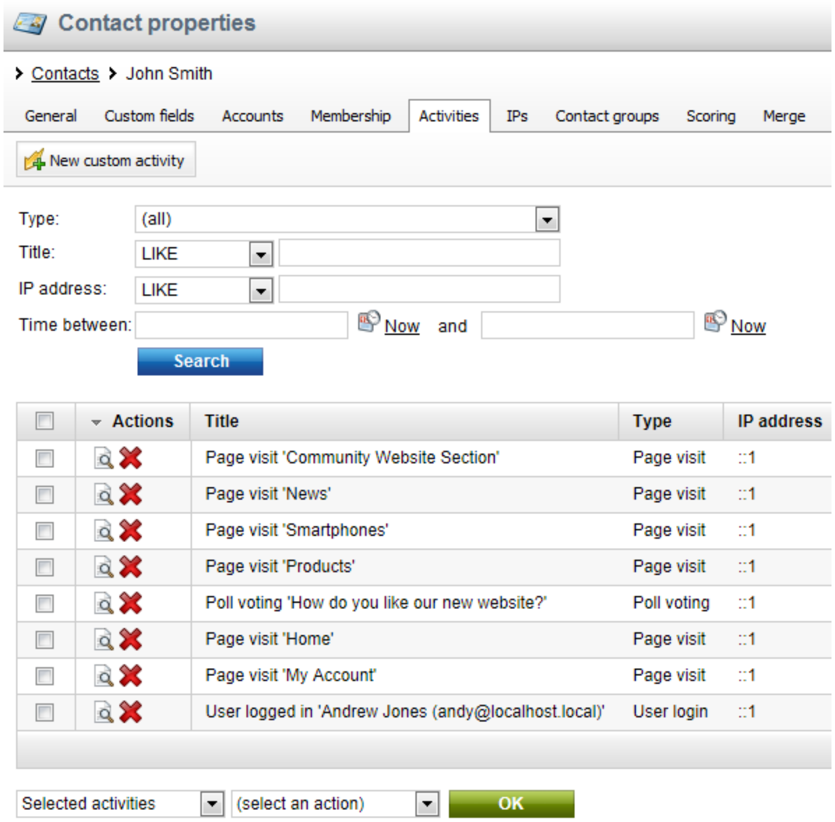


Activities tab

On this tab you can see a list of the current contact's activities, e.g. user registration, blog post subscription, poll voting etc. However, activities are logged only if the **Enable on-line marketing** setting is enabled in **Site Manager -> Settings -> On-line marketing** and the **Log activities** setting is enabled in **Site Manager -> Settings -> On-line marketing -> Contact management -> Activities**.

Individual activities can be **Viewed** () or **Deleted** (). The displayed list can be sorted by title, type, IP address and activity time.

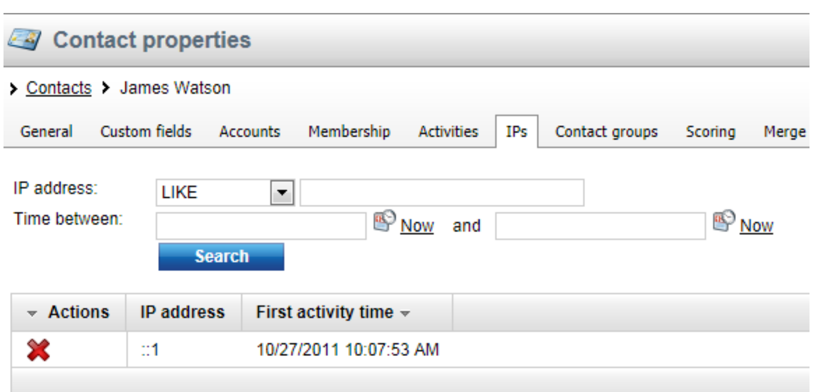
You may need to display only activities matching certain search criteria ,e.g., an activity type, title etc. To do so, fill in the fields as required and click **Search**.



IPs tab

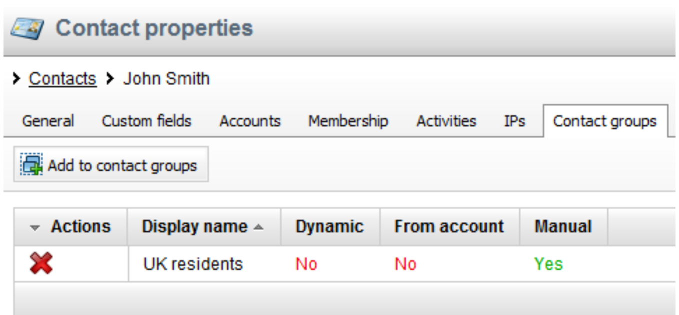
On this tab you can see a list of IP addresses from which the current contact accessed the website. The listed items can be **Deleted** ( ) and you can also **Export** (->) the items. The displayed list can be sorted by IP address and activity time.

You may need to display only IP addresses matching certain search criteria (IP address, time period). To do so, fill in the respective fields as required and click **Search**.



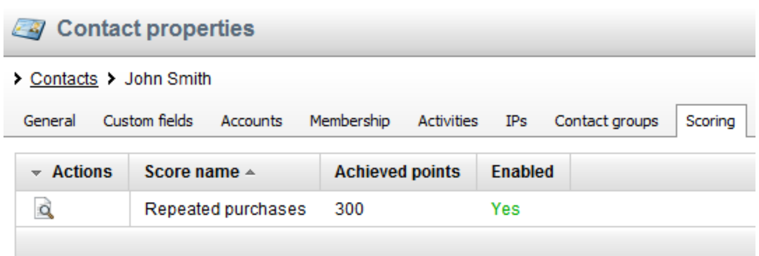
Contact groups tab

On this tab you can see a list of [contacts groups](#) to which the current contact is assigned. You can also add the contact to other contact groups by clicking **Add to contact groups**. Individual items can be **Removed** () from the list, which results in the current contact being removed from the respective contact group.



Scoring tab

On this tab you can see a list of scores in which the current contact achieved some points, together with the actual score point values. By clicking the **View** () icon, the respective score's management interface is opened in a pop-up window. The interface is identical to the one displayed when editing () a score in **CMS Desk -> On-line marketing -> Scoring**.



Merge tab

On this tab you can merge other contacts with the currently edited contact so that only the currently edited contact remains. This is useful in cases when you come to the conclusion that multiple contacts actually represent a single real-world person. On the tab, you can also view a list of contact that have been merged with it and split them back into separate contacts if needed. Detailed information on merging in Kentico CMS can be found in the Merging contacts and accounts chapter.

Data.com tab

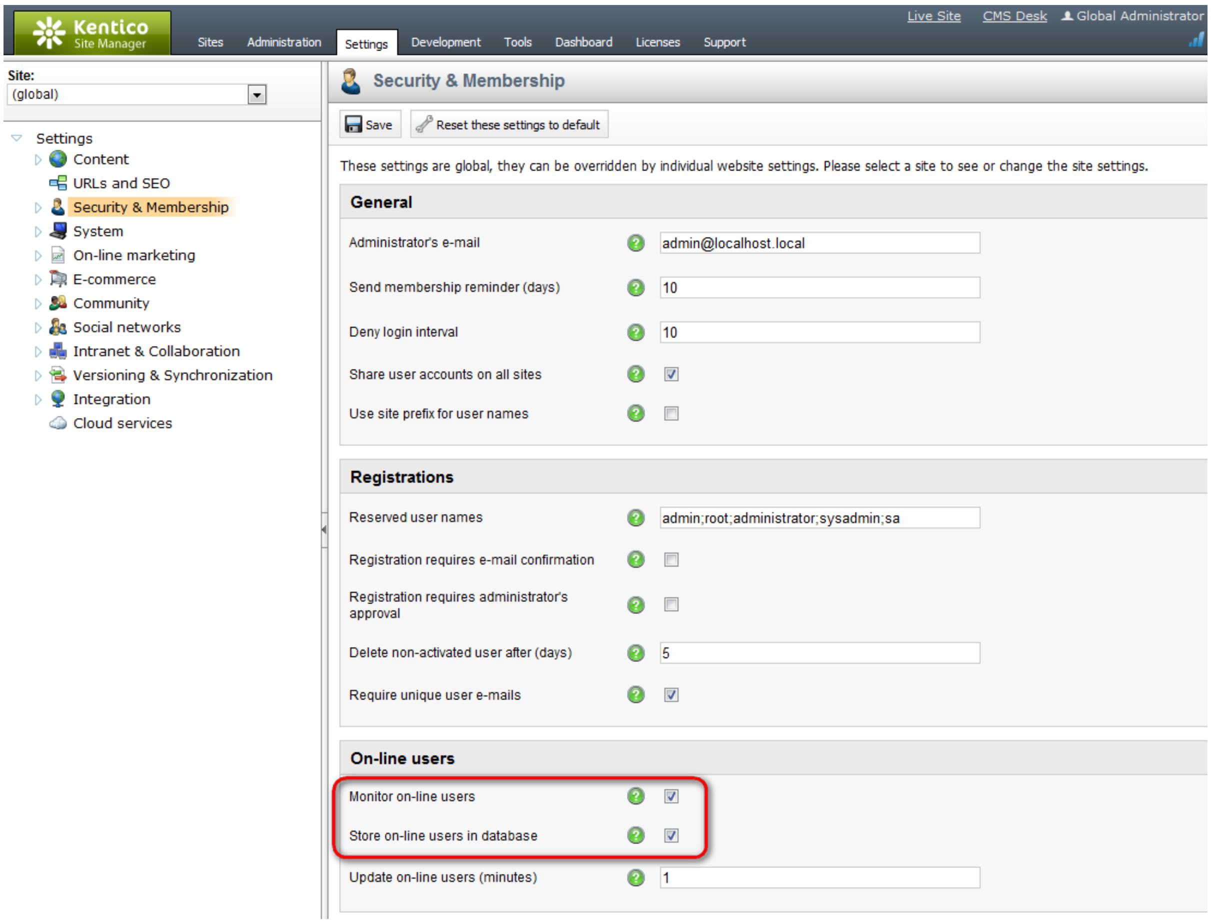
This tab allows you to compare the currently edited contact with matching contacts in the Data.com database and search for additional information. Data.com integration must be enabled and configured in order for this feature to be available. Refer to the Integration of 3rd party modules -> Data.com chapter to learn more.

6.2.3 Viewing on-line contacts

The information gathered by the contact management module may also be used to keep track of the users who are currently visiting the website. This allows you to monitor how many visitors the site has at any given time, check which pages are being viewed and manage the corresponding contact data as required.

To activate this feature, go to **Site Manager -> Settings -> Security & Membership** and enable both of the following settings:

- **Monitor on-line users**
- **Store on-line users in database**



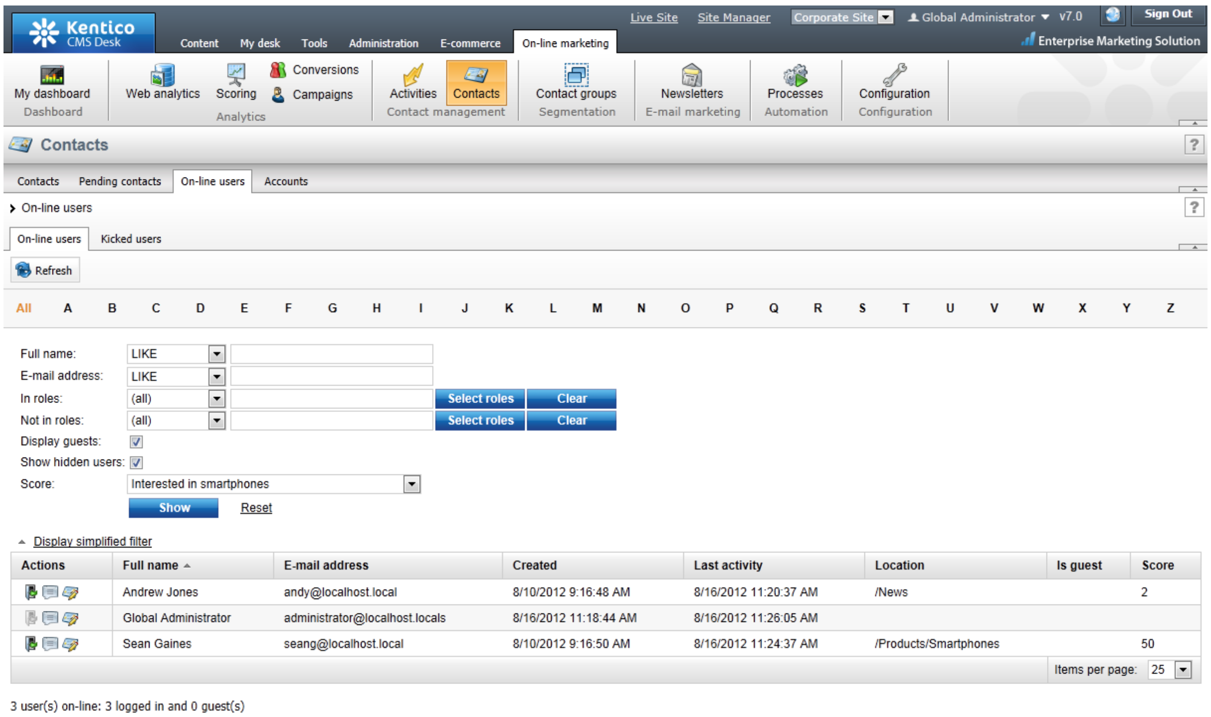
You can then access the list of current website visitors in **CMS Desk -> On-line marketing -> Contacts -> On-line users**.

The list provides basic information about each user, such as their name and e-mail address. For logged in users, this data is taken from the values entered during registration. If the information is unavailable, the system loads the values from the corresponding contact attributes. You can also see other data, such as the time of the user's last action and their current location (i.e. the path of the page that they are viewing).

The displayed users can be filtered according to their name, either by selecting one of the letters at the top of the page or by entering text into the search box. Click **Display advanced filter** to access more filtering options. This way, you can also limit which users are visible based on their e-mail address or the roles to which they belong. Additionally, several special options are available in the advanced filter:

- **Display guests** - if enabled, the list also shows individual visitors who are not logged in to the website. In the case of guests, the system fills in the name and e-mail information based on the data available for the corresponding contact. When disabled, only authenticated users are displayed.

- **Show hidden users** - determines whether the list includes users who are flagged as hidden (e.g. administrators or other internal user accounts).
- **Score** - by selecting a score, you can view the number of points that each on-line user (contact) has in the given score. To learn more about contact scoring, see the Scoring chapter.

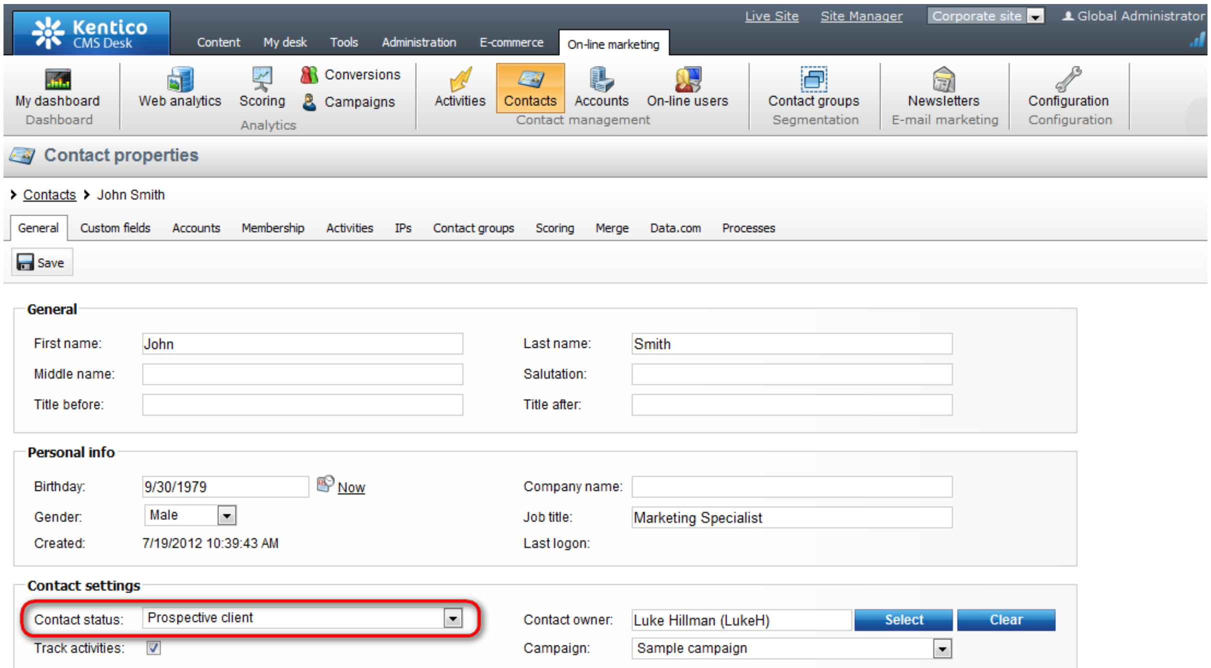


You may also perform the following actions for the displayed users:

- **Kick** - may be used to kick *authenticated* users from the website. This means that the user is logged out and will not be able to log back in for the duration specified in the *Site Manager -> Settings -> Security & Membership -> Deny login interval* setting. You can view a list of all users who are currently kicked out on the *Kicked users* tab.
- **Initiate chat** - allows you to start directly communicating with the given person through a chat window. To work correctly, support chat must be enabled for the website and supported by the page that the user is currently viewing. Please see the Modules -> Chat chapter in the Developer's Guide for more information.
- **View contact details** - can be used to edit the contact associated with the given user (the editing interface is the same as described in the Managing contacts topic).

6.2.4 Contact statuses

Contacts can be labeled with statuses in order to reflect their relationship to your organization or to indicate how they are perceived from the marketing point of view. Statuses can be assigned to individual contacts in the **Contact status** field on the General tab of each contact's editing interface. One contact can be labeled with only one status at a time.

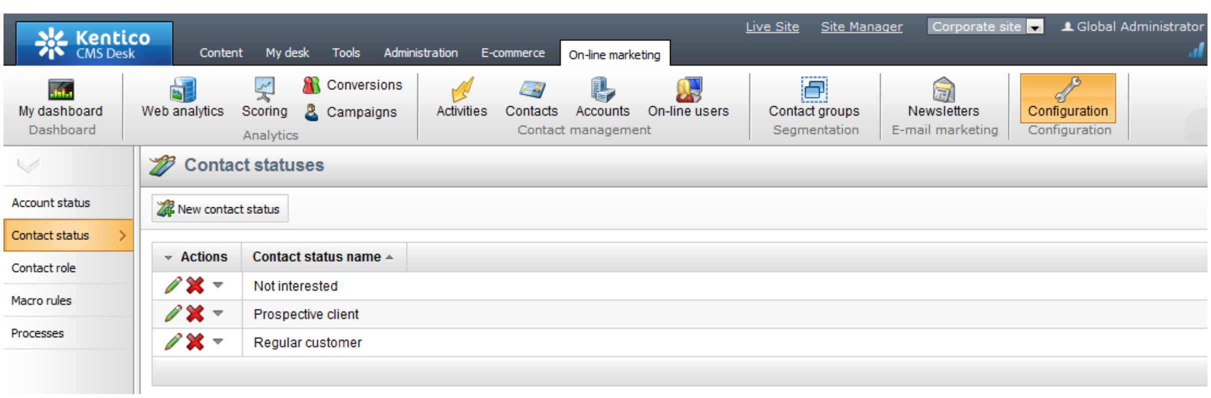


There is a predefined set of contact statuses for each website. The sets of statuses can be managed in the following sections of the user interface:

- **CMS Desk -> On-line marketing -> Configuration -> Contact status**
- **Site Manager -> Tools -> Contact management -> Configuration -> Contact status**

The user interface in CMS Desk allows management of contact statuses predefined for the currently edited website. The one in Site Manager allows management of contact statuses for all websites running in the system. If global contact statuses are enabled, they can be managed in both sections.

Properties of contact statuses listed in the mentioned sections of the user interface can be edited after clicking the **Edit** ( ) icon in a respective contact role's row. By clicking the **Delete** ( ) icon, you can delete the contact status so that it is no longer present and available in the system. The **Export** ( ) action available in the drop-down menu ( ) in each row enables you to export the respective status into a file and subsequently import it on another Kentico CMS instance.

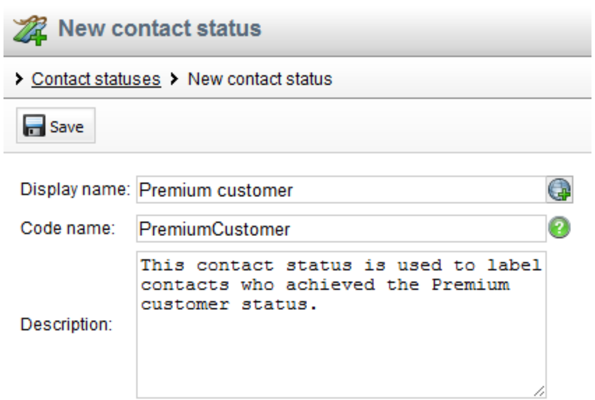


Creating a new contact status

If you want to add a new contact status to the set of listed statuses, click **New contact status**. Then, specify the following properties in the **New contact status** dialog:

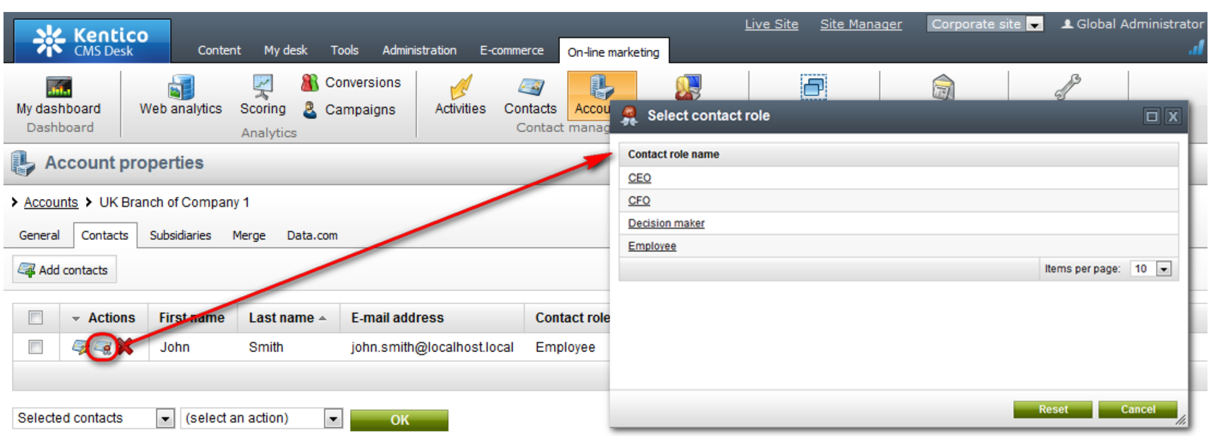
- **Display name** - name of the contact status used in the system's user interface.
- **Code name** - name of the contact status used in web project code.
- **Description** - text describing the contact status.

Once you have the properties specified, click **Save** to create the new status.



6.2.5 Contact roles

Contact roles enable you to specify the role of a contact in an account. It is typically a job position or a type of competence which a person has in the organization, e.g. *CEO*, *CFO*, *Decision maker*, etc. The role of a contact in an account can be specified after clicking the **Select contact role** ( ) icon on the Contacts tab of each account's editing interface. One contact can only have one role in one account.

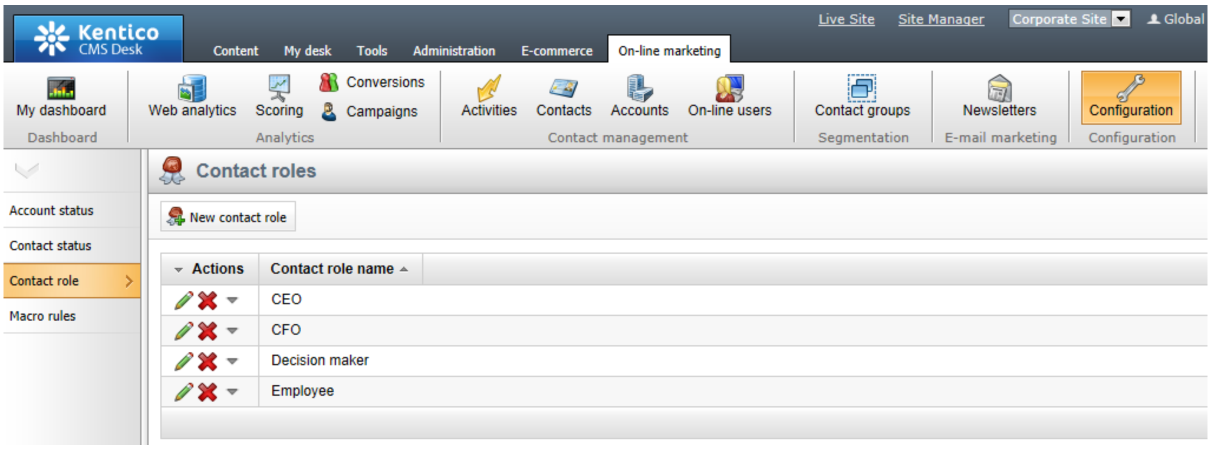


There is a pre-defined set of contact roles for each website. The sets of roles can be managed in the following sections of the system's user interface:

- **CMS Desk -> On-line marketing -> Configuration -> Contact role**
- **Site Manger -> Tools -> Contact management -> Configuration -> Contact role**

The user interface in CMS Desk allows management of contact roles pre-defined for the currently edited website. The one in Site Manager allows management of contact roles for all websites running in the system. If global contact roles are enabled, they can be managed in both sections.

Properties of contact roles listed in the mentioned sections of the user interface can be edited after clicking the **Edit** ( ) icon in a respective contact role's row. By clicking the **Delete** ( ) icon, you can delete the contact role so that it is no longer present and available in the system. The **Export** ( ) action available in the drop-down menu ( ) in each row enables you to export the respective status into a file and subsequently import it on another Kentico CMS instance.

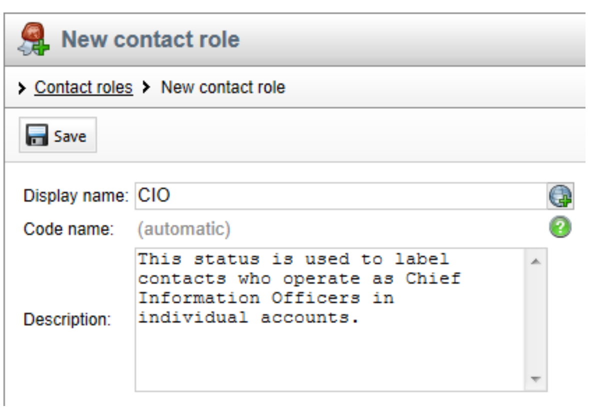


Creating a new contact role

If you want to add a new contact role to the set of listed roles, click the **New contact role** button above the list. After doing so, you need to specify the following properties in the **New contact role** dialog:

- **Display name** - name of the contact role used in the system's user interface.
- **Code name** - name of the contact role used as an identifier, for example in the API. You can leave the *(automatic)* option to have the system generate an appropriate code name based on the display name.
- **Description** - text describing the contact role.

Once you have the properties specified, click **Save** to create the new contact role.



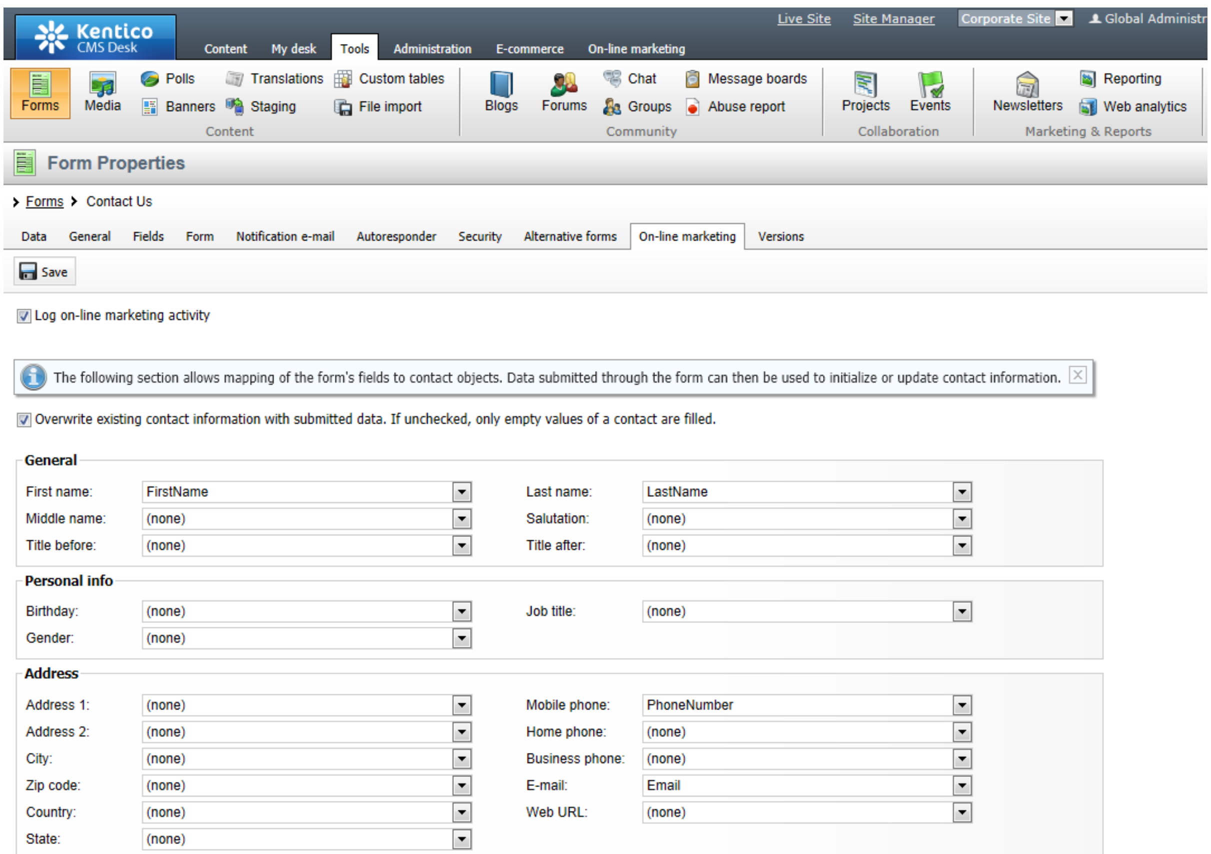
6.2.6 Mapping fields to contact attributes

When a new visitor arrives on the website for the first time, an anonymous contact without any data is created to represent them. If the user submits information on the site, for example during registration or by filling out a form, the system will automatically transfer the entered values into the data of the matching contact. The same logic may also be used to update the data of existing contacts if the corresponding user provides new information.

The purpose of field mapping is to precisely determine from where the values of individual contact attributes should be loaded. These mappings can be configured for various types of input forms and system objects.

On-line forms

In the case of on-line forms created using the Forms module, the related settings can be defined in **CMS Desk -> Tools -> Forms** by editing a given form and switching to its **On-line marketing** tab. Here you can assign specific form fields to contact attributes.



As you can see, the available options match the attributes that are offered when editing a contact on its **General** tab. For example, if the form has a field where users enter their e-mail address, you can choose this field for the **E-mail** attribute. When a user submits the form, the value of the field will automatically be saved as the e-mail address of the contact representing the given user.

The checkbox directly above the mapping settings is very important. If it is enabled, existing contact data will always be replaced by the submitted form values, otherwise only empty attributes will be filled.

System tables

In addition to on-line forms, the same mapping configuration options are available for various types of system objects that store user data. The default mappings for these objects should be suitable for most websites, but you can change them if you are using custom system fields or have some other type of specific requirement. It can be done by going to **Site Manager -> Development -> System tables** and editing one of the following system tables on its **On-line marketing** tab:

- User
- E-commerce - Customer
- Newsletter - Subscriber

For example, when a user registers on the live site or edits their profile settings, the system automatically updates the data of the corresponding contact according to the mappings set for the *User* system table. By default, the **First name** contact attribute is filled by the value of the *FirstName* field and so on. You can assign specific system fields to different contact attributes in the same way as for on-line forms.

The same also applies to the *Newsletter - Subscriber* and *E-commerce - Customer* tables, which are instead used when a contact fills out a newsletter subscription form or registers as a customer.

6.2.7 Example: Creating a sample contact

Here you will learn how to create a new contact.

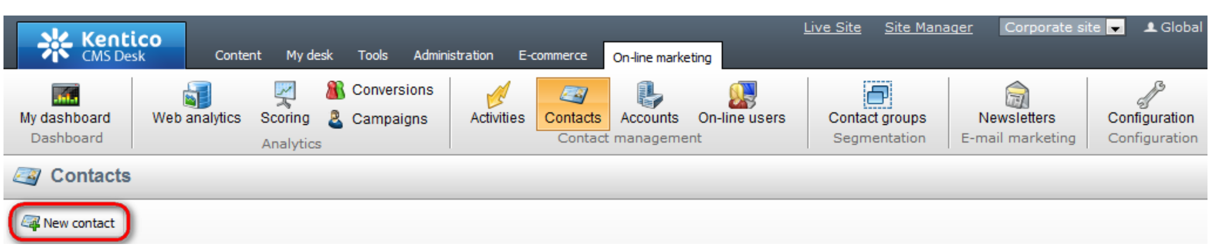
A new contact can be created automatically on the live site, e.g. if an anonymous visitor browses the website for some content or if a new user registers there, and manually in the administration interface.

On the live site

When an anonymous website visitor comes to the website for the first time, a contact is created for them automatically. When the same visitor registers on the live site as a new user, the contact's data is updated according to the registration details and it is associated with the given user account. An example of creating a new contact on the live site can be found in the Activities -> Example topic in the Contact management section of this guide.

In the administration interface

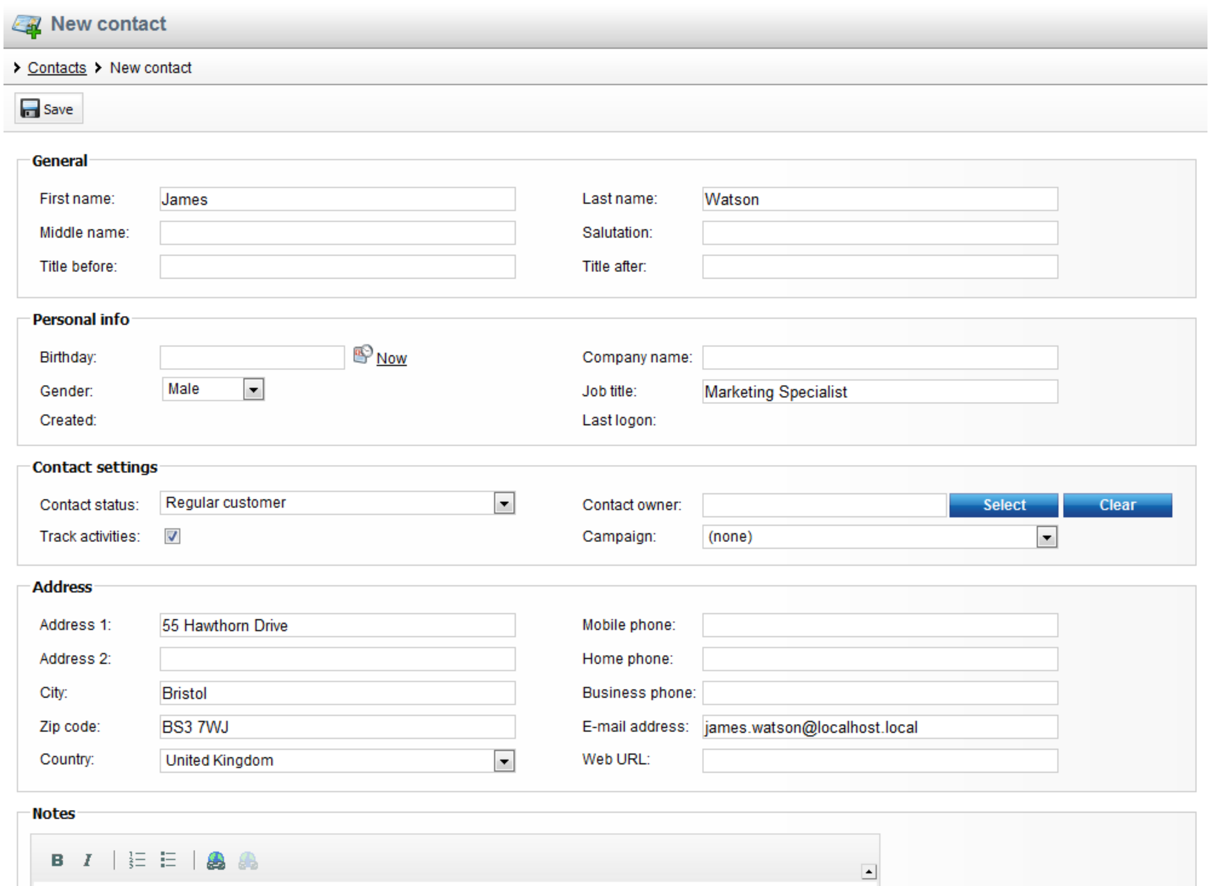
1. Sign in as an administrator, go to **CMS Desk -> On-line marketing -> Contacts** (alternatively, you can navigate to the **Contacts** tab in **Site Manager -> Tools -> Contact management**) and click **New contact**.



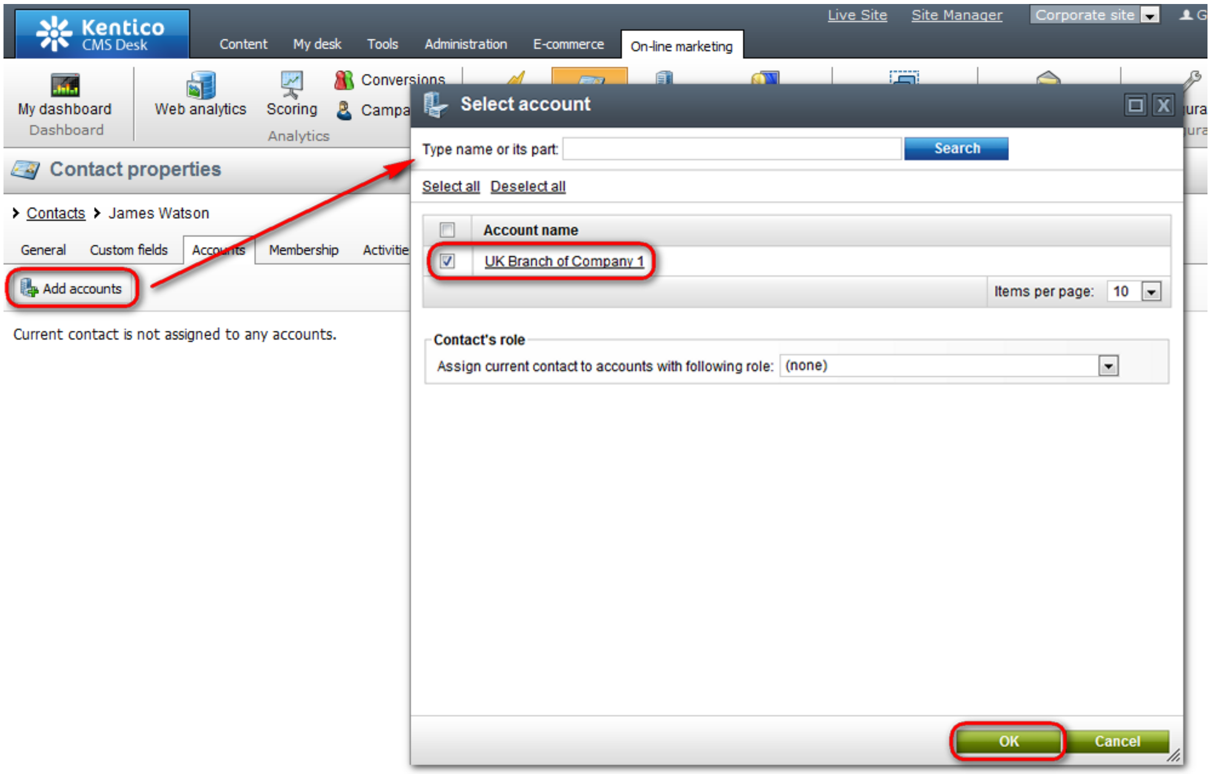
2. In the **New contact** dialog, enter the following properties:

- **First name:** James
- **Last name:** Watson
- **Contact status:** select a contact status, if no statuses were defined yet, please refer here to see how to do it
- **Gender:** Male
- **Job title:** Marketing specialist
- **Address 1:** 55 Hawthorn Drive
- **City:** Bristol
- **ZIP code:** BS3 7WJ
- **Country:** United Kingdom
- **E-mail:** james.watson@localhost.local

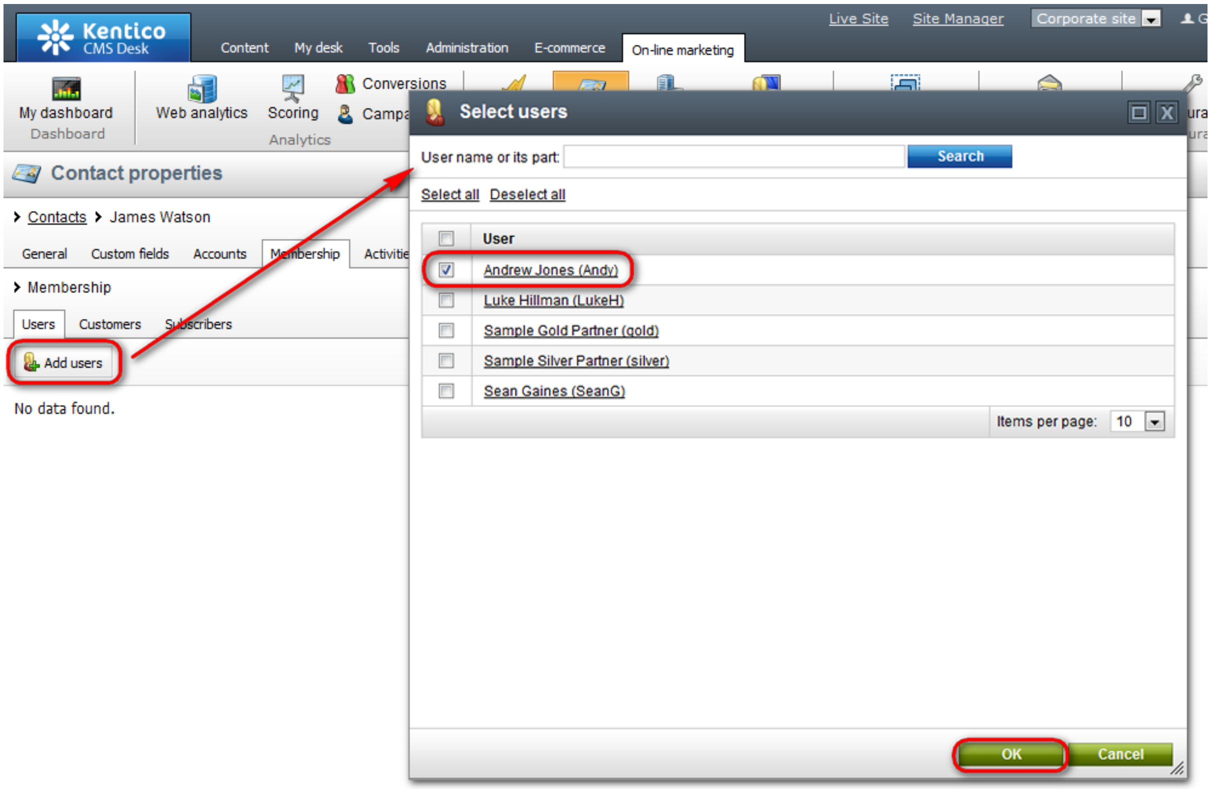
Once entered, click **Save** ( ) for the contact to be created.



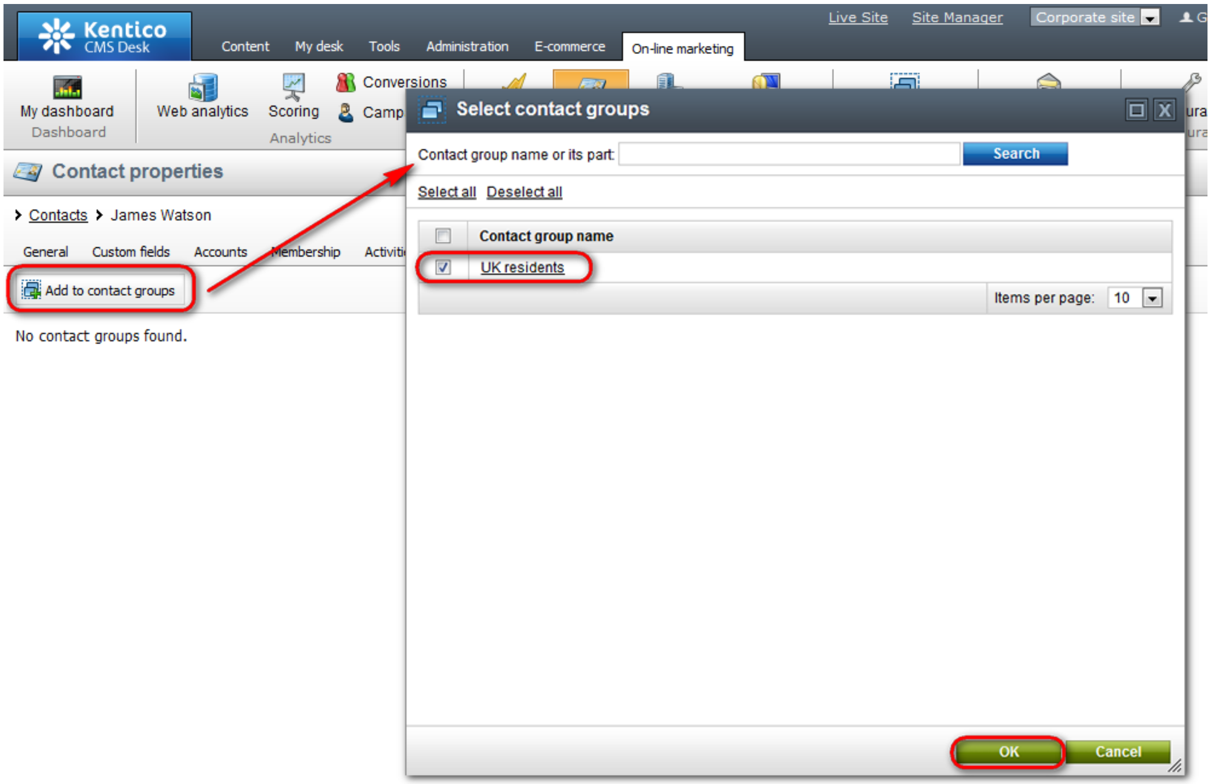
3. Optionally, switch to the **Accounts** tab, click on **Add accounts** and in the **Select account** dialog select accounts which should be assigned to the contact. Click **OK**. Please note that at least one account must be defined to be able to perform this step. If no accounts were defined yet, see this topic.



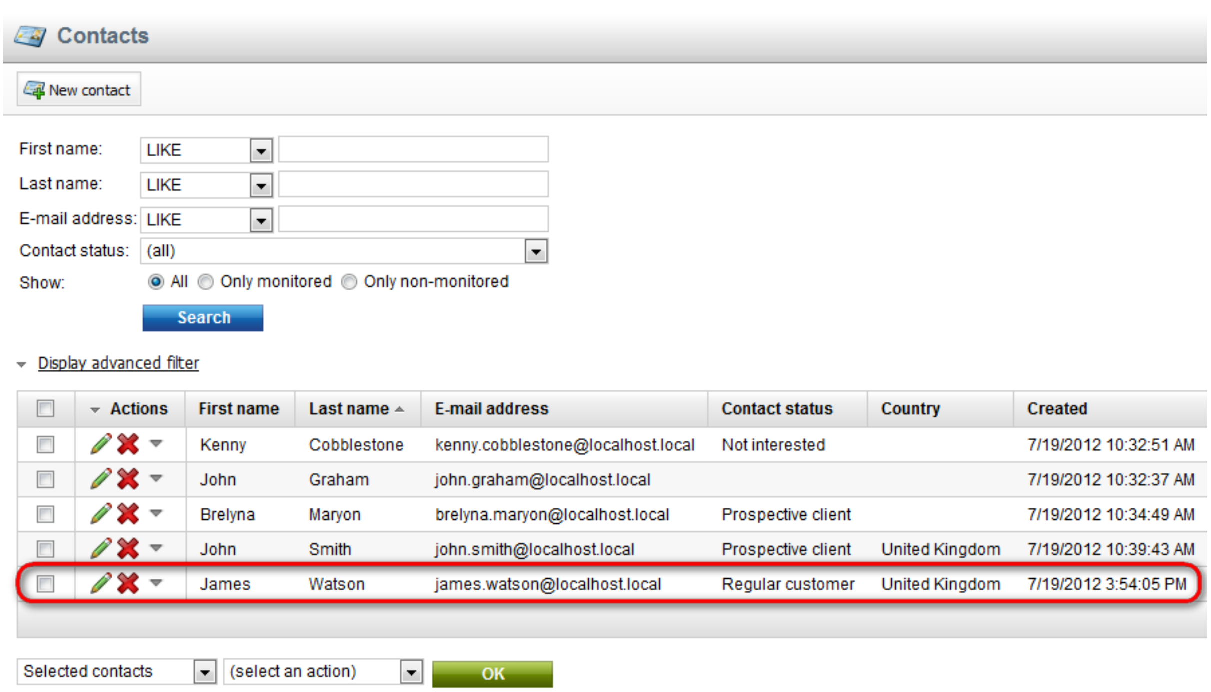
4. Optionally, switch to the **Membership** tab and on the respective tabs add users, customers or subscribers that should be associated with the new contact.



5. Optionally, switch to the **Contact groups** tab, click on **Add to contact groups** and in the **Select contact groups** dialog select contact groups to which the contact should be added. Click **OK**. Please note that at least one contact group must be defined to be able to perform this step. If no contact groups were defined yet, see this topic.



6. If you now view the list of contacts, you can see that the new contact is listed among other contacts.



6.3 Accounts

6.3.1 Overview

An account in Kentico CMS is an object representing a single organization, i.e. a company or other commercial or non-commercial body. The object can be site-specific or shared across more websites (global account). It contains multiple contacts, can have subsidiaries and can be assigned with an account status. Besides, stored as a system table, an account supports custom fields.

Further in this chapter, you will learn how to manage accounts and you will also see an example of adding a new account to your Contact management.

6.3.2 Managing accounts

In this topic you will learn how to manage accounts. Accounts can be managed in:

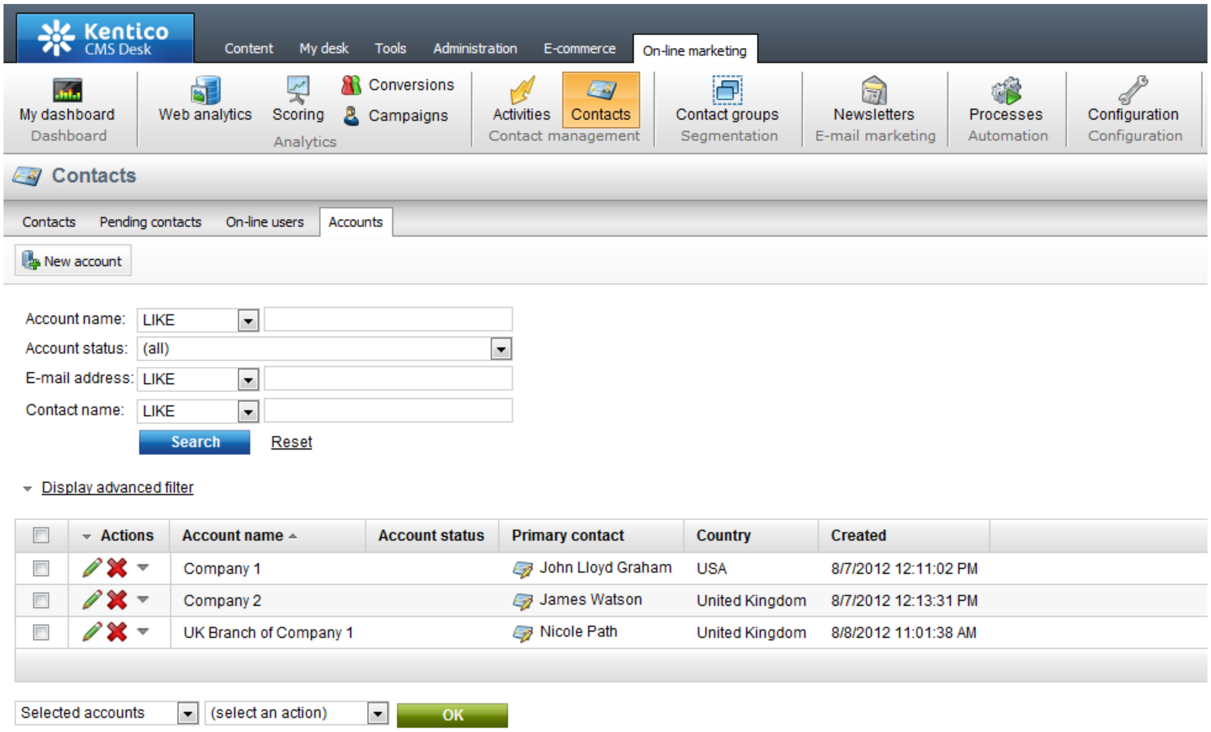
- **CMS Desk -> On-line marketing -> Contacts -> Accounts**
- **Site Manager -> Tools -> Contact management -> Accounts**

Here you can see a list of accounts available on the current site and you can also create new accounts by clicking **New account**. Please note that you can always create site-specific accounts and, if the use of global accounts is allowed, also accounts shared across more websites.

You may need to display only accounts matching certain search criteria (e.g. account name, account status, e-mail etc.). To do so, choose either the simplified or advanced filter, enter the required values and click **Search**.

Individual accounts can be **Edited** () or **Deleted** () and you can also perform **Other actions** such as **Export**, **Backup** or **Restore**. The displayed list can be sorted by account name, account status, primary contact and country.

Using the pair of drop-down lists below the list, you can perform certain contact management tasks with multiple accounts at once, specifically adding accounts to a contact group, deleting and merging accounts or changing the account status. To do this, choose either *Selected contacts* or *All contacts* from the first drop-down list, the required action from the second one and click **OK**.



When you edit an account, the following tabs are available to you:

- General tab
- Contacts tab
- Subsidiaries tab
- Merge tab

General tab

On this tab you can edit properties of a selected account. When you have modified the values, click **Save** to save the account.

General

- **Account name** - the name of the account. Accounts usually represent companies or other commercial or non-commercial bodies.
- **Account status** - allows to select an account status. If defined, the account can be labeled with an account status, e.g. a V.I.P. account.
- **Subsidiary of** - allows to select an account, i.e. company or other commercial or non-commercial body, of which the current account is a subsidiary.
- **Account owner** - allows to define an account owner, i.e. user responsible for management of the current account.

Address

- **Address line 1** - the first line of the account's address.
- **Address line 2** - the second line of the account's address.
- **City** - allows to define the city where the account has its address.
- **Zip code** - allows to define the zip code of the area where the account has its address.
- **Country** - allows to define the country where the account has its address.
- **Phone** - the phone number of the account.

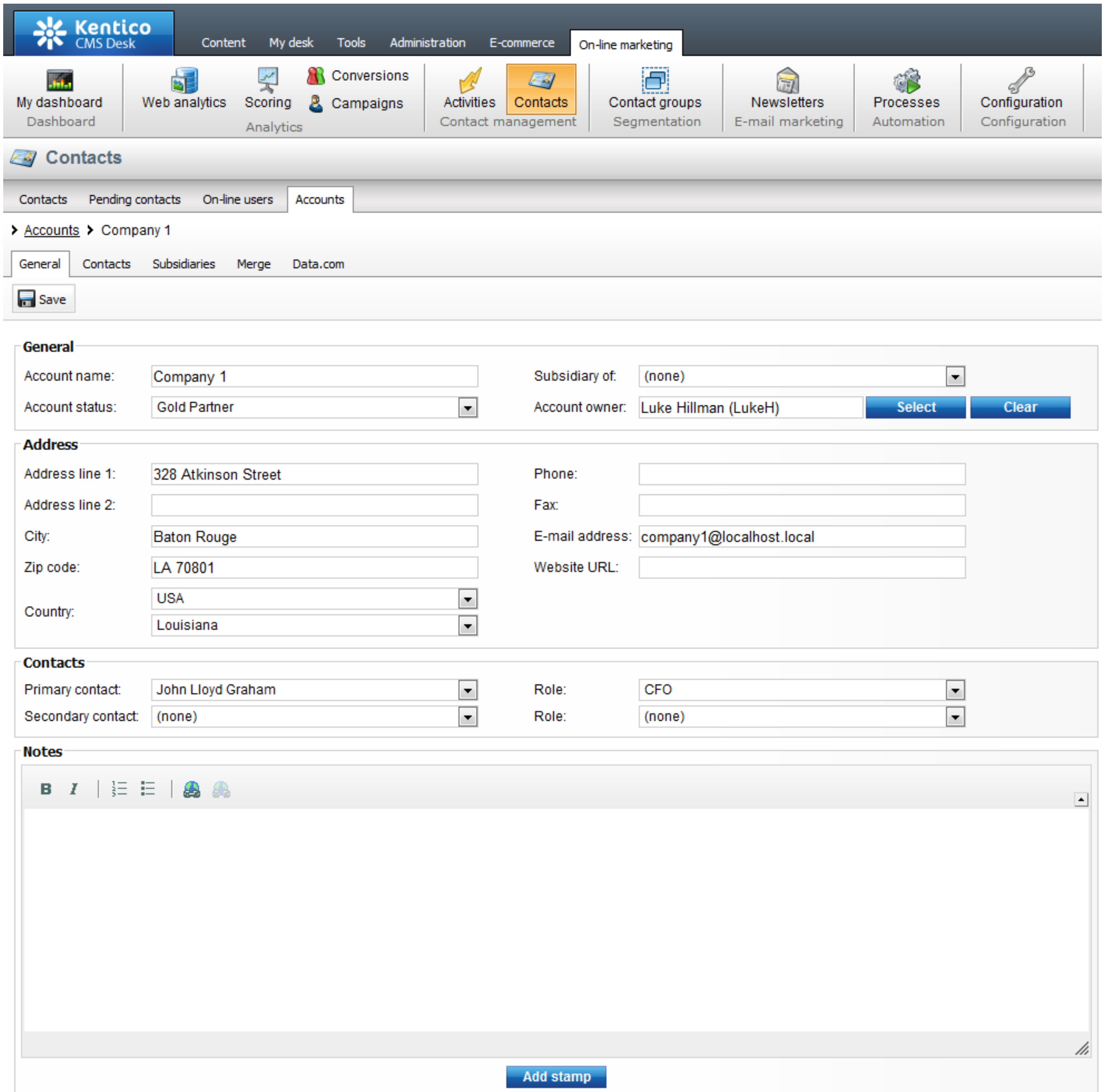
- **Fax** - the fax number of the account.
- **E-mail** - the e-mail address of the account.
- **Website URL** - the website URL of the account.

Contacts

- **Primary contact** - the most important contact assigned to the current account; e.g. a company's CEO. Please note that you can select a primary contact only if at least one contact is assigned to the current account.
- **Role** - if defined, allows to assign the primary contact with a contact role, e.g. *CEO*.
- **Secondary contact** - the second most important contact assigned to the current account; e.g. a company's CFO. Please note that you can select a secondary contact only if at least one contact is assigned to the current account.
- **Role** - if defined, allows to assign the secondary contact with a contact role, e.g. *CFO*.

Notes - you can leave account notes either as plain text or as text with basic formatting, and you can also add your stamp by clicking the **Add stamp** button.

Please note that if you are viewing properties of a merged account, **Split from parent** is available on the page, enabling you to split the current account from the parent one.



Custom fields tab

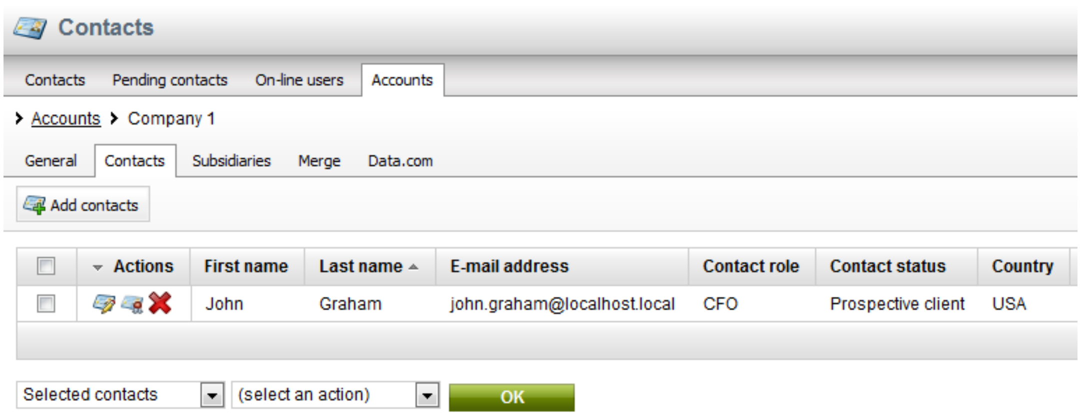
On this tab you can enter values into the custom fields defined for accounts in **Site manager -> Development -> System tables -> Edit (** **)** ***Contact management - Account*** on the **Fields** tab. When you have entered the values, click **Save** to confirm the changes.

Contacts tab

On this tab you can see a list of contacts assigned to the current account and you can also add contacts by clicking **Add contacts**. If you decide to do so, the **Select contact** dialog opens, enabling you to assign selected contacts to the current account; optionally also with a selected role (if defined). Please note that if you edit a global account, only global contacts are offered. Similarly, if you edit a site account, only current site contacts are offered.

You can **View contact details** () of individual contacts and you can also **Select contact role** () or **Remove** () a contact. The displayed list can be sorted by contact first name, last name, e-mail, contact role, contact status and country.

Using the pair of drop-down lists below the list, you can perform certain management tasks with multiple contacts at once, specifically *Selecting contact role* or *Removing* contacts. To do this, choose either *Selected contacts* or *All contacts* from the first drop-down list, the required action from the second one and click **OK**.

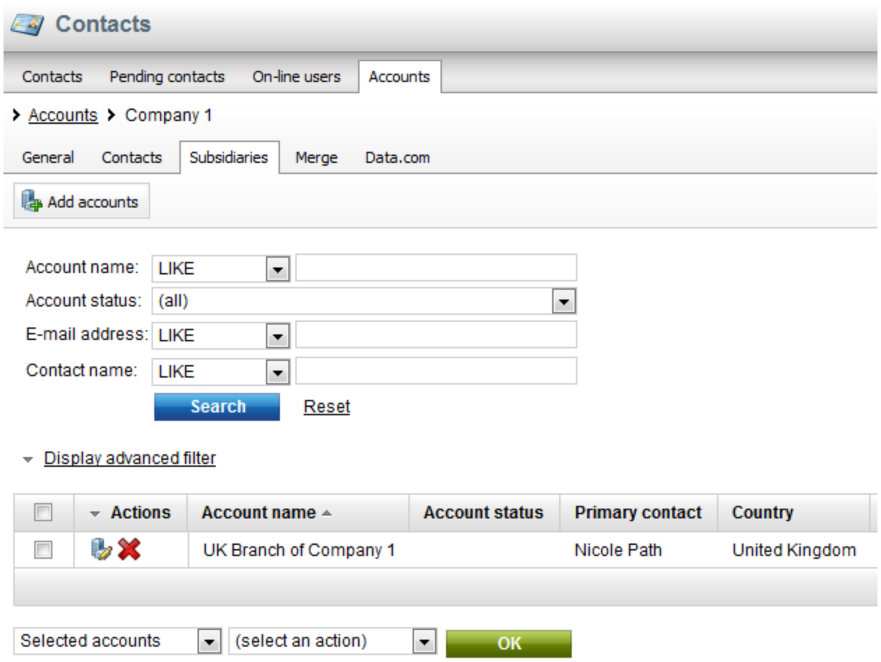


Subsidiaries tab

On this tab you can see a list of accounts which are subsidiaries of the current account. You can also add accounts to be recognized as subsidiaries of the current account by clicking **Add accounts**. If you decide to do so, the **Select account** dialog opens, enabling you to assign these accounts. Please note that if you edit a global account, only global accounts are offered; similarly, if you edit a site account, only current site accounts are offered.

You can **View details** () of individual accounts or **Remove** () them. The displayed list can be sorted by account name, account status, primary contact and country.

Using the pair of drop-down lists below the list, you can perform the *Remove* action for multiple accounts at once. To do this, choose either *Selected accounts* or *All accounts* from the first drop-down list, the action from the second one and click **OK**.

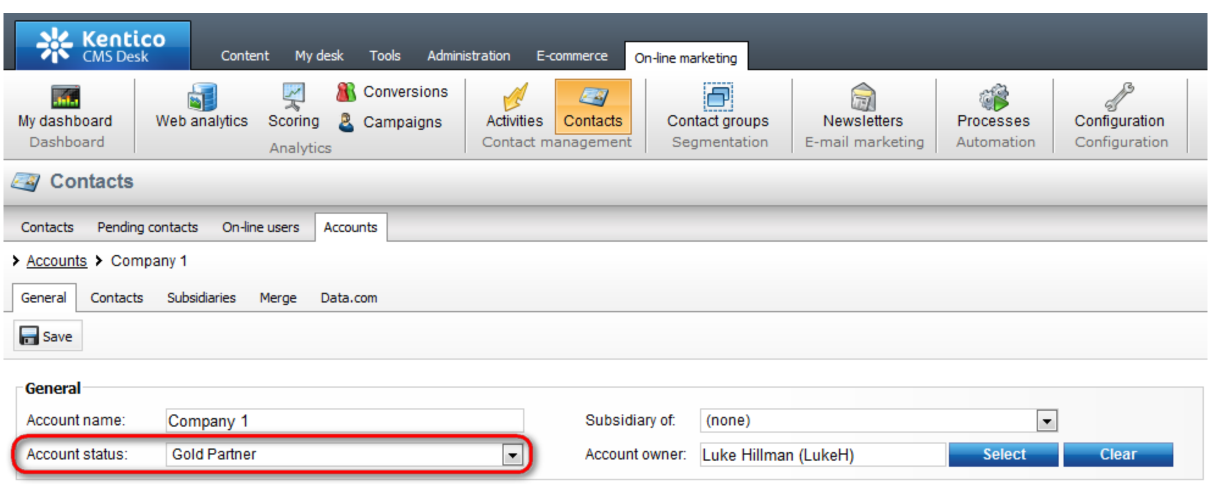


Merge tab

On this tab you can merge other accounts with the currently edited account so that only the currently edited account remains. This is useful in cases when you come to the conclusion that multiple accounts actually represent a single real-world organization. On the tab, you can also view a list of accounts that have been merged with it and split them back into separate accounts if needed. Detailed information on merging in Kentico CMS can be found in the Merging contacts and accounts chapter.

6.3.3 Account statuses

Accounts can be labeled with account statuses in order to reflect their relationship to your organization or to indicate how the account is perceived from the marketing point of view. Statuses can be assigned to accounts in the **Account status** field on the General tab of each account's editing interface. One account can be labeled with only one status at a time.

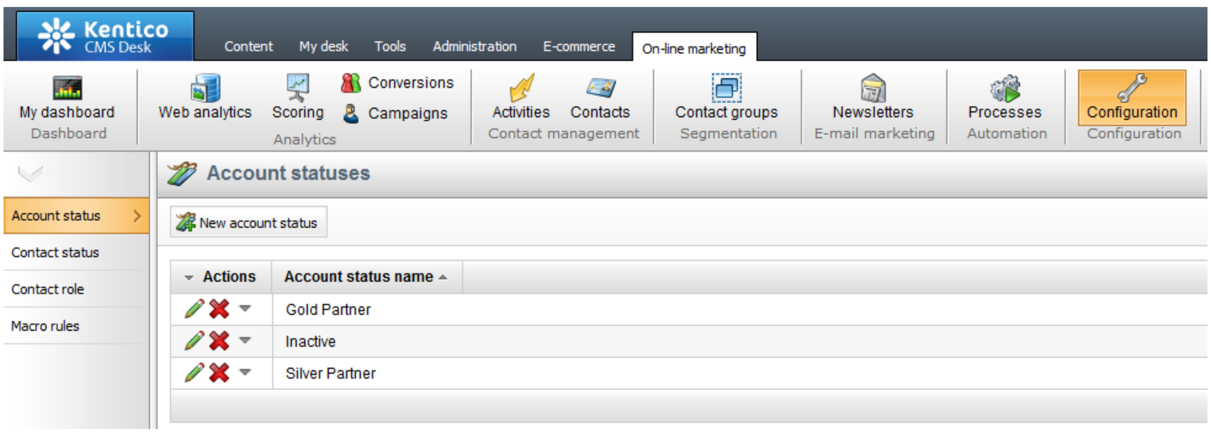


There is a pre-defined set of account statuses for each website. The sets of statuses can be managed in the following sections of the user interface:

- **CMS Desk -> On-line marketing -> Configuration -> Account status**
- **Site Manager -> Tools -> Contact management -> Configuration -> Account status**

The user interface in CMS Desk allows management of account statuses pre-defined for the currently edited website. The one in Site Manager allows management of account statuses for all websites running in the system. If global account statuses are enabled, they can be managed in both sections.

Properties of account statuses listed in the mentioned sections of the user interface can be edited after clicking the **Edit** ( ) icon in a respective account status's row. By clicking the **Delete** ( ) icon, you can delete the account status so that it is no longer present and available in the system. The **Export** ( ) action available in the drop-down menu ( ) in each row enables you to export the respective status into a file and subsequently import it on another Kentico CMS instance.

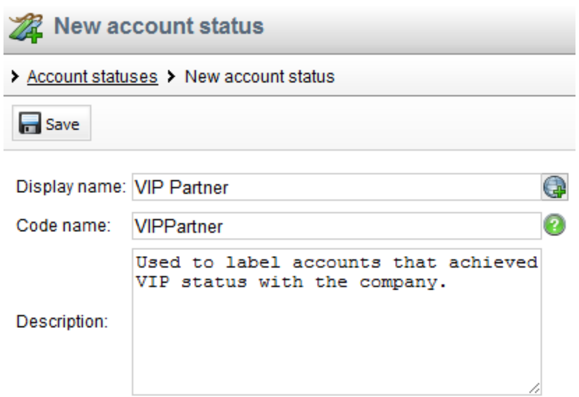


Creating a new account status

If you need to add a new account status to the set of listed statuses, click **New account status**. After doing so, you need to specify the following properties in the **New account status** dialog:

- **Display name** - the name of the account status used in Kentico CMS user interface.
- **Code name** - the name of the account status used in the code.
- **Description** - the description of the account status.

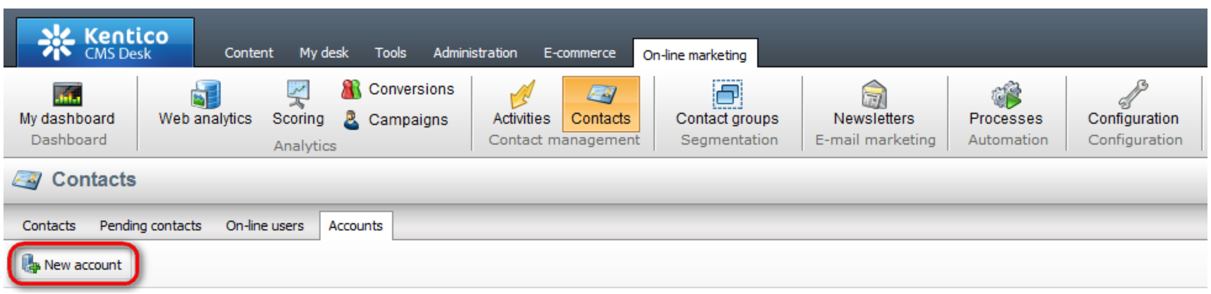
Once you have the properties specified, click **OK** for the new status to be created.



6.3.4 Example: Creating a sample account

Here you will learn how to create a new account.

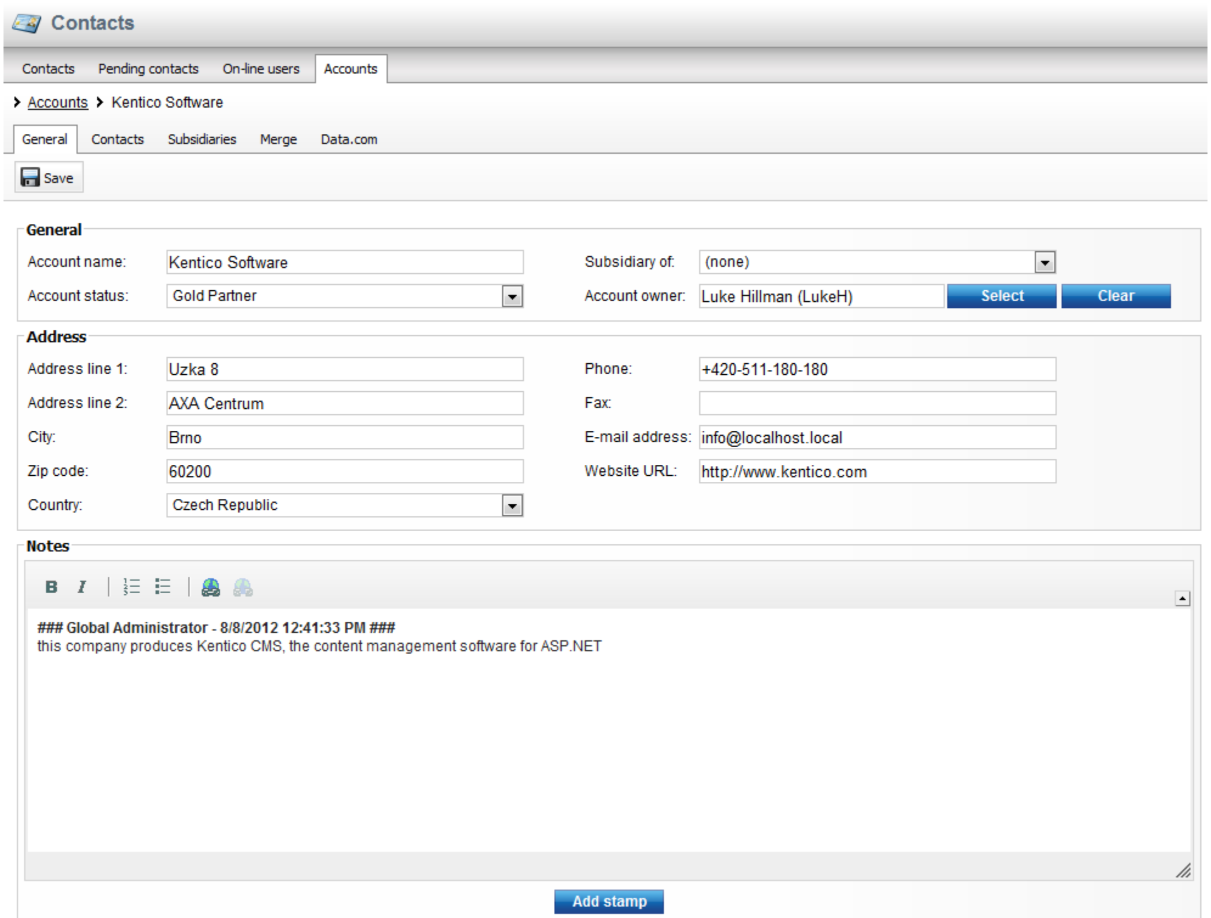
1. Sign in as an administrator and go to **CMS Desk -> On-line marketing -> Contacts -> Accounts** and click on **New account**.



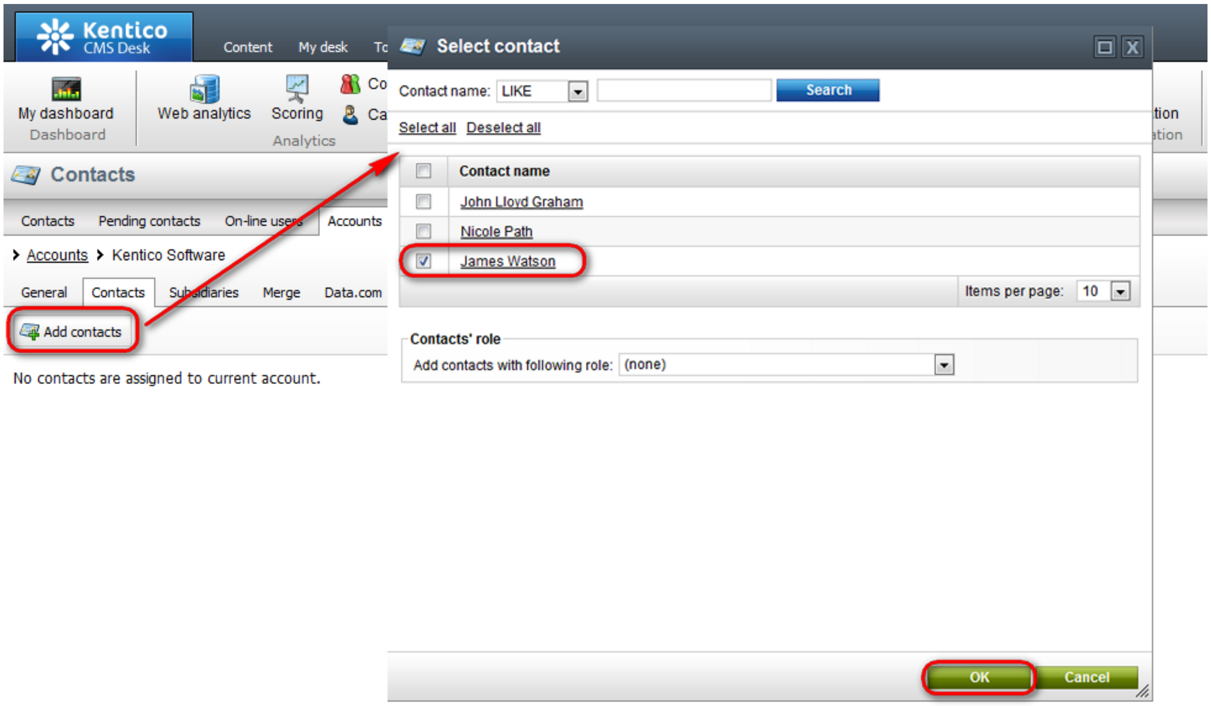
2. In the **New account** dialog, enter for example the following properties:

- **Account name:** Kentico Software
- **Account status:** select an account status, if no statuses were defined yet, please refer here to see how to do it
- **Account owner:** Luke Hillman (LukeH)
- **Address line 1:** AXA Centrum
- **City:** Brno
- **Zip code:** 60200
- **Country:** Czech Republic
- **Phone:** +420-511-180-800
- **E-mail:** info@localhost.local

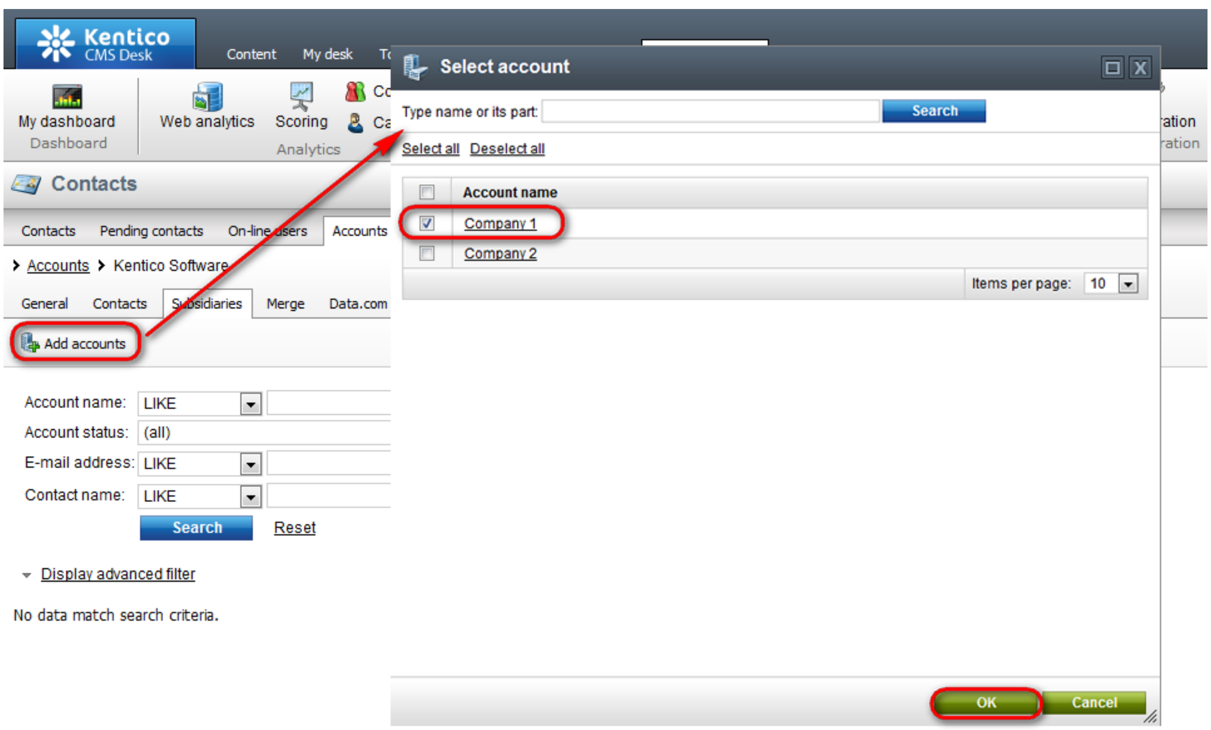
Once entered, click **Save** for the new account to be created.



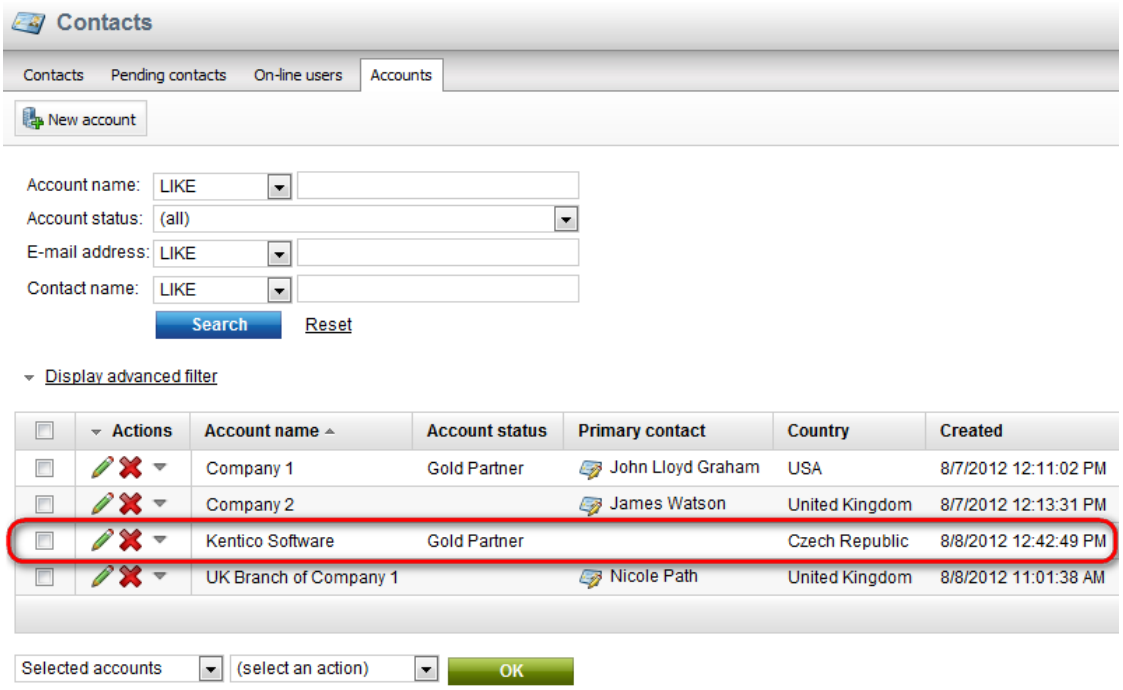
3. The account is now created and you are redirected to the General tab of its editing interface. If you want to add contacts to it, switch to the **Contacts** tab and click **Add contacts**. In the **Select contact** dialog which pops up, select contacts who should belong to the account, e.g. *James Watson* defined in Contacts -> Example: Creating a sample contact, and click **OK**.



4. Optionally, switch to the **Subsidiaries** tab, click **Add accounts** and in the **Select account** dialog select accounts which should be recognized as subsidiaries of the current account. Click **OK**.



6. If you now view the list of accounts in **CMS Desk -> On-line marketing -> Contacts -> Accounts**, you can see that the new account is listed among other accounts.



6.4 Contact groups

6.4.1 Overview

You can organize contacts into segments called *contact groups*. Segmentation allows you to divide your website's visitors and users into groups of people with shared characteristics, which you can then accurately target in your on-line marketing efforts (such as E-mail marketing). You can either add contacts to groups manually or set up the system to assign contacts to groups automatically based on conditions.

Contact groups can be site-specific or shared across all websites in the system (global contact groups). Contacts can belong to any number of different contact groups.

Topics:

- Managing contacts groups
- Example: Creating a sample contact group

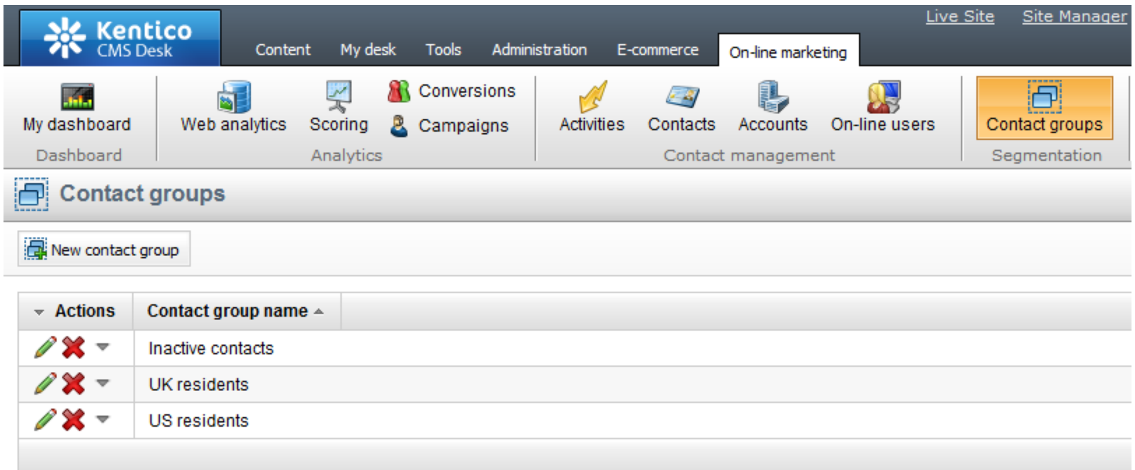
6.4.2 Managing contact groups

In this topic you will learn how to manage contact groups. Contact groups can be managed in:

- **CMS Desk -> On-line marketing -> Contact groups**
- **Site Manager -> Tools -> Contact management -> Contact groups**

Here you can see a list of contact groups available on the current site and you can also create new contact groups by clicking **New contact group**. Please note that you can always create site-specific contact groups and, if the use of global contact groups is allowed, also contact groups shared across more websites.

Individual contact groups can be **Edited** () or **Deleted** () and you can also perform **Other actions** such as **Export**, **Backup** or **Restore**. The displayed list can be sorted by contact group name and, optionally, site.



If editing a contact group, the following tabs are available:

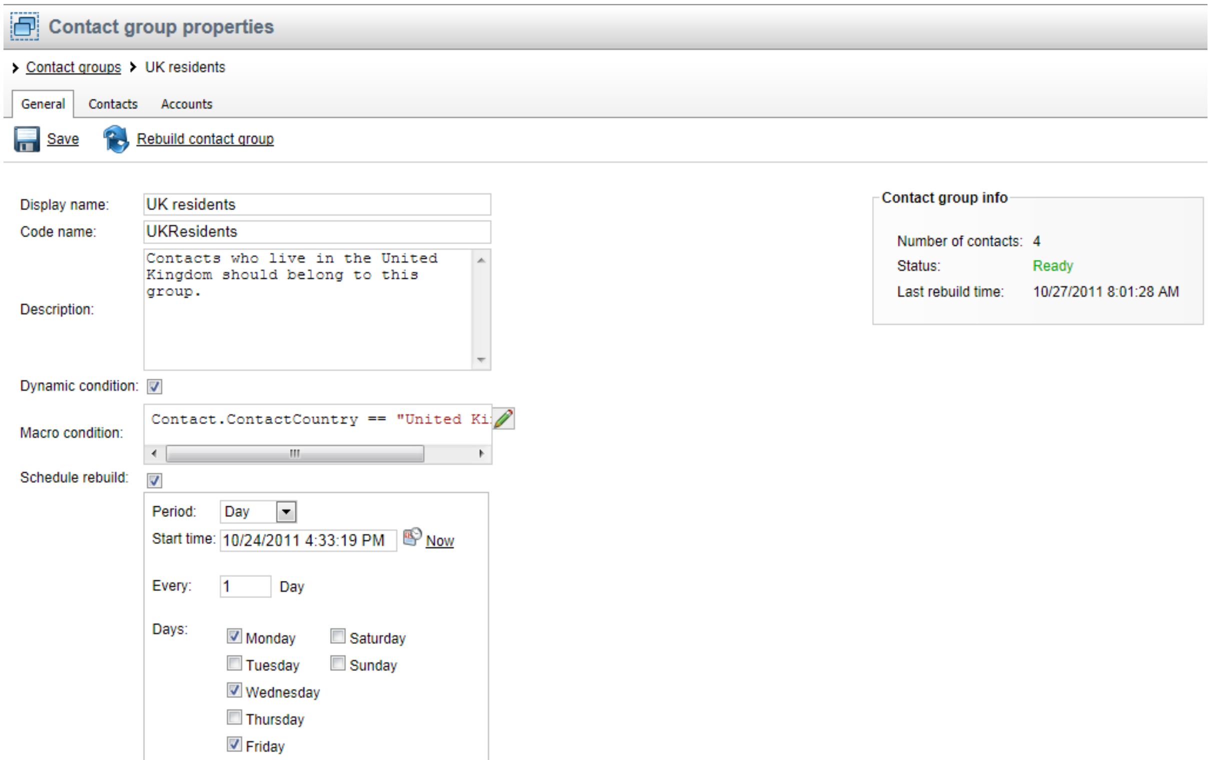
- General tab
- Contacts tab
- Accounts tab

General tab

On this tab you can edit properties of a selected contact group. When you have modified the values, click **Save** to save the contact group.

- **Display name** - the name of the contact group used in Kentico CMS user interface.
- **Code name** - the name of the contact group used in the code.
- **Description** - the description of the contact group.
- **Dynamic condition** - allows to define a macro condition. If defined, a contact which fulfills the given condition is assigned to the current contact group.
- **Macro condition** - use the text area to define a macro condition, e.g. contact's city is *London*. Alternatively, you can click **Edit** () to edit the macro condition in the **Edit macro condition** dialog window. For more details on using macros in Kentico CMS, please refer to the Macro expressions chapter in the Developer's Guide.
- **Schedule rebuild** - if enabled, the current contact group is automatically rebuilt in accordance with the defined macro condition.

You can also manually rebuild the contact group according to a macro condition (if defined) by clicking on **Rebuild contact group** or view **Contact group info** containing information on the number of contacts in the current contact group, its status and last rebuild time.



Contacts tab

On this tab you can see a list of contacts assigned to the current contact group and you can also add contacts to the group by clicking the **Add contacts** link. If you decide to do so, the **Select contact** dialog opens, enabling you to assign selected contacts to the current contact group. Please note that if you edit a global contact group, current site contacts and optionally global contacts are offered; however, if you edit a site contact group, only current site contacts are offered.

You can **View contact details** () of individual contacts and you can **Remove** () contacts. The displayed list can be sorted by contact first name, last name, e-mail, contact status, country etc. Besides, the list provides information on how a particular contact was added to the contact group:

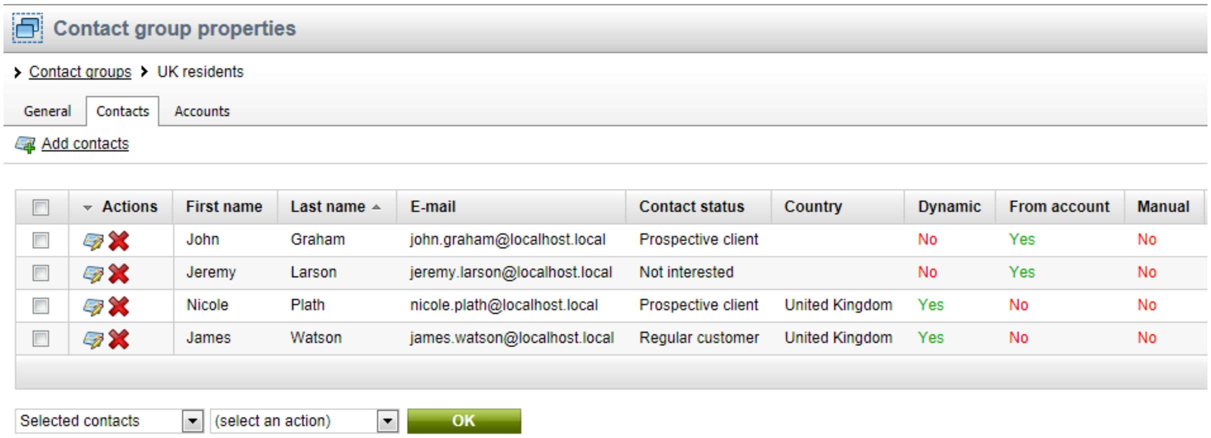
- **Dynamic** - indicates if the contact was added based on a dynamic condition.
- **From account** - indicates if the contact was added together with an account with which it is associated.
- **Manual** - indicates if the contact was added manually, using the **Select contact** dialog window.

Using the pair of drop-down lists below the list, you can perform the *Remove* action for multiple contacts at once. To do this, choose either *Selected contacts* or *All contacts* from the first drop-down list, the action from the second one and click **OK**.

Please note

Contacts added to the current contact group otherwise than by assigning an associated account, i.e. either dynamically based on a condition or manually, remain in the contact group after removing the respective account from it.

Similarly, this holds true also to contacts added to the group dynamically providing the respective dynamic condition ceased to apply.

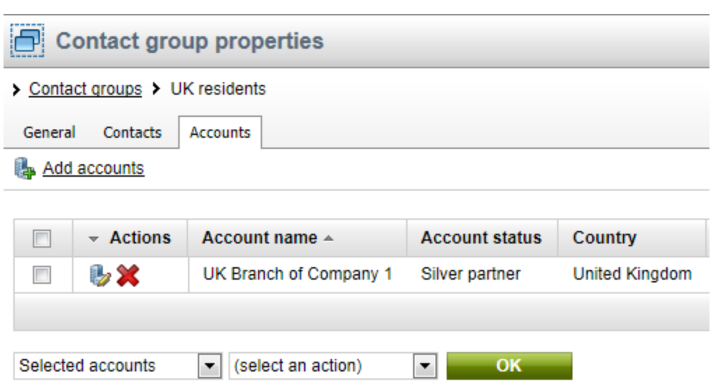


Accounts tab

On this tab you can see a list of accounts assigned to the current contact group and you can also **Add accounts**. Please note that adding an account to a contact group results in adding all contacts associated with this account to the group. If you decide to add accounts, the **Select account** dialog opens, enabling you to assign selected accounts to the current contact group. Please note that if you edit a global contact group, current site accounts and optionally global accounts are offered; however, if you edit a site contact group, only current site accounts are offered.

You can **View account details** () of individual accounts and you can **Remove** () accounts. The displayed list can be sorted by account name, account status, country etc.

Using the pair of drop-down lists below the list, you can perform the *Remove* action for multiple accounts at once. To do this, choose either *Selected accounts* or *All accounts* from the first drop-down list, the action from the second one and click **OK**.

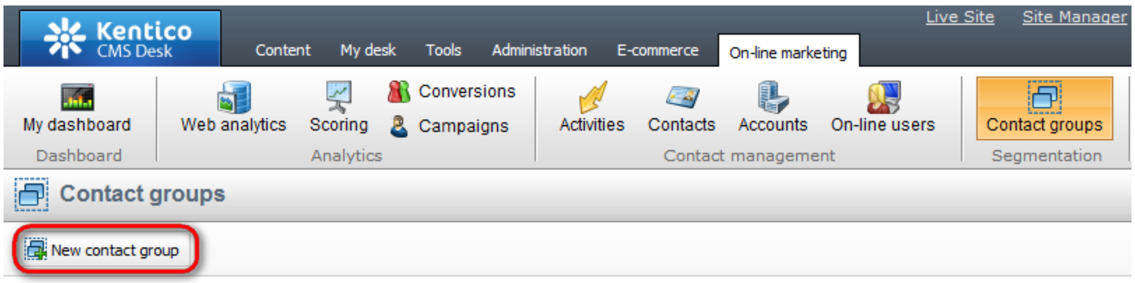


6.4.3 Example: Creating a sample contact group

The following steps demonstrate how to create a contact group with a dynamic condition:

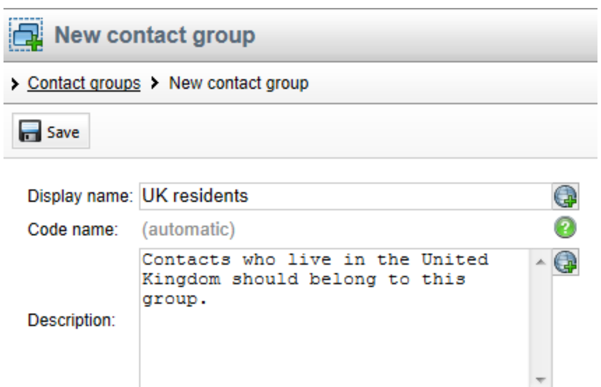
1. Sign in to CMS Desk and go to **On-line marketing -> Contact groups**.

2. Click **New contact group**.



3. In the **New contact group** dialog, enter the following properties:

- **Display name:** UK residents
- **Description**: Contacts who live in the United Kingdom should belong to this group.

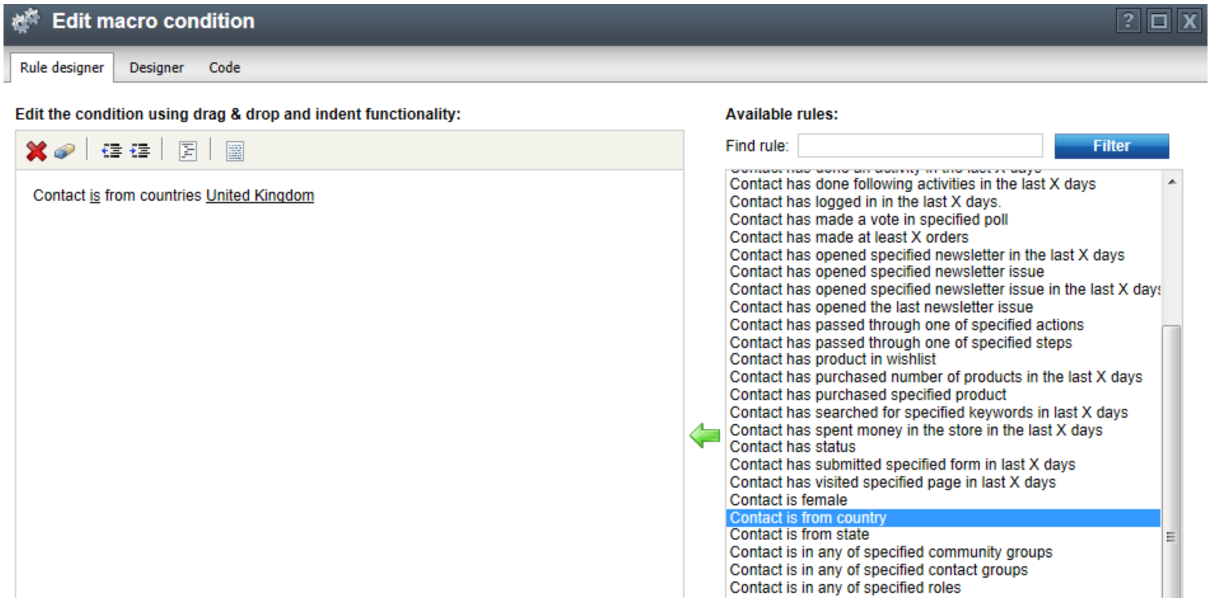


4. Click **Save**.
   - The **General** tab of the new contact group opens.

5. Check the **Dynamic condition** box and click edit () next to the **Macro condition** field.

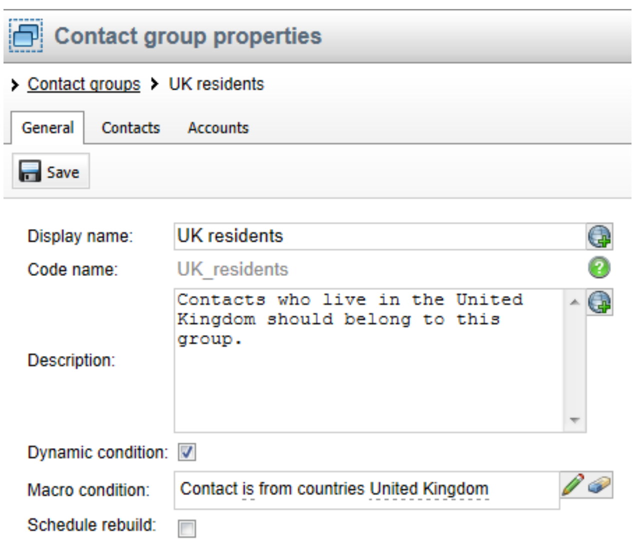
6. In the **Edit macro condition** dialog:

a. Select the **Contact is from country** rule.
b. Click **Add rule** ().
c. Click the select countries parameter in the text of the rule clause.
d. In the **Set parameter value** window, click **Select**, choose *United Kingdom* and click **OK**.



This condition ensures that all contacts who have *United Kingdom* specified in the **Country** field of their properties are added automatically to the contact group.

7. Click **OK**.
   - The editor closes and inserts the condition.



8. Click **Save**.

The contact group is now ready.

Adding contacts to the contact group

You can add contacts to groups in three different ways:

- Automatically based on a dynamic conditions

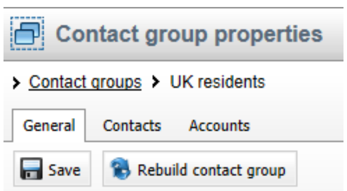
- Manually add members of specific accounts
- Manually add individual contacts

Adding contacts based on the condition

To try out adding of contacts based on the dynamic condition, make sure that you have at least one contact with *United Kingdom* specified in the **Country** field.

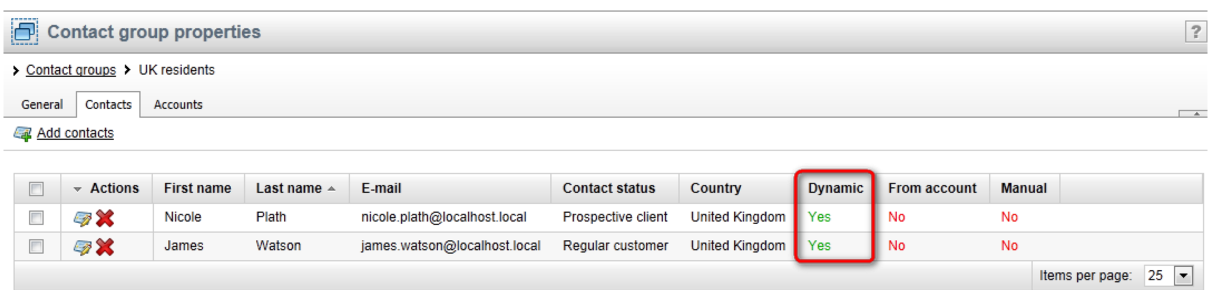
1. Open the **General** tab of the contact group.

2. Click **Rebuild contact group** next to the **Save** button.
   - The system adds all contacts that match the group's condition.



3. Switch to the **Contacts** tab.

You should see that the matching contacts were added to the contact group. The *Yes* value in the **Dynamic** column indicates that the contacts were added to the contact group based on a dynamic condition.



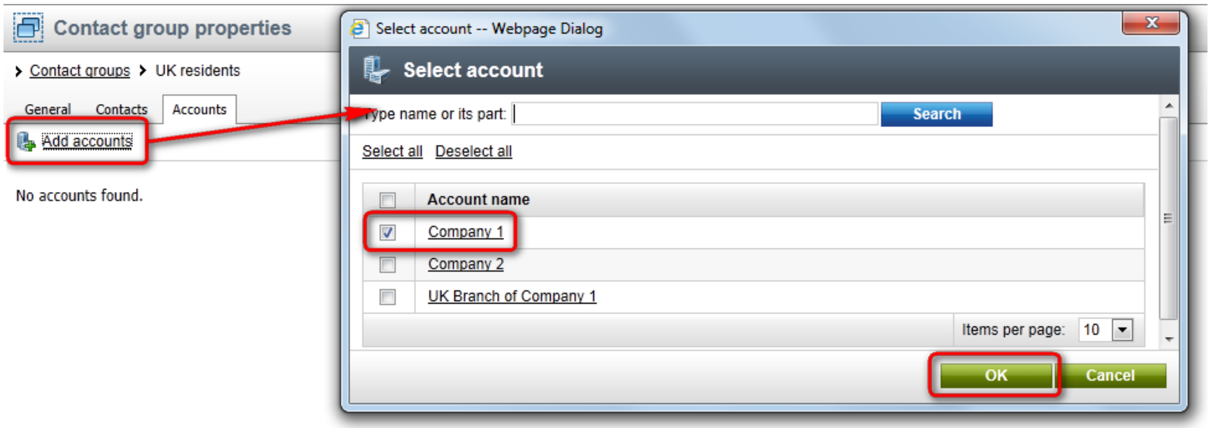
Adding accounts

Adding an account to the contact group assigns all contacts that belong into the account. Before proceeding, make sure that you have at least two contacts assigned to one account in the system.

1. Open the **Accounts** tab of the contact group.

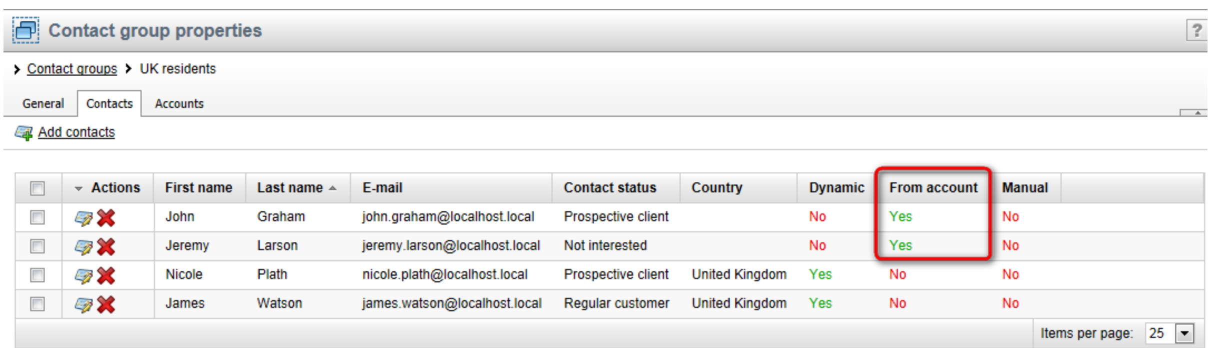
2. Click **Add accounts**.

3. Select the account and click **OK**.



4. Switch back to the **Contacts** tab.

You can see that all contacts associated with the previously added account are listed. The *Yes* value in the **From account** column indicates that the contacts were added automatically because they belong to the added account.



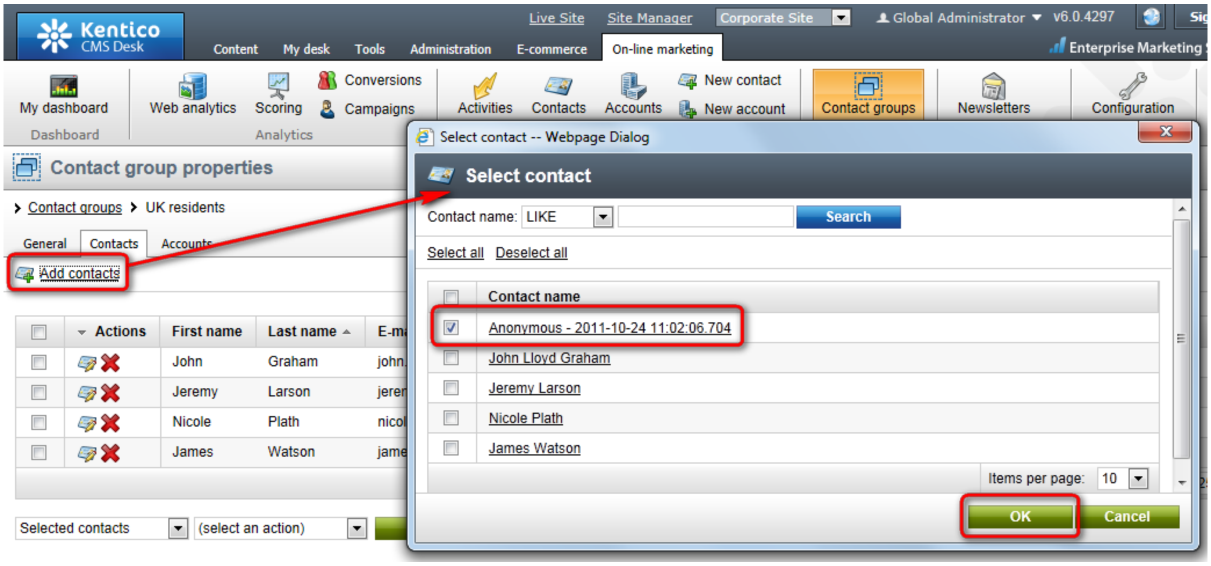
Adding contacts individually

You can manually add specific contacts to contact groups.

1. Open the **Contacts** tab of the contact group.

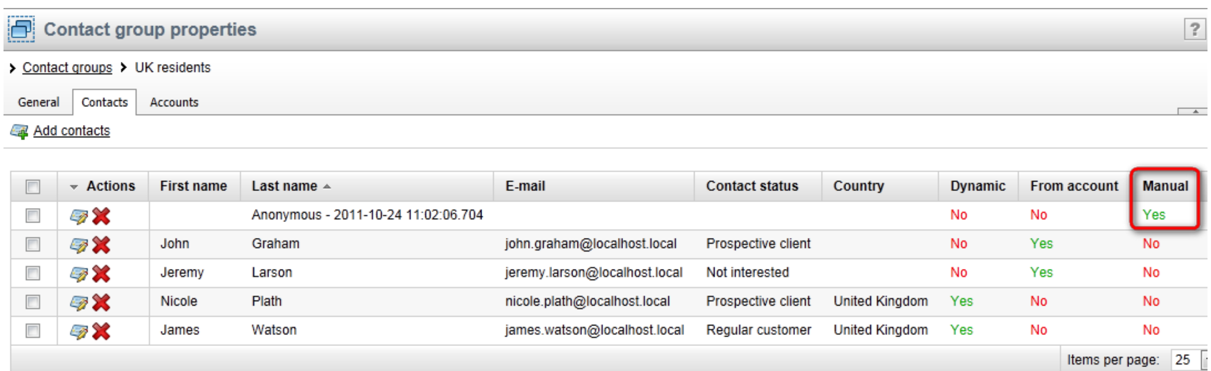
2. Click **Add contacts** above the list of contacts.

3. Select a contact that has not been added to the contact group yet and click **OK**.



4. Switch back to the **Contacts** tab.

You should see the last contact listed with the *Yes* value in the **Manual** column, indicating that it was added to the contact group manually.



6.5 Merging contacts and accounts

6.5.1 Overview

Multiple [contacts](#) in the system may actually represent the same real-world person, or you may have more than one [account](#) that represents a single real-world organization. Merging functionality allows you to get rid of duplicates by combining multiple contacts or accounts into a single object.

There are two ways to perform merging:

- [Manual merging](#) - available for both contacts and accounts. Can be performed on the **Merge** tab when editing contacts or accounts. You can select any contacts or accounts and merge them with the currently edited object. The interface provides merge suggestions for contacts with identical parameters.
- [Automatic merging](#) - available only for contacts. The system automatically merges contacts that are

associated with the same E-commerce customer or newsletter subscriber, based on the settings in **Site Manager -> Settings -> On-line marketing -> Contact management -> Global data & merging**.

The result of merging is a single object (called a parent). However, the merged objects remain stored in the database.

You can view a list of all contacts or accounts that have been merged with a parent on the **Merge -> Split** tab of the contact or account editing interface. It is possible to split merged contacts or accounts from the parent back into separate objects. See: Splitting

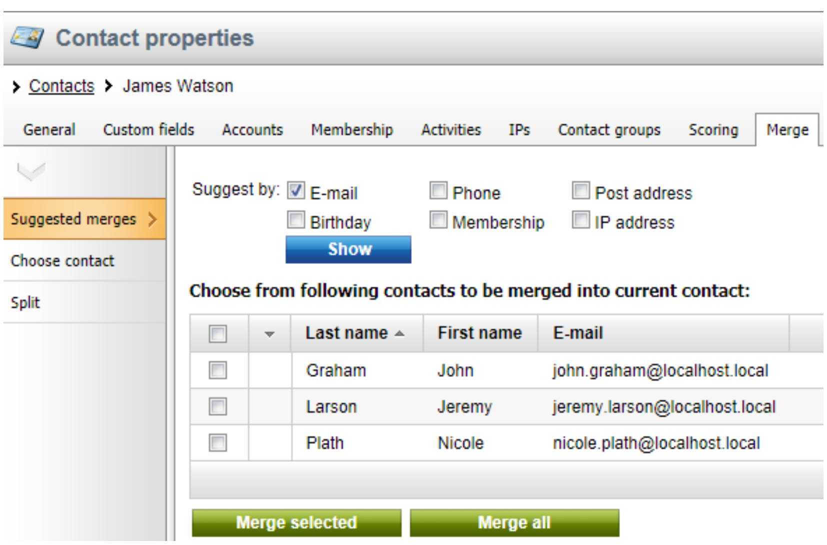
6.5.2 Manual merging

Manual merging of contacts or accounts can be performed on the **Merge** tab of their editing interfaces. Here, you can either let the system suggest you which contacts/accounts appear to be duplicate to the currently edited one, or choose the contacts/accounts to be merged with it manually from a list of all contacts/accounts that are available.

Suggested merges

On the **Suggested merges** sub-tab, you can let the system suggest you which contacts/accounts have identical values of certain properties. To get the suggestions, select the parameters to be compared using the **Suggest by** check-boxes and clicking **Show**. A list of contacts/accounts that have values of the selected properties identical to the currently edited one is displayed below.

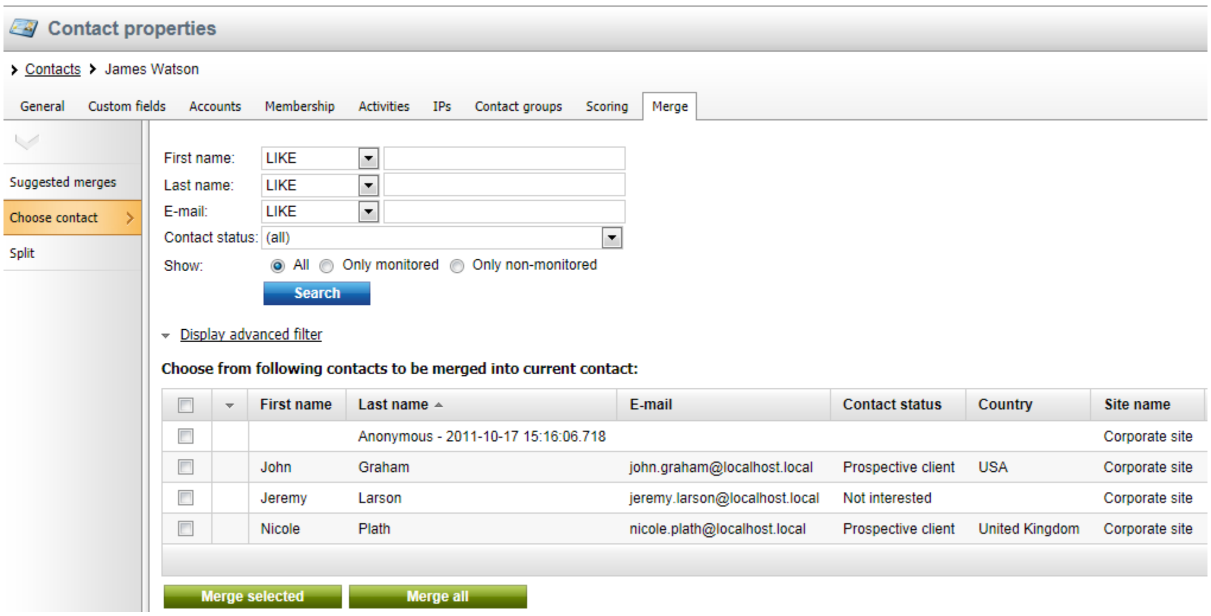
To merge all suggested contacts/accounts into the currently edited one, click the **Merge all** button. If you want only some of them to be merged, select them using the check-boxes in their rows and click the **Merge selected** button. In both cases, the Merge collisions dialog window will be opened, enabling you to specify values of individual properties and resolve potential value conflicts.



Choose contact/account

On the **Choose contact/account** sub-tab, you can choose contacts/accounts to be merged with the currently edited one manually. By default, a list of all available contacts/accounts is displayed, while you can display only those that match specified criteria using the filter above the list.

To merge all displayed objects into the currently edited one, click the **Merge all** button. If you want only some of them to be merged, select them using the check-boxes in their rows and click the **Merge selected** button. In both cases, the Merge collisions dialog window will be opened, enabling you to specify values of individual properties and resolve potential value conflicts.

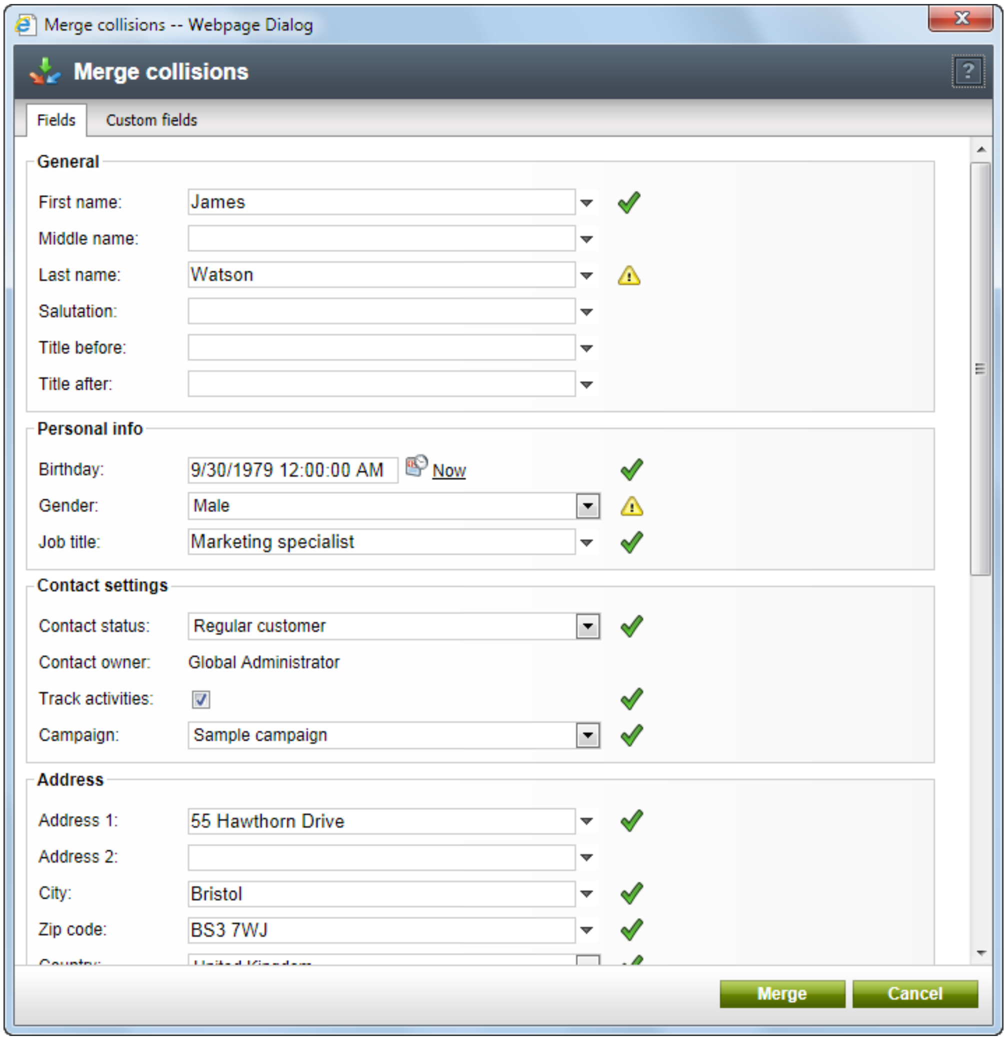


Merge collisions

In the **Merge collisions** dialog window, you can adjust property values that the contact/account will have after merging. Properties whose values are not empty in at least one of the merged contacts/ accounts are pre-filled, while the following icons are displayed next to them:

- - indicates that the system didn't detect any collision of this field's values.
- - indicates a detected collision, while the pre-filled value is is taken from the parent. If collisions are detected only between child contact properties while the parent has the property value empty, the pre-filled value taken from the colliding contact that is retrieved from the database first. By clicking the icon next the fields, you can choose the colliding value from a different contact.

You can adjust values of all the properties according to your needs. Once finished, click **Merge** to perform the merging.



6.5.3 Automatic merging

You can set up the system to merge contacts automatically. This helps you save time by merging contacts that have clear ties to the same person. Automatic merging can occur when new contacts are created or when the data of existing contacts is updated.

The system always merges contacts that are associated with the same user account.

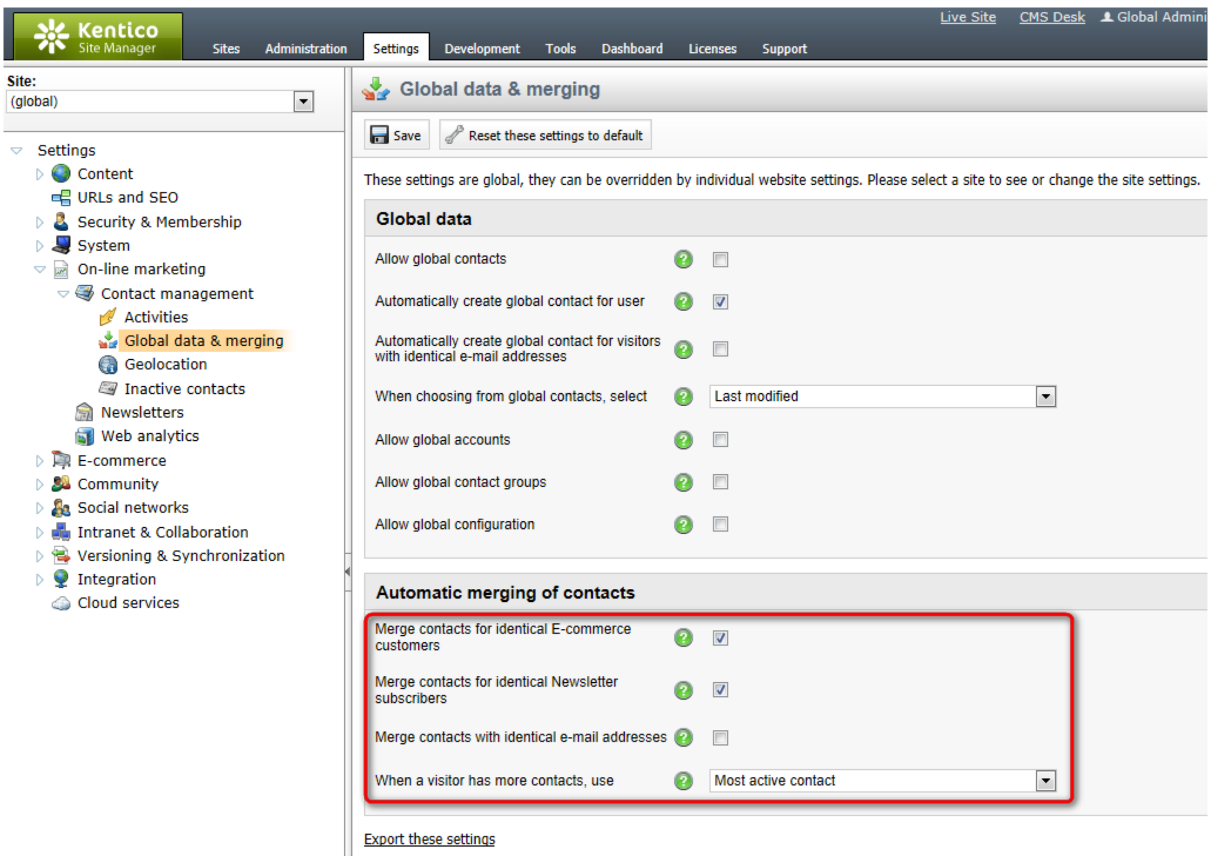
To configure additional types of automatic merging, adjust the following settings in **Site Manager -> Settings -> On-line marketing -> Contact management -> Global data & merging**:

- **Merge contacts for identical E-commerce customers** - if enabled, the system automatically merges contacts that are associated with the same e-commerce customer.
- **Merge contacts for identical Newsletter subscribers** - if enabled, contacts associated with the same newsletter subscriber will be merged automatically.
- **Merge contacts with identical e-mail addresses** - if enabled, the system automatically merges contacts that are associated with the same newsletter subscriber.
- **When a visitor has more contacts, use** - determines which contact the system selects as the parent when automatically merging:

- **Last logged contact -** the contacts with the most recently logged activity
- **Most active contact** - the contact with the highest number of logged activities
- **Create a new contact -** the system creates a new contact as the parent a merges all matching contacts into it

Automatic merge collisions

When attribute collisions occur during automatic merges, the system uses the values of the parent contact. If a collision is detected between the attributes of child contacts and the parent contact does not contain a value in the given field, the priority of the values is undetermined (the system uses the values of the first contact loaded from the database).



Setting up automatic contact merging

6.5.4 Splitting

On the **Merge -> Split** tab of each contact's or account's editing interface, you can see a list of all objects which were merged into the currently edited one. It is possible to split these objects from the parent and make them separate objects again.

To split some of the merged objects back into separate objects, select them using the check-boxes and click the **Split selected** button. Before splitting, you may enable the following options in the **Split**

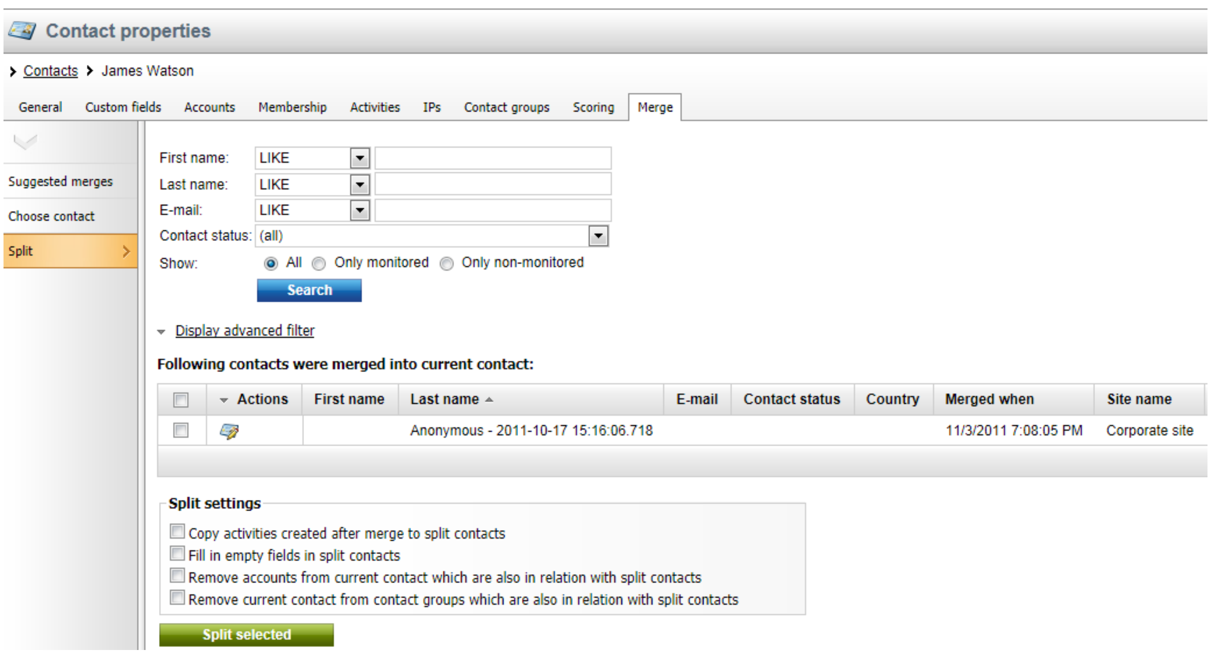
settings section for additional actions to be performed when splitting:

The following options are available when splitting merged contacts:

- **Copy activities created after merge to split contacts** - enable to copy the activities logged for the parent contact after the merge into the split contacts.
- **Fill in empty fields in split contacts** - enable to copy the parent contact's attribute values to the split contacts. Only affects fields that were empty before the merge.
- **Remove accounts from current contact which are also in relation with split contacts** - enable to remove the parent contact from accounts that were originally associated only with the split contacts before the merge.
- **Remove current contact from contact groups which are also in relation with split contacts** - enable to remove the parent contact from contact groups that were originally associated only with the split contacts before the merge.

The following options are available when splitting merged accounts:

- **Fill in empty fields in split accounts** - enable to copy the parent account's attribute values to the split accounts. Only affects fields that were empty before the merge.
- **Remove contacts from current account which are also in relation with split accounts** - enable to remove all contacts from the parent account that were only part of the split accounts before the merge.
- **Remove current account from contact groups which are also in relation with split accounts** - enable to remove the parent account from all contact groups that were originally associated only with the split accounts before the merge.



6.6 Tracking contact activities

6.6.1 Overview

Activities represent actions that contacts perform on the website. For example, the system logs activities when visitors view pages, post comments, purchase products, register for events, subscribe to newsletters etc.

The data gathered in the activity log allows you to evaluate the behavior of your website's users and visitors. You can also use activities in combination with other on-line marketing features:

- Automatically calculate scores for contacts based on their activities
- Use activities as triggers for starting Marketing automation processes for contacts
- Personalize your website's content for contacts who have performed specific activities

Topics:

- Enabling activity tracking
- Viewing the activity log
- Example: Logging activities
- Adding custom activities
- Reference: Activity types

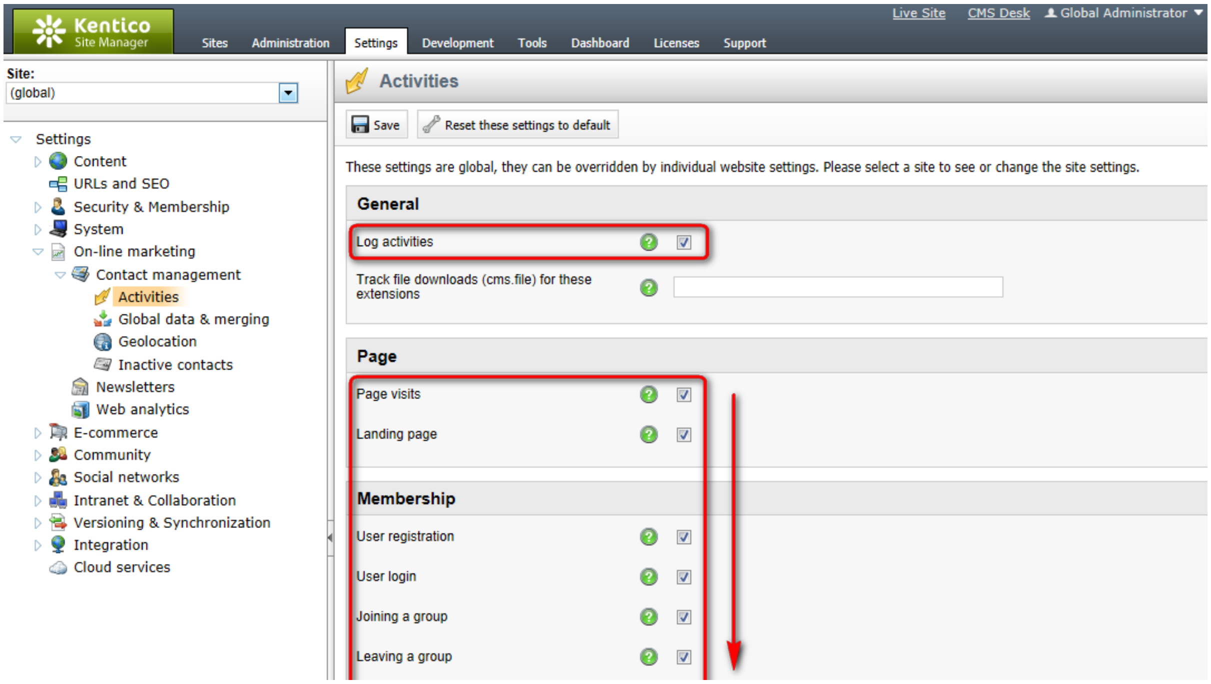
6.6.2 Enabling tracking of activities

To allow tracking of activities for contacts, you need to:

1. Go to **Site Manager -> Settings -> On-line marketing** and ensure that **Enable on-line marketing** is checked for your website.

2. Navigate to the **On-line marketing -> Contact management -> Activities** settings category and enable **Log activities**.

3. Enable or disable logging of specific activity types using the other settings in the category.



Enabling activities in the website settings

4. **Save** the settings.

You can additionally configure logging of activities for specific pages, visitors and certain features. The system only tracks activities within a given scope if all related settings are enabled. See the sections below for more information.

Logging activities immediately

By default, the system logs activities into temporary files and batch-processes them periodically using a scheduled task. As a result, there is a slight delay of up to several minutes before activity-related data is updated in the user interface.

You can configure the system to log activities directly into the database. Immediately logging activities can generate a very high database load, so it is only recommended when evaluating or testing the functionality of activities, not for live websites.

To enable immediate logging of activities, administrators with access to the project's file system need to add the following key to the *appSettings* section of the *web.config* file:

```
<add key="CMSLogActivityImmediatelyToDB" value="true" />
```

Tracking file downloads

The system can track file downloads as *Page visit* activities for files stored as *CMS.File* documents in the content tree of a website. The **Track file downloads (cms.file) for these extensions** setting in the **General** section specifies which types of files the tracking includes.

Enter the allowed file types as a list of extensions separated by semicolons, for example: *pdf;docx;png*

If left empty, the system tracks all file types.

Enabling activities for specific pages

You can enable or disable logging of page-related activities (e.g. *Page visit*, *Landing page* or *Content rating* activity types) for individual pages on your website.

1. Select the document representing the page in the CMS Desk content tree.
2. Open the **Properties -> General** tab.
3. Toggle the **Log on-line marketing activity** checkbox on or off.
   - If the **Inherit** field is checked, the document loads the value from the parent document.
4. Click **Save**.

By default, documents inherit the value of the **Log on-line marketing activity** property from their parent document. This state allows you to configure activity logging for all documents on the website through the root document.

Selecting the logging method for page activities *[Only available after applying hotfix 7.0.17 or newer]*

The system provides two different ways of logging activities related to website traffic (*Page visit* and *Landing page*):

Processing activities on every web request	Using JavaScript
Advantages: • Provides statistics for all types of web requests.	**Advantages**: • Filters out activities generated by non-human tools such as RSS readers and web crawlers. • Provides more relevant data for most public-facing websites.
Disadvantages: • May log irrelevant activities for non-human contacts such as web crawlers, RSS readers and other bots.	**Disadvantages**: • Ignores users with browsers that do not support JavaScript or have it disabled (typically less than 1% of all visitors). • May not work correctly on pages that run custom JavaScripts (if conflicts occur). The logging is compatible with all default Kentico CMS scripts.

In most cases, it is recommended to use JavaScript logging. Switching to JavaScript logging does not reduce the performance of the website.

To enable JavaScript logging:

1. Go to **Site Manager -> Settings -> On-line marketing -> Web Analytics**.
2. Enable the **Log via JavaScript snippet** setting.
3. Save the settings.

Note: The **Log via JavaScript snippet** setting also determines how the system logs web analytics.

Disabling activity logging for particular visitors

By default, the system tracks activities for all contacts. You can disable activity tracking for individual contacts or the user accounts of registered users.

Contacts:
1. Go to **CMS Desk -> On-line marketing -> Contacts**.
2. Find and edit the given contact.
3. Disable **Track activities** on the **General** tab.
4. Click **Save**.

Users:
1. Go to **Administration -> Users**.
2. Find and edit the given user account.
3. Open the **Settings** tab.
4. Disable the **Log activities** property.
5. Click **Save**.

The activities performed by the visitor (or registered user) will no longer be recorded.

Enabling activity logging for specific features

You can choose whether the system logs the related activity types for individual instances of the following items:

Feature	Related activity types	To enable/disable activities
Newsletters	• Newsletter subscription • Newsletter unsubscription • Opened newsletter e-mail • Clicked newsletter link	1. Go to **CMS Desk -> Tools -> Newsletters**. 2. Edit a newsletter. 3. Set **Log on-line marketing activities** on the **Configuration** tab.
Forums	• Forum post • Forum post subscription	1. Go to **CMS Desk -> Tools -> Forums**. 2. Edit a forum group or specific forum. 3. Set **Log on-line marketing activity** on the **General** tab.
Message boards	• Message board	1. Go to **CMS Desk -> Tools -> Message boards**.

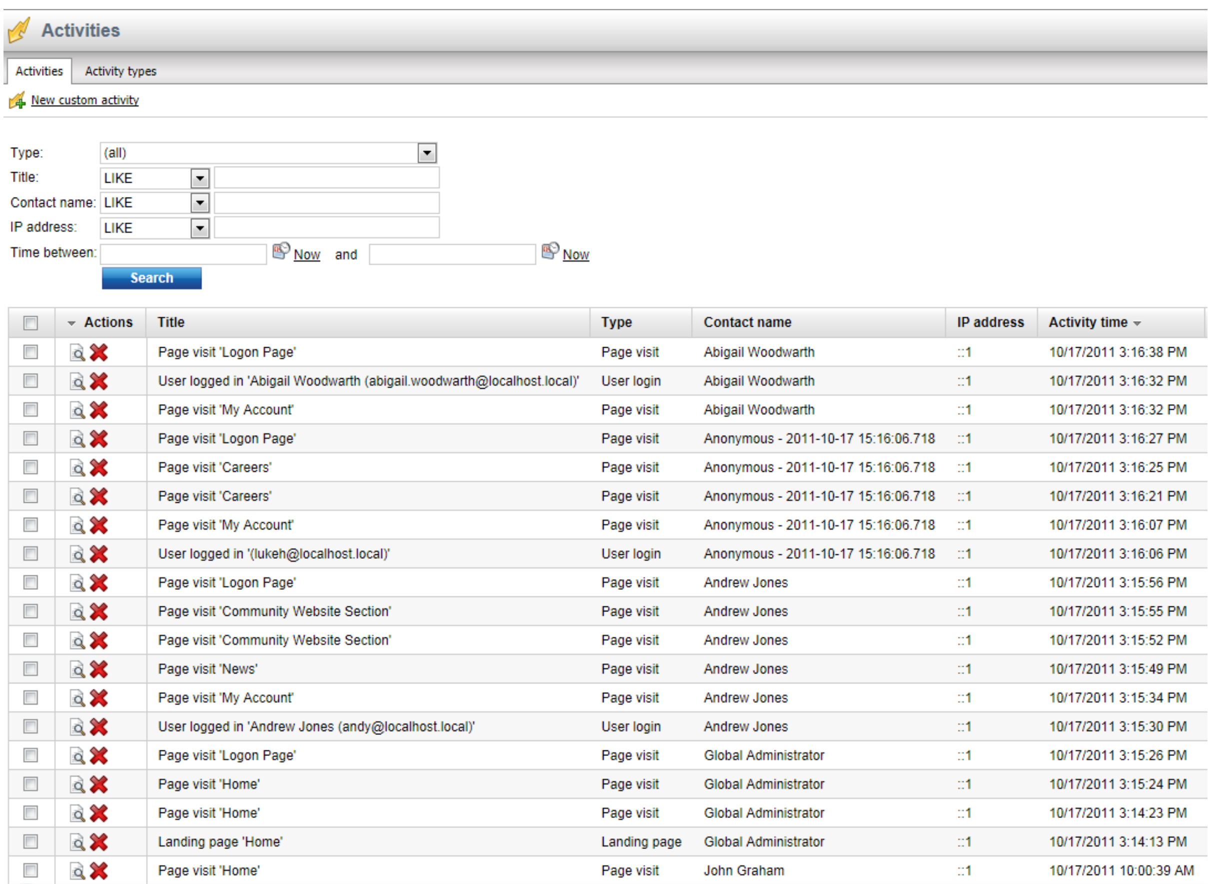
	comment • Message board subscription	2. Select the **Boards** tab. 3. Edit a message board. 4. Set **Log on-line marketing activity** on the **Configuration** tab.
Polls	• Poll voting	1. Go to **CMS Desk -> Tools -> Polls**. 2. Edit a poll. 3. Set **Log on-line marketing activity** on the **General** tab.

6.6.3 Viewing the activity log

Once you have enabled tracking of activities for your website, you can monitor the activities of contacts in:

- **CMS Desk -> On-line marketing -> Activities**
- **Site Manager -> Tools -> On-line marketing -> Contact management -> Activities**

The activity log shows you which actions visitors have performed on the website, such as user registrations, page visits, blog post subscriptions or product purchases. You can filter the list according to the activity type and title, the contact who performed the action, or the date when the action occurred.



Tip: To find the activities of a specific contact, switch to the **Contacts** section, edit () the given contact and open the **Activities** tab.

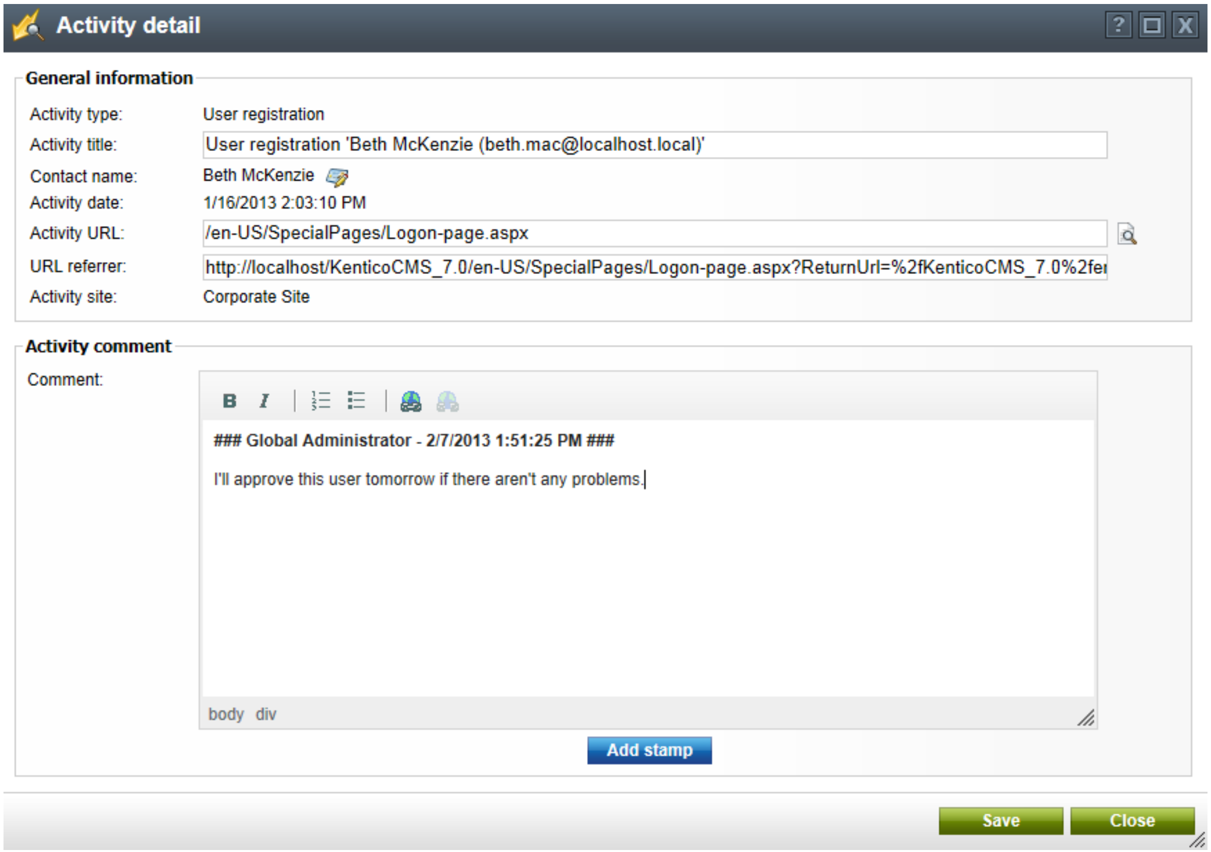
Viewing activity details

Every activity record stores additional details related to the logged event. To see the details of a specific activity, click **View** ( ) next to the given record.

The following activity details are available:

General information	
Activity type	The type of the activity.
Activity title	The activity's title contains basic information about the event. The default text depends on the activity type. You can manually edit this field.
Contact name	The name of the contact associated with the activity. Click to view and edit the contact's details.
Activity date	The date and time when the activity occurred.
Activity URL	The URL of the page where the activity occurred. Click to view the web page on the live site. You can manually edit this field.
Activity URL referrer	The URL from which the contact who performed the activity arrived on the page. You can manually edit this field.
Activity site	The website where the activity occurred.
Activity comment	
Comment	Here you can add any additional information related to the activity, either as plain text or with basic formatting. Click **Add stamp** to insert your stamp into the comment. The default stamp contains the full name of your user account and the current date and time.

Click **Save** to confirm any changes.



Viewing/Editing the details of a logged activity

Removing activity records from the log

To permanently remove activity records, click **Delete** ( ) next to individual items in the log.

You can remove multiple records using the options below the activity list:

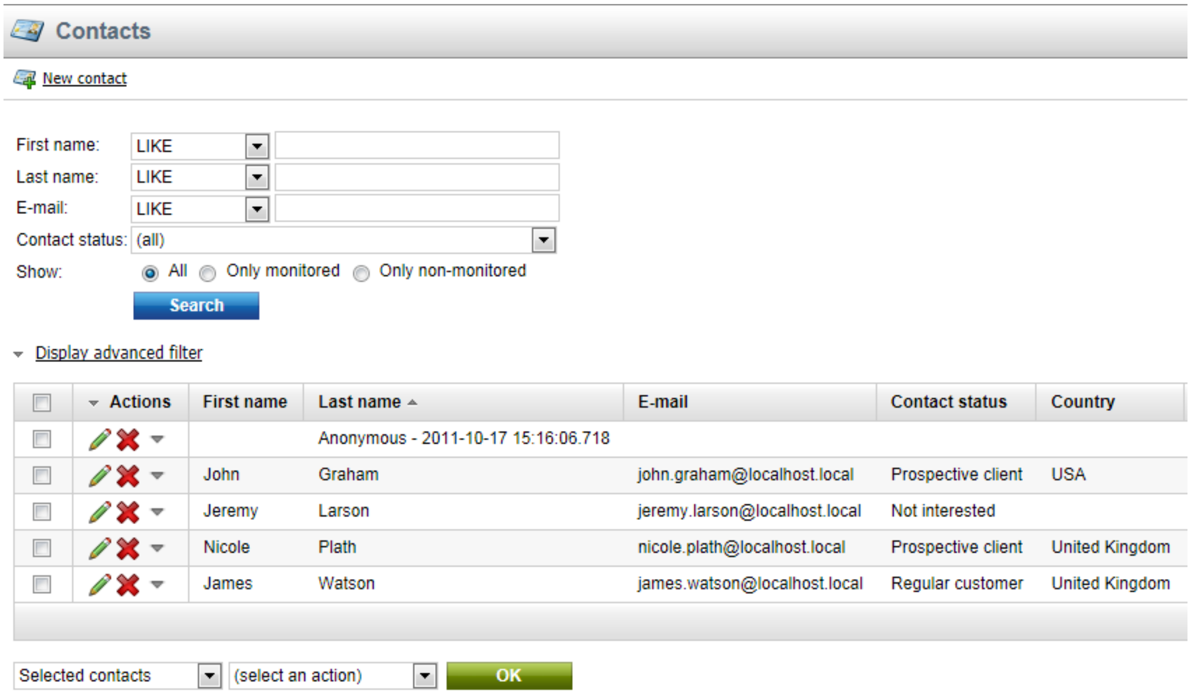
1. Choose which activities to delete:
   a. *All activities*
   b. *Selected activities* - mark specific activities using the checkboxes on the left side of the list.
2. Select the *Delete* action.
3. Click **OK**.

6.6.4 Example: Logging activities

The following example demonstrates how the system logs activities for anonymous and registered users. You can try out the example on the sample Corporate site.

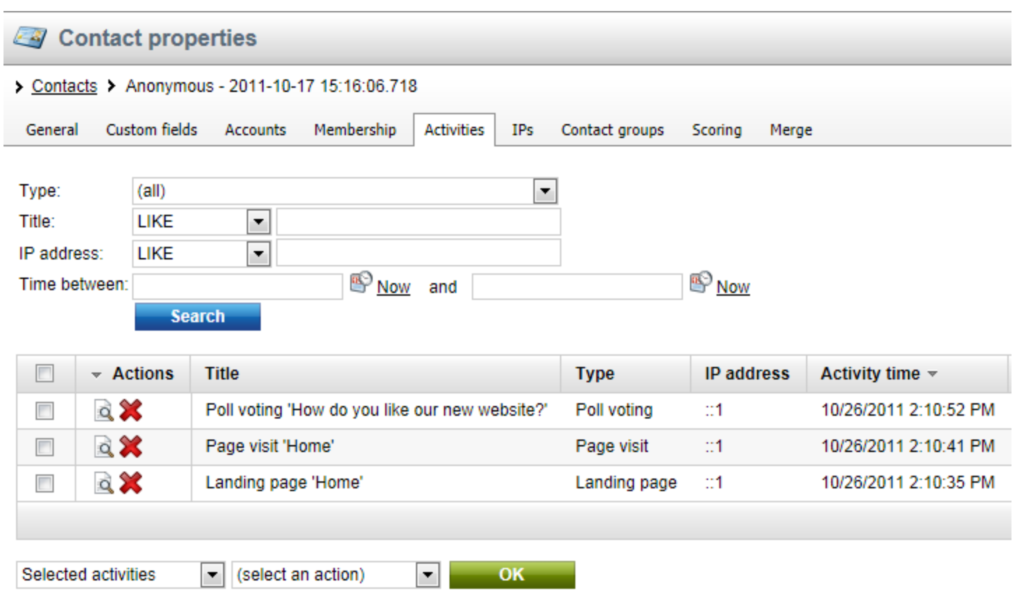
1. Visit the live website as an anonymous visitor and perform some actions, for example view the Home page and vote in the poll.

2. Sign in to CMS Desk as an administrator and navigate to **On-line marketing -> Contacts**.
   - The system has created an anonymous contact.



New Anonymous contact created for the public visitor

3. **Edit** (✏) the new contact and switch to the **Activities** tab.
   - You can see that the system logged the Home page visit and the poll voting activity.



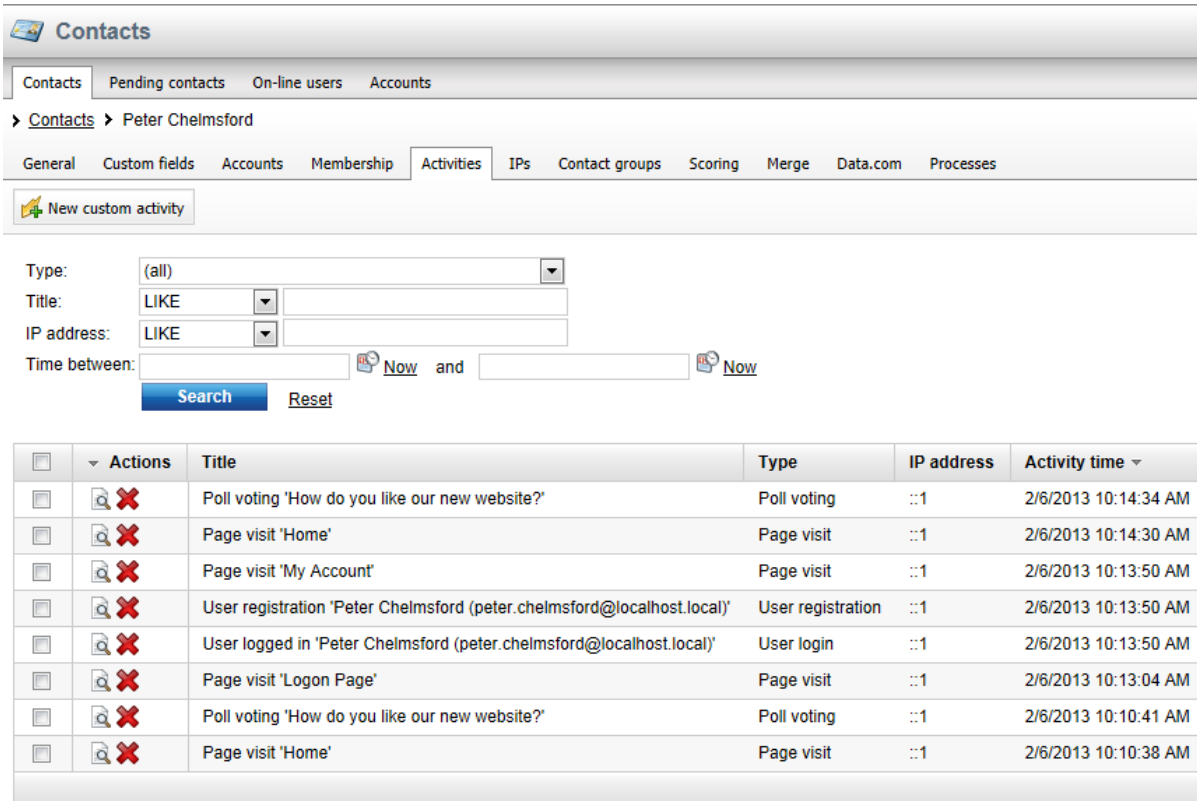
Viewing the contact's activity log

4. Sign out from CMS Desk to view the live site as an anonymous visitor again.

5. Go to the **My account** page and sign up as a new user, for example *Peter Chelmsford*.

6. Perform actions like in Step 1, i.e. view the Home page and vote in the poll.

If you view the list of contacts in the administration interface, you can see that system has renamed original anonymous contact according to the entered registration data. Editing () the contact and opening the **Activities** tab allows you to view the updated activity log for the new user.



Activity log showing the registration of a new user

6.6.5 Adding custom activities

In addition to the default activity types, you can also define custom activity types for tracking any other actions on your website.

Creating new activity types

To add custom activity types:

1. Go to one of the following locations:

- **CMS Desk -> On-line marketing -> Activities -> Activity types**
- **Site Manager -> Tools -> On-line marketing -> Contact management -> Activities -> Activity types**

2. Click **New activity type**.

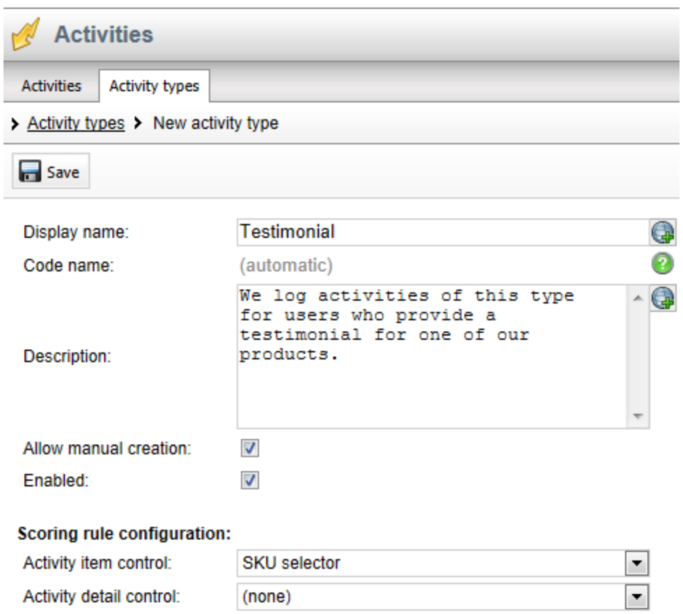
3. Fill in the following fields as required:

Display name	The name of the activity type used in the Kentico CMS user interface.

Code name	Serves as a unique identifier for the activity type (for example in the API) . You can leave the *(automatic)* option to have the system generate an appropriate code name based on the display name.
Description	Text description about the purpose of the activity type.
Allow manual creation	If you enable this option, users with access to the on-line marketing administration interface are allowed to manually log activities of this custom type.
Enabled	Indicates whether the system logs activities of this type.

4. (Optional) Set the **Activity item** and **Activity detail** field options in the **Scoring rule configuration** section.

- The system offers the *Activity detail* and *Activity item* fields when defining scoring rules based on the custom activity.
- For example, an activity type representing testimonials has an *SKU selector* item control. This allows the score administrators to create different scoring rules for testimonials related to different products.
- The *(none)* value hides the given field when creating scoring rules based on the custom activity.
- The *(default)* value provides standard comparison options for text and numeric values.



Defining a new activity type for tracking testimonials

5. Click **Save**.

Your custom activity type is now ready. You can either log the custom activities manually in the user interface, or set up the system to log them automatically.

Manually logging custom activities

To log a single activity of a custom type for a contact:

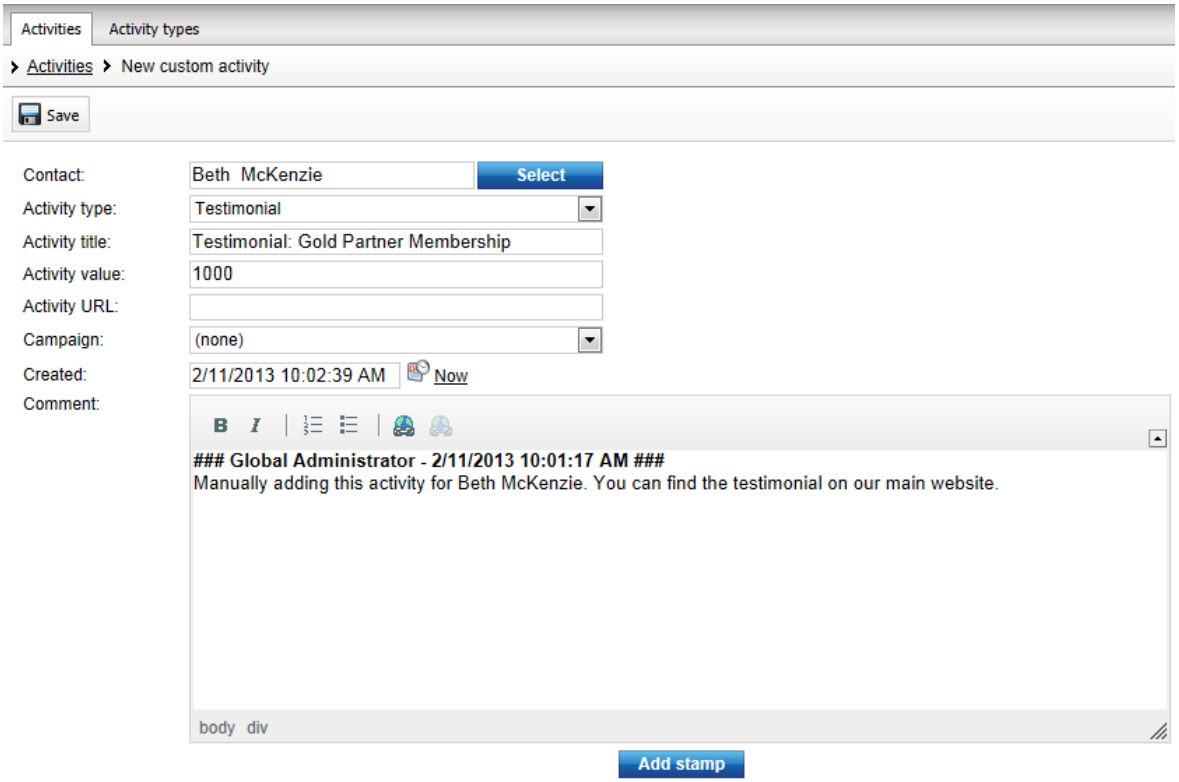
1. Open the activity log in one of the following locations:

- **CMS Desk -> On-line marketing -> Activities**
- **CMS Desk -> On-line marketing -> Contacts -> (edit contact) -> Activities**
- **Site Manager -> Tools -> On-line marketing -> Contact management -> Activities**

2. Click **New custom activity**.

3. Fill in the details of the activity:

Contact	Select the contact associated with the activity.
Activity type	Select one of your custom activity types.
Activity title	Add basic information about the event that the activity represents.
Activity value	Here you can assign a value to the activity. You can associate values with activities such as ratings or e-commerce purchases.
Activity URL	The URL of the page where the activity occurred.
Campaign	The marketing campaign associated with the activity. Used e.g. if you need to monitor the behavior of the visitors of your website.
Created	The time when the activity occurred.
Comment	Here you can add any additional information related to the activity, either as plain text or with basic formatting. Click **Add stamp** to insert your stamp into the comment. The default stamp contains the full name of your user account and the current date and time.



Manually logging a custom activity

4. Click **Save**.

You can now see the custom activity logged for the given contact.

Automatically logging custom activities

In most cases, logging activities manually one-by-one is not a practical option. If you need the system to automatically record custom activities for contacts who perform a specific action, you can use one of the following approaches:

- Log the activity as part of a marketing automation process via the **Log custom activity** action step.
- Develop custom components for your website that log the activity through the API.

6.6.6 Reference: Activity types

By default, you can track the following types of activities for contacts:

Activity type	Logged when
Abuse report	Visitors submit a report using the Abuse report module.
Accept initiated chat request	Visitors accept a chat request initiated by website personnel. See Modules -> Chat -> Support chat in the Developer's Guide.
Blog comment	Visitors comment on a blog post.

Blog post subscription	Visitors subscribe to blog post comments.
Chat request support	Visitors request real time support via chat.
Chat request support using offline form	Visitors submit a support request via e-mail (this option is available when no one from the chat support staff is on-line).
Clicked newsletter link	Subscribers click a tracked link in a newsletter.
Content rating	Visitors rate content (see Content rating).
Customer registration	Customers register during the checkout process.
Event booking	Visitors sign up for an event using the Booking system.
External search	Visitors arrive on the website after searching on a search engine, such as Google.
Form submission	Visitors submit an on-line form.
Forum post	Visitors post on a forum.
Forum post subscription	Visitors subscribe to a forum post.
Internal search	Visitors search for text using the on-site search.
Joining a group	Users join a community group or a workgroup (see Groups).
Landing page	Visitors arrive on the website for the first time on a specific page.
Leaving a group	Users leave a community group or a workgroup.
Message board comment	Visitors post a comment on a message board.
Message board subscription	Visitors subscribe to a message board.
Newsletter subscription	Visitors subscribe to a newsletter.
Newsletter unsubscription	Users unsubscribe from a newsletter.
Opened newsletter e-mail	Subscribers open a tracked newsletter issue.
Page visit	Visitors view a page.
Poll voting	Visitors vote in a poll.
Product added to shopping cart	Visitors add a product to the shopping cart.
Product added to wishlist	Visitors add a product to their wishlist.
Product removed from shopping cart	Visitors remove a product from the shopping cart.
Purchase	Customers make a purchase.
Purchased product	Customers buy a specific product.
User contribution - content created	Users create new content via the user contributions (Wiki) module.
User contribution - content deleted	Users delete content via the user contributions module.

User contribution - content updated	Users update content via the user contribution module.
User login	Visitors log in to a site.
User registration	Visitors register as a member of a site.

6.7 Settings

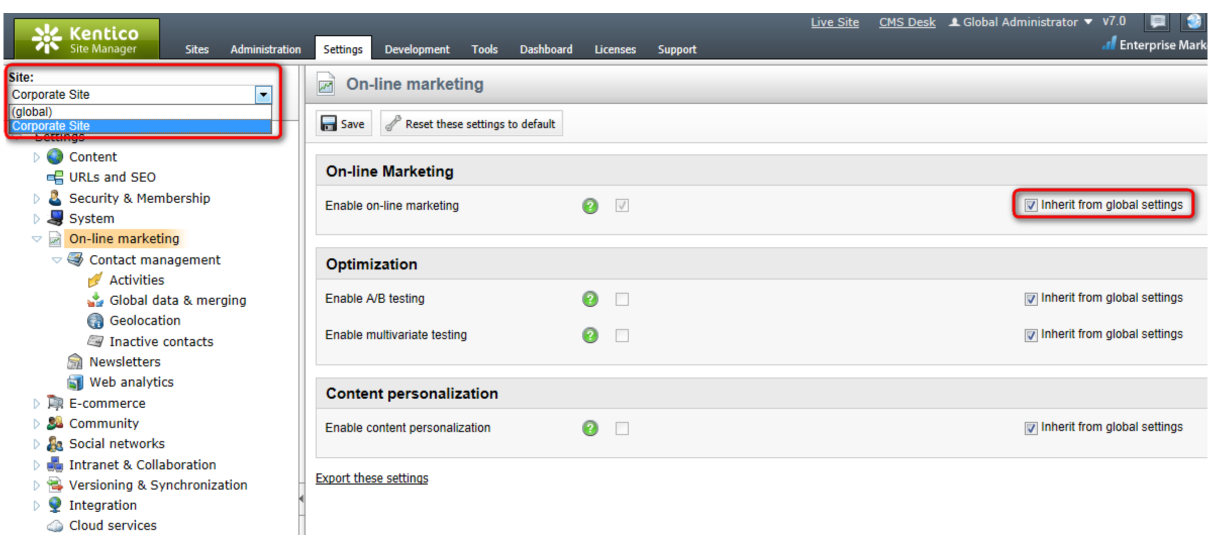
6.7.1 Overview

In this sub-chapter, you will learn about settings which influence and modify the behavior of the Contact management module. These settings can be adjusted in the **Site Manager -> Settings** section of the system's user interface. Here, they are located in the **On-line marketing** settings category and its sub-categories.

The following topics in this sub-chapter contain explanations of settings with related functionality:

- General contact management settings - explains general settings related to the Contact management module.
- Enabling tracking of activities - explains which settings need to be adjusted in order for activity tracking to be functional.
- Allowing the use of global objects - explains what global Contact management objects are, how their use can be enabled.
- Geolocation - explains how you can configure geolocation to automatically fill in contact data based on IP addresses.
- Automatic deleting of contacts - explains which settings may be used to configure the system to periodically delete unnecessary contacts and their activities.

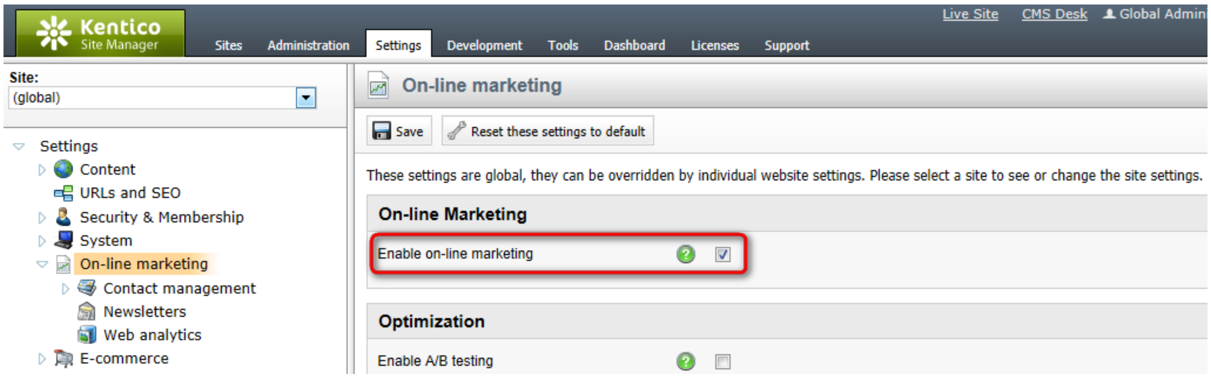
Depending on your selection in the **Site** drop-down list, you can either adjust the settings for a particular website, or globally if the **(global)** option is selected. Global settings can be inherited by individual sites if the **Inherit from global settings** check-box next to a site-specific setting is enabled.



As you can notice if you browse the settings categories, there is a number of other settings not listed in this sub-chapter. These settings have no direct relation to the Contact management features and are explained in respective chapters of this guide dedicated to the features which they influence.

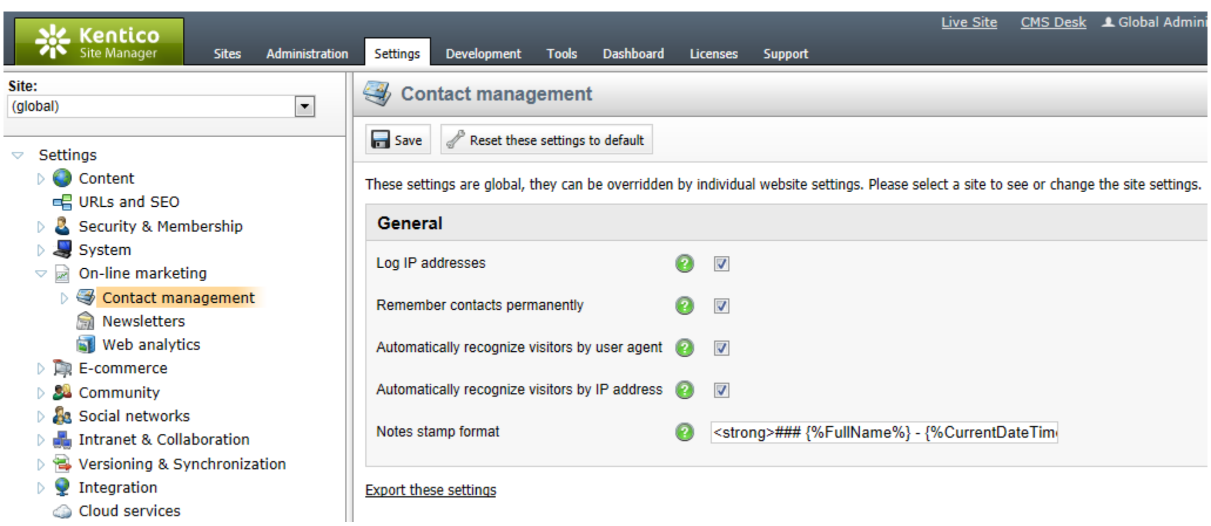
6.7.2 General contact management settings

To enable on-line marketing features, among others including tracking of live site users as contacts and logging of their activities, go to **Site Manager -> Settings**, select the **On-line marketing** category and check the **Enable on-line marketing** box.



General settings related to the **Contact management** module can be adjusted in the corresponding sub-category. The following configuration options are available:

- **Log IP address** - indicates if logging of contacts' IP addresses should be allowed.
- **Remember contacts permanently** - indicates how long contacts should be remembered. If checked, a contact is kept until cookies are deleted or another contact is determined. If unchecked, a contact is kept until the current session expires.
- **Automatically recognize visitors by user agent** - indicates if a visitor's browser user agent information should be used to determine what contact should be assigned to the current anonymous visitor.
- **Automatically recognize visitors by IP address** - indicates if visitor's IP address should be used to determine what contact should be assigned to the current anonymous visitor.
- **Notes stamp format** - defines the format of the stamp added to the **Notes** field after clicking the **Add stamp** button when editing contacts or accounts.



Contact management and web bots

Because they are usually not relevant from a marketing point of view, web bots (such as search engine crawlers) are not tracked via the Contact management module by default.

If you wish to have bots logged in the system as contacts, you can change the default behavior by adding the following key to the *appSettings* section of your projects's *web.config* file:

```
<add key="CMSEnableContactBots" value="true" />
```

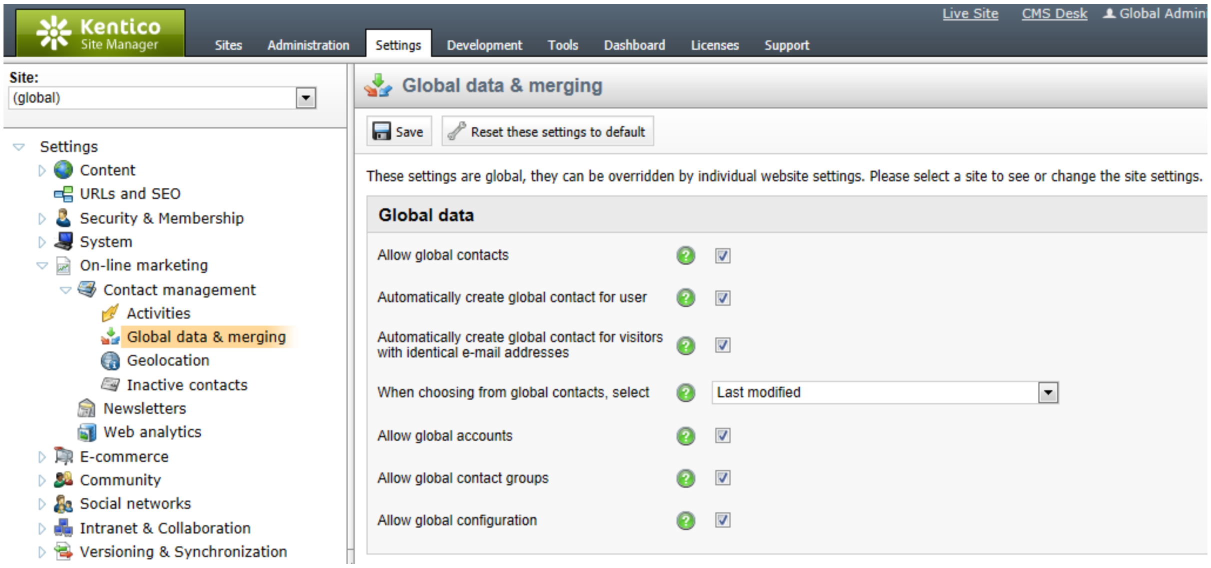
6.7.3 Allowing the use of global objects

The following objects can either be site-specific, i.e. available only for a single website, or global, which means that they are shared across all websites running in the system:

- Contacts
- Accounts
- Contact groups
- Account statuses
- Contacts statuses
- Contact roles

The use of global objects is disabled by default. To enabled it, go to **Site Manager -> Settings -> On-line marketing -> Contact management -> Global data & merging** and adjust the following settings in the **Global data** section:

- **Allow global contacts** - enables the use of global contacts.
- **Automatically create global contact for user** - if enabled and the system detects that there are multiple contacts associated with a user registered on more than one website running in the system, a new global contact will be created and the detected ones merged into it. If one of the detected contacts is already global, the others are merged into it instead of creating a new one.
- **Automatically create global contact for visitors with identical e-mail addresses** - if there are users with the same e-mail address across multiple websites, a global contact will be created for all site contacts associated with the given user.
- **When choosing from global contacts, select** - if there are multiple global contacts associated with a user, this setting determines which of them should be used as the primary one. The following options are available:
  - **Last modified** - the contact that was most recently modified.
  - **First created** - the oldest contact.
  - **Create a new contact** - a new contact will be created and the other associated global contacts will be merged into it.
- **Allow global accounts** - enables the use of global accounts.
- **Allow global contact groups** - enables the use of global contact groups.
- **Allow global configuration** - enables the use of global contact statuses, account statuses and contact roles.

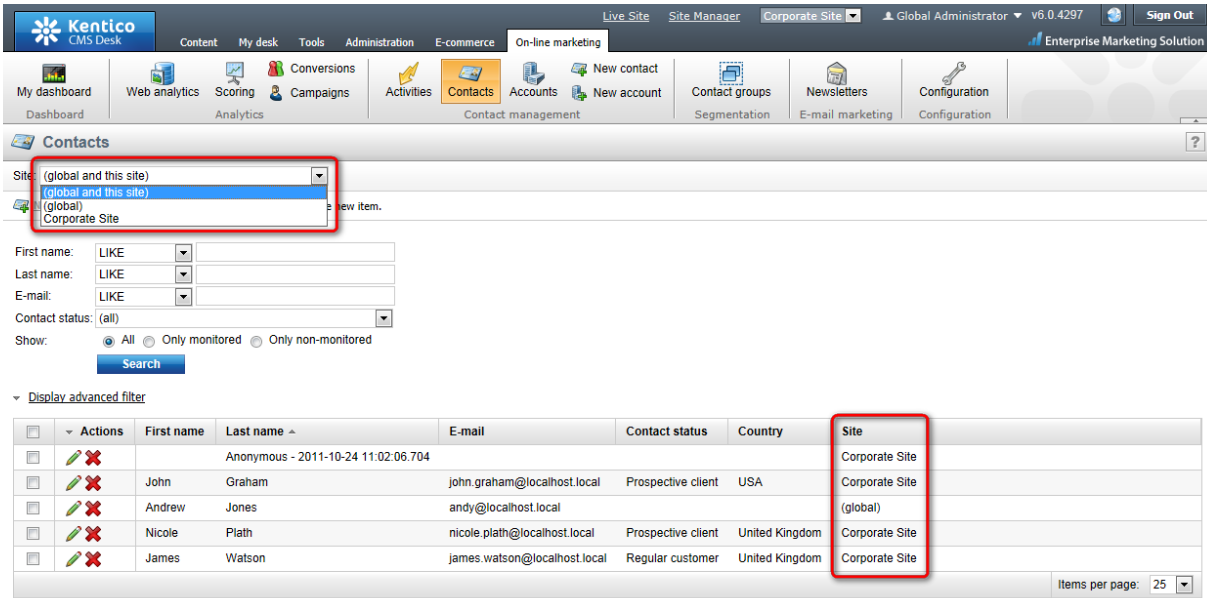


User interface differences when global objects are enabled

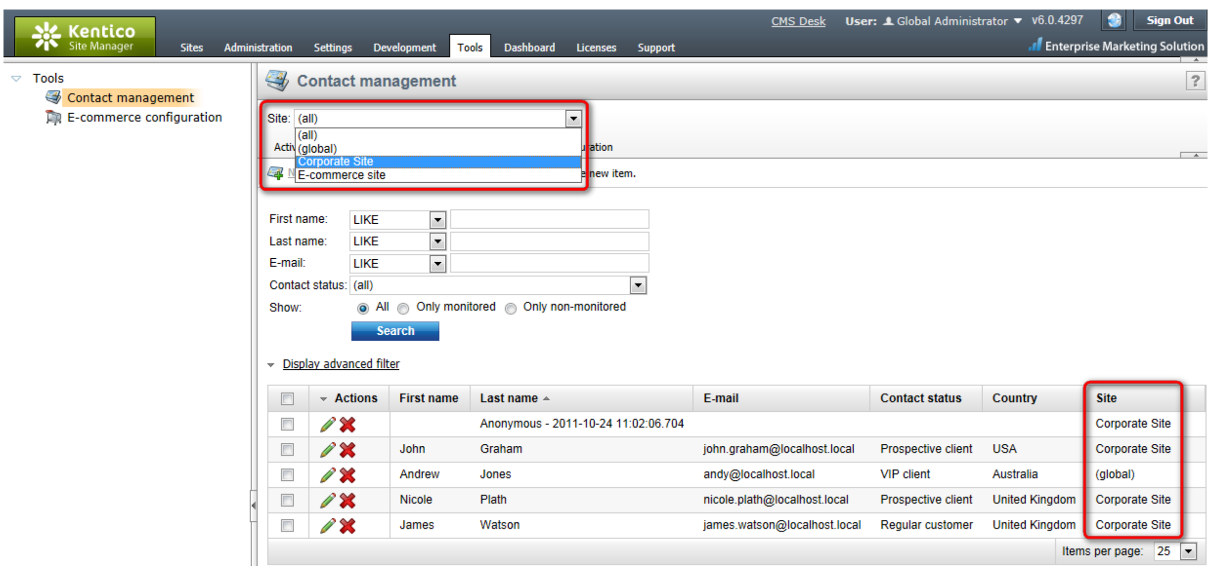
If global objects are allowed, the **Site** drop-down list is visible above the list of objects in the respective sections of CMS Desk (e.g. **CMS Desk -> On-line marketing -> Contacts**). The **Site** drop-down list offers you the following options:

- **(global and this site)** - both global and current site objects are listed.
- **(global)** - only global objects are listed.
- **current site** - only current site objects are listed.

When the **(global and this site)** options is selected, the **Site** column is added to the grid, indicating if the object is site-specific or global. It is also not possible to create new objects in this case since it would not be clear whether the new object should be global or site-specific. To create a new object, you either need to have a particular website selected in the **Site** drop-down list, in which case you create an object specific for the website, or you need to select the **(global)** option, in which case you create a global object.



If you manage contact management objects in **Site Manager -> Tools -> Contact management**, the **Site** drop-down list is always present and allows you to manage both global objects and objects of all websites in the system. The same conditions as described above apply here as well.



6.7.4 Geolocation

In this topic you can find information about the configuration and use of Geolocation in Kentico CMS. You can use Geolocating to determine the origin of your website's visitors based on their IP address. To obtain the information, the feature makes use of MaxMind's GeoIP databases. GeoLite City, the more accurate of the free IP geolocation databases, is used by default. You can extend the accuracy of the feature by downloading additional databases.

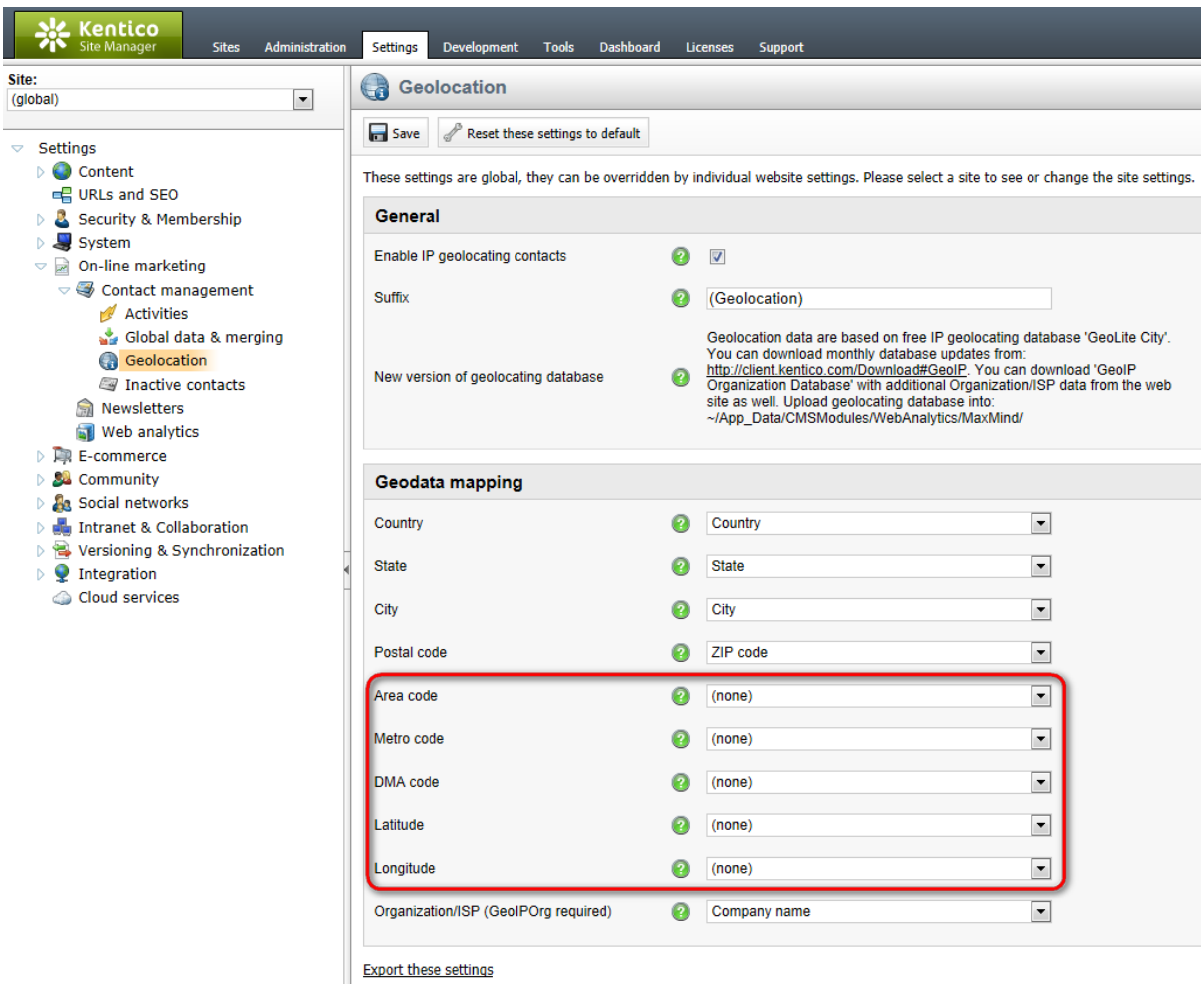
Please note

The origin of a contact is determined only during its first visit to the website. That is, the information does not get updated when the same contact accesses the website again from a different location.

Configuration

To use the Geolocation feature, access the settings in **Site Manager -> Settings -> On-line marketing -> Contact management -> Geolocation**. There, configure the mapping of the GeoIP fields to Kentico CMS contact fields. Use the drop-down lists to assign the desired target columns into which the information will be mapped. Use the other settings in the **General** section as follows:

- **Enable IP geolocating contacts** - turn the checkbox on to track the location of contacts based on their IP address.
- **Suffix** - type in an optional suffix, which will be added to each of the contact's field values obtained using Geolocating. The suffix will be added to text-based fields only.



Geolocation settings

The fields highlighted in the preceding image do not have an equivalent in Kentico CMS system table fields. To create new fields to map into, navigate to **Site Manager -> Development -> System tables** and **Edit ( )** the **OM.Contact** class, where you can add new fields. Refer to Developer's Guide -> Development -> System tables and custom fields for more information.

Mind the following type restrictions when assigning target fields and creating custom fields to assign later. Also keep in mind that only text-based fields can have a suffix added to them.

Field	Only accepts
Country	Integer-based and text-based columns. When mapping to an integer-based column, *Country ID* is used. *Country display name* is mapped into a text-based column.
State	Integer-based and text-based columns. When mapping to an integer-based column, *State ID* is used. *State display name* is mapped into a text-based column.
City	Text-based columns.
Postal code	Integer-based and text-based columns.
Area code	Integer-based and text-based columns.
Metro code	Integer-based and text-based columns.
DMA code	Integer-based and text-based columns.
Latitude	Decimal-based and text-based columns.
Longitude	Decimal-based and text-based columns.
Organization/ISP (GeoIPOrg required)	Accepts text-based columns. The specified column will only be updated if the GeoIP Organization Database is installed.

Managing your MaxMind databases

You can improve the functionality of geolocation by using more detailed and accurate databases and by keeping your current databases up to date.

Adding GeoIP City

The **GeoIP CIty** database provides more accurate results than the default GeoLite City database.

1. Obtain the GeoIP City database.
2. Place the database file into the *~/App_Data/CMSModules/WebAnalytics/MaxMind/* folder of your web project.
3. Configure the system to use the new geolocation database by adding the following key to the appSettings section of your web.config file.
    - Specify the name of the database file as the key's value.

```
<add key="CMSGeoIPLocationFileName" value="GeoIPCity.dat" />
```

Geolocation now uses the GeoIPCity.dat database instead of the default GeoLiteCity.dat database.

Adding GeoIP Organization

The GeoIP organization database allows the system to determine the organization name of visiting contacts.

1. Obtain the GeoIP organization database.
2. Place the database file into the *~/App_Data/CMSModules/WebAnalytics/MaxMind/* folder of your web project.

Note: The name of the database file must be GeoIPOrg.dat.

Geolocation now supports organization detection.

Updating existing databases

To ensure that the geolocation data is accurate, you need to regularly update your MaxMind databases. You can download the newest versions of the databases from the Kentico Client Portal. The databases are updated in the following intervals:

- **GeoLite City -** the first Tuesday of each month
- **GeoIP City** - weekly
- **GeoIP Org** - monthly

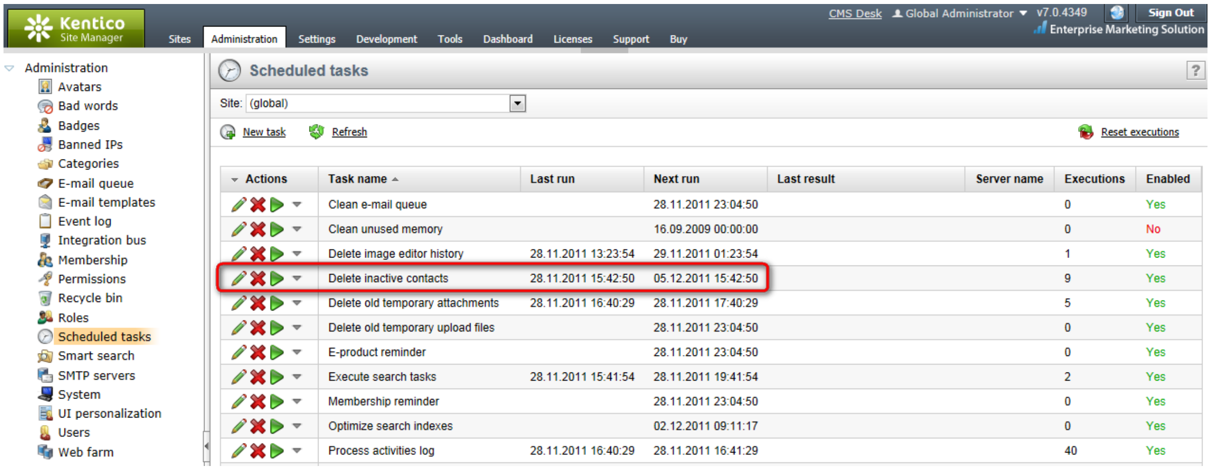
To update your databases, download the newest version and upload the database file to the *~/App_Data/CMSModules/WebAnalytics/MaxMind/* folder of your web project.

6.7.5 Automatic deleting of contacts

The contact management module generates a very large amount of data, particularly on high-traffic websites. Contacts are created and activities logged for every visitor, which may be overwhelming for your marketers. For this reason, the system can be configured to regularly remove contacts that meet certain conditions (and the activities logged by these contacts). This way, you can filter out inactive, outdated or otherwise unnecessary contacts without having to remove them individually.

This functionality is provided by a global Scheduled task named **Delete inactive contacts**, which is executed once per week by default. If you wish, you can change this interval (and other properties) by editing the task in **Site Manager -> Administration -> Scheduled tasks**. You can also run the task manually at any time.

The task runs in a separate thread by default and the deleting itself is done by the database server, so it should not have a negative impact on the performance of your website, even when removing a large number of contacts.



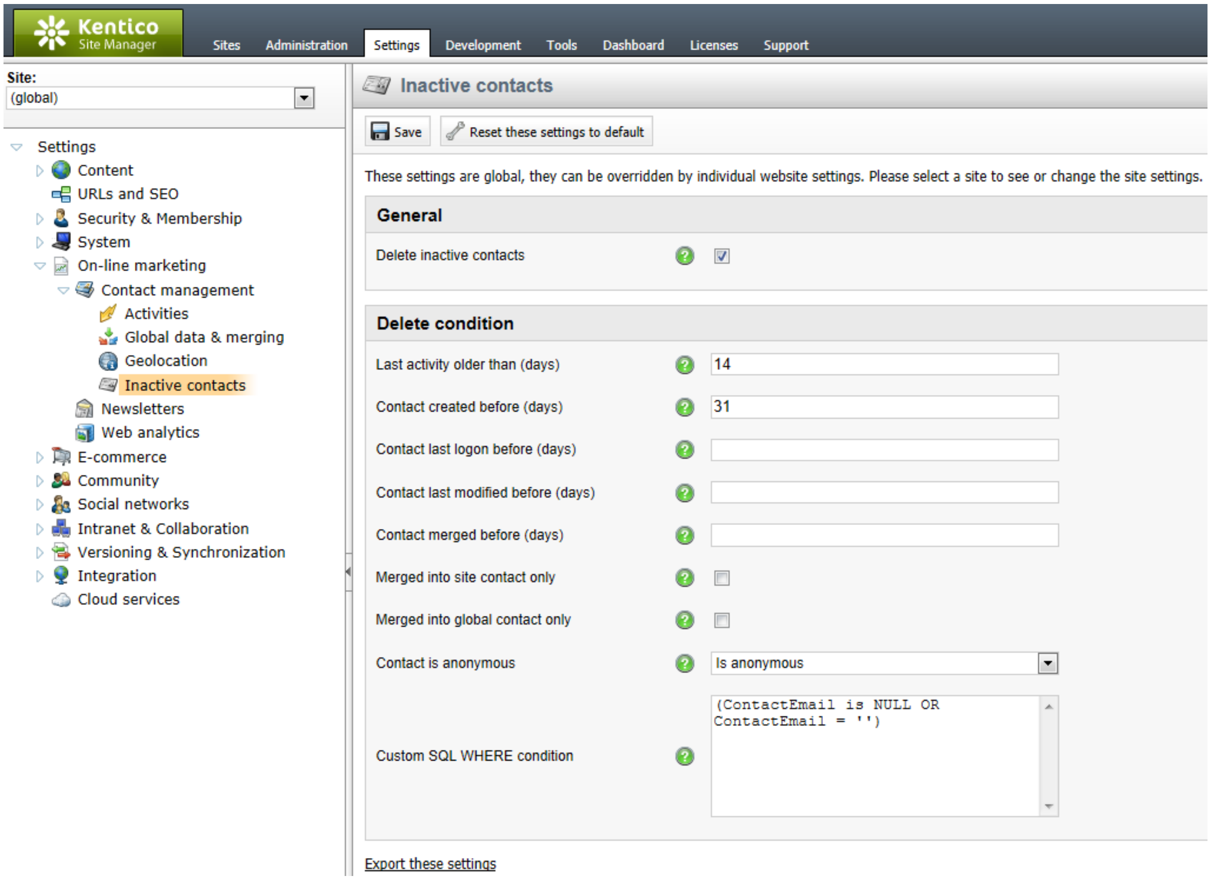
When the system executes this task, it starts a deleting process for each site according to the rules defined by the settings in the **Site Manager -> Settings -> On-line marketing -> Contact management -> Inactive contacts** category. The task does *not* delete global contacts.

The **Delete inactive contacts** setting determines whether the contacts associated with the given site should be affected by the scheduled task. If disabled, the site's contacts will only be removed if they are deleted manually. Additionally, the **Enable on-line marketing** setting must also be enabled for every site where you wish to use this task.

The remaining settings in the section are used to set the conditions that specify which contacts should be deleted. At least one condition must be entered in order for any contacts to be deleted. If multiple conditions are set, only those contacts that fulfill all of them will be removed. The following conditions are available:

Last activity older than (days)	Can be used to delete contacts that do not have any recent activities logged. Contacts whose latest activity is older than the specified number of days are removed. For example, entering *14* means that the task removes all contacts which do not have any activities logged within the last two weeks.
Contact created before (days)	Can be used to clear out old contacts. All contacts that are older than the specified number of days are removed. For example, setting the value to *365* means that the task removes all contacts created more than a year ago.
Contact last logon before (days)	This condition is only applied to contacts that are not anonymous (i.e. only those that are associated with a specific user account). It can be used to delete contacts who have not logged into the website recently. For example, entering *31* means that the task removes all contacts who have not logged in within the last month.
Contact last modified before (days)	Can be used to remove contacts that were not edited recently (e.g. had their contact address changed). Contacts whose latest modification is older than the specified number of days are deleted.

	For example, entering *31* means that the task removes all contacts which were not modified within the last month.
Contact merged before (days)	Can be used to delete contacts that were merged into another contact a certain number of days ago. A merged contact is one that was combined into another contact, not the contact which is the actual result of a merge operation. For example, entering *7* means that the task removes all contacts which were merged more than one week ago.
Merged into site contact only	Determines whether the task should delete all contacts that were merged into another contact associated with the given site.
Merged into global contact only	Determines whether the task should delete all contacts that were merged into a global contact.
Contact is anonymous	Can be used to choose whether the task should remove all contacts that are anonymous, or the opposite (those that are related to a specific user account). Please note: Contacts that are only related to a customer or subscriber are still considered as anonymous. The **Doesn't matter** option is the equivalent of an empty value in this condition. (i.e. if selected, the task does not delete any contacts unless at least one other condition is specified).
Custom SQL WHERE condition	Allows you to enter an SQL WHERE condition that determines which contacts are deleted by the scheduled task. For example: *(ContactEmail is NULL OR ContactEmail = '')* With this custom condition, the **Delete inactive contacts** task removes contacts that have an empty e-mail address (and fulfill all other delete conditions).



It is recommended to be very careful when configuring these conditions. Make sure that the system only deletes those contacts that are no longer necessary, otherwise you may lose valuable marketing data.

6.8 Database separation

6.8.1 Overview

When using contact management on high-traffic websites, the system will need to store a very large volume of data in order to keep records of all contacts, their activities and other related information. This may complicate database administration and make it difficult to perform regular backups. For these reasons, Kentico CMS provides a way to move all tables and database objects associated with contact management to a separate database (including the currently stored data). Doing so allows easier management of the application's main database, without the need to deal with the contact data.

Please note that there can only be one separated contact management database for the entire system, even when hosting multiple websites. It is not possible to create a different database for each site.

The steps needed to perform the database separation are described in the Separating the contact management database topic. If you wish to reverse the process and return everything back to a single database, you can do so by following the instructions in Rejoining the databases.

The separation procedure affects the following database tables:

- OM_Account

- OM_AccountContact
- OM_AccountStatus
- OM_Activity
- OM_ActivityType
- OM_Contact
- OM_ContactGroup
- OM_ContactGroupMember
- OM_ContactRole
- OM_ContactStatus
- OM_IP
- OM_Membership
- OM_PageVisit
- OM_Rule
- OM_Score
- OM_ScoreContactRule
- OM_Search
- OM_UserAgent
- Newsletter_Link
- Newsletter_OpenedEmail
- Newsletter_SubscriberLink

Refer to the database information in the Internals and API chapter to learn more about specific tables.

6.8.2 Separating the contact management database

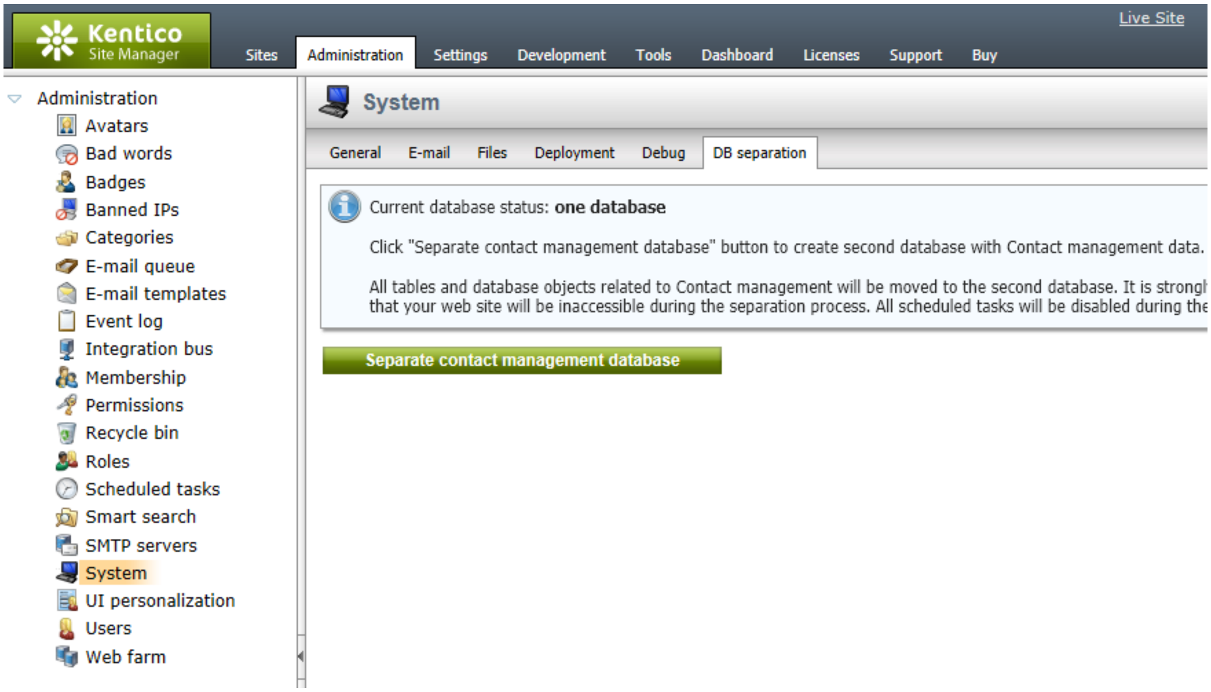
This topic demonstrates how to perform the separation of the system's contact management data into a different database. It is strongly recommended to backup your database before starting this process. Additionally, the database will be unavailable while the actual separation is being carried out, so it should be done at a time when no visitors or editors are accessing the websites in the system.

Database separation on Windows Azure

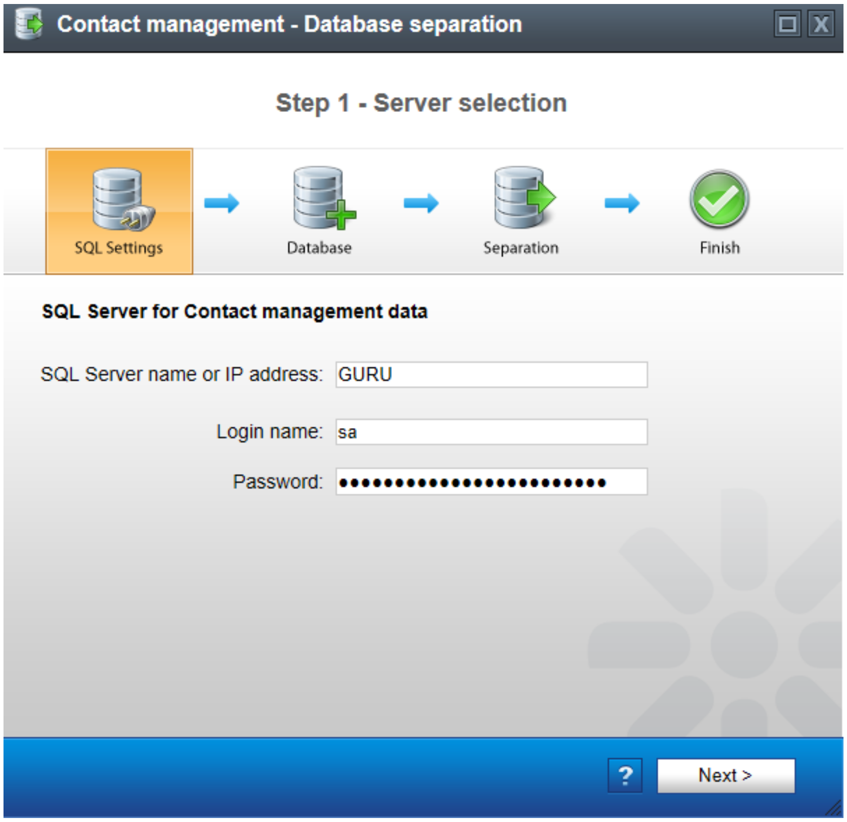
If your application runs on Windows Azure, you must manually perform some parts of the procedure that would normally be automatic:

- Create a new database before you launch the separation. The process cannot automatically create the separated database.
- Move data to the separated database. The process does not move the data, it only creates the database structure.
- Delete the separated tables from the original database.

1. Log in to **Site Manager** and go to **Administration -> System**. Select the **DB separation** tab and click the **Separate contact management database** button. This starts the database separation wizard.



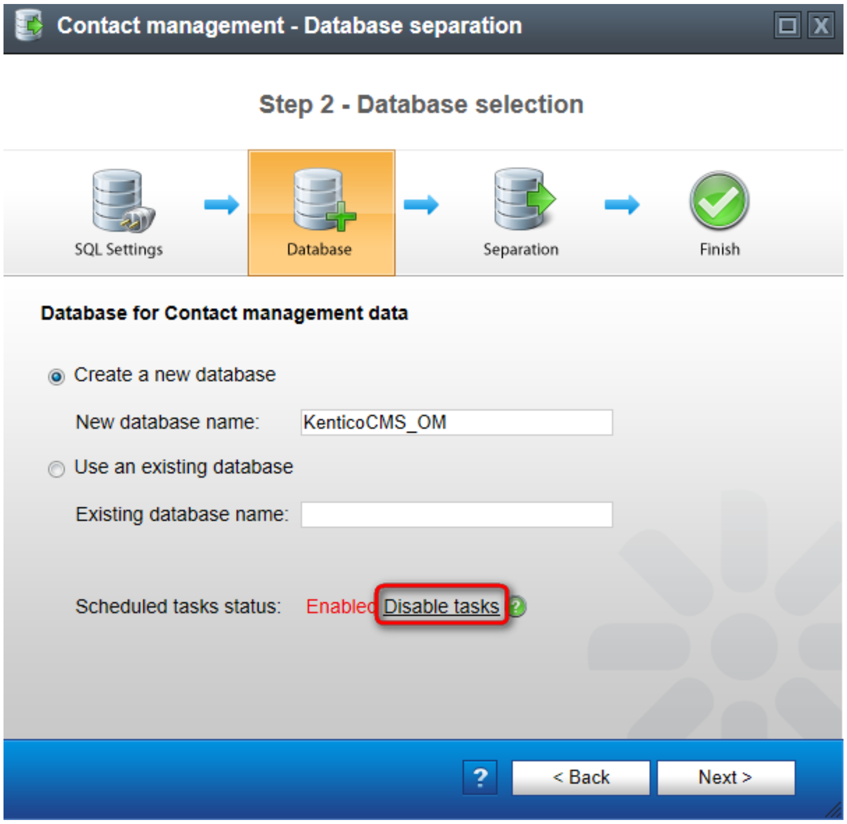
2. In the first step of the wizard, specify the SQL server to which you wish to transfer the contact management database and enter the credentials used to connect to it. The server can be identified through its name or IP address. It is possible to use a different server than the one containing the main database. The separated database may be hosted on all server types supported by Kentico CMS, i.e. Microsoft SQL Server 2005 or 2008 (including the free Express Edition).



Continue by clicking the **Next** button.

3. In the next step, choose to either create a new database for the contact management DB objects and data, or import them into an existing one on the server specified in the previous step. In both cases, you need to enter the name of the database into the appropriate field. When using an existing database, make sure it does not contain any of the tables that are included in the separation.

Before continuing, it is necessary to prevent all Scheduled tasks in the system from running, because the database will be unavailable during the separation process. You can stop all tasks by clicking the **Disable tasks** button (in case there are any enabled). If a task is currently being executed when the button is clicked, the disabling action waits until it is finished. The tasks will automatically be enabled again when the separation is completed.



Once the target database is specified and the scheduled tasks are disabled, click **Next** to begin the separation.

4. A log is displayed while the database operations are being performed. During this stage, all websites in the system are temporarily switched to Off-line mode. Any regular visitors who arrive at this time will see a message informing that the site is down due to maintenance. For global administrators, the current state of the separation wizard is shown.

5. When the process is completed, all contact management data is transferred to the new database. If you view your application's *web.config* file, you can notice that an additional connection string for the new contact management database is now present in the <*connectionStrings*> section:

```
<add name="CMSOMConnectionString" connectionString="..." />
```

6. By clicking the **Finish** button in the final step of the wizard, you confirm that the separation was successful and that the data (including the database table structure and other related objects) can be deleted from the main database. The system brings all websites back on-line and enables the inactive scheduled tasks.

Database separation on web farms

If your application is running on multiple servers in a Web farm environment, you need to manually copy the **CMSOMConnectionstring** key into the *web.config* files of all other servers, otherwise they will not be functional.

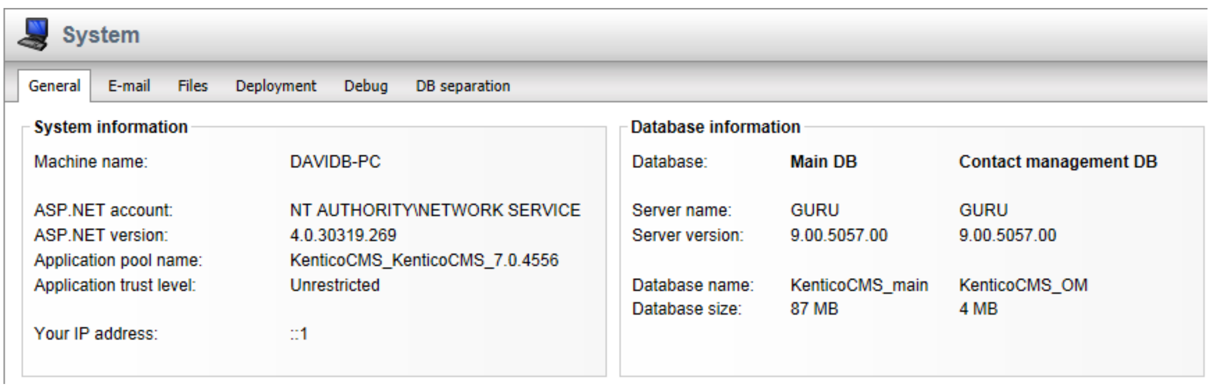
Database separation in a medium trust environment

If your application is running in a medium trust environment, the system cannot enable the scheduled tasks and return the off-line sites back on-line. To manually perform these tasks:

1. Go to **Site Manager -> Settings -> System** and enable the **Scheduled tasks enabled** setting.
2. In **Site Manager -> Sites**, edit your websites on the **Off-line mode** tab.
3. Click **Bring the site on-line**. Repeat the same for each site.

Result

The new database status is reflected on the **General** tab of the **Site Manager -> Administration -> System** page in the **Database information** section.

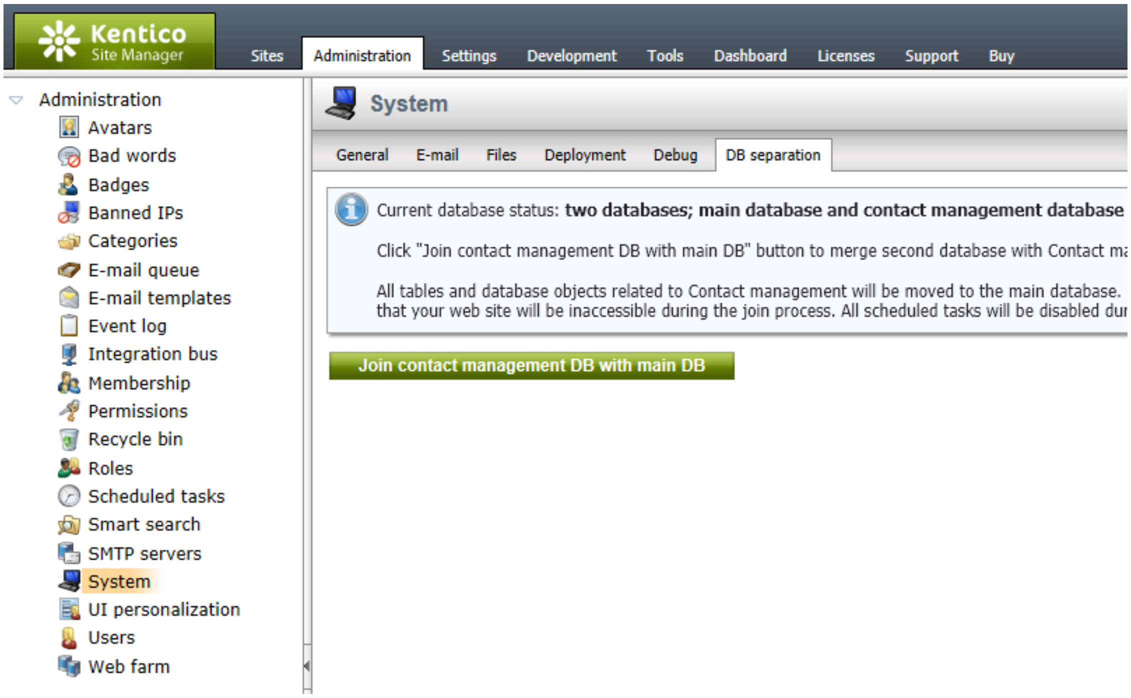


All functionality of the website and administration interface remains the same as before the separation.

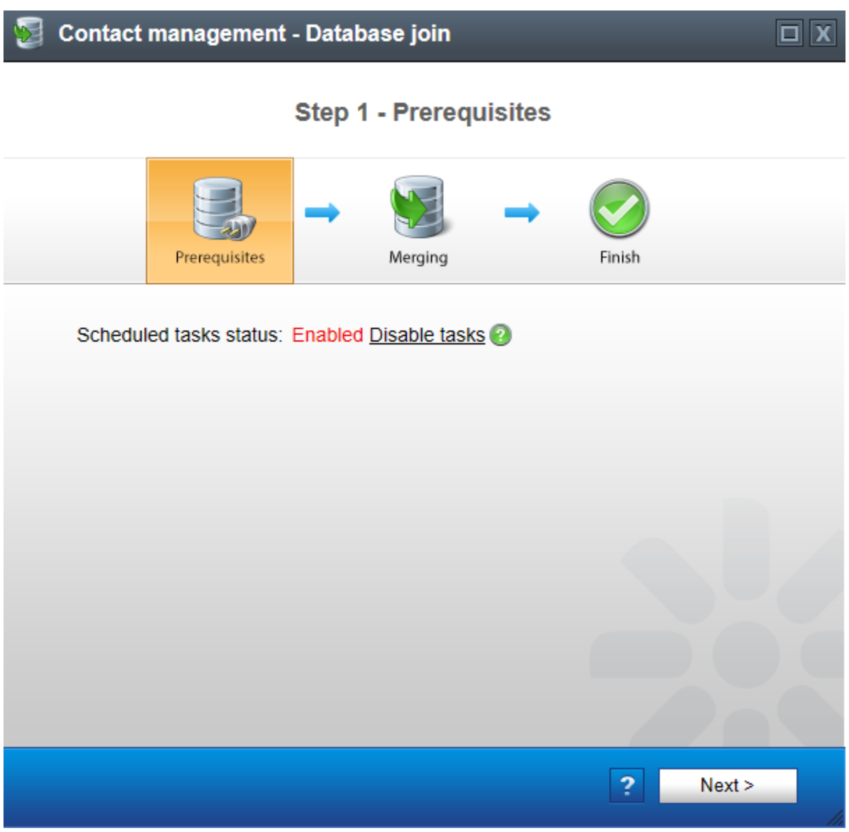
6.8.3 Rejoining the databases

The following steps describe how to return the separated contact management database objects and data back to the application's main database. We recommend backing up both databases before starting the procedure. Additionally, it should be performed at a time when no visitors or editors are accessing the websites in the system.

1. Go to **Site Manager -> Administration -> System**, select the **DB separation** tab and click the **Join contact management DB with main DB** button to open the **Database join** wizard.

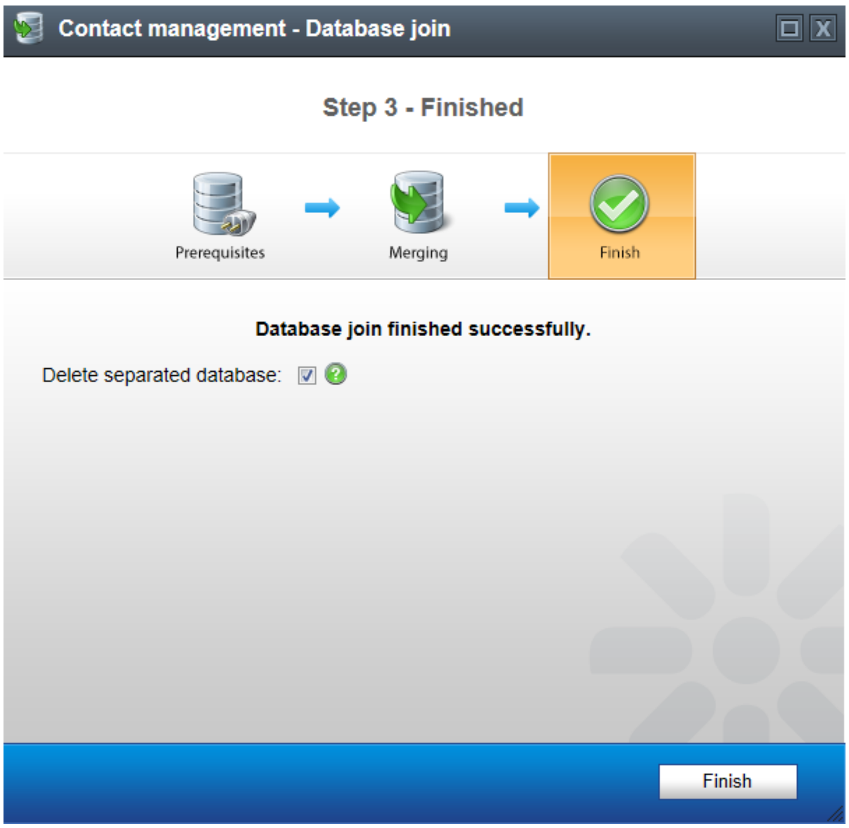


2. Like when separating the database, all [Scheduled tasks] need to be stopped before the joining process can begin. To do this, click the **Disable tasks** button. The tasks will automatically be enabled again after the databases are merged.



Click **Next** to continue. A log is displayed showing the progress of the required database operations. During this stage, all websites in the system are temporarily unavailable.

3. When the process is successfully finished, all tables and data from the dedicated contact management database will again be a part of the application's main database. If you leave the **Delete separated database** option checked in the final step, the separated database is completely removed from the server after you click **Finish**. The system brings all websites back on-line and enables the inactive scheduled tasks.



Completing the wizard automatically deletes the **CMSOMConnectionString** connection string key from the application's *web.config* file.

Database rejoining on web farms

When reversing the database separation for an application running on a Web farm, it is necessary to manually remove the **CMSOMConnectionstring** from the *web.config* files of all other servers.

Database rejoining in a medium trust environment

If your application is running in a medium trust environment, the system cannot enable the scheduled tasks and return the off-line sites back on-line. To manually perform these tasks:

1. Go to **Site Manager -> Settings -> System** and enable the **Scheduled tasks enabled** setting.
2. In **Site Manager -> Sites**, edit your websites on the **Off-line mode** tab.

3. Click **Bring the site on-line**. Repeat the same for each site.

Database rejoining on Windows Azure

If your application runs on Windows Azure, you must manually remove the separated tables after you rejoin the database.

Part VII

Marketing automation

7 Marketing automation

You can automate, optimize and analyze your campaigns using marketing automation. Marketing automation allows you to nurture your website's visitors and leads — represented by contacts in Kentico.

Marketing campaign are controlled through one or more automation processes. Each process consists of a set of steps that you can fully customize using a built-in visual designer. You can either start processes manually for individual contacts or groups of contacts, or have the system start processes automatically using condition-based triggers. For example, you can:

- Create a trigger that moves every new contacts from a specific country into an automation process, which then performs a targeted marketing campaign.
- Automatically move contacts who have reached a certain score into an advanced automation process.

Topics:

- Creating processes
- Working with the process designer
  - Managing steps
  - Connecting steps
  - Defining condition steps
  - Defining user choice
  - Defining automatic decisions
  - Defining timeouts and waiting
  - Configuring security
  - Adding action steps
- Creating triggers
- Manually running automation processes
- Managing the flow of contacts in processes
- Viewing process reports
- Versioning
- Defining custom actions
- Walkthrough: Creating a sample process

7.1 Creating processes

This topic describes how to add new marketing automation processes.

1. Log in to **CMS Desk** and navigate to **On-line marketing -> Processes**.
   - You can alternatively create new processes in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. Click **New process**.

3. Choose a name for the new process and type it as the **Display name**.

4. Click **Save**.
   - The system creates the process and opens the **General** tab of its editing interface.

5. Under **Start the process**, choose one of the following process recurrence options:

Note: Process recurrence determines when the process is allowed to run for each contact.

- **Always as a new instance** - the system can create and start new instances of the process at any time for the same contact.
- **If it hasn't been run before** - the process can only run once for a given contact (i.e. starts only if it has never been run for the contact before).
- **If the same process is not running concurrently** - the process starts only if there is no instance of the same process running for the contact at the same time.

> **Warning!**
>
> If you select the **Always as a new instance** recurrence option, the system may run multiple unnecessary instances of the process concurrently (depending on the implementation of the process and its triggers).
>
> This can lead to increased system load and deliver the same marketing messages to the same contact multiple times.

6. Click **Save**.

Once you create a process, you can proceed to working with the process designer or create triggers.

Tip: You can change the recurrence settings of a process at any time by editing () the process in **On-line marketing -> Processes**.

7.2 Working with the process designer

This chapter describes the process designer, a built-in tool that provides a visual representation of automation processes. The designer allows you to define the steps that implement the functionality of automation processes.

Accessing the process designer

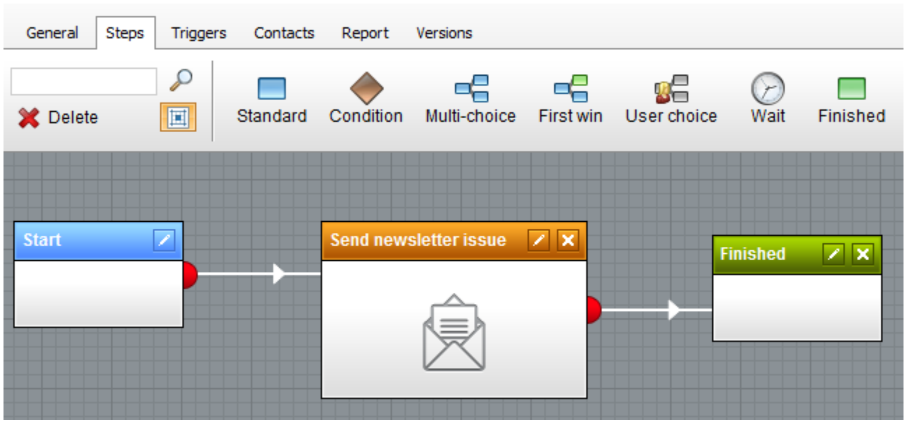
To open the process designer for an automation process:

1. Log in to **CMS Desk** and navigate to **On-line marketing -> Processes**.
    - You can also access the process designer in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. **Edit** () the process for which you want to access the designer.

3. Switch to the **Steps** tab.

The following image shows the appearance of the designer for an automation process consisting of the *Start*, *Send newsletter issue* and *Finish* steps. The steps are connected by transition lines. The direction of the arrows indicates which way the process moves forward.



Process with Start, Send newsletter issue and Finish steps

Tip: If your graph doesn't fit into the designer area, click an unoccupied area of the grid and drag to expose additional free space.

Standard steps

You can use the following types of basic steps to design automation processes:

Step	Description
Standard	Basic approval step. Allows designated users or roles (e.g. marketing managers) to decide whether the process continues to the next step.
Condition	Splits the process into two branches based on a condition. The process automatically transitions to the next step according to the result of the condition. See: Defining condition steps
Multi-choice	Splits the process into any number of branches based on a set of conditions. • If exactly one condition is met, the process automatically continues through the given branch. • If multiple conditions are fulfilled, designated users need to manually choose the next step. See: Defining automatic decisions
First win	Splits the process into any number of paths based on a set of conditions. The process automatically continues through the branch whose condition is met first. See: Defining automatic decisions
User choice	Splits the process into any number of branches. Designated users need to manually choose the next step for each contact in the process.

Wait	Halts the process for a specified amount of time before continuing to the next step. See: Defining timeouts and waiting
Finished	Represents the end of the process.

You can also embed actions into processes. When a contact reaches an action step in a process, the system automatically performs the given action.

Action step	Description
Change contact account	Adds or removes the contact from an account.
Change contact group	Adds or removes the contacts from a contact group.
Delete contact	Deletes the contact.
Import to Salesforce	Replicates the contact into a lead in a target Salesforce organization. You need to have Salesforce replication set up for your website to use this action.
Log custom activity	Logs a custom activity for the contact.
Newsletter subscription	Subscribes or unsubscribes the contact from a newsletter.
Send e-mail	Sends an e-mail (to any address).
Send newsletter issue	Sends a newsletter issue to the contact. **Note**: When performing this action step, the process automatically adds the contact as a subscriber for the newsletter containing the specified issue.
Set contact property	Inserts a new value into one of the contact's properties (fields).
Set contact status	Changes the contact's status.
Start process	Triggers a different marketing automation process for the contact. Based on the recurrence settings of the target process, this action can result in multiple unnecessary instances of the process running concurrently.
Update from Data.com	Updates the contact's information with the data found in the Data.com database. You need to have Data.com integration set up for your website to use this action.

Note: If your developers create custom actions, you can add them through the designer as well.

7.2.1 Managing steps

Every marketing automation process consists of a set of interconnected steps. Using steps, you can define automatic (conditional) or user-made branching decisions and various other actions throughout the flow of the process.

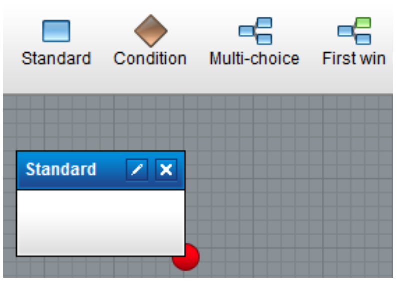
For example, you can designate a user to decide whether the contacts in a given process are prospective clients and should continue through the rest of the process. You can also make the process decide automatically based on each contact's previous behavior, such as purchases made on the website or newsletter subscriptions.

To design the steps of an automation process, edit the process in **On-line marketing -> Processes** and open the **Steps** tab.

Placing steps into the designer

1. Click a step on the designer toolbar and hold down the mouse button.
2. Drag the step onto the grid.
3. Release the mouse button.

The step appears in the grid.



A step placed onto the process designer

To learn more about using specific kinds of steps, see:

- Defining condition steps
- Defining user choice
- Defining automatic decisions
- Defining timeouts and waiting
- Adding action steps

Placing steps onto existing connections

To add a new step between two existing steps:

1. Click a step on the designer toolbar and hold down the mouse button.

2. Drag the step onto an existing connection between two steps.
   - The connection increases in thickness once it is in the correct position under your mouse pointer.

3. Release the mouse button.

The new step breaks the existing connection and automatically creates two new connections with the original steps.

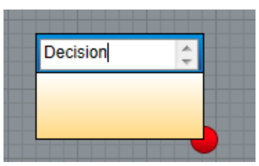
Splitting a connection by inserting a new step

Note: You cannot split connections when moving existing steps, only when adding new steps from the designer toolbar.

Labeling steps in the designer

To change the names of steps in the process designer:

1. Double-click the step's name in the grid.
2. Type in a new name.
3. Press **Enter** to save the new name.



Renaming a step

Deleting steps

Note: You cannot delete steps that are currently active for a contact in a running instance of the process. Before you can delete such steps, you need to remove the given contacts from the process or move them into a different step.

To remove steps from a process:

1. Select the step in the designer grid (left click).
2. Click **Delete** on the designer toolbar or press the **Delete** key.
3. Click **OK** in the confirmation box.

Deleting a step also removes the related connections. You need to manually reconnect the steps that were originally connected to the deleted step.

7.2.2 Connecting steps

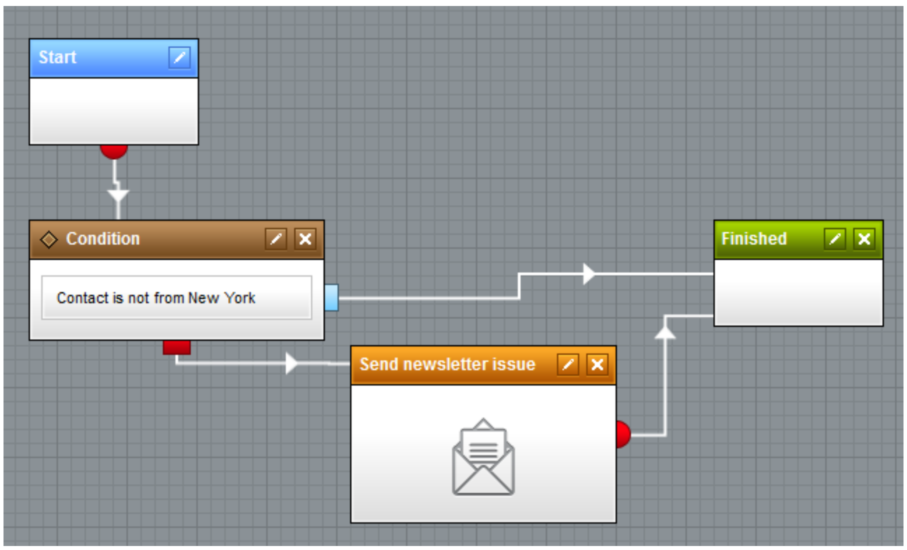
To allow an automation process to move from one step to another, you need to connect the steps. Each step has one or more source points, which you can drag to another step to create a connection.

Working with source points

The red points next to steps are **source points**.

Connecting steps by dragging a source point

Certain types of steps have multiple source points.



Condition step with two source points

For example, a *Condition* step branches the process into two paths, so it has two source points. You can use the condition step to branch the process into an if/else condition using its two source points. The process continues through the blue source point if the condition is met. The red point indicates the branch used when the condition is false.

Steps that offer multiple conditions have multiple blue source points, one for each condition.

Creating connections between steps

You can drag source points on top of other steps to create a connection between the two steps.

1. Place the steps you want to connect onto the grid.
2. Click the source point of the step you want the connection to lead from and hold the mouse button.
3. Drag the source point over the step you want to connect.
    - Moving a source point on top of a target step highlights it in the designer.

4. Release the mouse button.

After you create the connection, it stays selected. You can immediately change the connection by dragging again or remove it by pressing the **Delete** key.

Newly created connections remain highlighted

Changing connections

1. Click the connection that you want to change and hold the mouse button.
2. Drag the connection to the new target step.
3. Release the mouse button.

This removes the original connection and creates a connection with the new target step instead.

Connecting source point to a different step

Deleting connections

1. Click the connection to select it.
   - The connection changes its color and increases in thickness.
2. Click **Delete** on the designer toolbar or press the **Delete** key.
3. Click **OK** in the confirmation box.

Deleting a connection resets the source point to its default position.

7.2.3 Defining condition steps

Marketing automation allows you to split processes into multiple branches. You can define conditional steps, which automatically determine how contacts continue through the process.

Adding conditions

Condition steps split the process into two branches — one if the condition is true, the other when false. The step automatically moves contacts forward to the next step based on the result of the condition.

To add a condition step:

1. Place the **Condition** step onto the grid from the designer toolbar.

2. Edit the step by clicking the pencil icon next to its name or by double-clicking the step box.
   - The **Process step properties** dialog opens.

3. Specify the condition in the **Condition** field.
   - If you write the macro condition manually, your expression must return either a *true* or *false* value.

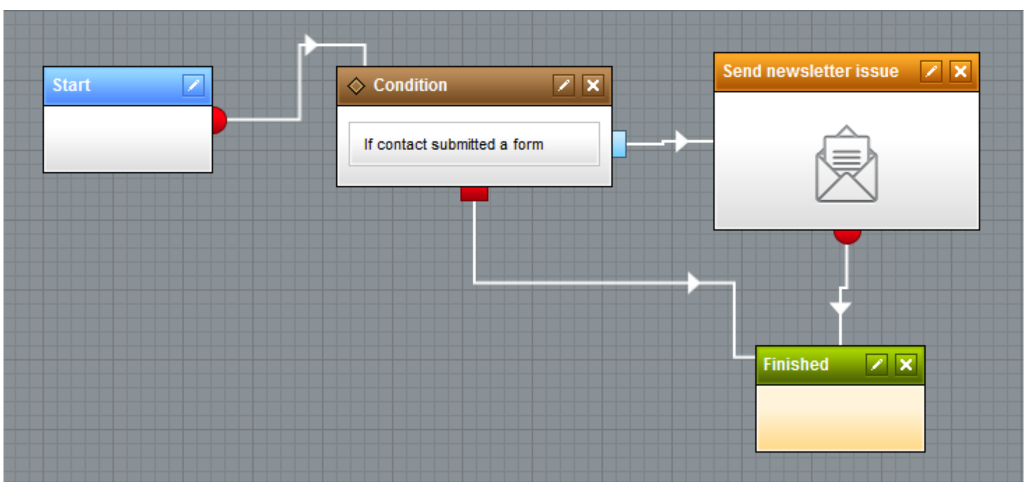
4. (Optional) Type a text description of the condition into the **Designer label** field.

5. Click **Save** and close the dialog.

6. Integrate the condition into the flow of the process by connecting a preceding step.

7. Connect both source points of the condition to other steps in the process.

For example, the process shown below uses a condition to send a newsletter issue only to contacts who fulfill a specific condition (have submitted a certain form). If the contact in the process does not meet the condition, the process moves directly to the **Finished** step.



Condition step that moves the process to the Send newsletter issue step if true

7.2.4 Defining user choice

You can add steps to automation processes that allow human users to decide how individual contacts go through the process. Designated users (such as members of the website's marketing staff) can manage individual instances of automation processes for specific contacts.

Adding approval steps

To add an approval step to your automation processes:

1. Place the **Standard** step onto the grid from the designer toolbar.
2. Integrate the approval step into the flow of the process by connecting a preceding step.
3. Connect the step's source point to the next step in the process.
4. (Optional) Edit the step by clicking the pencil icon next to its name or by double-clicking the step box:

   a. Fill in the **User action text** and **User action tooltip** for the approval step.
   b. Add an additional **Condition** — the process allows users to approve the step only if the condition is fulfilled.
   c. Click **Save**.
   d. Open the **Security** tab and configure which users or roles are allowed to approve the step.

When a contact reaches the step, the process waits for approval from a designated user before continuing to the next step.

Branching processes based on user decisions

The *User choice* step can split the process into any number of branches. Designated users need to manually choose the next step for each contact in the process.

1. Place the **User choice** step onto the grid from the designer toolbar.
2. Integrate the step into the flow of the process by connecting a preceding step.
3. Click **Add new choice** (plus sign in the step header) to create the required number of user choices.
4. (Optional) Edit individual choices in the step by clicking their pencil icon:

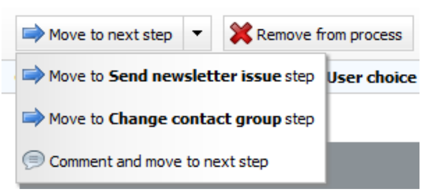
   a. Fill in the **User action text** and **User action tooltip** for the choice.
   b. Add an additional **Condition** — the process allows users to to select the choice only if the condition is fulfilled.
   c. Click **Save**.
   d. Open the **Security** tab and configure which users or roles are allowed to select the choice.

5. Connect the source points of all choices to other steps in the process.

Setting the user action text for steps

When users manage automation processes for specific contacts, they can move the contact between steps via action buttons. You can edit the text captions of the buttons for individual steps in your processes.

For example, the following image shows the process management interface for contacts in a User choice step with two available choices.



User action button with two possible step choices

To edit the action button text of a step:

1. Edit the step by clicking the pencil icon next the its name.
   - The **Process step properties** dialog opens.

2. Fill in the following fields:

- **User action text** - the caption of the action button.
- **User action tooltip** - text displayed when a user hovers over the action button.

3. Click **Save**.

If the step contains multiple cases (choices):

1. In the **Process step properties** window, switch to the **Cases** tab.
2. **Edit** ( ) each of the cases.
3. Fill in the user actions fields.
4. Click **Save**.

7.2.5 Defining automatic decisions

Marketing automation allows you to split processes into multiple branches. You can define conditional steps, which automatically determine how contacts continue through the process.

The process designer provides two types of steps that branch the process based on multiple conditions - multi-choice and first win.

Adding Multi-choice steps

Multi-choice steps can split the process into any number of branches. You need to define a set of cases for the step, each with its own condition. When a contact reaches the step in the process, the system automatically evaluates the conditions of the cases:

- If the condition is fulfilled for exactly one case, the process automatically moves the contact to the step connected to the corresponding source point.
- If multiple conditions are fulfilled, designated users need to manually choose the next step for the given contact.
- If none of the conditions are fulfilled, the process continues through the *else* branch.

To add Multi-choice steps to your process:

1. Place the Multi-choice step onto the grid from the designer toolbar.
2. Integrate the step into the flow of the process by connecting a preceding step.
3. Define cases for the step.
4. Connect the *else* source point (red) to another step.
5. Connect the source points of all cases to other steps.

Adding First win steps

First win steps can split the process into any number of branches. You need to define a set of cases for the step, each with its own condition. When a contact reaches the step in the process, the system evaluates the cases according to their order. If the step finds a matching condition, the contact moves forward to the step connected to the corresponding case (the remaining cases are not evaluated). If none of the conditions are fulfilled, the process continues through the *else* branch.

To add First win steps to your process:

1. Place the **First win** step onto the grid from the designer toolbar.
2. Integrate the step into the flow of the process by connecting a preceding step.
3. Define cases for the step.
4. Connect the *else* source point (red) to another step.
5. Connect the source points of all cases to other steps.

Defining cases

Cases represent individual branches of *Multi-choice* and *First win* steps. To set up cases for a step:

1. Click **Add new case** (plus sign in the step header) to create the required number of branches.

2. Click **Edit case properties** (pencil icon) for each case.
   - The **Process step properties** dialog opens.

3. Specify the case's condition in the **Condition** field.

4. (Optional) Fill in the User action text and User action tooltip for the case.
Users can see these text values when manually managing the flow of the process for contacts in the given step.

5. Open the **Security** tab and choose which users or roles are allowed to manually move contacts through the given case.
   - By default, cases inherit the security settings of the step.

6. Click **Save** and close the **Process step properties** dialog.

You can see the cases listed inside the step. Each case provides a separate source point for connecting the decision step to the appropriate branch of the process.

7.2.6 Defining timeouts and waiting

The process designer allows you to define timeouts, after which the system automatically moves the process to the next step.

You have two options how to define timeouts:

- Setting a timeout in a step's properties
- Adding a Wait step into the process

Setting timeouts for steps

You can set timeouts to ensure that processes do not indefinitely stay in steps that require a human decision. If nobody moves a contact within a process to another step before the timeout expires, the system advances the contact automatically. The following types of steps support timeouts:

- Standard
- User choice
- Multi-choice
- First win

To add a timeout limit for a step:

1. Edit the step by clicking the pencil icon next to its name.
    - The **Process step properties** dialog opens.

2. Under **Timeout settings**, specify when you want the process to move to the next step:

- **Specific interval** - set the length of the timeout interval. The time starts running when a contact moves into the step. Every instance of the process has a separate timeout counter.
- **Specific day** - set the exact date (and optionally time) when you want the process to move into the next step. Applies to all instances of the process.

3. Choose to which step the process continues if the timeout expires via the **Leave through** selector.
    - For **Standard** and **User choice** steps, you can select a special timeout source point.

4. Click **Save** and close the properties dialog.

5. If you selected a separate timeout source point as the **Leave through** option, connect the new source point to another step in the process designer.

Adding Wait steps

Wait steps allow you to halt automation processes for a specified time interval. When contacts arrive in a wait step, the process does nothing until the wait time expires. After the wait is complete, the process automatically moves the contact to the next step.

While the process is in a Wait step, designated users can manually move contacts forward.

For example, you can use wait steps to:

- Add delays between automatic messages that the process sends to contacts
- Periodically check if the contact meets a condition by creating a loop containing a wait step within the flow of the process

To add a wait step to your process:

1. Place the **Wait** step onto the grid.

2. Integrate the wait step into the flow of the process by connecting a preceding step.

3. Edit the step by clicking the pencil icon next to its name.
    - The **Process step properties** dialog opens.

4. Under **Timeout settings**, specify when you want the process to move to the next step:

- **Specific interval** - set the length of the waiting interval. The time starts running when a contact moves into the wait step. Every instance of the process has a separate wait timer.
- **Specific day** - set the exact date (and optionally time) when you want the process to move into the next step. Applies to all instances of the process.

5. Click **Save**.

7.2.7 Configuring security

Step security allows you to configure which users are allowed to manually move contacts between steps in automation processes. This allows users to:

- Make user choices
- Override the automatic decisions of steps

You can set security options for the following types of steps:

- **Standard**
- **Condition**
- **Multi-choice**
- **First win**
- **User choice**
- **Wait**

The system applies step security when users manage instances of the automation process for contacts.

Marketing automation permissions

Users can override the security settings of specific automation process steps if they have the following on-line marketing permissions:

- **Manage processes** - allows users to move contacts to the previous and next steps.
- **Move to specific step** - allows users to move contacts to any other step in the given automation process.

The permissions allow users to move contacts for all steps in any automation process, regardless of the step security settings.

Allowing users to move contacts from steps

1. Edit the step by clicking the pencil icon next to its name.
   - The **Process step properties** dialog appears.

2. Switch to the **Security** tab.

3. Select which roles are allowed to move contacts to the previous or next step:

- **None**

- **Only listed** - click **Add roles** to choose which roles are allowed to move contacts from the step.
- **All except listed** - click **Add roles** to block roles from managing the step. Using this option automatically allows all other roles to move contacts from the step.

4. Configure the step security options for individual user accounts:

- **No extra users**
- **Include the following users** - click **Add users** to choose which users are allowed to move contacts from the step.
- **Exclude the following users** - click **Add users** to block specific users from managing the step.

The combination of the user and role settings defines which users are allowed to move contacts from the given step to the previous or next step.

Setting security for steps with multiple branches

For steps that branch into multiple cases (choices), you can configure different security options for individual cases. By default, cases inherit the settings of their step.

1. Edit individual cases:
- Directly in the process designer
- On the **Cases** tab in the **Process step properties** dialog

2. Open the **Security** tab of the given case.

3. Select which roles are allowed to move contacts forward through the corresponding branch:

- **Use step settings** - uses the security settings of the main step.
- **Only listed** - click **Add roles** to choose which roles are allowed to manually select the case for contacts in the step.
- **All except listed** - click **Add roles** to block roles from selecting the case. Using this option automatically allows all other roles to select the case for contacts.

4. Configure the case security options for individual user accounts:

- **Use step settings** - uses the security settings of the main step.
- **Include the following users** - click **Add users** to choose which users are allowed to manually select the case for contacts in the step.
- **Exclude the following users** - click **Add users** to block specific users from selecting the case.

The combination of the user and role settings defines which users are allowed to manually move contacts forward to the process branch represented by the given case.

7.2.8 Adding action steps

Automation processes can perform various types of marketing actions. By integrating actions into processes, you can automate tasks that users would otherwise need to do manually. For example, you can have a process send a newsletter issue to contacts that meet a certain condition or move contacts between different accounts.

You can add actions into flow of processes via **action steps**. When a contact reaches an action step in a process, the system automatically performs the given action.

The following action steps are available:

Action step	Description
Change account	Adds or removes the contact from an account.
Change group	Adds or removes the contacts from a contact group.
Delete contact	Deletes the contact currently going through the automation process.
Import to Salesforce	Replicates the contact into a lead in a target Salesforce organization. You need to have Salesforce replication set up for your website to use this action.
Log custom activity	Logs a custom activity for the contact.
Newsletter subscription	Subscribes or unsubscribes the contact from a newsletter.
Send e-mail	Sends an e-mail (to any address).
Send newsletter issue	Sends a newsletter issue to the contact. **Note**: When performing this action step, the process automatically adds the contact as a subscriber for the newsletter containing the specified issue.
Set contact property	Inserts a new value into one of the contact's properties (fields).
Set contact status	Changes the contact's status.
Start process	Triggers a different marketing automation process for the contact. Based on the recurrence settings of the target process, this action can result in multiple unnecessary instances of the process running concurrently.
Update from Data.com	Updates the contact's information with the data found in the Data.com database. The step connects to Data.com using a specified account. See also: Data.com integration

Managing contact accounts

1. Place the **Change account** action step onto the process designer grid.

2. Integrate the step into the flow of the process by connecting a preceding and following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Specify the step's parameters:

- **Account name** - select the account that the action manages.
- **Contact role** - you can have the process assign a specific role within the account to the contact.

- **Action -** you can either add the contact to the specified account or remove it from the account.

5. Click **Save**.

See also: Accounts

Managing contact groups

1. Place the **Change group** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Specify the step's parameters:

- **Contact group -** select the contact group that the action manages.
- **Action -** you can either add the contact to the specified group or remove it.

5. Click **Save**.

See also: Contact groups

Setting contact properties

1. Place the **Set contact property** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Select which contact property the action changes (**Property name**).

5. Type the new **Property value**.

6. Click **Save**.

See also: Managing contacts

Setting contact statuses

1. Place the **Set contact status** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Select the new **Contact status** that the action assigns to contacts.

5. Click **Save**.

See also: Contact statuses

Deleting contacts

1. Place the **Delete contact** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Specify the step's parameters:

- **Delete merged** - indicates if the step also deletes merged child contacts. If not checked, the process splits all child contacts before deleting the parent contact.
- **Move activities** - check to move activities, relationships, IPs, and UserAgents to the parent contact when deleting merged contacts.

5. Click **Save**.

Replicating contacts to Salesforce

1. Place the **Import to Salesforce** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Specify the step's parameters:

- **Deferred replication** - if checked, the action step delays the replication of the contact until the system executes the scheduled replication process. When disabled, the action replicates the contact immediately. Using deferred replication helps reduce the number of Salseforce API calls made by your application.

5. Click **Save**.

See also: Salesforce integration

Logging custom activities

1. Place the **Log custom activity** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Specify the step's parameters:

- **Activity type** - select one of your custom activity types.
- **Activity title** - add basic information about the event that the activity represents.
- **Activity value** - you can assign a value to the activity. You can associate values with activities such as ratings or e-commerce purchases.

- **Activity URL** - enter the URL of the page where the activity occurred.
- **Campaign** - you can select a marketing campaign associated with the activity.
- **Comment** - here you can add any additional information related to the activity.

5. Click **Save**.

See also: Tracking contact activities, Adding custom activities

Managing newsletter subscriptions

1. Place the **Newsletter subscription** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Select the **Newsletter**.

5. Select the **Action** - you can either subscribe the contact to the newsletter or unsubscribe the contact from the newsletter.

6. Click **Save**.

See also: Newsletters

Sending e-mails

Prerequisite: You must have at least one SMTP server configured for your website.

1. Place the **Send e-mail** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Enter the sender's address into the **From** field.

5. Specify the recipients of the e-mail in the **To** field.
   - You can enter multiple addresses separated by semicolons.
   - Use macro expressions to load the address dynamically. For example, *{% Contact.ContactEmail %}* gets the address of the current contact in the process.

6. Choose the content type of the e-mail:

- **E-mail template** - the content of the e-mail is based on the selected E-mail template.
- **HTML formatted text** - manually write the *Subject* and *Body* of the e-mail.

7. Click **Save**.

Sending newsletter issues

1. Place the **Send newsletter issue** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Select the **Newsletter issue** that the action step sends.
   - You need to choose both the newsletter and specific issue.

5. Click **Save**.

See also: Newsletters

Starting automation processes

1. Place the **Start process** action step onto the grid.

2. Connect the step with a preceding step and optionally with a following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Select the automation **Process** that you want to start for the contact.
   - You cannot start a new instance of the same process that you are designing (to avoid the possibility of process loops).

5. Click **Save**.

Updating contact information Data.com

1. Place the **Update from Data.com** action step onto the grid.

2. Integrate the step into the flow of the process by connecting a preceding and following step.

3. Edit the step by clicking the pencil icon next to its name.

4. Fill in the **E-mail address** and **Password** of a valid Data.com account.

5. Click **Set**.

6. Configure the step's parameter:

- **Buy enabled** - if checked, the step automatically buys contacts that the assigned Data.com account does not not own. The step buys contacts only if there is one exact match in the Data.com registry.

7. Click **Save**.

Note

> Every **Update from Data.com** step stores the credentials of a Data.com account. The step uses the account to search for data and buy contacts. Steps cannot update contact data if the assigned Data.com credentials become invalid.
>
> To change a step's Data.com account:
>
> 1. Edit the step in the process designer.
> 2. Click **Remove credentials**.
> 3. Fill in the authentication form with the e-mail and password of the new Data.com account.
> 4. Click **Set**.

See also: Data.com integration

Adding custom actions

Kentico allows developers to create custom action steps for use in automation processes. Developing actions step requires knowledge of programming and the Kentico CMS API. See Defining custom actions for more information.

7.3 Creating triggers

By defining triggers, you can configure processes to automatically start for contacts. Every trigger has a set of conditions that determine under which circumstances the system runs the related process.

For example, you can create a trigger that automatically initiates a marketing process for all new contacts from a certain country.

Creating triggers

To create a new trigger for an automation process:

Note: You can add any number of triggers for a single process.

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
   - You can also create triggers in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. **Edit** ( ) the process for which you want to create the trigger.

3. Switch to the **Triggers** tab.

4. Choose whether you want to create a *(global)* trigger or only for a specific **Site**.

5. Click **New trigger**.

6. Enter a **Display name** for the new trigger.

7. Select the trigger **Type**. You can choose from four different types:

Trigger type	Description
Contact created	Activates the process for new contacts created in Kentico CMS. You can specify an additional **Condition** that must be fulfilled for the trigger to start the process. You can only create *Contact created* triggers for specific sites.
Contact changed	Activates the process whenever the data of a contact is updated. You can specify an additional **Condition** that must be fulfilled for the trigger to start the process. You can only create *Contact changed* triggers for specific sites.
Activity performed	Activates the process when contacts perform a specific activity on the website. • Select the **Activity type** that contacts must perform to initiate the process. • You can specify an additional **Condition** that must be fulfilled for the trigger to start the automation process. Use the *Activity...* rules to specify detailed requirements for the trigger activity.
Score exceeded	Activates the process for contacts that exceed a specified score value. • You can specify an additional **Condition** that must be fulfilled for the trigger to start the process. • Select which score the trigger monitors using the **Score name** field. • Type the **Score value** that contacts must reach to initiate the process.

8. **Save** the new trigger

The process now automatically starts for all contacts that fulfill the conditions of the trigger.

Modifying existing triggers

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
2. **Edit** () the process whose triggers you want to modify.
3. Switch to the **Triggers** tab.
4. Use the **Site** selector to access a list of either *(global)* triggers or triggers created for specific sites.
5. **Edit** () the trigger.
6. **Save** the modified trigger.

7.4 Manually running automation processes

You can manually start marketing automation processes for specific contacts or entire contact groups.

Note: For full automation, define triggers for your processes. This allows the system to automatically run processes for contacts based on dynamic conditions.

To manually run processes, you need to have the appropriate permissions for the On-line marketing module. The settings need to be set for the site under which you want to run the process.

Running processes for specific contacts

1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.
2. **Edit** () the contact for which you want to run the process.
3. Switch to the **Processes** tab.
   - Here you can see a list of all automation processes currently running for the contact.
4. Click **Start new process**.
5. Select the process that you want to start for the contact.

After starting the process, you can see its status in the list on the contact's **Processes** tab.

Running a process for a contact from within the process

You can also start a process for a single contact directly while editing the process:

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
2. **Edit** () the process you want to run for a contact.
3. Switch to the **Contacts** tab.
4. Click **Start new process**.
5. Choose the contact for which you want to run the process.

Running processes for contact groups

1. Navigate to **CMS Desk -> On-line marketing -> Contact groups**.
2. **Edit** () the contact group for which you want to run the process.
3. Switch to the **Contacts** tab.
4. Click **Start new process**.
5. Select the process that you want to start.

The process starts for all the contacts in the contact group separately.

7.5 Managing the flow of contacts in processes

Automation processes run in a separate instance for each contact. You can move contacts through steps that require human input or manually override the automatic decisions of the process.

The system provides the following lists of contacts in automation processes:

- **Pending contacts** - all contacts that you are allowed to move from their current step within any running automation process
- The contacts within a specific process (including those in the *Finished* step)

Note: You need to have the **Read contacts** Contact management permission to view the lists of contacts. If you only have the permission for a specific site, you can only see contacts from the given site.

Managing pending contacts:

1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.
2. Open the **Pending contacts** tab.

The page shows a list of contacts within any active automation process (i.e. not in the *Finished* step). You can only see contacts that you are allowed to move from their current step in the process. To move contacts, you need to be included in the step's security settings or have the **Manage processes** On-line marketing permission.

Tip: You can also manage pending contacts through the **My pending contacts** widget, which you can add to your dashboard.

Managing contacts in a specific process:

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
2. **Edit** ( ) the process whose contacts you want to manage.
3. Open the **Contacts** tab.

The page shows a list of all contacts in the process (including those in the *Finished* step).

Moving contacts between steps in the process

When viewing the list of pending contacts or the contacts within a specific process, click **Manage** ( ) next to a specific contact. A diagram appears where you can see the current position of the contact within the process.

The following actions are available:

Note: You can only use the actions allowed by your on-line marketing permissions and the security settings of individual steps.

Action	Description
Move to next step	• Directly moves the contact to the next step in the process if there is only one possible option. • If the current step branches into multiple steps, clicking the button offers you all available options. Each option represents one of the possible steps to which the contact can continue.
Move to previous step	Moves the contact back to the previous step in the process. The contact's previous step is not necessarily the step that precedes the current step in the process designer, for example if the contact was manually moved using the *Move to specific step* action.
Move to specific step	Moves the contact to any step of your choice.
Comment and move to step	Available after clicking next to the *Move to next step, Move to previous step* or *Move to specific step* buttons.

	Opens a dialog where you can enter a comment for the step transition. Choose the target step using the selector above the comment box. The comment is then visible in the **Process history**.
Remove from process	Ends the automation process for the given contact. This deletes the entire process instance running for the contact. For example, you can use the remove action if a contact gets stuck in a step that has incorrectly set conditions or connections.

Tip: You can view the history of all step transitions made by the contact in the **Process history** section below the process diagram.

Tracking marketing automation in the Event log

By default, the system logs an event log entry whenever a contact moves to a different step in a marketing automation process. You can view the event log in **Administration -> Event log**.

Disabling event logging for marketing automation

[Only possible after applying hotfix 7.0.18 or newer]

When running automation processes for very large numbers of contacts, logging of step transitions may lead to a cluttered event log or even website performance problems.

Administrations with access to the web project's file system can disable the logging of these events by adding the following key to the */configuration/appSettings* section of the *web.config* file:

```
<add key="CMSLogMATransitions" value="false" />
```

7.6 Viewing process reports

You can view a report for each automation process. The report shows how many contacts are are currently in particular steps of the process, including the **Finished** step. The relative ratio of these contacts is displayed in a pie chart.

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
   - You can also view the reports in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. **Edit** ( ) the process for which you want to view the report.

3. Switch to the **Report** tab.

You can perform the following actions with a report:

- **Save** the current state of the report for later use. You can find the saved reports in *CMS Desk -> Tools -> Reporting -> Online marketing ->* Automation.
- **Print** the report.
- **Subscribe** to the report to have it sent in specified intervals via e-mail.
- **Refresh** the report.
- Right-click the graph or table to export the data to files in various formats.

Tip

You can configure or customize the automation process report in **CMS Desk -> Tools -> Reporting** under the **Online marketing -> Automation** report category.

For more information about the Reporting module, see its documentation.

7.7 Versioning

The system stores the version history of automation process as they are modified by users. Versioning allows you to view and compare process versions and revert processes back to previous versions.

Version numbering

Every version of an automation process has an identification number. By default, the system assigns numbers to versions according to the following rules:

- When you create a process, its version number is **0.1**.
- When you modify a process, the system creates a minor version and increases the number by a decimal point. For example, from **0.2** to **0.3**, then to **0.4** and so on.
- Click **Make current version major** promotes the latest minor version to a major version. This increases the number before the period and resets the decimal number to zero. For example, **0.4** becomes **1.0**.

Viewing and comparing process versions

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
    - You can also view process versions in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. **Edit** () the process for which you want to view the versions.

3. Switch to the **Versions** tab.

4. Click **View** next to the version you want to view.

5. Select a version for comparison using the **Compare to** drop-down list. The older version appears on the left.

Rolling back versions

1. Navigate to **CMS Desk -> On-line marketing -> Processes**.
   - You can also roll back versions in **Site manager -> Tools -> On-line marketing -> Automation processes**.

2. **Edit** () the process that you want to roll back to a previous version.

3. Switch to the **Versions** tab.

4. Click **Roll back** next to a version.

The system makes a new copy of the version and marks the copy as the latest version.

7.8 Defining custom actions

Kentico CMS enables you to write your own actions to incorporate into an automation process. First, you need to write the action's code and then register it as a new action in automation processes. Note that writing custom actions requires certain programming experience.

Writing actions

1. Create a new class and make it inherit from *CMS.WorkflowEngine.AutomationAction* for general automation actions and *CMS.OnlineMarketing.ContactAutomationAction* for actions that work with contacts.

2. (Optional) If you created the class in the App_Code folder (or the Old_App_Code folder if you're using web application), register it as described in Developer's Guide -> API programming and Kentico CMS internals -> Registering custom classes in App_Code.

3. Override the *Execute()* method from the *base* class.

4. Write the code that you want executed by the action into the *Execute* method's body.

Action steps can have parameters (settings) specified that you can use to modify their behavior. You can then enter the parameter values when configuring an action step in the workflow designer. If you want your custom action step to rely on parameters, you can specify them when editing or creating an action on the **Actions** tab in **Site Manager -> Tools -> On-line marketing -> Automation processes**. To access the parameter's values in the code of an action, use the *GetResolvedParameter* method.

Registering actions in the system

Once you create the code for the custom action, register it.

1. Navigate to **CMS Desk -> On-line marketing -> Processes** and switch to the **Actions** tab.

2. Click on **New action**.

3. Fill in the following mandatory fields:

- **Display name** - name of the action step.
- **Assembly name** - name of the library that contains the action's code.
- **Class name** - full name of the class that contains the action's code.

4. (Optional) Select the module the action will belong to.

5. (Optional) Define the action's parameters on the **Parameters** tab.

Note that you can register actions in **Site manager -> Tools -> On-line marketing -> Automation processes** as well.

7.9 Walkthrough: Creating a sample process

This chapter provides a step-by-step example that you can follow to create a new marketing automation process from start to end. You can quickly get acquainted with marketing automation using this walkthrough.

The walkthrough consists of four topics. Follow the topics in sequential order:

- Preparing the prerequisites
- Defining the process
- Designing the process
- Managing the process

7.9.1 Preparing the prerequisites

Before you create the automation process, you need to:

- Prepare the users and roles that you will later use to manage the process.
- Create a pair of newsletters.
- Configure Kentico CMS to allow you to run all the features demonstrated in the walkthrough.

Creating testing users

Create two user accounts for administering the automation process:

1. Navigate to **Site Manager -> Administration -> Users**.

2. Click **New user** ( ).

3. Set the following details for the user:

- **User name**: Travis
- **Full name**: Travis McCoy
- **Enabled**: yes
- **Is editor**: yes

4. Click **Save**.

5. Go to the **Sites** tab and assign the user to your current website.

6. Create another user with the following details:

- **User name**: Matt
- **Full name**: Matt McGinley
- **Enabled**: yes
- **Is editor**: yes

7. Click **Save**.

8. Go to the **Sites** tab and assign the user to your current website.

Setting the roles and permissions

For the users to be able to perform their tasks, you need to add them to roles with appropriate permissions.

Preparing the Marketing Manager role

Assign Travis to the default **Marketing Manager** role:

1. Navigate to **Site Manager -> Administration -> Roles** and choose the appropriate site from the **Site** drop-down-list.

2. **Edit** () the **Marketing Manager** role and switch to the **Users** tab.

3. Click **Add users**.

4. Select Travis McCoy (Travis) and confirm by clicking **OK**.

5. Switch to the **Permissions** tab.

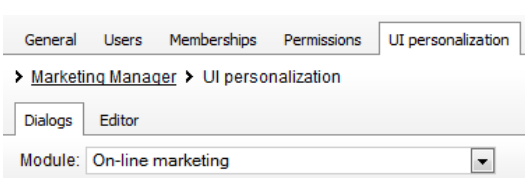
6. Choose **Module** in the first drop-down list and **On-line marketing** in the second.

7. Make sure the **Marketing Manager** role has all automation process permissions allowed:

- **Manage processes**
- **Start process**
- **Remove process**
- **Move to specific step**

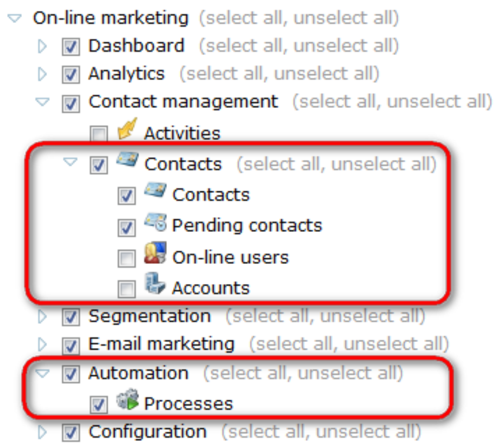
8. Switch to the **UI personalization** tab.

9. Choose **On-line marketing** in the **Module** drop-down list.



10. Check the boxes in the UI tree for all items in the following sections:

- **Contact management -> Contacts**
- **Automation -> Processes**



The *Travis* user account now has full control over automation processes, i.e. can design the entire sample process and fully modify its instances.

Preparing the Process Administrator role

Create a **Process administrator** role and assign Matt into the role:

1. Navigate to **Site Manager -> Administration -> Roles** and choose the appropriate site from the **Site** drop-down-list.

2. Click **New role** ( ) type in *Process administrator* as the **Role display name**.

3. Click **Save**.

4. Switch to the **Users** tab and click **Add users**.

5. Select *Matt McGinley (Matt)* and click **OK**.

6. Switch to the **Permissions** tab and choose **Module** in the first drop-down list.

7. Allow the following permissions for the **Contact management** module:

- **Read contacts**
- **Read global contacts**

8. Allow the **Read** permission for the **Content** module.

9. Switch to the **UI personalization** tab.

10. Choose **CMS Desk** in the **Module** drop-down list and check **On-line marketing**.

11. Choose **On-line marketing** in the **Module** drop-down list and check the following boxes in the UI tree:

- **Contact management**
- **Contact management -> Contacts**
- **Contact management -> Contacts -> Contacts**
- **Contact management -> Contacts -> Pending contacts**
- **Automation**
- **Automation -> Processes**

The *Matt* user account can now access the on-line marketing interface for modifying instances of running automation processes. You can always expand or reduce the permissions.

See also: Developer's Guide -> Development -> Membership, permissions and security

Creating testing newsletters

Create two newsletter issues that will be used by the sample automation process:

1. Navigate to **CMS Desk -> On-line marketing -> Newsletters**.

2. Click **New newsletter** and enter the following details:

- **Newsletter display name**: *Coffee Co Merchandise Newsletter*
- **Newsletter name**: leave the *(automatic)* option;
- **Subscription confirmation**: Subscription confirmation template
- **Unsubscription confirmation**: Unsubscription confirmation template
- **Sender name**: Coffee Co
- **Sender e-mail**: Enter your e-mail address
- Check **Template-based newsletter**.

3. Click **Save**.
   - You are now on the **configuration** tab of the new newsletter.

4. Under the **On-line marketing** settings group, check both **Track opened e-mails** and **Track clicked links**.

5. Click **Save**.

6. Switch to the **Issues** tab and click **Create new issue**.
   - The **New issue** wizard opens.

7. Enter *Coffee Co Merchandise Newsletter #1* as the subject.

8. Click **Insert/Edit link** () and create a link to a page on your website in the **Content** section.

9. Click **Next**.

10. Select **Send newsletter manually later**.

11. Click **Finish**.

12. Create another newsletter issue and use *Coffee Co Merchandise Newsletter #2* as the subject.
    - Insert a link to a different page into the content of the issue.

13. Return to the main **Newsletters** tab and create another newsletter.
    - Name the newsletter **Coffee Co Monthly Newsletter**.
    - Leave the rest of the details as in the first newsletter.
    - Don't create any issues for the second newsletter.

You now have two newsletters prepared for later use in the automation process.

See also: Developer's Guide -> Modules -> Newsletters

Enabling On-line marketing

You need to track contacts in order to test the functionality of the sample process. To track contacts, enable on-line marketing:

1. Navigate to **Site manager -> Settings -> On-line marketing**.

2. Check the **Enable on-line marketing** box.

3. Click **Save**.

Configuring your SMTP server

To be fully able to test the process, you need to have a SMTP server configured correctly in Kentico. You can either configure it yourself as described in Developer's guide -> Installation and deployment -> Additional configuration tasks -> SMTP server configuration or contact your administrator.

7.9.2 Defining the process

With the testing users and roles prepared, you can start creating the new automation process.

Creating the automation process

Add the process itself and set its recurrence settings:

1. Log in to your website's **CMS Desk** as **Travis**.
    - Go to *http://<yourdomain>/CMSDesk* and enter *Travis* as the user name.

2. Navigate to **On-line Marketing -> Processes**.

3. Click **New process** and type *Coffee Merchandise Lead Nurturing* as the **Display name**.

4. Click **Save**.
    - The system creates the process and opens the **General** tab of the process editing interface.

5. Under **Start the process**, choose **If it hasn't been run before**.

6. Click **Save**.

By setting the recurrence setting to **If it hasn't been run before**, you make sure that the process cannot run more than once for the same contact. This ensures that contacts will not receive the same marketing information twice.

See also: Creating processes

Creating a trigger

Create a trigger that runs the automation process whenever a new contact is created on your website:

1. Open the **Triggers** tab of the process's editing interface.

2. Click **New trigger** and set the details as follows:

- **Display name**: UK and US Contacts
- **Type**: Contact created

3. Click **Edit** () next to the **Condition** field.
   - The **Edit macro condition** dialog opens.

4. In the **Available rules** list, select *Contact is from country* and click **Add rule** ().
   - You can now see the rule in the designer on the left side of the dialog.

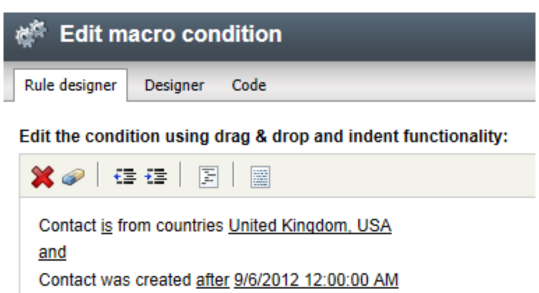
5. Click the select countries parameter.
   - The **Set parameter value** dialog opens.

6. Click **Select**, choose **USA** and **United Kingdom** (you can use the filter at the top of the dialog) and confirm by clicking **OK** twice.

7. Select *Contact was created* in the **Available rules** list and click **Add rule** () again.

8. Click the select date parameter.

9. Click **Now** and **OK**.



10. Close the **Edit macro condition** dialog by clicking **OK** again.

The trigger runs the process for contacts from the UK and USA created from now onwards.

See also: Creating triggers

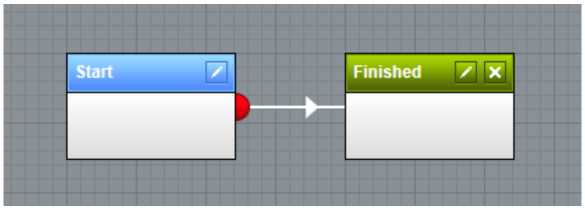
7.9.3 Designing the process

Now that you have created the trigger that automatically runs the process for certain contacts, use the process designer to define the flow of the process:

1. Log in to **CMS Desk** as *Travis* and navigate to **On-line marketing -> Processes**.

2. **Edit** () the *Coffee Merchandise Lead Nurturing* process.

3. Switch to the **Steps** tab.
   - You can see the two default steps, **Start** and **Finish**, connected together.



4. Click the connection and press the Delete key to remove the connection.

Adding a Condition step

The process starts only for contacts created with UK or USA as their *country* attribute. Let's assume you need to distinguish customers from New York City, so you can send them an e-mail inviting them to your local cafes.

1. Place the **Condition** step onto the grid.

2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

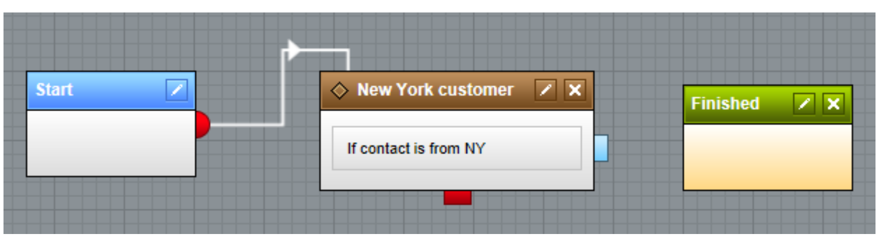
- **Display name**: New York customer
- **Designer label**: If contact is from NY
- **Condition**: Add *Contact field contains value* condition. Specify the field to *City* and the value to *New York*.

4. Click **Edit** () next to the **Condition** field.

5. Add the *Contact field contains value* rule. Set the field parameter to *City* and the value to *New York*.

6. Click **Save** and **Close**.

7. Connect the **Start** step to the **New York customer** step.



Adding a Send e-mail step

You are now able to distinguish users that come from New York, so you can send them an e-mail informing about the cafes you have set up in the area.

1. Place the **Send e-mail** step onto the grid.

2. Edit the step by clicking the pencil icon next to its name.

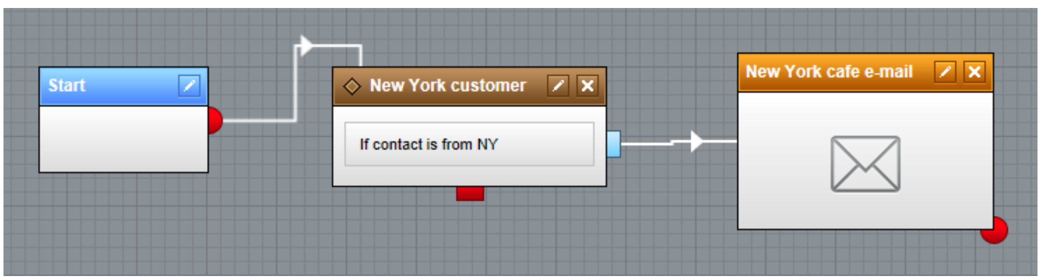
3. Modify the values as follows:

- **Display name**: New York cafe e-mail
- **From**: Coffee co
- **To**: {% Contact.ContactEmail %}
- **Based on**: HTML template
- **Subject**: Visit one of our Cafes!
- **Body**: any text

The value you entered into the **To** field is a macro expression that dynamically loads the e-mail address of the contact in the process.

4. Click **Save** and **Close**.

5. Connect the IF source point (blue) of the **New York customer** step to the **New York cafe e-mail** step.



Adding Wait step #1

Add a waiting interval to give contacts time to receive and read the e-mail:

1. Place the **Wait** step onto the grid.

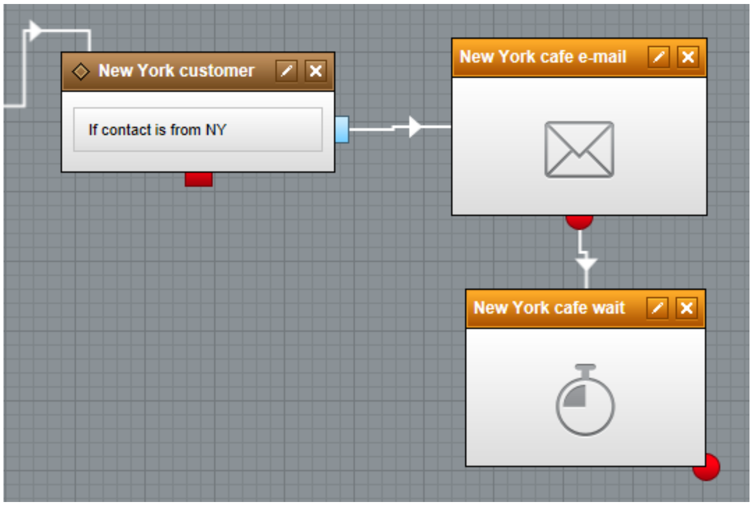
2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: New York cafe wait
- **Timeout settings**: Select **Specific interval** and enter **7 days.** You can set a shorter interval for testing purposes (e.g. 3 **minutes**).

4. Click **Save** and **Close**.

5. Connect the **New York cafe e-mail** step to the **New York cafe wait** step.



Adding Send newsletter issue step #1

Add a step that sends the first newsletter issue that you created earlier to both the contacts that are and aren't from New York:

1. Place the **Send newsletter issue** step onto the grid.

2. Edit the step by clicking the pencil icon next to its name.

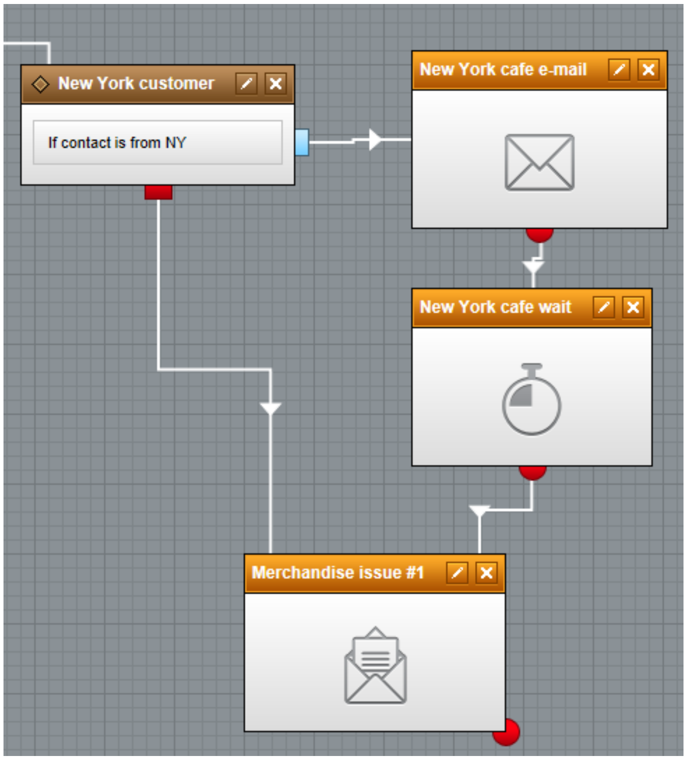
3. Modify the values as follows:

- **Display name**: Merchandise issue #1
- **Newsletter issue:** Click *Select,* choose the correct site, *Coffee and Co Merchandise Newsletter* and *Coffee Co Merchandise #1*.

4. Click **Save** and **Close**.

5. Connect the ELSE source point (red) of the **New York customer** step to the **Merchandise issue #1** step.

6. Connect the **New York cafe wait** step to the **Merchandise issue #1** step.



Adding Wait step #2

Add another **Wait** step into the process to give contacts time to receive the newsletter:

1. Place the **Wait** step onto the grid.

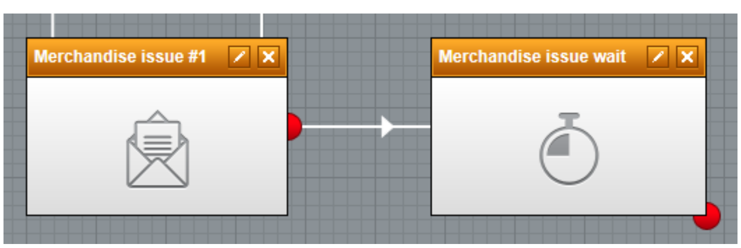
2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Merchandise issue wait
- **Timeout settings**: Select **Specific interval** and enter **3 days.** You can set a shorter interval for testing purposes (e.g. 3 **minutes**).

4. Click **Save** and **Close**.

5. Connect the **Merchandise issue #1** step to the **Merchandise issue wait** step.



Adding Send newsletter issue step #2

Add a step that sends the second newsletter issue from the newsletter that you created earlier:

1. Place the **Send newsletter issue** step onto the grid.

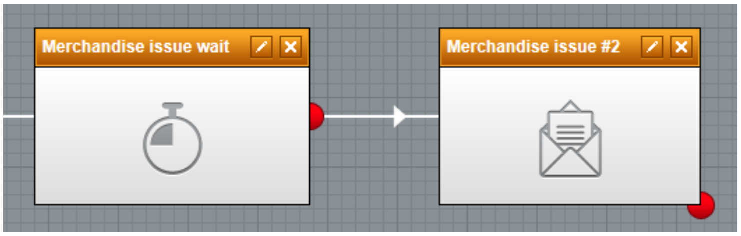
2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Merchandise issue #2
- **Newsletter issue:** Click *Select,* choose the correct site, *Coffee and Co Merchandise Newsletter* and *Coffee Co Merchandise #2*

4. Click **Save** and **Close**.

5. Connect the **Merchandise issue wait** step to the **Merchandise issue #2** step.



Adding Wait step #3

Add another **Wait** step into the process to give contacts time to receive the second newsletter:

1. Place the **Wait** step onto the grid.

2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Merchandise issue wait 2
- **Timeout settings**: Select **Specific interval** and enter **3 days.** You can set a shorter interval for testing purposes (e.g. 3 **minutes**).

4. Click **Save** and **Close**.

5. Connect the **Merchandise issue #2** step to the **Merchandise issue wait 2** step.

Adding a Multi-choice step

Add a **Multi-choice step** to decide where the process moves next based on whether the contact clicked on the links in the previous newsletters:

1. Place the **Multi-choice** step onto the grid.

2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Newsletter link decision
- **User action text:** Choose the next step.
- **User action tooltip:** *Choose whether you want to subscribe the contact to the Coffee Co monthly newsletter or set the contact's status to Prospective client.*

4. Click **Save**.

Modifying the first case

1. Switch to the **Cases** tab.

2. **Edit** () Case 1 and modify the values as follows:

- **Designer label**: Subscribe to monthly newsletter
- **User action tooltip:** Subscribe the contact to the Coffee Co monthly newsletter.

3. Click **Edit** () next to the **Condition** field.

4. Add the *Contact has clicked a links in newsletter issue* rule. Specify the *Coffee Co Merchandise #1 issue* under the *Coffee Co Merchandise Newsletter*.

5. Confirm by clicking **OK** twice.

6. **Save** the changes.

Modifying the second case

1. **Edit** () Case 2 and modify the values as follows:

- **Designer label**: Set status to prospective client
- **User action tooltip:** Set the contact's status to *Prospective client*.

2. Click **Edit** () next to the **Condition** field.

3. Add the *Contact has clicked a links in newsletter issue* rule. Specify the *Coffee Co Merchandise #2 issue* under *Coffee Co Merchandise Newsletter*.

4. Confirm by clicking **OK** twice.

5. **Save** the changes.

Configuring security for the Multi-choice step

1. Switch to the **Security** tab.

2. Under *Roles which can move object to different step*, select **Only listed** and your website in the **Select site** drop-down list.

3. Click on **Add roles** and select the **Process administrator** role.

4. Click **Save** and **Close**.

5. Connect the **Merchandise issue wait 2** step to the **Newsletter link decision** step.

The multi-choice step automatically decides how the process continues. Users in the **Process administrator** role can decide how the process continues for contacts that satisfy both conditions (by clicking on both newsletter links). You can modify security for each case separately, but by default both cases inherit the settings of the main step.

Adding an alternative Finished step

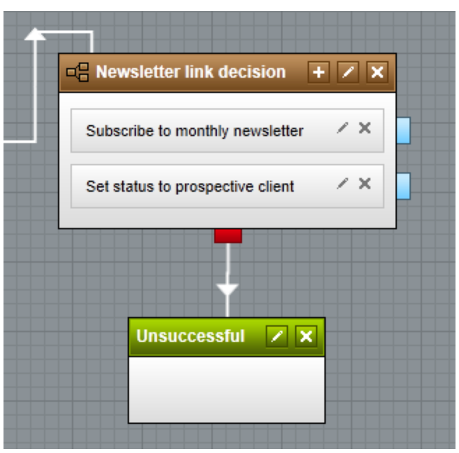
Add a Finished step that concludes the marketing process for contacts who don't click any of the newsletter links:

1. Place the **Finished** step onto the grid.

2. Double-click the step's name in the header.

3. Rename the step to **Unsuccessful** and press **Enter**.

4. Connect the ELSE source point (red) of the **Newsletter link decision** step to the **Unsuccessful** step.



Adding a Newsletter subscription step

Add a step that subscribes the contact to the Coffee Co monthly newsletter:

1. Place the **Newsletter subscription** step onto the grid.

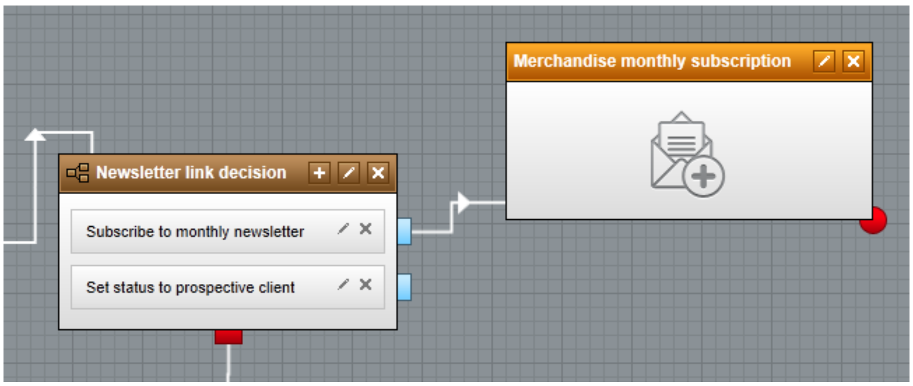
2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Merchandise monthly subscription
- **Newsletter**: Select the *Coffee Co monthly newsletter.*
- **Action**: Choose Subscribe to newsletter

4. Click **Save** and **Close**.

5. Connect the first blue source point of the **Newsletter link decision** step to the **Merchandise monthly subscription** step.



Adding a Set contact status step

Add a step that sets the contact's status to *Prospective client*:

1. Place the **Newsletter subscription** step onto the grid.

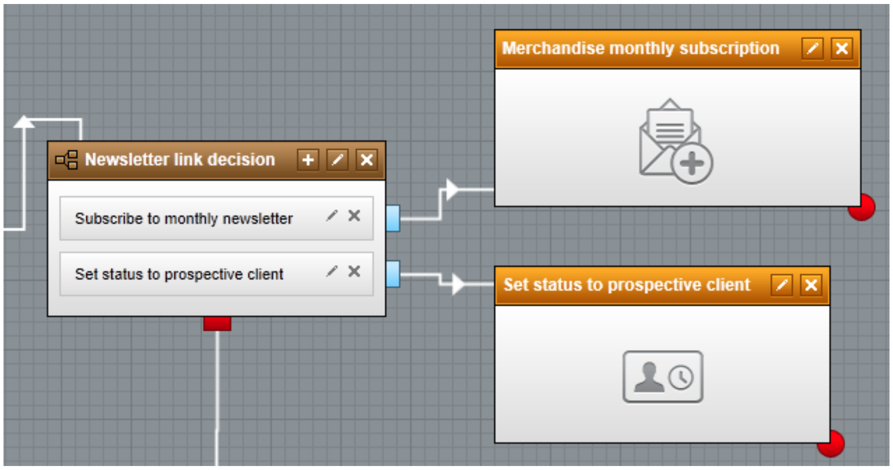
2. Edit the step by clicking the pencil icon next to its name.

3. Modify the values as follows:

- **Display name**: Set status to prospective client
- **Contact status**: Select *Prospective client* status

4. Click **Save** and **Close**.

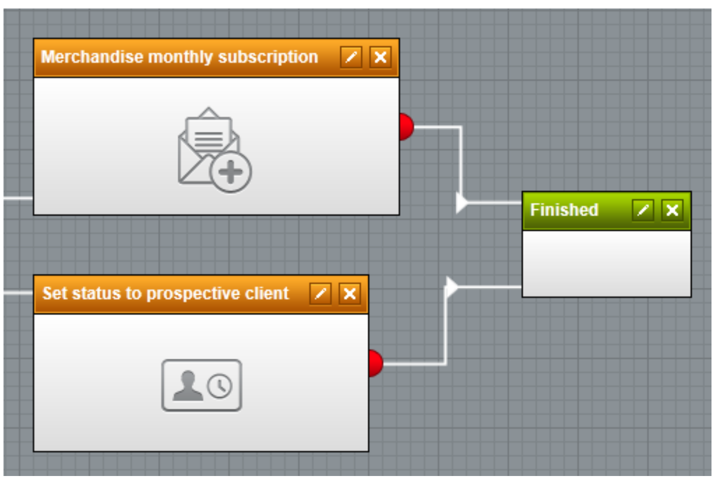
5. Connect the second blue source point of the **Newsletter link decision** step to the **Set status to prospective client** step.



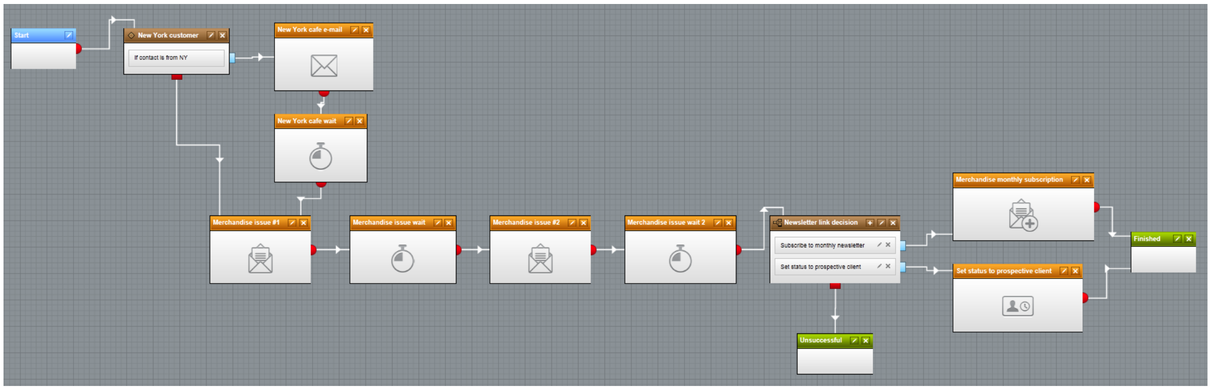
Finishing the process design

You have now finished creating all the necessary steps. You only need to connect the last two steps to the original **Finished** step.

1. Connect the first source point of the **Merchandise monthly subscription** step to the **Finished** step.
2. Connect the first source point of the **Set status to prospective client** step to the **Finished** step.



The design of the process is now complete.



See also: Working with process designer

7.9.4 Managing the process

The sample automation process is now ready for testing. You can run the process by creating a contact that meets the trigger's conditions.

Creating a contact to trigger the process

Create a contact that meets the trigger's condition to initiate the process:

1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.

2. Click **New contact**.

3. Modify the values as follows:

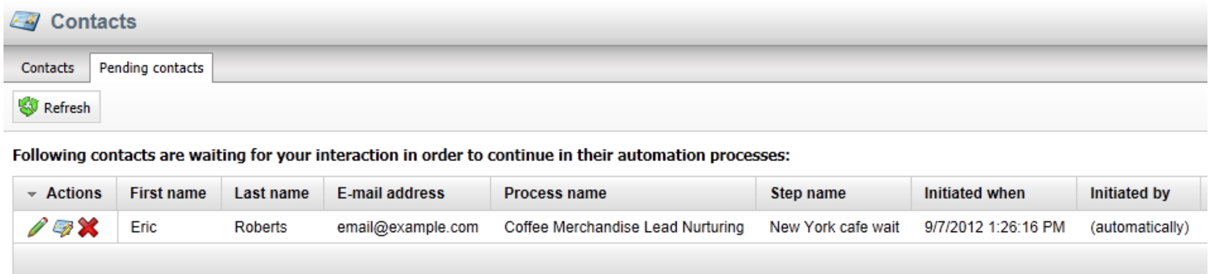
- **First name:** Eric
- **Last name:** Roberts
- **City:** New York
- **Country:** USA
- **E-mail:** your e-mail address

4. Click **Save.**

Managing the process using the Marketing Manager role

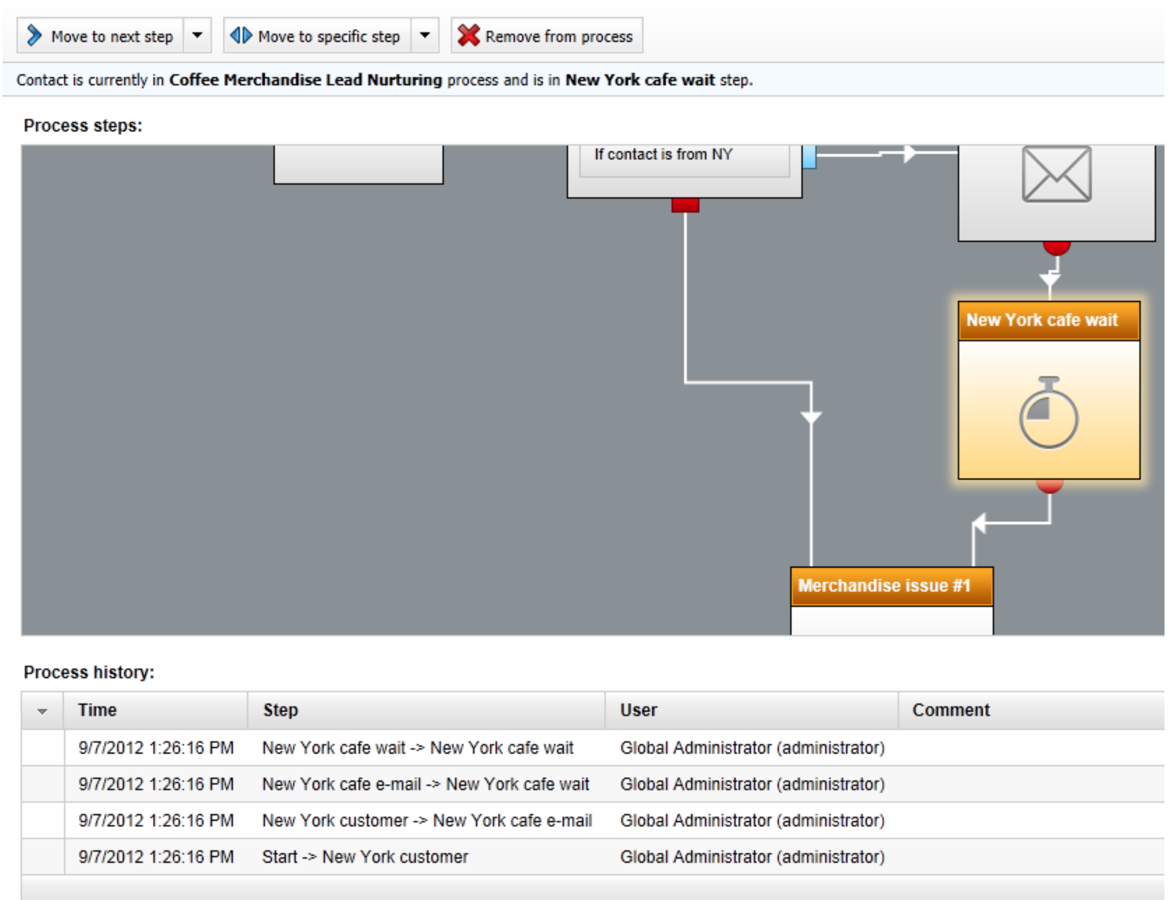
You can fully administer the process through the user account that you used to create the process:

1. Switch to the **Pending contacts** tab.
   - You can see that the process has been started for the contact.



2. Click **Edit** (✎) next to the contact.

You can see the current state of the process for the given contact. The **Move to next step**, **Move to specific step** and **Remove process** buttons are available to you.



Tip: If you haven't set the wait steps for a short time interval, use the **Move to specific step** button to move to the newsletter steps manually.

Administering the process using the Process administrator role

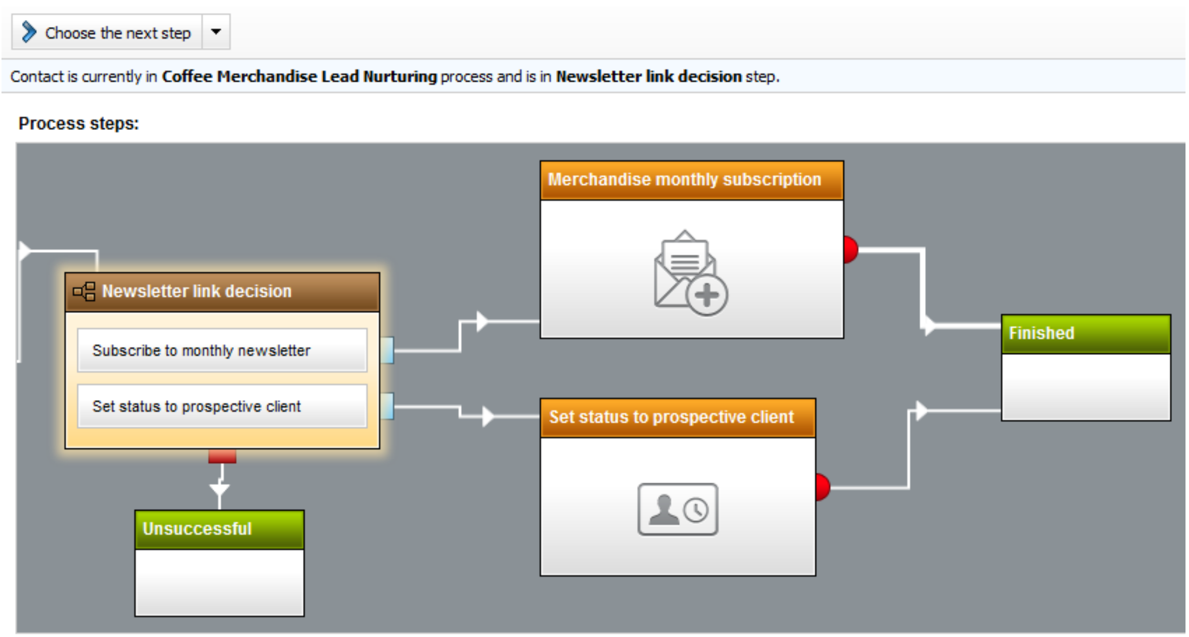
Once the instance of the process is in the **Newsletter link decision** step, you can manage the instance through the **Process administrator** role. The contact only stays in this step if you entered your own e-mail address when creating the contact and clicked the links in both newsletter issues that you received.

1. Log in to your website's **CMS Desk** as **Matt**.
   - Go to *http://<yourdomain>/CMSDesk* and enter *Matt* as the user name.

2. Navigate to **On-line Marketing -> Contacts.**

3. Open the **Pending contacts** tab.

4. Click **Edit** () next to the contact.
   - You can see the process is in the **Newsletter link decision** step.



5. Click the **Move to Set status to prospective client step** button.
   - The process finishes.

6. Switch to the **Contacts** tab.

7. **Edit** () the contact that you created for the process.
   - You can see that the contact's status changed to **Prospective client**.

You have created a complete automation process and tested some of the functionality that you can use in your own marketing automation processes.

Part VIII

Scoring

8 Scoring

8.1 Overview

Scoring (also referred to as *lead scoring* or *engagement scoring*) is an on-line marketing method that assigns numeric values to individual contacts. The system automatically calculates scores for contacts based on their properties and activities performed on the website. The purpose of scoring is to numerically evaluate contacts according to various criteria, deduce conclusions from the scores and perform appropriate marketing actions.

You can define any number of *scores*, each with a separate set of *rules*. The system assigns *score points* to individual contacts based on the rules of each score. Contacts that fulfill the conditions of the scoring rules either gain or lose points within the given score.

You can access the scoring user interface in **CMS Desk -> On-line marketing -> Scoring**. See Managing scores for additional information. The Example: Using contact scoring topic provides a step-by-step example of how to define a sample score with several rules and demonstrates how the system adds score points for contacts that match the rules.

Tip

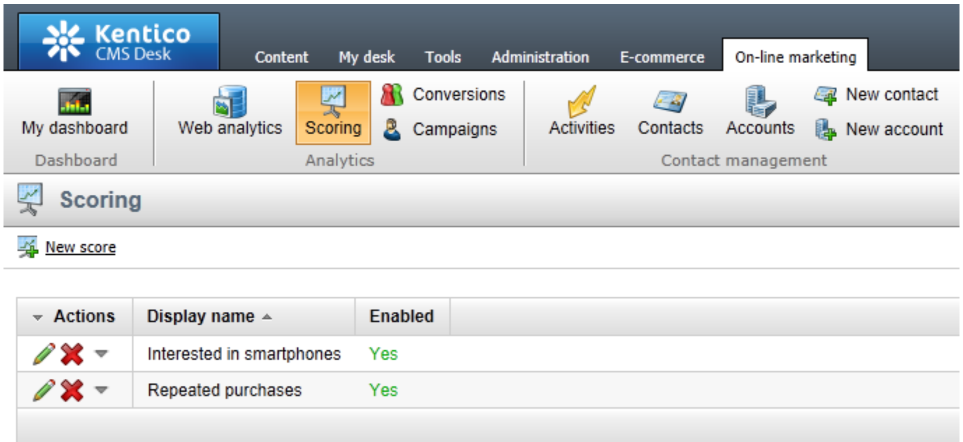
Visit the Kentico lead scoring web page, where you can also download the *Lead Scoring for Success* white paper for more information on the subject.

8.2 Managing scores

User interface of the Scoring module is located in **CMS Desk -> On-line marketing -> Scoring**. On the initial screen of the interface, you can see a list of defined scores. New scores can be added using **New score**. Existing scores have the following actions available in the **Actions** column:

- **Edit** - opens the score's editing interface where properties of the score can be modified, score rules defined and score points that individual contacts received within the score can be viewed.
- **Delete** - deletes the score.

It is also possible to **Export, Backup** and **Restore** listed scores by choosing the respective action from the **Other actions** drop-down menu.



The editing interface of each score, displayed after clicking the **Edit** () icon, is split into three tabs — **General**, **Rules** and **Contacts**.

General tab

On the **General** tab, you can edit general properties of the score. The following of them are also available when creating a new score after clicking **New score** mentioned above:

- **Display name** - name of the score used in the system's administration interface.
- **Code name** - name of the score used in web project code.
- **Description** - text describing the score.
- **Enabled** - indicates if the score is enabled, i.e. if contacts are evaluated with score points within the score.

Other properties are only available when editing an already existing contact:

E-mail notifications

The two properties listed below enable you to set up e-mail notifications about contacts reaching a certain number of score points:

- **Send notification at score** - number of score points that a contact has to reach in order for the e-mail notification to be sent.
- **Notification e-mail address** - e-mail addresses to which the notification e-mails will be sent. You can enter multiple addresses separated by semicolons (;).

The e-mail notifications are based on the **Scoring - Notification e-mail** e-mail template.

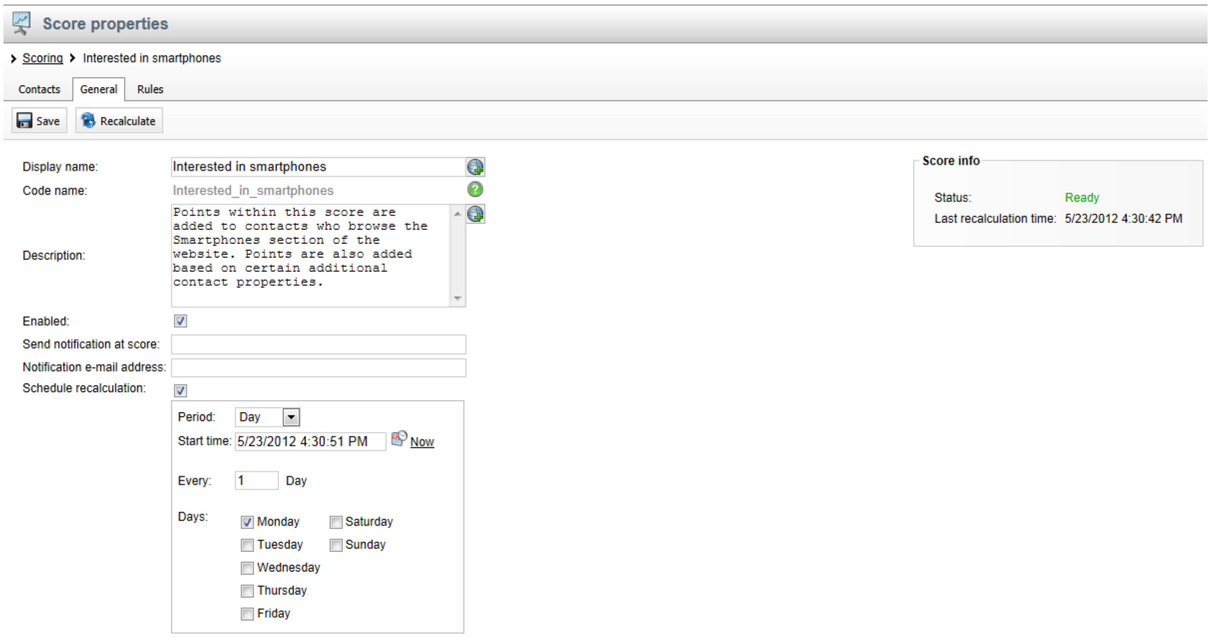
Score recalculation

When rules within the score are added or modified, it is necessary to recalculate score points previously reached by individual contacts with regards to the new rules. This can either be done manually by clicking **Recalculate**, or automatically by means of a dedicated scheduled task.

A scheduled task named **Score '<score display name>' recalculation** is created automatically in **Site Manager -> Administration -> Scheduled tasks** for every created score. It is disabled by default and can be enabled through the **Schedule recalculation** checkbox. After doing so, a set of controls is displayed below the check-box, letting you specify the period after which recalculation will be regularly performed.

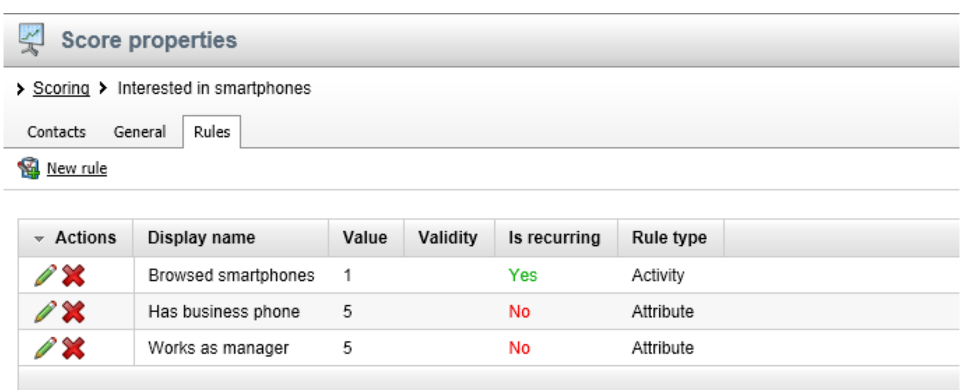
In the **Score info** box on the right, two values related to score recalculation are displayed:

- **Status** - when rules within the score were added or modified, the **Recalculation needed** status is displayed here. When the current score points reflect the currently defined rules, you should see the **Ready** status.
- **Last recalculation time** - date and time of last score points recalculation.



Rules tab

On the **Rules** tab, you can see a list of defined rules based on which score points are added to contacts within the current score. New rules can be defined after clicking **New rule** above the list. Listed rules can be **Edited** () or **Deleted** () by clicking the respective icons in the **Actions** column.

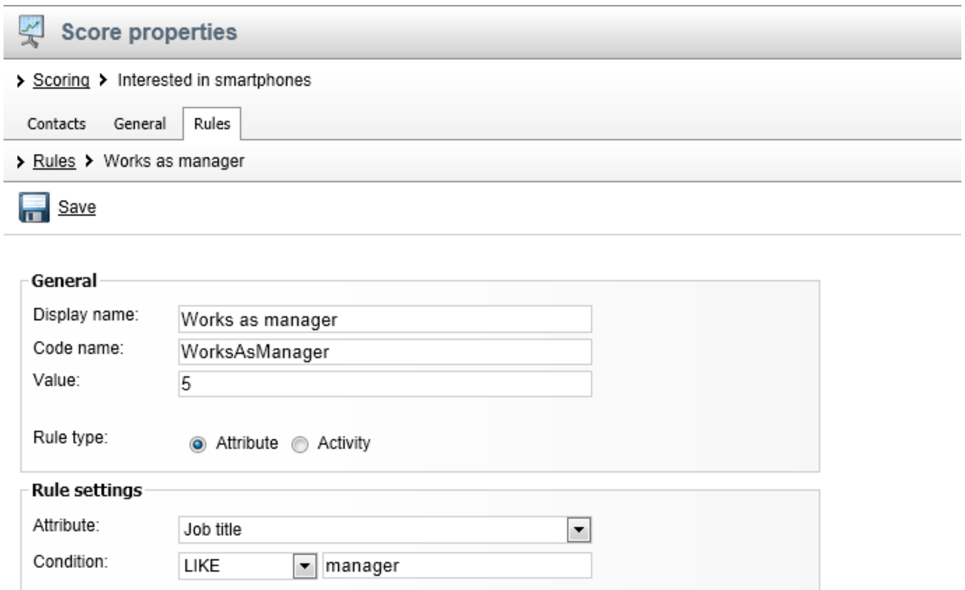


When defining a new rule or editing an existing one, the following properties can be specified:

- **Display name** - name of the rule used in the system's administration interface.
- **Code name** - name of the rule used in web project code.
- **Value** - number of score points that will be added to a contact for matching the rule.
- **Rule type** - you can choose from the following types of rules:
  - **Attribute** - score points are added to contacts whose properties match a specified value.
  - **Activity** - score points are added to contacts who perform a specified activity.

If you select the **Attribute** type, the following two fields can be used to define the rule:

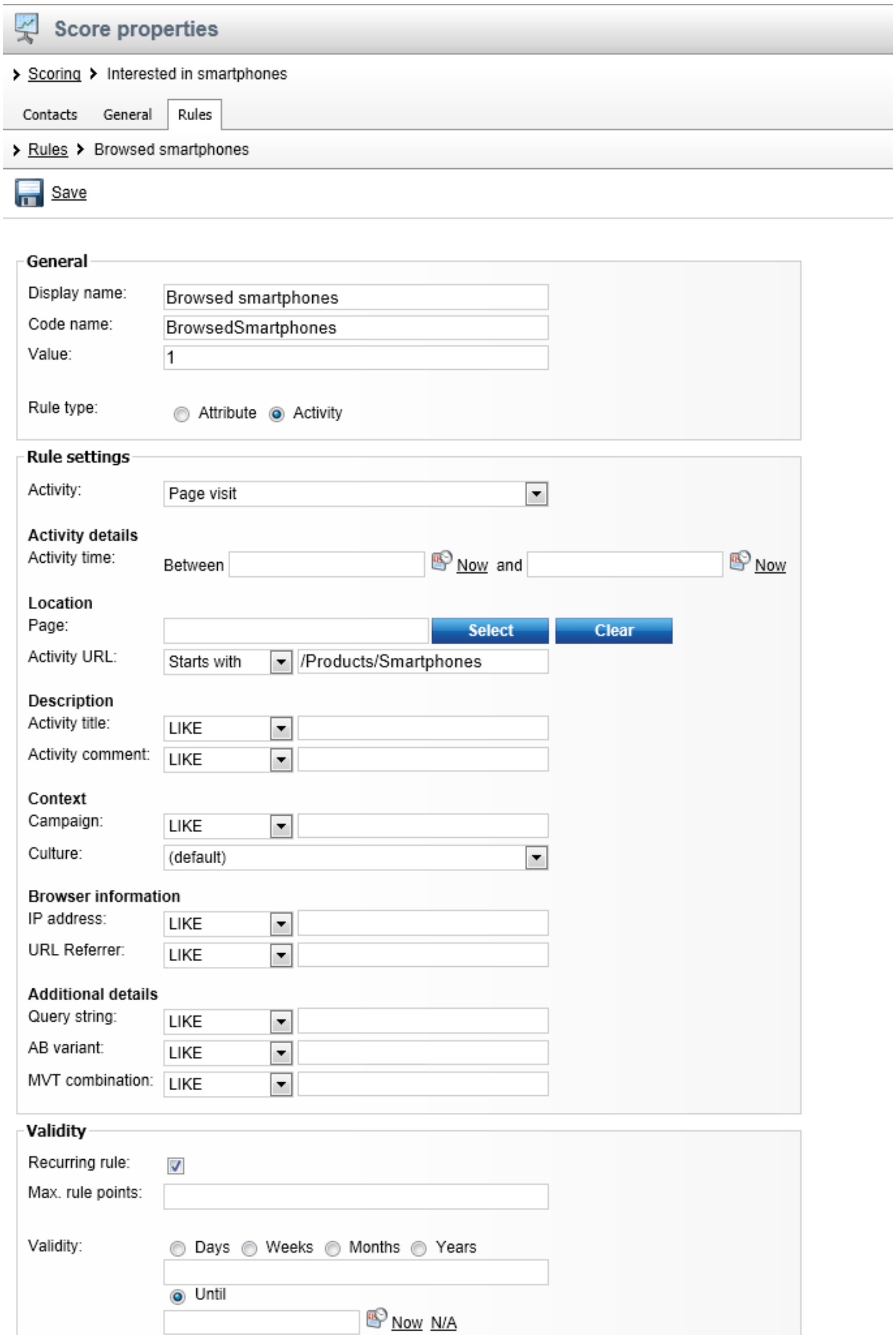
- **Attribute** - contact property that has to match the condition specified below in order for contacts to receive the score points.
- **Condition** - condition that the attribute must match in order for contacts to receive the score points.



If you select the **Activity** type, the following fields can be used to specify the required activity:

- **Activity** - type of activity that the contact needs to perform in order to receive the score points.
- **Activity time** - time period within which contacts receive score points for performing the activity.
- **Page** - path to a page where the activity needs to be performed.
- **Activity URL** - URL where the activity needs to be performed.
- **Activity title** - title of the activity.
- **Activity comment** - text entered as comment of the activity.
- **Campaign** - web analytics campaign through which the user came to the website.
- **Culture** - cultural version of the website in which the activity needs to be performed.
- **IP address** - IP address from which the activity needs to be performed.
- **URL referrer** - URL from which the user needs to come to the page where they perform the activity.
- **Recurring rule** - indicates if score points can be added to contacts for performing the activity repeatedly. If disabled, score points are added only for the first time it is performed by each contact.
- **Max. rule points** - maximum number of points that can be added to a single contact for performing the activity
- **Validity** - indicates how long the score points will be added to a contact after performing the activity. After the specified time period, the score points will be subtracted from the user's total score points.

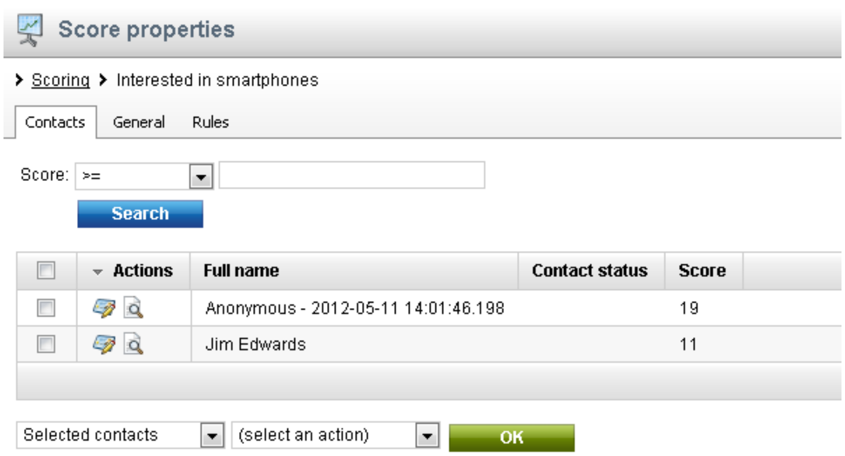
Depending on the selected type of activity, other specific properties can be displayed along with the ones listed above.



Contacts tab

On the **Contacts** tab, you can see a list of contacts who received points within the current score and the actual numbers of score points that they received (the **Score** column). Please note that points received for matching activity rules are added about one minute after the activity is performed, unless the *CMSLogActivityImmediatelyToDB* key is added to the *appSettings* section of the project's *web.config* file and set to *true*. In this case, score points for activity rules are added almost immediately.

When large numbers of contacts are displayed, you can filter them by their score points value using the **Score** filter above the list. By clicking the **View contact details** ( ) icon next to a contact, the contact's editing interface identical to the one available in **CMS Desk -> On-line marketing -> Contacts** is displayed in a new pop-up window.



You can also **View score details** ( ) for each of the contacts. You will see a new window, listing additional information about each of the rules that have contributed to the contact's total score value. You can see three values next to each of the listed rules:

- **Rule value** - this is a value you set when creating the rule. It is added to the contact's score each time it meets the rule's criteria.
- **Quantity** - shows how many times has the **Rule value** been added to the contact's score.
- **Total value** - a total of the previous two values.

8.3 Example: Using contact scoring

The following step-by-step example defines a score on the sample Corporate site and demonstrates how the system adds score points for contacts that match the rules.

Prerequisites:

- Enable activity logging for the website.

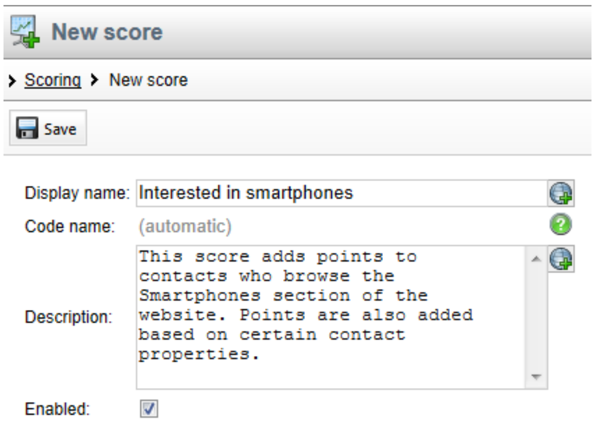
Creating the sample score

1. Go to **CMS Desk -> On-line marketing -> Scoring**.

2. Click **New score**.

3. Enter the following details in the **New score** dialog:

- **Display name**: Interested in smartphones
- **Description**: enter text describing the score
- **Enabled**: yes (checked)



4. Click **Save**.
   - The system creates the score and opens the **General** tab of the score's editing interface.

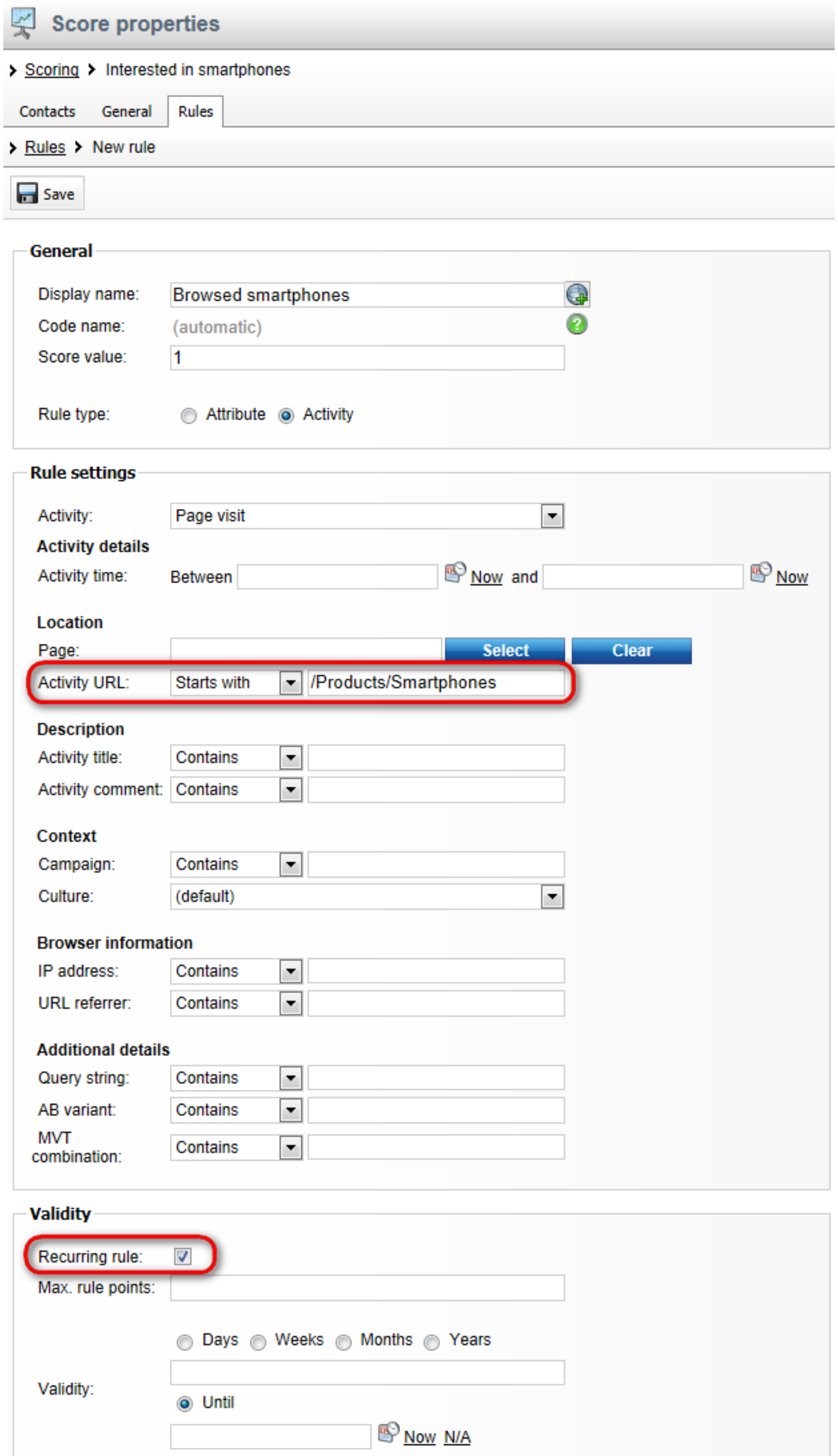
5. Switch to the **Rules** tab and click **New rule**.

6. Specify the following configuration options in the **New rule** dialog:

- **Display name**: Browsed smartphones
- **Score value**: 1
- **Rule type**: Activity

- **Activity**: Page visit
- **Activity URL**: *Starts with* /Products/Smartphones
- **Recurring rule**: yes (checked)

This rule adds one score point to contacts who view any page in the */Products/Smartphones* section of the website (i.e. the list of available smartphone products and individual smartphone detail pages). The rule is recurring, which means that points are added even for repeated visits of the pages.



7. Leave the remaining options blank or with their default values and click **Save**.

8. Click **New rule** next to the **Save** button and enter the following configuration:

- **Display name**: Works as manager
- **Score value**: 5
- **Rule type**: Attribute

- **Attribute**: Job title
- **Condition**: *Contains* manager

This attribute rule adds 5 score points to contacts who have the word *manager* in their job title (such contacts are likely to need smartphones in their job).

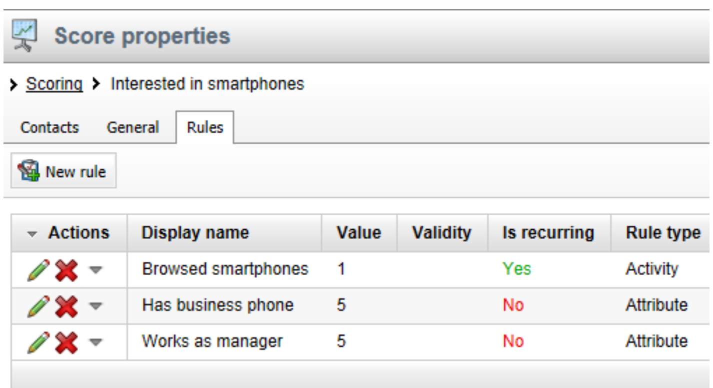
9. Save the rule and repeat the procedure one more time to create the final rule:

- **Display name**: Has business phone
- **Value**: 5
- **Rule type**: Attribute

- **Attribute**: Business phone
- **Condition**: *Is not empty*

This rule adds 5 score points to contacts who have a business phone number (such contacts can also be considered as potential smartphone buyers).

If you switch back to the score's **Rules** tab, you can see a list of the defined rules.



Sample score with three rules

Trying out the scoring functionality

Now that you have defined the rules for the score, you can test how the system adds points to contacts.

This example uses two new contacts, which the system creates automatically when you access the website as a public visitor from two different browsers. If you already have contact information stored in your browsers (for example after going through Example: Creating a sample contact), your contact data may be different, but the scoring functionality still works as demonstrated in the example.

Delay in activity rule points

By default, the system adds score points for activity rules about one minute after the activity occurs. When testing scores, you can enable immediate logging of activities, which ensures that activity score points are updated almost instantly.

To enable immediate logging of activities, administrators with access to the project's file system need to add the following key to the *appSettings* section of the *web.config* file:

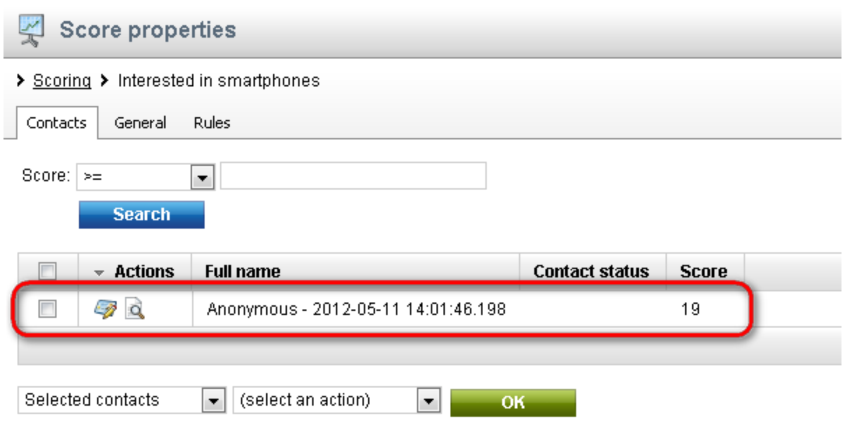
```
<add key="CMSLogActivityImmediatelyToDB" value="true" />
```

Note: To maintain optimal database performance, remove this web.config key before deploying the live version of the website.

1. Sign out of the administration interface, navigate to the **/Products/Smartphones** section of the Corporate Site, and view several product detail pages of smartphones.

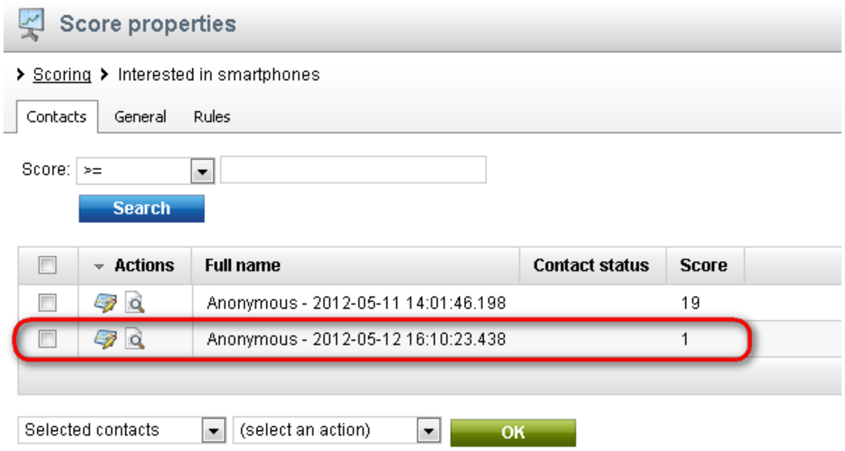
2. Log back in to **CMS Desk**, go to **On-line marketing -> Scoring** and edit ( ) the **Interested in smartphones** score.

3. Open the **Contacts** tab.
   - You should see a new contact with a score that matches the number of pages that you viewed in the Smartphones section.



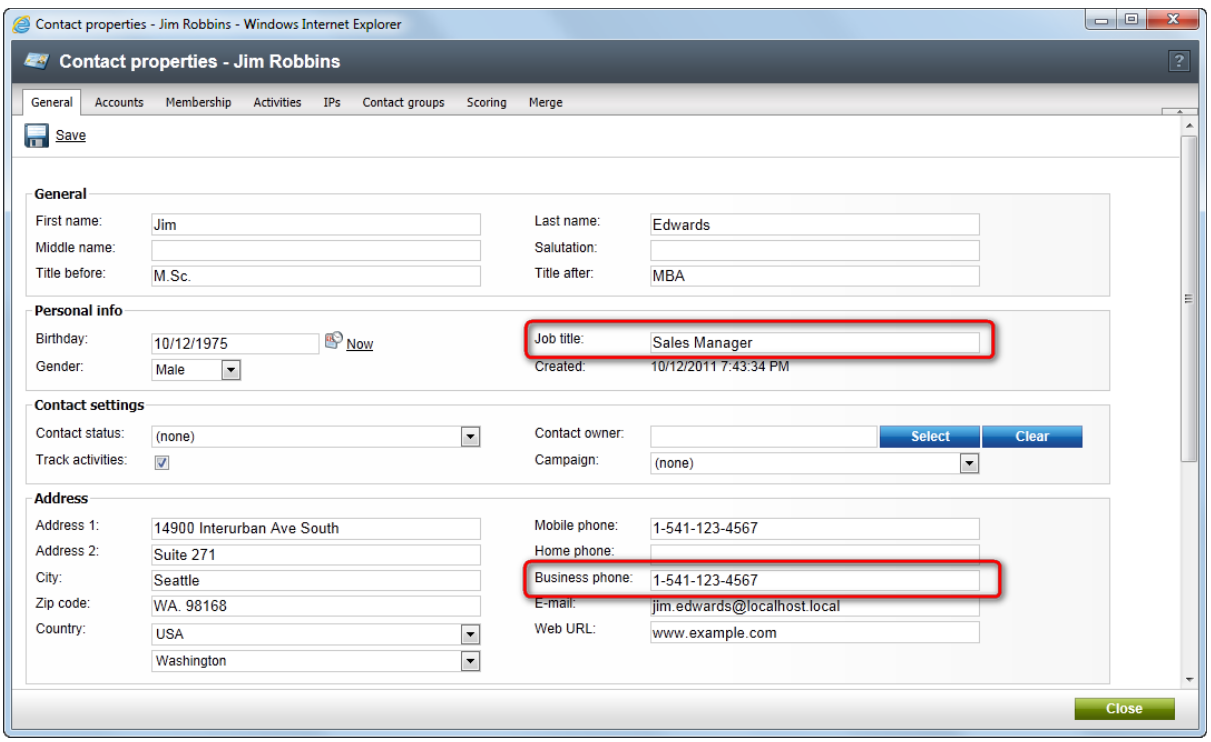
4. Open a different web browser and access the website's **/Products/Smartphones** section as an anonymous visitor again. This time, only view the list of smartphones once.

5. Switch back to the first browser and refresh the **Contacts** tab.
   - You should see another contact, with only one point added for viewing the list of smartphones.

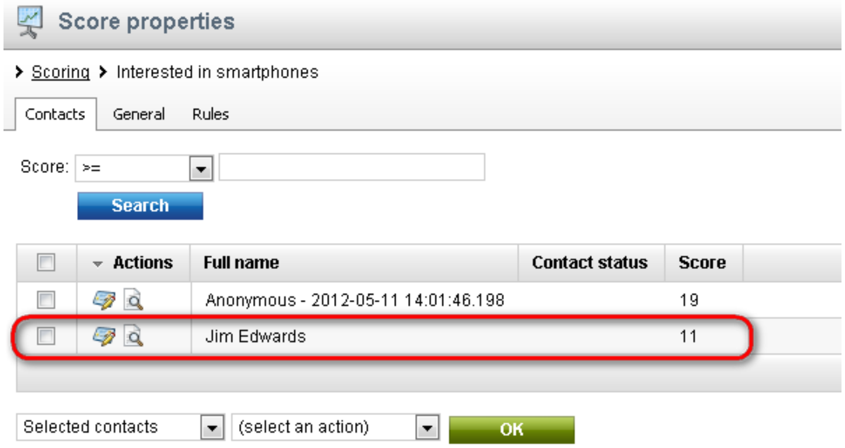


6. Click **View contact details** ( ) next to the second contact.
   - A dialog appears where you can modify the properties of the given contact. The options are the same as the main contact editing interface (see the Contacts chapter for details).

7. On the **General** tab, fill in a **Business phone** number and a **Job title** containing the word *manager*.



8. Click **Save** and **Close** the dialog.
   - Back on the **Contacts** tab, you should see that the second contact now has 11 points (you may need to refresh the tab). The system added 5 points because the contact has a business phone and another 5 due to the word *manager* in the contact's job title.



9. Click **View score details** ( ) next to the contact.
   - The score details dialog shows exactly which rules have contributed to the contact's total score.

Part IX

Website optimization

9 Website optimization

9.1 Overview

Making improvements on a website can be a difficult process, since it is often not possible to know ahead of time whether changes will have a positive effect, or which modification out of several options will bring the best results. You can resolve these issues by running optimization tests for pages.

Optimization testing allows you to create different versions of a page (or specific parts of a page) and evaluate them according to the behavior of the website's visitors. You can confirm which changes are actually beneficial and use the content that works best for the users who visit your website. The testing process does not interfere with browsing on the website. Visitors do not need to give any feedback manually.

There are two different techniques that you can use to optimize pages. Each type of testing has its advantages and is intended for different scenarios:

A/B testing	Multivariate testing (MVT)
• Divides incoming traffic between two or more different variants of a page • Tracks the results for each page variant as a whole (measures the combined effect of all changes made to the page) **Pros & Cons**: • Usually more straightforward, easier to set up and evaluate • Most suitable for situations where you need to test a single variable element or a full redesign that changes the entire appearance of a page • Each variant is a separate document in the content tree	• Allows you to make multiple modifications to the content of a single page • Tracks the results for each specific change on the page **Pros & Cons**: • Better option for evaluating multiple variables on a single page • Allows you to monitor the effects of individual modifications with a greater degree of detail • Typically involves more tested variants, so may require more time (or site traffic) than A/B testing to get meaningful results

Both types of optimization testing measure results by tracking the activity of users after they access the tested page and view one of the different content versions. Actions that are desired from visitors are represented in the system as ***Conversions***. Typical examples of conversions are product orders, registrations, newsletter subscriptions, views of special pages etc. When a user performs the action tracked by a conversion, the system logs a ***Conversion hit***. To learn more about conversions and how they can be implemented on your website, please read the Conversions chapter.

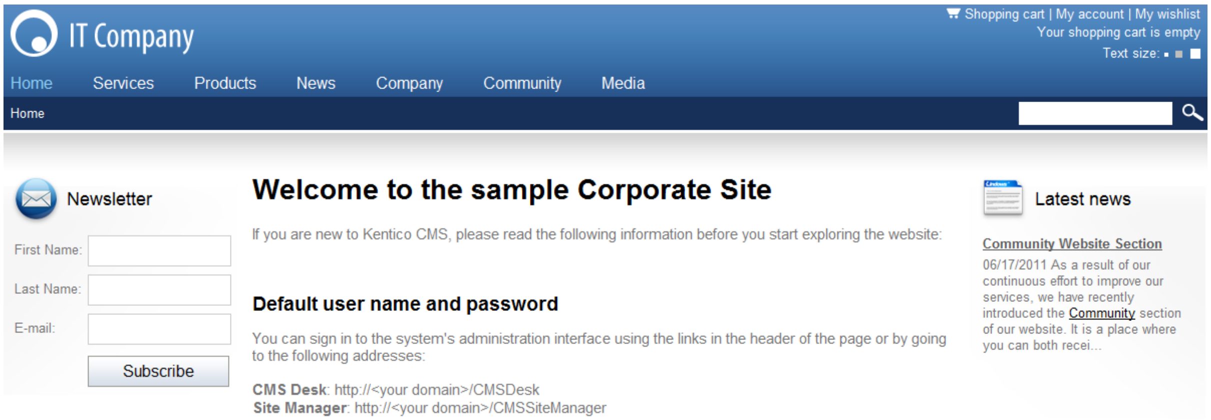
Testing is most recommended for key sections of the website that receive the most traffic, such as the default home page (landing page).

9.2 A/B testing

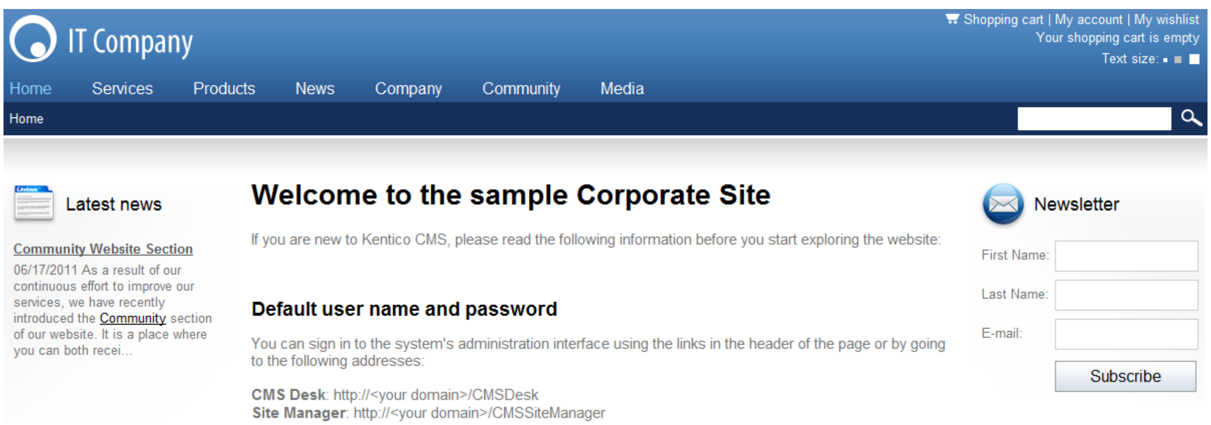
9.2.1 A/B testing overview

A/B testing is an on-line marketing technique used to optimize the pages of a website according to the reactions of visitors. This is achieved by creating one or more modified versions of a given page and running them all at the same time during a designated testing period. The system then divides traffic between individual page versions and tracks how the changes affect the user experience and activity of visitors.

The following images show two versions of a sample home page — the original and an alternative page with the position of the left and right content columns reversed.



Some visitors will be assigned to the second version and will see the following when they view the home page:



There are no limitations placed on the modifications that can be made to different page options. They may include anything from subtle changes to using a completely different page.

In Kentico CMS, the basic objects that provide this functionality are ***A/B tests***, which can be created for specific website pages. The page options defined for a test are called ***Page variants***. Each variant is linked to a document in the content tree of the website that can be designed and configured using the standard web development process. Usually, the ***Original page*** for which the test was created will be included as one of the variants, so that potential improvements can be compared with the baseline statistics (those of the unmodified page).

For a practical scenario, imagine that you have an e-commerce page on your website where visitors can purchase products and you wish to fine-tune it to be more user friendly in order to increase the amount of sales. By utilizing A/B testing, you can create multiple versions of this page with any type of modifications such as a slightly altered layout, graphical design or text content. These alternative pages will then serve as variants of the A/B test. While the test is running, it will automatically display different page variants to different visitors and keep track of all product purchases as conversions. When the test is completed after a certain amount of time, you can evaluate which page variant encouraged more users to make purchases on the website and set the most successful one as the permanent page.

Please see the Managing A/B tests topic for detailed information about how an A/B test can be defined for a page. If you wish to try out a live A/B testing scenario, you can do so on the sample Corporate Site according to the instructions in the Example: Sample A/B test topic. The data gathered by A/B tests is displayed using reports that allow you to view and analyze several types of conversion metrics, as described in Analyzing A/B test results.

With A/B testing, results are tracked for entire page variants, which means that the measured statistics reflect the combined effect of all changes made to each variant. If you wish to monitor how individual page modifications affect the behaviour visitors, you may use Multivariate testing, which is another website optimization feature provided by Kentico CMS.

How it works

When a visitor navigates to a page that has a running A/B test defined, one of the page variants configured for the given test will be displayed to them. The variant is chosen randomly for every user. With a large enough visit sample size, each page variant should receive roughly the same amount of traffic during the course of the test.

A persistent cookie is stored in the visitor's browser, used to identify which variant was assigned by the given A/B test. The name of the cookie uses the following format: *CMSAB<A/B test code name>*. It saves the code name of the assigned page variant as its value. This cookie expires either within 30 days after the last visit on the tested page, or on the date when the test is configured to end.

Any conversions performed on the website by users who have passed through an A/B test page will be logged under the assigned page variant, which is taken from the value of the A/B testing cookie. The logging of conversion hits is provided by the Web analytics module. In addition to monitoring conversions, the cookie also ensures that returning visitors are always shown the page variant that was previously assigned to them, which helps avoid confusion by maintaining a consistent appearance of the tested page.

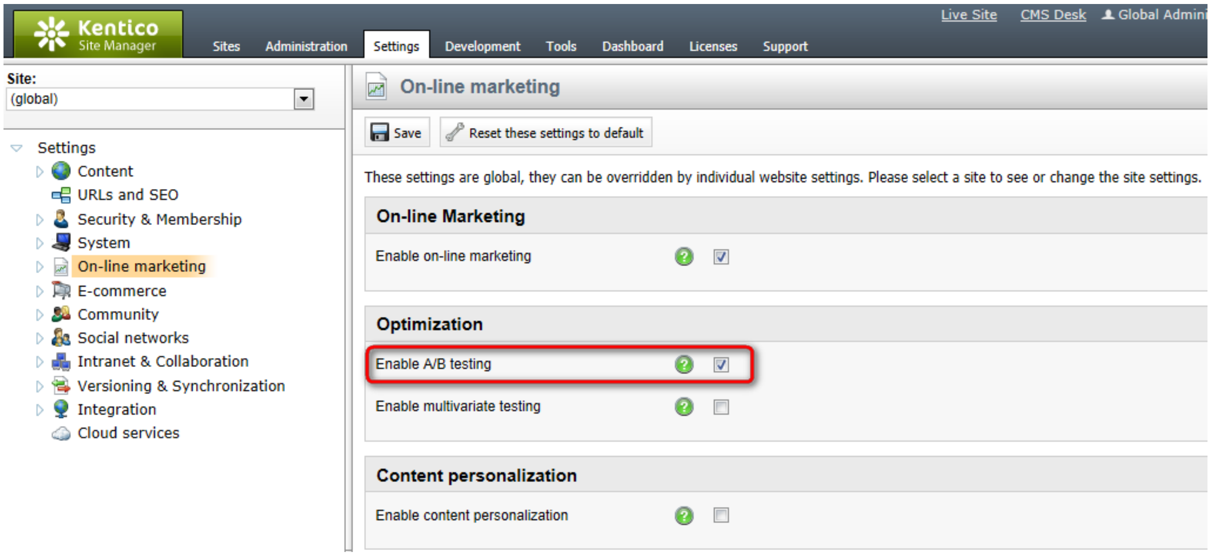
It is possible to run multiple A/B tests concurrently for different pages on the same website. Conversions will be logged for all tests defined for pages visited by a given user, according to the cookies present in the browser.

9.2.2 Managing A/B tests

The sections below describe the general steps that need to be taken in order to successfully set up an A/B test on your website.

Enabling A/B testing

Before you can start creating A/B tests for pages, it is first necessary to enable the module by going to **Site Manager -> Settings -> On-line marketing** and checking the **Enable A/B testing** field.



Since web analytics are used to log conversion statistics for individual page variants, the **Enable web analytics** and **Track conversions** fields must additionally be enabled in the **On-line marketing -> Web analytics** sub-category of the settings. All other requirements for the correct functioning of the web analytics module also need to be fulfilled.

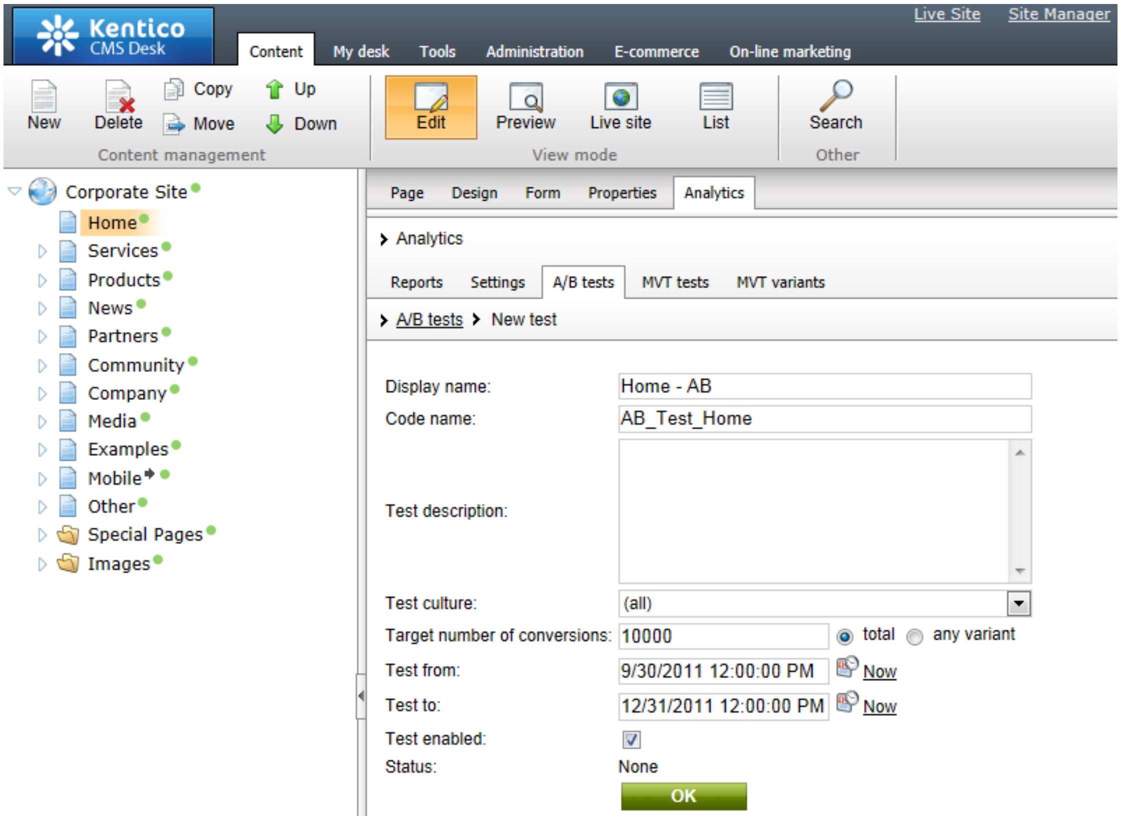
Defining an A/B test for a page

To start optimizing a page via A/B testing, create and configure an *A/B test* object for it. This can be done in the **Content -> Edit** mode of **CMS Desk** by selecting the appropriate document from the content tree and switching to its **Analytics -> A/B tests** tab. Here you can find a list of all A/B tests assigned to the current document and manage them as required. Use **New test** to add an A/B test.

When creating or editing A/B tests, the properties described in the table below are available:

Display name	The name of the A/B test displayed in the administration interface.
Code name	Sets a code name that serves as an identifier for the test. It is also used in the name of the browser cookie used to store which of the test's page variants was assigned to a visitor.
Test description	Can be used to enter a text description for the test in order to give information about its purpose, goals etc.
Test culture	Used to selects which culture version(s) of the document should be included in the test.
Target number of conversions	Sets the number of conversion hits that must be logged to complete the test. Once this number is reached, the A/B test will automatically stop working and switch to the **Finished** status. If the **total** option is selected, then the number will be compared with the total sum of the conversion hits logged for all test page variants. If **any variant** is chosen, then the test

	will be concluded when the specified number is reached by one variant (the one with the most conversion hits). Leaving this property empty or setting it to *0* means that there will be no conversion hit limit for the test.
Test from/to	Sets the time interval during which the test should occur. The **Calendar** button () can be used to select the exact date and time.
Test enabled	This property may be used to manually start or stop the test.
Status	Displays the current status of the A/B test. The following statuses are possible: • **Disabled** - indicates that the test is not active. The original tested page will always be displayed to visitors and conversions will not be logged for the test's variants. • **Not running** - indicates that the test is not active yet. Used when the test is enabled, but the *Test from* date is still in the future. • **Running** - indicates that the test is currently active. • **Finished** - this status is automatically assigned after the *Test to* date passes or when the current number of conversions reaches the *Target number of conversions*. Tests with this status are no longer active. Each page may have only one A/B test running at a given time. It is however possible to have multiple finished or inactive (disabled) tests assigned to a page, which can be used to archive data from previous tests or when preparing future tests. Different culture versions of a page may each have a different test running at the same time.



Managing A/B tests through the Web analytics interface

If you wish to manage all tests assigned to different pages from a single location, go to **CMS Desk -> On-line marketing (or Tools) -> Web analytics**. Then expand the **Optimization -> A/B tests** category, select any of the contained items (reports) and switch to the **A/B tests** tab.

Here, all A/B tests defined on the current website and their page variants may be managed in the same way as described for individual documents. The only difference is that the **Original page** property is additionally available when editing a test, which can be used to assign tests to specific pages.

Adding page variants to an A/B test

Test variants are created the same way as any other pages, since each variant is represented by a separate document in the website's content tree and can be managed in **CMS Desk -> Content**. Because of this, you can utilize all page configuration and design options provided by Kentico CMS. Please refer to the Development -> Web development overview chapter of the Developer's Guide if you require basic information about page development.

Usually, the page variants of an A/B test will all use the same basic concept, but have one key difference that sets them apart from the original page, such as:

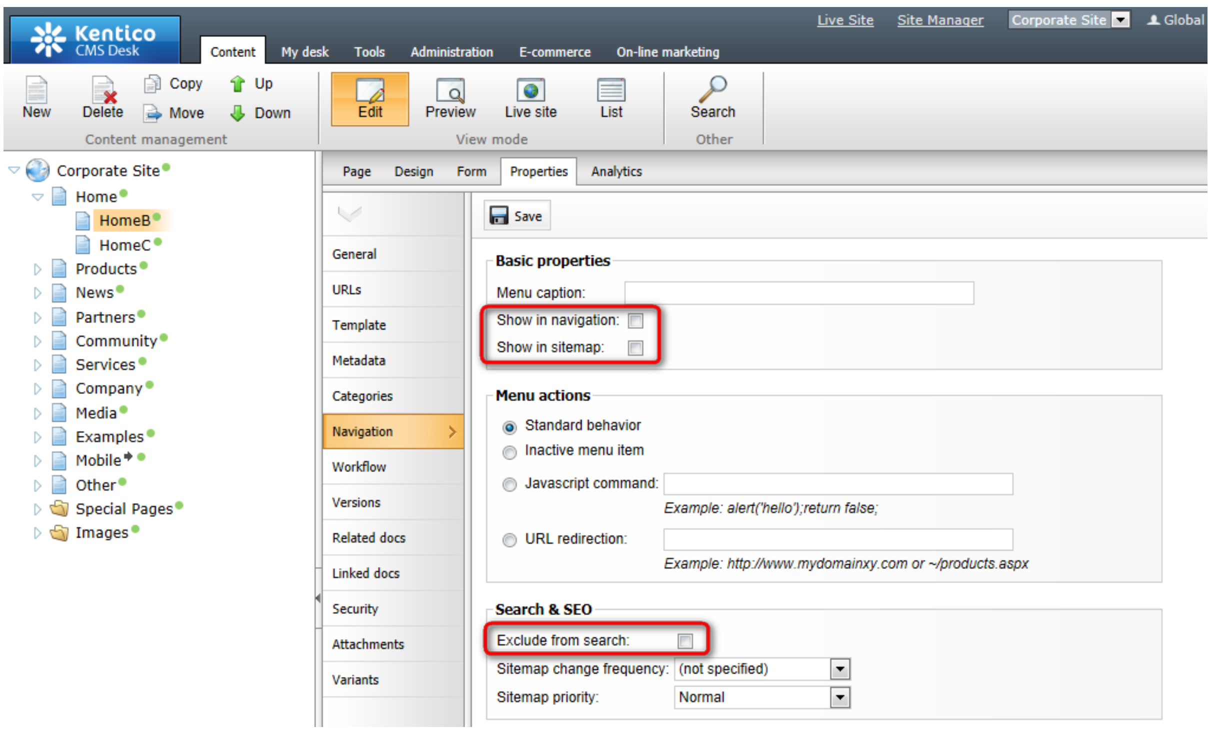
- The position of an important page element.

- A different color scheme intended to highlight a part of the page.
- Alternative text content of headings or descriptions that could potentially be more interesting for visitors.

These are only basic examples of ideas that can be tested by individual variants. You can implement any other type of modifications according to your specific requirements.

You may also leverage existing functionality to help present all variants defined for an A/B test as a single page. The following two properties are available on the **Properties -> Navigation** tab when editing any document:

- **Show in navigation** - indicates if the page variant should be displayed by navigation controls and web parts (i.e. in the website's menus).
- **Show in site map** - indicates if the page variant should be displayed by the Site map web part and included in the website's Google Sitemap.

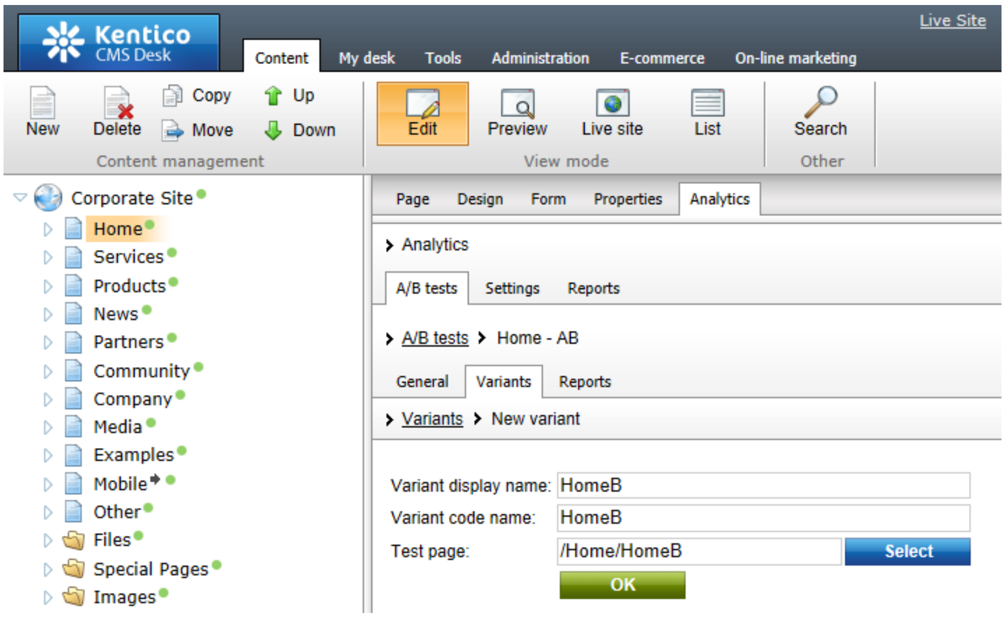


In most cases, it is recommended to have these properties disabled for page variants in order to prevent users from encountering multiple pages with nearly identical content. The appropriate variant will automatically be displayed to users when they view the test's original page, so standard navigation is not necessary.

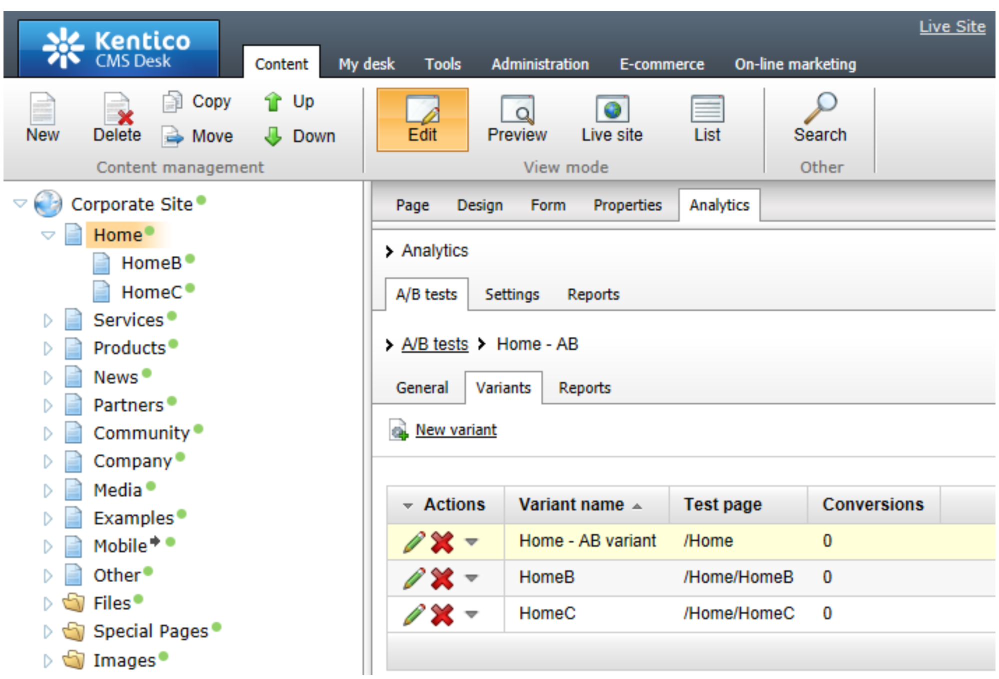
Additionally, the **Exclude from search** flag can be enabled for page variants under the **Search & SEO** section. This will ensure that the given variant will not be returned in the results of searches performed on the website.

Once the modified versions of the original document are added to the content tree, they must be registered as variants of the given A/B test. To do this, select the original document, go to **Analytics -> A/B tests**, edit ( ) the test and switch to its **Variants** tab. Click **New variant** and enter the following properties:

- **Variant display name** - the name of the variant displayed in the administration interface.
- **Variant code name** - sets a code name that serves as an identifier for the page variant. It is also used as the value of the A/B testing cookie stored for visitors assigned to this variant.
- **Test page** - must contain the path of the associated page (document). When users assigned to the variant access the original page of the given A/B test, they will see the page specified here instead. The *Select* button may be used to choose an existing page from the website's content tree.



Repeat this process for all documents that should be included as possible page options for the A/B test. The original document is automatically added as a variant when the A/B test is created, and highlighted in the list by a yellow background.

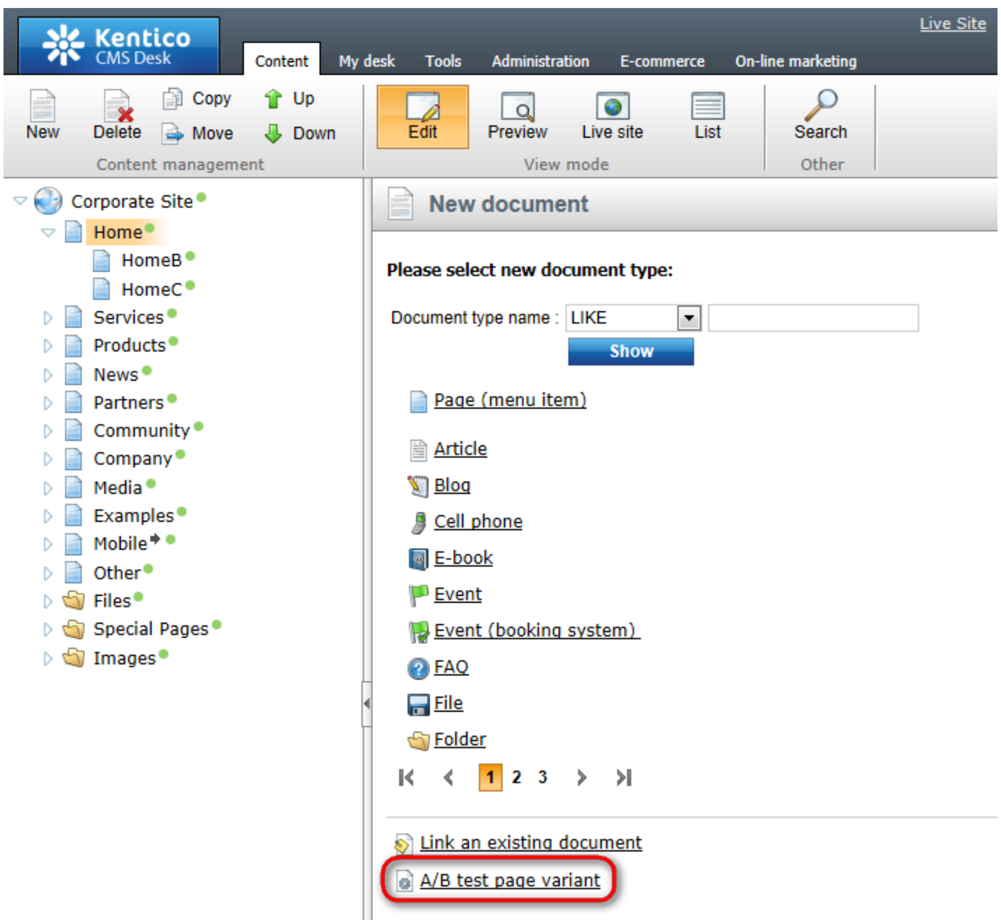


Important!

When editing a running A/B test, it is necessary to always keep the variant **Test page** paths up-to-date and ensure that the target documents actually exist in the content tree. Otherwise visitors may encounter a *Page not found* error if the path of their variant is not valid.

To preserve conversion statistics, variants remain in the system even if their associated document is deleted.

It is possible to save time when defining page variants by using an action that combines the two steps described above. Simply select the A/B test's original page in **CMS Desk -> Content** and click the **New** button in the menu above the content tree. Instead of selecting a document type, choose the **A/B test page variant** option below the list.



This can also be done by right-clicking the document in the content tree and selecting the **New -> A/B test page variant** item from the context menu. A dialog with the following options will be opened:

- **Document name** - sets the name of the new document that will be associated with the page variant.
- **Save under location** - sets the path of the document under which the page variant will be placed.

The *Select* button may be used to choose a parent page directly.
- **Assign to A/B test** - selects the A/B test to which the variant will be added.
- **Show in navigation, Show in site map, Exclude from search** - these options may be used to directly set the corresponding properties of the new document. The default values will hide the page variant from all standard website navigation.

This will create a copy of the selected document and automatically add it as a page variant of the specified A/B test. Now you can implement the required modifications that will distinguish the variant from the original page. Please keep in mind that this copy will use the same page template as the original document, so the template must be *Cloned as ad-hoc* on the document's **Properties -> Template** tab if you wish to make any changes to the design or layout of the page.

Configuring conversions

Conversions allow you to track the behaviour and actions of the website's visitors, so they must be used in order to get A/B testing results. They work on a site-wide level and do not need to be assigned to individual A/B tests in any way. All that needs to be done is define conversion objects and link them with the actions that you wish to log for the test. There are many ways to specify that an action should be logged as a conversion. You can find information about conversion management in the Conversions chapter of this guide.

When an action designated as a conversion is performed anywhere on the website, the system checks if the given user has passed through a page with an A/B test (according to the presence of a cookie). If this is the case, the conversion hit will be logged for the page variant assigned to the user.

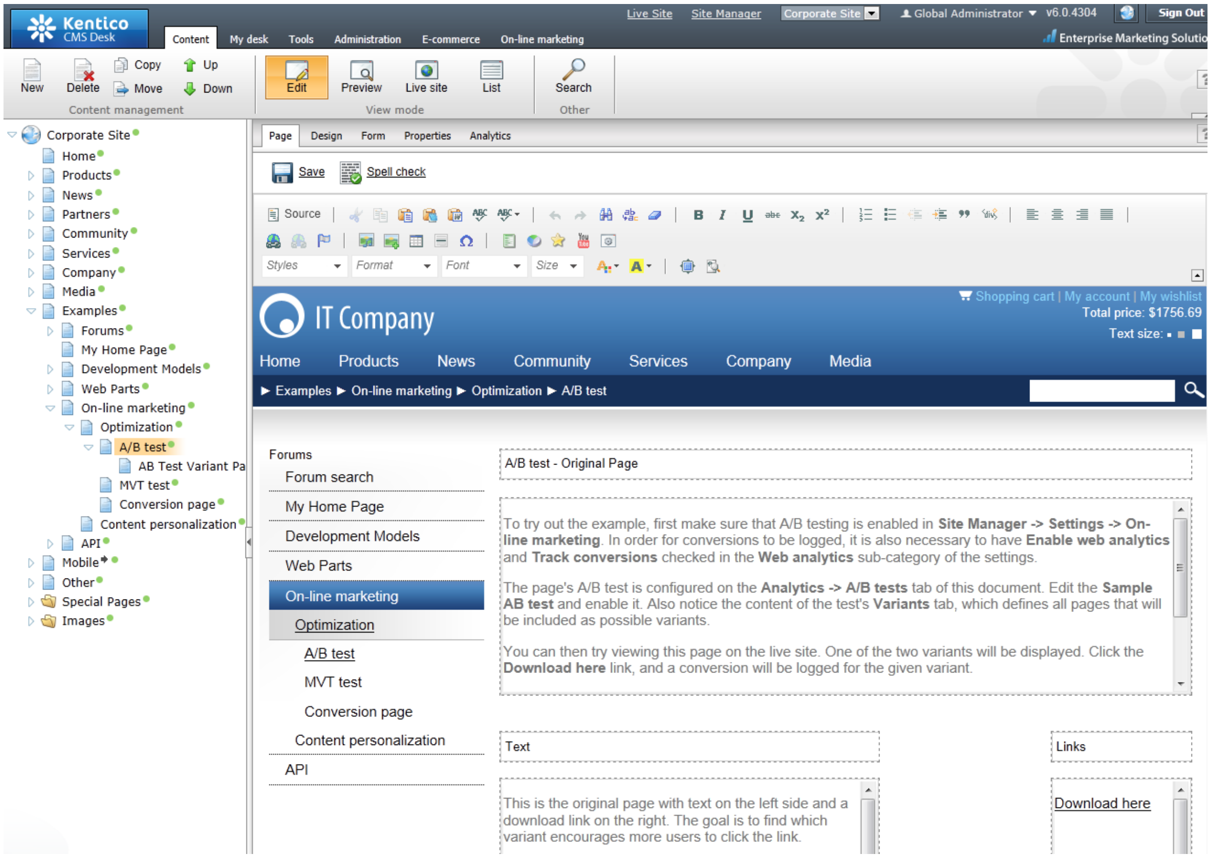
Once the A/B test starts running on the live website, you can monitor the conversion statistics for individual page variants using pre-defined reports, as described in the Analyzing A/B test results topic.

9.2.3 Example: Sample A/B test

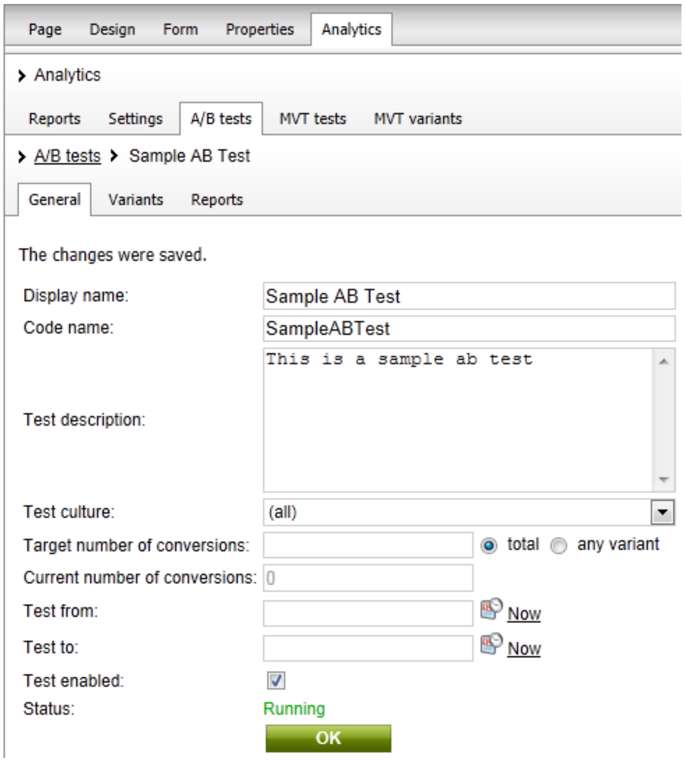
The following example demonstrates how the sample A/B test included on the Corporate Site works. The first part shows how the test looks from the live site perspective once it is started. In Part 2, you can learn how to create a new page variant and add it to the test.

Part 1: Live testing example

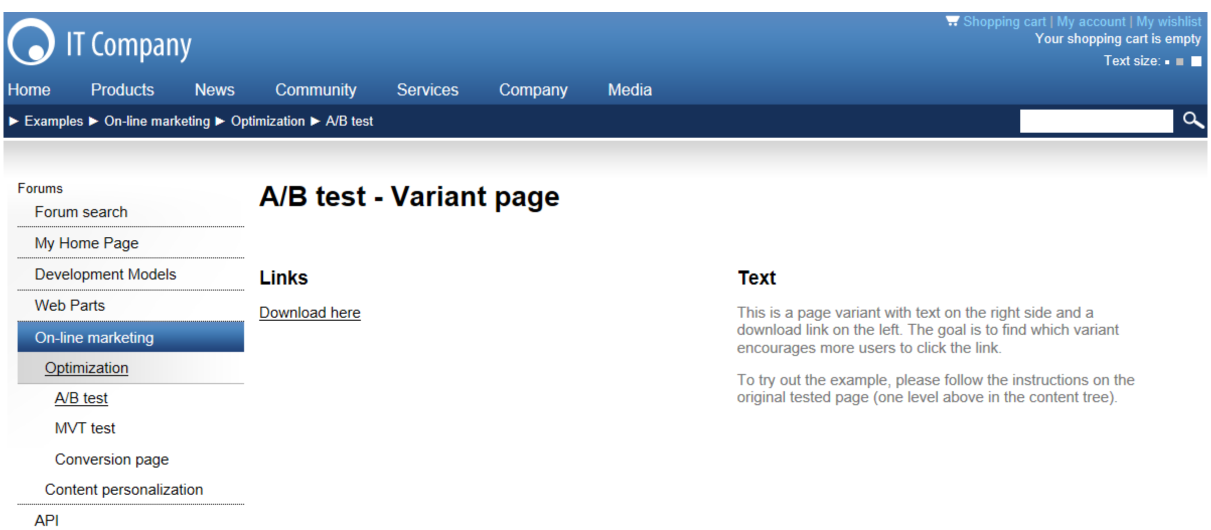
1. Go to **CMS Desk -> Content -> Edit** and select the **A/B test** document, which can be found under **Examples -> On-line marketing -> Optimization**. This document already has an A/B test defined and one page variant added as a child document. The testing scenario is very basic, with the page being composed of a text column on one side and a link on the other. The layout is reversed on the page variant and the goal of the test is to find out which version gets more link clicks from visitors.



2. To start the test, switch to the **Analytics -> A/B tests** tab of this document, edit (✏) the **Sample AB Test**, check the **Test enabled** box on the **General** tab and click **OK**.



3. Now try viewing the page on the live site. To access it, sign out of CMS Desk, open the **Home** page of the Corporate Site, scroll down to the bottom and click the **Examples** link under the **Where to learn more?** section. Then, use the menu on the left to navigate to **On-line marketing -> Optimization -> A/B test**. When you view the tested page, one of the two possible variants will be randomly selected and displayed.



4. Click on the **Download here** link, and a conversion will be logged for the given page variant by the sample **Conversion page**. If you return to the A/B test page, the same variant that was assigned during the first visit will always be displayed.

5. Now open the same page using a different browser. Again, one of the page variants will automatically

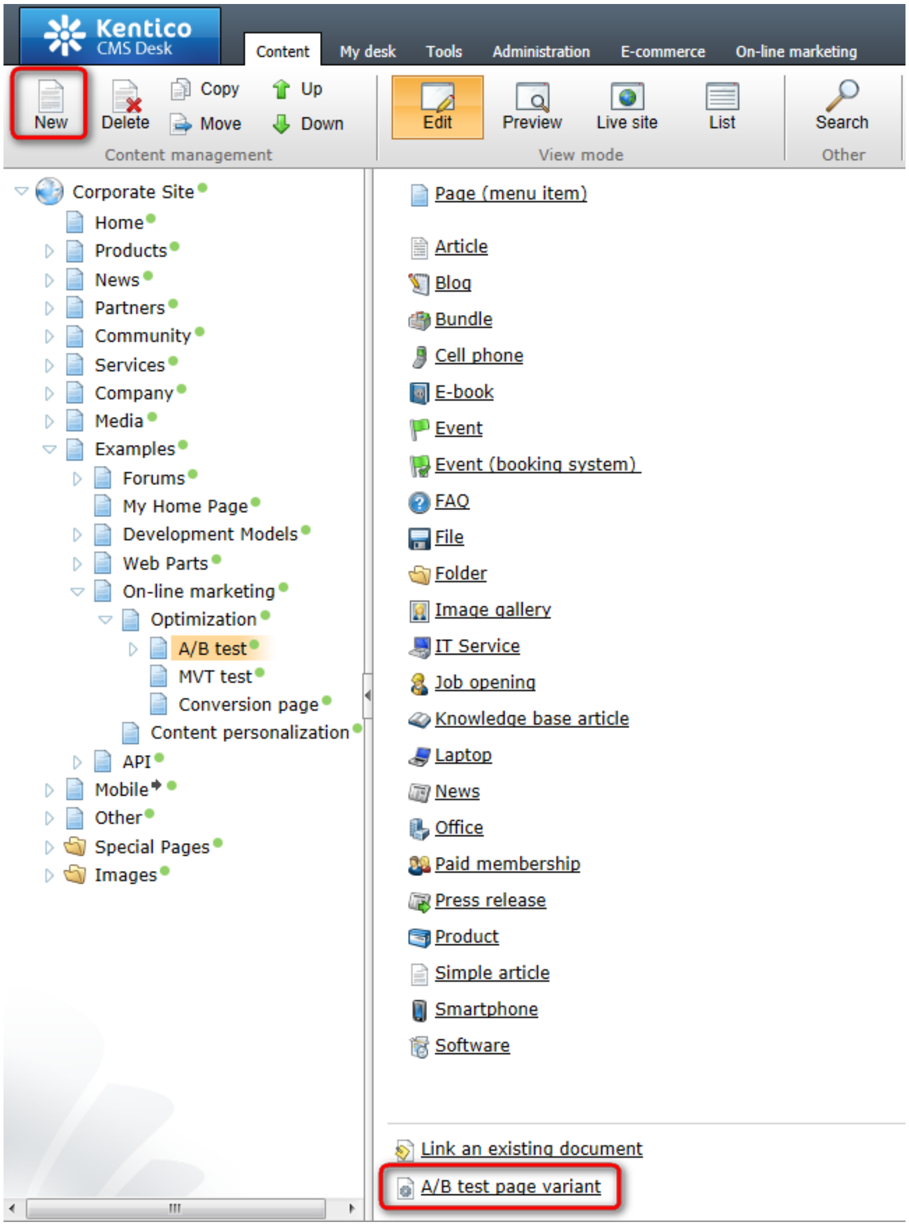
be chosen and assigned. Please note that it may be the same variant as before, since the selection process is entirely random. You can simulate visits from multiple users by clearing the cookies from your browser between views of the tested page. Try to use this approach to log several conversions for each variant.

Please continue reading in the Analyzing A/B test results topic to see how you can view the results measured for the sample test performed in the previous steps.

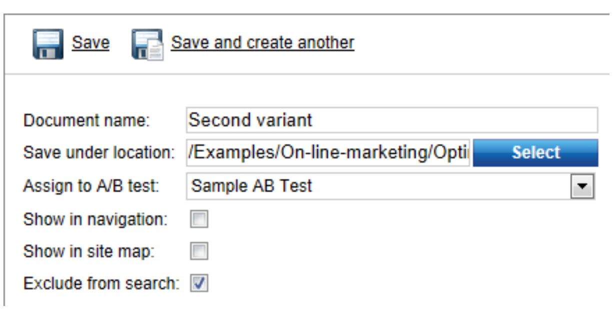
Part 2: Creating an A/B testing page variant

This part of the example demonstrates how you can add another variant to the test used in the section above. Please note that it is generally not recommended to add, remove or otherwise modify the variants of a running test, since this may affect the accuracy of the results (this is not a concern in the case of this example).

6. Log in to **CMS Desk -> Content -> Edit** and select the **A/B test** document again (under **Examples -> On-line marketing -> Optimization**). Then click the **New** action located in the menu above the content tree and choose the **A/B test page variant** option below the list of possible document types.

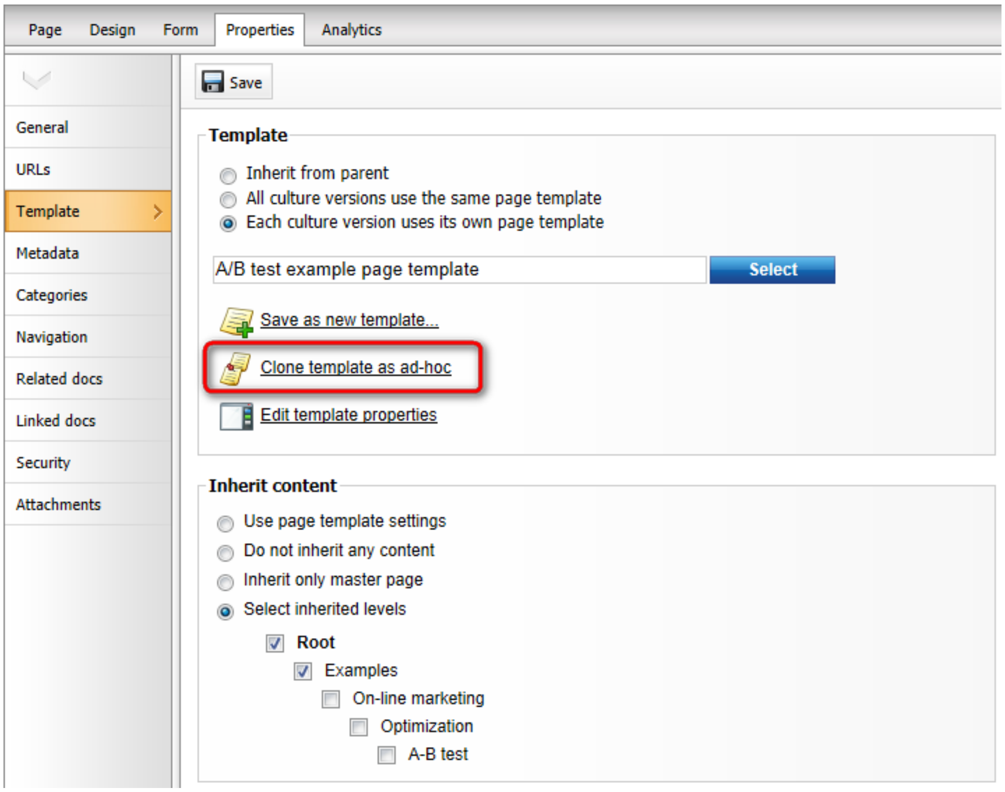


7. A dialog will appear, where you can specify the settings of the page variant. Enter *Second variant* into the **Document name** field and leave the default values for the remaining properties.

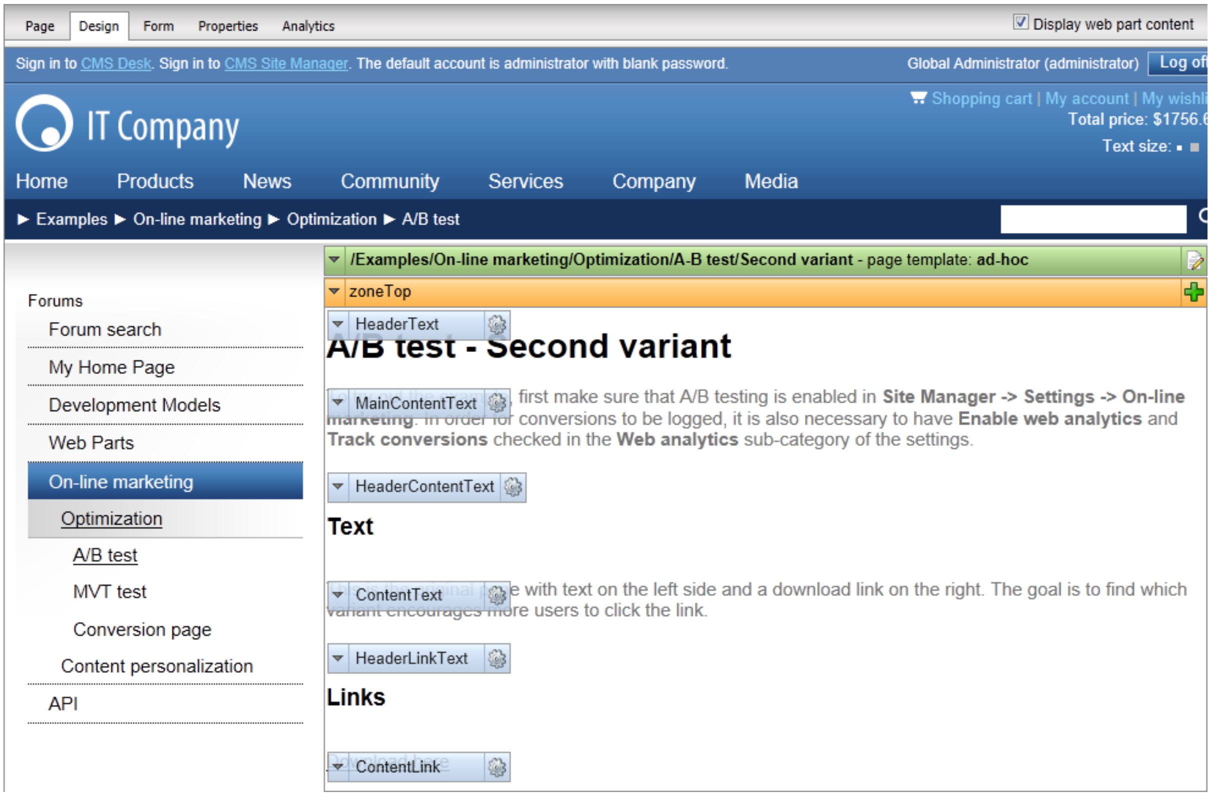


Click the **Save** button and the variant will be created as a child under the currently selected document. Additionally, it will automatically be assigned to the *Sample AB Test*.

8. By default, new variants added this way use the same page template as the parent document, so you need to assign a unique template before you can start modifying the design of the page variant. To do this, select the **Second variant** document, go to the **Properties -> Template** tab and click on the **Clone template as ad-hoc** action.



9. Now that the document no longer shares its template with the original page, switch to the **Design** tab. In this example, the new variant will test a single column layout where the "Download link" is placed directly below the text. To achieve this, simply drag the web parts from the *zoneLeft* and *zoneRight* zones into *zoneTop* as shown below:

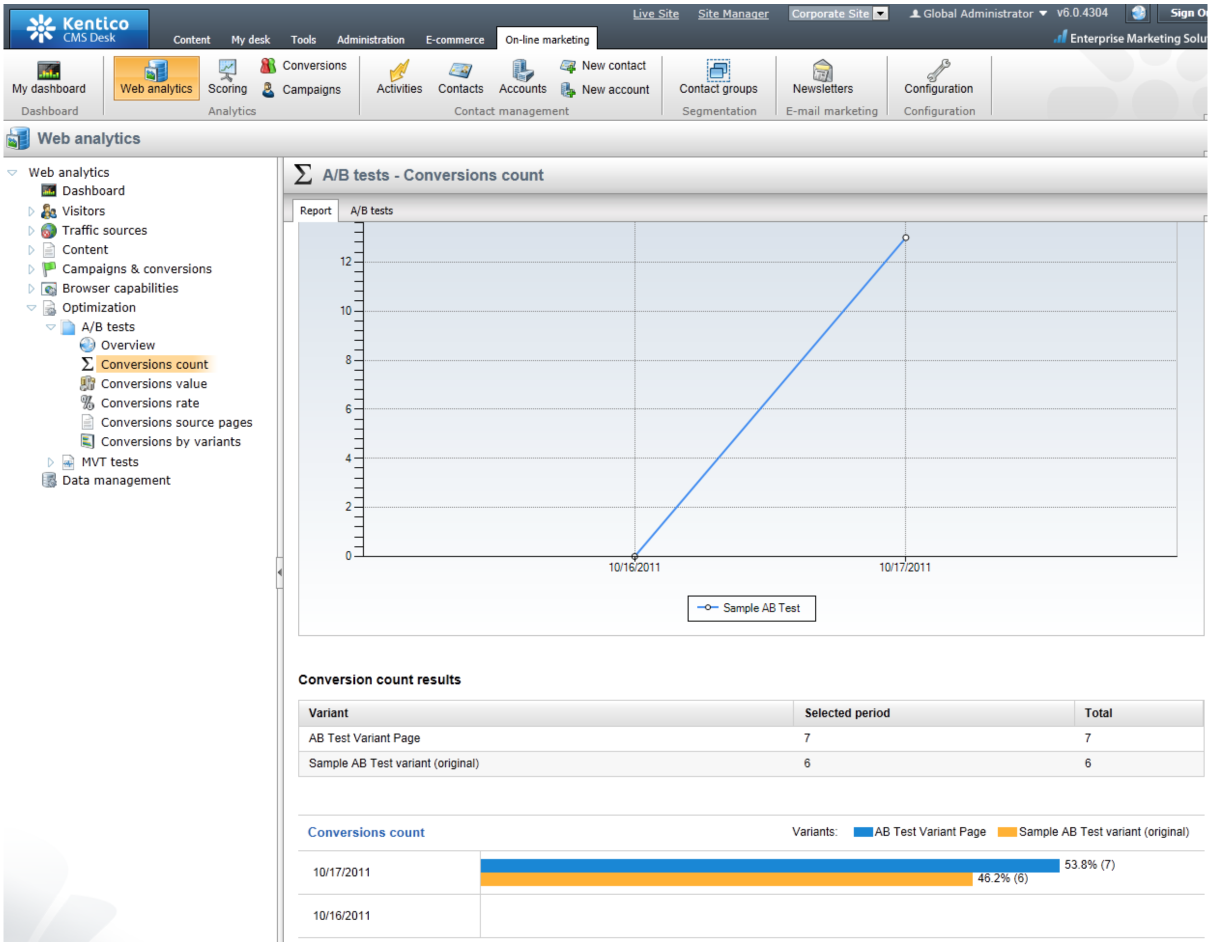


You can also make any necessary changes to the text content on the **Page** tab.

The variant will now serve as a third possible option when the tested page is viewed. If you wish, you can use the approach described in the first part of this example to try out the functionality of the A/B test now that it has an additional page variant.

9.2.4 Analyzing A/B test results

The data gathered during the course of A/B testing may be viewed in reports that display various types of conversion metrics. To access the main interface containing these reports, go to **CMS Desk -> On-line marketing -> Web analytics** and expand the **Optimization -> A/B tests** category.



Here you can check the results of the A/B testing actions performed according to the instructions in the Example: Sample A/B test topic. Because of the way the analytics log is processed, you may have to wait approximately one minute before conversion hits are reflected in the reports.

A/B testing reports use the same basic format as all other web analytics reports, so please refer to Web analytics -> Using web analytics reports for details about the actions that can be used to filter or otherwise manage the displayed data.

The **Conversions** drop-down list may be used to select which conversion's statistics should be displayed. Please note that the system logs *all* conversion hits generated by visitors who have passed through a page where an A/B test is running. If there are many conversions defined on your website, only those that can somehow be affected by the differences in the A/B test's page variants will contain relevant data.

Data in the reports is represented by two possible types of graphs. The line charts show the progress of a certain conversion statistic over time and display combined data for all of the A/B test's variants. The bar graphs contain details for individual units of time according to the selected report type (hours, days, months etc.). You may also view the conversion data in a table located between the graphs. Each row in the table shows the data logged for a specific page variant, both for the time period currently displayed by the report and the entire duration of the test.

The following reports are available for A/B testing:

Overview	This report can be used to view the progress of the primary metrics measured for the site's A/B tests from a single location.
Conversions count	Displays the number of conversion hits logged for the selected A/B test during the specified time interval. In the bar graph, the number of conversion hits is divided into categories that represent individual page variants. This allows you to compare the A/B test's variants and determine which one generated the most conversions (in absolute terms).
Conversions value	Displays the sum of the conversion values logged for the selected A/B test during the specified time interval. In the bar graph, the conversion values are divided into categories that represent individual page variants, which allows you to determine which variant generated the highest total conversion value. This way you can easily evaluate an A/B test's variants when using weighted conversions that have a different level of importance.
Conversions rate	Used to indicate how many visitors who access the tested page perform a conversion. The conversion rate is calculated as the amount of logged conversion hits divided by the total number of visitors on the variants of the tested page. If you select the *(all)* option from the **Conversions** drop-down list, then the rate will be measured for all possible conversions, i.e. as the percentage of visitors who generated at least one conversion hit of any type. The conversion rate in the bar graph is displayed for individual page variants. This allows you to compare the A/B test's variants and determine which one encouraged the highest share of its visitors to perform a conversion.
Conversions source pages	Displays hit statistics for individual conversions that were logged as part of the selected A/B test during the specified time interval. The data logged for the chosen conversion is categorized according to the page variants defined for the given A/B test. This allows you to easily determine which variant generated the most conversion hits of the selected type.
Conversions by variants	Displays details of the number of hits logged for each conversion by individual page variants defined for the selected A/B test. You can select the variant that you wish to evaluate from the **Variants** drop-down list. The hits logged for the chosen variant are divided into categories that match individual conversions. This allows you to easily measure which conversions are performed most commonly by visitors assigned to the selected page variant.

Reports for individual A/B tests

These reports can also be viewed when editing the original page of an A/B test in **CMS Desk -> Content -> Edit -> Analytics -> A/B testing -> Reports**.

The same options are available as described for the web analytics interface, but statistics are only displayed for the currently edited A/B test.

9.3 Multivariate testing

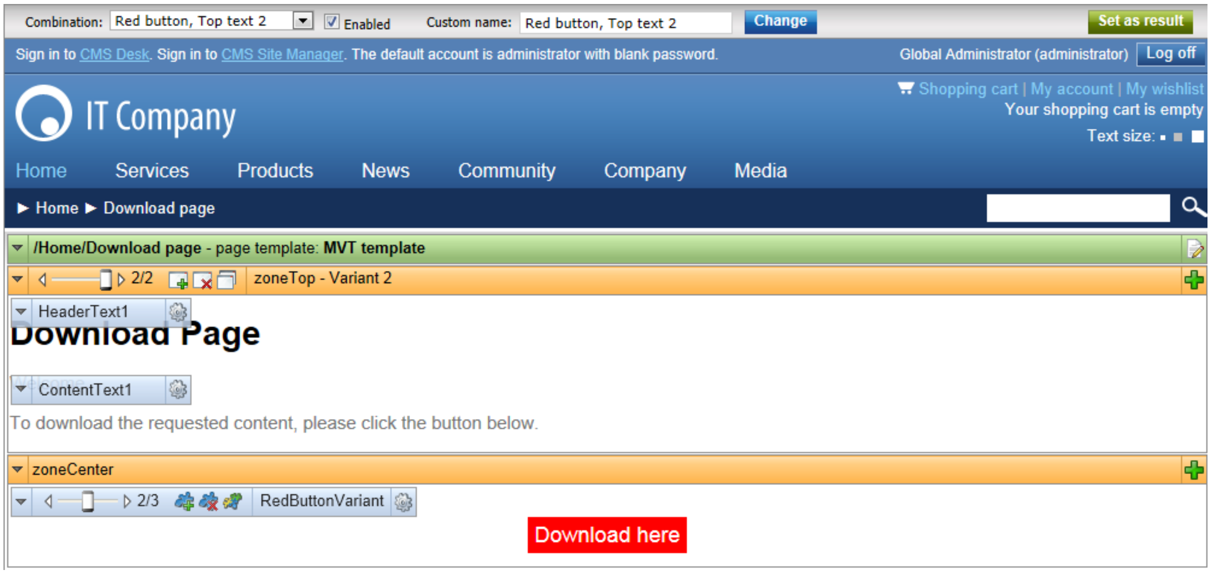
9.3.1 Multivariate testing overview

Multivariate testing (MVT) provides a way to optimize a website's pages based on the behaviour of its visitors. It allows you to define any number of elements on a page as variables and create different versions for each one. Once the test is started, users who view the page will see one of the possible versions of its content and their subsequent activity will then be tracked to determine which modifications produce the best results.

The basic objects used to manage this functionality are ***MVT tests***, which can be created for specific documents (pages) on the website. You can implement the changes that you wish to test by creating ***Variants*** of the standard design objects that make up the content of the given page. This includes Web parts, entire web part zones and Widgets in the page's editor zones.

Testing is not done for individual variants, but rather for ***Combinations*** of the variants on the page. Because scenarios with multiple object variables can lead to a very large amount of possible combinations, you can limit the scope of the test by disabling those that you are not considering as options. The default page is also available as a combination, so you can compare potential improvements with the baseline statistics (those of the page with its original, unmodified content).

The image below shows how a page with a defined multivariate test could look in the design interface. There are two different versions of the top zone's content and three possible variants of the web part that displays the download button in the bottom zone. That makes six total combinations of the page's content which can be included in the MVT test.



To learn how you can define an MVT test for a page and create different testing variants of its content, please see the Managing MVT tests topic. If you wish to try out a live multivariate testing scenario, you can do so on the sample Corporate Site according to the instructions in the Example: Sample MVT test topic. The data gathered by MVT tests can be viewed in various reports that allow you to analyze several types of conversion metrics, as described in Analyzing MVT test results.

With multivariate testing, results are tracked for specific combinations of the variants defined on a single page, which represent individual modifications. If you wish to monitor the aggregate effect of all changes made to an entire page as a single variable, you may use A/B testing, which is another optimization feature provided by Kentico CMS.

How it works

When a visitor navigates to a page that has a running MVT test on the live site, one of the enabled content combinations will be displayed. The combination is chosen randomly for every user. With a large enough visit sample size, each active combination should receive roughly the same amount of traffic during the course of the test.

A persistent cookie is stored in the visitor's browser, used to identify which combination was assigned to the user by the given MVT test. The name of the cookie uses the following format: *CMSMVT<MVT test code name>*. It saves the internal code name of the assigned combination as its value. This cookie expires either within 30 days after the last visit on the tested page, or on the date when the test is configured to end.

Any conversions performed on the website by users who have passed through a page where an MVT test is running will be logged for the given test under the assigned combination, which is taken from the value of the MVT testing cookie. The logging of conversion hits is provided by the Web analytics module. In addition to monitoring conversions, the cookie also ensures that returning visitors are always shown the same content combination that was previously assigned to them, which helps avoid confusion by maintaining a consistent appearance of the tested page.

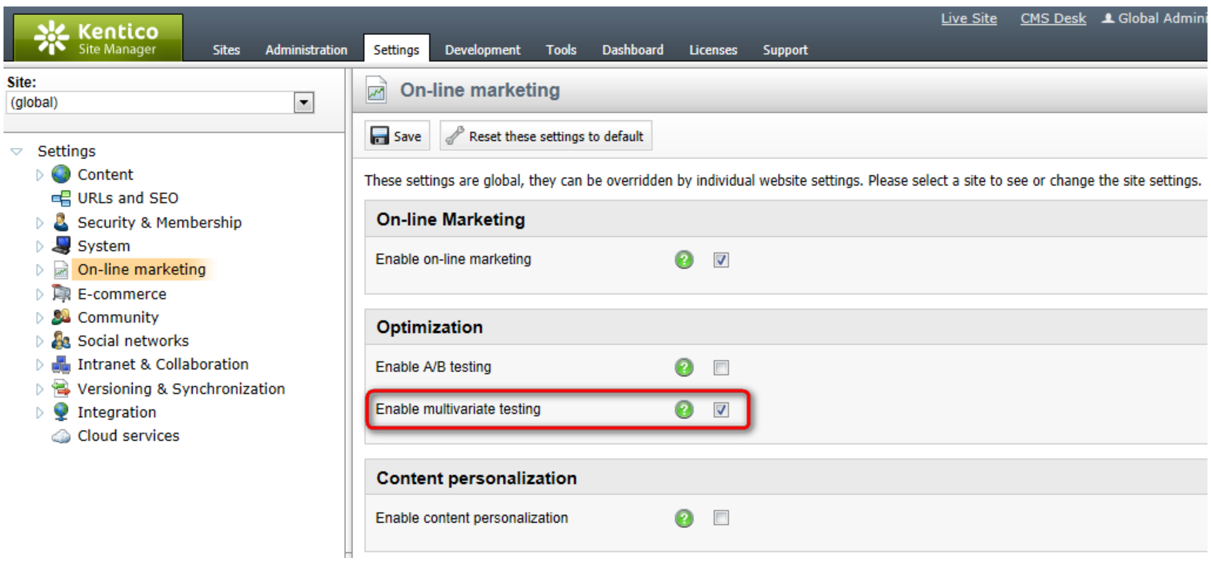
It is possible to run multiple MVT tests concurrently for different pages on the same website. Conversions will be logged for all tests defined on the pages visited by a given user, according to the cookies present in the browser.

9.3.2 Managing MVT tests

The sections below describe the general steps that need to be taken in order to successfully set up a multivariate test on your website.

Enabling multivariate testing

Before you can start creating MVT tests for pages, it is first necessary to enable the module by going to **Site Manager -> Settings -> On-line marketing** and checking the **Enable multivariate testing** field.



Since web analytics are used to log conversion statistics for individual versions of tested pages, the **Enable web analytics** and **Track conversions** fields must additionally be enabled in the **On-line marketing -> Web analytics** sub-category of the settings. All other requirements for the correct functioning of the web analytics module also need to be fulfilled.

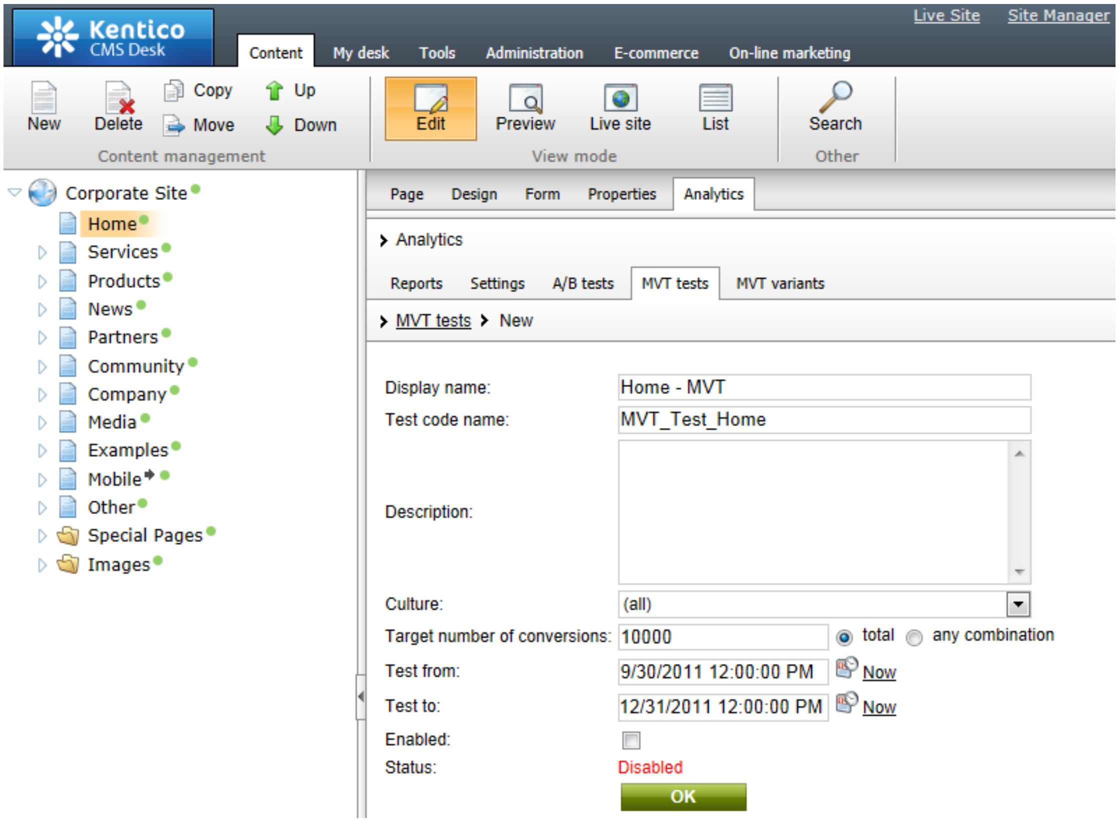
Defining an MVT test

To start optimizing a page on your website via multivariate testing, create and configure an *MVT test* object for it. This can be done in the **Content -> Edit** mode of **CMS Desk** by selecting the appropriate document from the content tree and switching to its **Analytics -> MVT tests** tab. Here you can find a list of all MVT tests assigned to the current document and manage them as required. The **New MVT test** link may be used to add an MVT test.

When creating or editing MVT tests, the properties described in the table below are available:

Display name	The name of the MVT test displayed in the administration interface.
Test code name	Sets a code name that serves as an identifier for the test. It is also used in the name of the browser cookie used to store which of the test's variant combinations was assigned to a visitor.

Description	Can be used to enter a text description for the test in order to give information about its purpose, goals etc.
Culture	Used to select which culture version(s) of the document should be included in the test.
Target number of conversions	Sets the number of conversion hits that must be logged to complete the test. Once this number is reached, the MVT test will automatically stop working and switch to the **Finished** status. If the **total** option is selected, then the number will be compared with the total amount of conversion hits logged for all of the test's combinations. If **any combination** is chosen, then the test will be concluded when the specified number is reached by one combination (the one with the most conversion hits). Leaving this property empty or setting it to *0* means that there will be no conversion hit limit for the test.
Test from/to	Sets the time interval during which the test should occur. The **Calendar** button () can be used to select the exact date and time.
Enabled	This property may be used to manually start or stop the test.
Status	Displays the current status of the MVT test. The following statuses are possible: • **Disabled** - indicates that the test is not active. The default content of the tested page will always be displayed to visitors and conversions will not be logged for the test's combinations. • **Not running** - indicates that the test is not active. Used when the test is enabled, but the *Test from* date is still in the future. • **Running** - indicates that the test is currently active. • **Finished** - this status is automatically assigned after the *Test to* date passes or when the *Current number of conversions* reaches the *Target number of conversions*. Tests with this status are no longer active. Each page may have only one MVT test running at a given time. It is however possible to have multiple finished or disabled tests assigned to a page, which can be used to archive data from previous tests or when preparing future tests. Different culture versions of a page may each have a different test running at the same time.



Managing MVT tests through the Web analytics interface

If you wish to manage all tests assigned to different pages from a single location, go to **CMS Desk -> On-line marketing (or Tools) -> Web analytics**. Then expand the **Optimization -> MVT tests** category, select one of the contained items (reports) and switch to the **MVT tests** tab.

Here, all MVT tests defined on the current website may be managed in the same way as described for individual documents. The only difference is that the **Page** property is additionally available when editing a test, which can be used to assign tests to specific pages.

Creating testing variants on the page

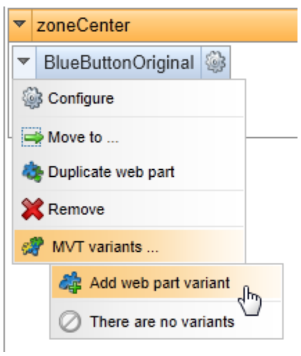
Once a page contains an MVT test, you can start creating the content options that you wish to evaluate. This is done by defining variants for the elements that make up the content of the page. It is possible to use the following objects as variables:

- **Web parts** - these variants are separate instances of the original web part. Each variant's properties may be configured differently and an alternative Web part layout can be specified.
- **Web part zones** - in this case, variants are added as entire web part zones. A zone variant may contain any type or number of child web parts as required. The basic properties of the zone may also be set differently. When a new variant is added to a zone, the content of the original is automatically copied into it, so you do not have to rebuild the zone from scratch if you only need to make small

modifications. Please note that creating variants for individual web parts inside zone variants is not supported.
- **Editor widgets** - like with web parts, each variant is a widget of the same type as the original that provides the option to set different values for its properties.

If you are not familiar with the basics of page development in Kentico CMS, it is recommended to read through the Development -> Web parts and Widgets chapters in the Kentico CMS Developer's Guide before you continue.

To add an MVT variant to a web part or zone, open the page on the **Design** tab, expand the context menu of the given object by right clicking its header (or through the ▾ icon), hover over the **MVT variants** option and select **Add <object type> variant** from the second level of the menu.

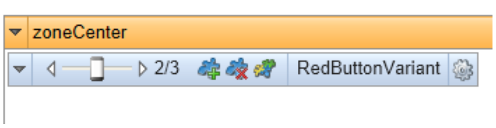


Then, fill in the following properties in the displayed dialog:

- **Display name** - the name used in lists of MVT variants in the administration interface.
- **Code name** - sets a code name that serves as an identifier of the variant.
- **Description** - can be used to enter a text description of the variant.
- **Enabled** - indicates if the variant should be considered as a possible testing option. If you disable an MVT variant, all testing combinations that include this variant will also be disabled.

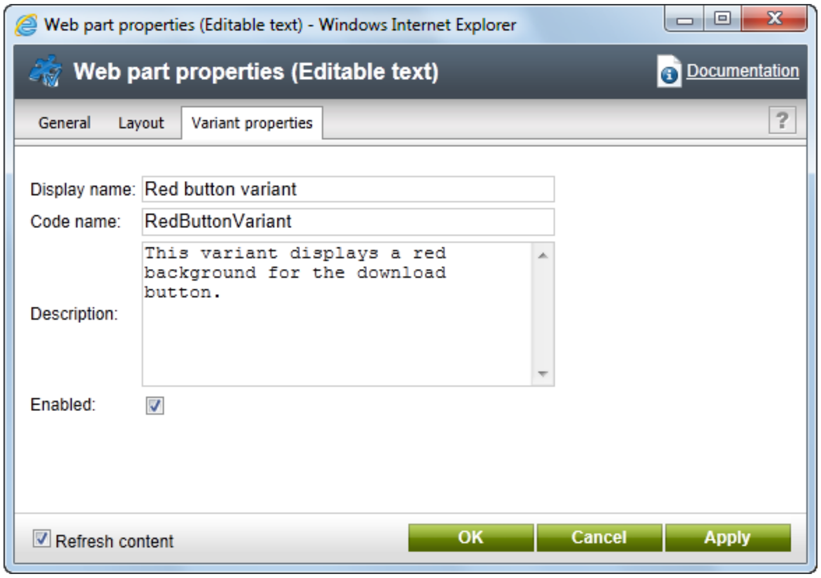
After you enter and confirm these basic values for the new variant, a configuration dialog will be opened, just like when creating a standard web part or widget. Here, you can set up the available properties (according to the type of the given object) so that the variant generates the content that you wish to test. By default, the values set for the original will be loaded into the properties, so you only have to change the parts of the configuration that are unique for the given variant.

Each object for which testing variants are created will have a slider available in its header as shown below:



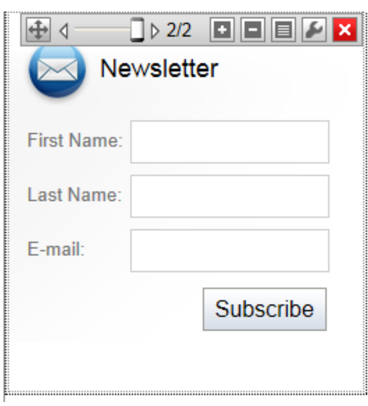
This slider can be used to switch between individual variants as needed (including the original). The content specified by the current selection on the page's variant sliders will be displayed in CMS Desk on the **Design** and **Page** tabs, and in **Preview** mode. You can also switch between different versions of the page's content by using the combination panel, which is described below in the next section. If you wish to view the content of variants while cycling through the slider on the **Design** tab, make sure that the **Display web part content** checkbox on the right side of the **Edit** mode header is checked.

The buttons on the right of the slider allow you to add new variants ( ), remove the currently selected one ( ) or open a list ( ) of all variants that the given object currently has. At any time, you can **Configure** ( ) the properties of the variant currently chosen on the slider.



When editing a variant like this, you can access the properties that affect the content on the **General** tab. The specific settings related to the MVT testing functionality of the specific variant (the same options that were offered when creating a new variant) can be changed on the additional **Variant properties** tab.

In the case of editor widgets, multivariate testing variants are handled using a very similar approach. The only difference is that editing is done on the **Page** tab of **CMS Desk** and the slider and actions for variant management are located on the pop-up menu of individual widgets.



You can access a list of all multivariate testing variants defined on a given document (page), by selecting it from the content tree in **CMS Desk -> Content -> Edit** and going to **Analytics -> MVT variants**. The variants of all three object types are included here, and they can be removed or edited as necessary.

There are some general rules that apply to all MVT variants:

- If you delete an object from the page, any variants that it might have will be removed along with it.
- Variants are not linked to a specific MVT test. Instead, they are either stored on the page template

used by the given document (in the case of web parts and zones) or bound to the document itself (editor widget variants). This means that existing variants may be used by other MVT tests performed on the given page at another time. Also keep this in mind if you wish to Export a multivariate testing scenario to another website. Variants will be transferred along with documents or page templates, not MVT test objects.

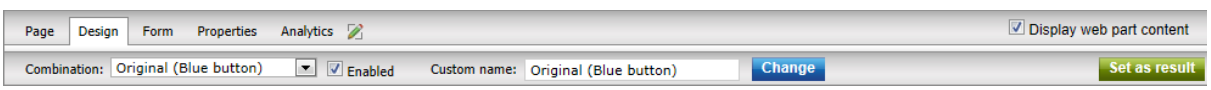
Multivariate testing and Content personalization

Please note that it is not possible to create multivariate testing variants of web parts, zones or widgets that already have variants defined for Content personalization.

Setting up testing combinations

Individual testing scenarios are represented by combinations of the MVT variants created on the page. By default, all possible combinations of the page's content will be included in a multivariate test, but you may perform some additional configuration to fine-tune your test as required.

Combinations can be managed for documents that have an MVT test defined via a panel located at the top of the page editing interface in CMS Desk. This panel is available on both the **Page** and **Design** tabs, and in **Preview** mode.



You can choose any of the available combinations through the **Combination** selector. Doing so will cause the page to display the content defined by the variants that make up the given combination. The selection made through the combination panel is linked with the positions set on the MVT sliders of variable objects on the page. If you switch to a different variant through an object's slider, the current combination will also change accordingly and vice versa.

Pages where there are several variants for multiple objects will have a large amount of possible combinations. For this reason, it is recommended to carefully choose which combinations should be included in the page's MVT test. You can include or exclude a combination by toggling the **Enabled** checkbox located on the panel next to the selector. There is a close relationship between the status of a combination and its variants. If an individual variant is disabled through its properties, all combinations that include it will also be disabled. If a disabled combination is enabled at a later point, all of its variants will be enabled automatically.

A good way to evaluate potential combinations is to check them in the **Preview** mode of CMS Desk, and disable those where the variants clash visually, have conflicting logic or are otherwise incompatible. We recommend leaving the *Default page* combination enabled, so you can compare any improvements with the baseline statistics of the original page.

Another best practice is to set a descriptive custom name for each active combination, so that you can easily differentiate between combinations in the editing interface and identify them in the test's result reports. To do this, simply select a combination through the panel, enter the appropriate text into the **Custom name** field and click the **Change** button. The default names assigned to new combinations are composed of a number indicating their order, followed by a list of the display names of all variants that are included in the given combination.

Once the page's variants and combinations are all configured as needed, you can enable your MVT test on the **Analytics -> MVT tests** tab of the document. Making further modifications on the page while an MVT test is running is not recommended, since this may affect the accuracy and relevance of the measured statistics. A warning will be displayed below the combination panel on pages where a test is running.

The **Set as result** action should only be used after the conclusion of the test, once you have analyzed the logged conversion data and identified the combination that provides the best results. The action allows you to easily set the winning combination as the permanent content of the page. Simply select the given combination via the combination panel, then click the button and confirm the action. This will replace the original web parts, zones and widgets with the variants included in the currently selected combination and remove all other MVT variants from the page.

Configuring conversions

Conversions allow you to track the behaviour and actions of the website's visitors, so they must be used in order to get MVT testing results. They work on a site-wide level and do not need to be assigned to individual MVT tests in any way. All that needs to be done is define conversion objects and link them with the actions that you wish to log for the test. There are many ways to specify that an action should be logged as a conversion. You can find information about conversion management in the Conversions chapter of this guide.

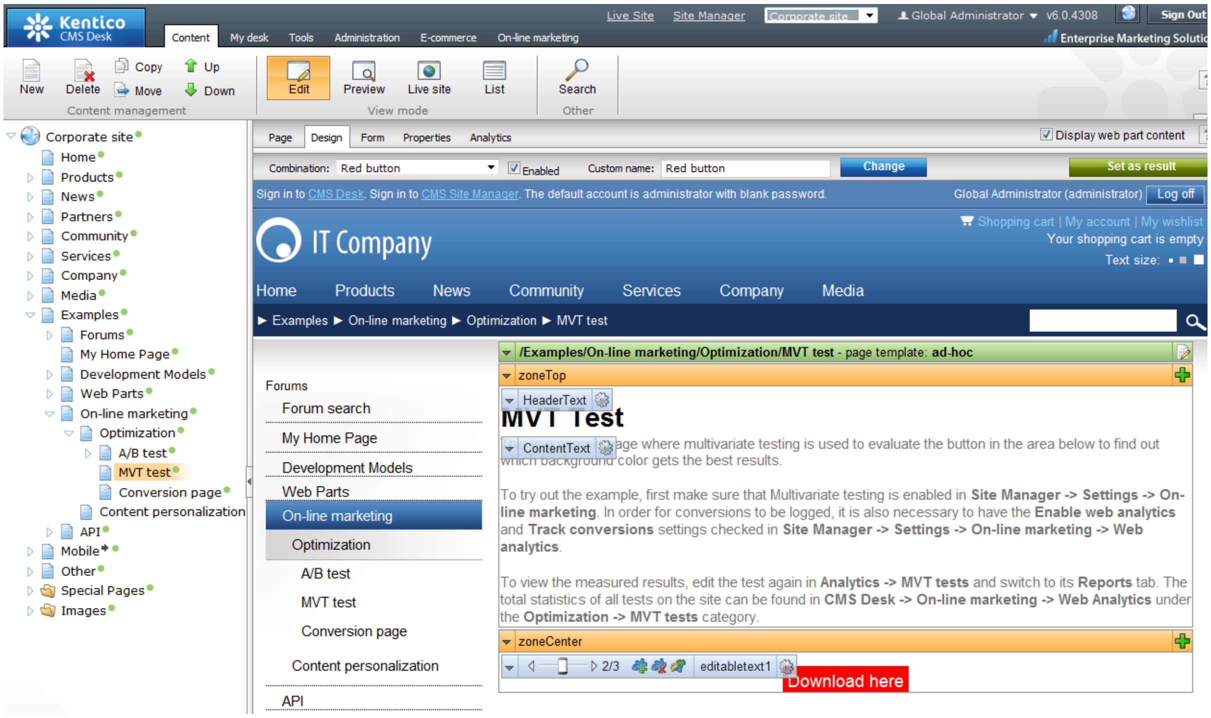
When an action designated as a conversion is performed anywhere on the website, the system checks if the given user has passed through a page with a running MVT test (according to the presence of a cookie). If this is the case, the conversion hit will be logged for the content combination that was assigned to the user.

Once you enable the MVT test and it starts running on the live site, the conversion statistics for individual combinations can be monitored using pre-defined reports, as described in the Analyzing MVT test results topic.

9.3.3 Example: Sample MVT test

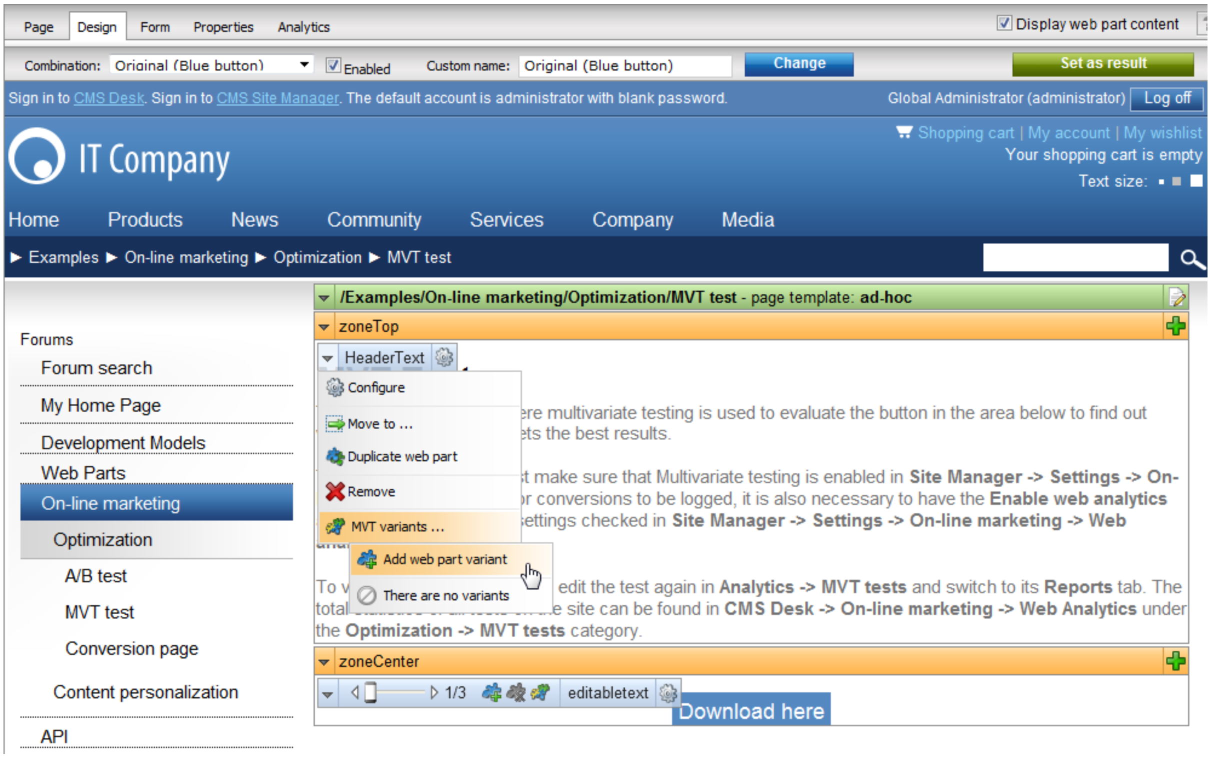
This example demonstrates the functionality of the sample multivariate test included on the Corporate Site and shows how you can expand it by adding another testing variant.

1. Go to **CMS Desk -> Content -> Edit** and select the **MVT test** document, which can be found under **Examples -> On-line marketing -> Optimization**. This document already has a pre-defined MVT test. The page is composed of a header, content text and a link at the bottom. If you switch to the **Design** tab, you can see that there are three versions of the Editable text web part that displays the link (the original and two MVT variants). Each one uses a different background color. The goal of the test is to find out which variant gets more link clicks from visitors.



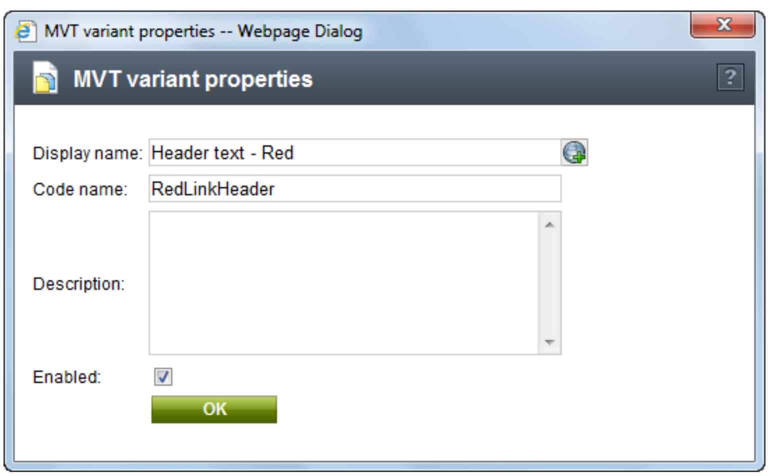
Try switching between the content of individual variants using the slider in the web part's header. Also notice the combination panel at the top of the page. It shows which combination of variants is currently being viewed and may also be used to switch between individual combinations or configure their behavior. Since the default test page only contains variants of a single web part, each "combination" represents the same page with a different link color.

2. To better demonstrate how MVT combinations work, we will need to add a variant to another element on the page. Right click the **HeaderText** web part, hover over the **MVT variants** option in the context menu and select **Add web part variant** from the second level.



3. Enter the following values for the variant's properties in the displayed dialog:

- **Display name**: Header text - Red
- **Code name**: RedLinkHeader
- **Enabled**: Yes (leave it checked)

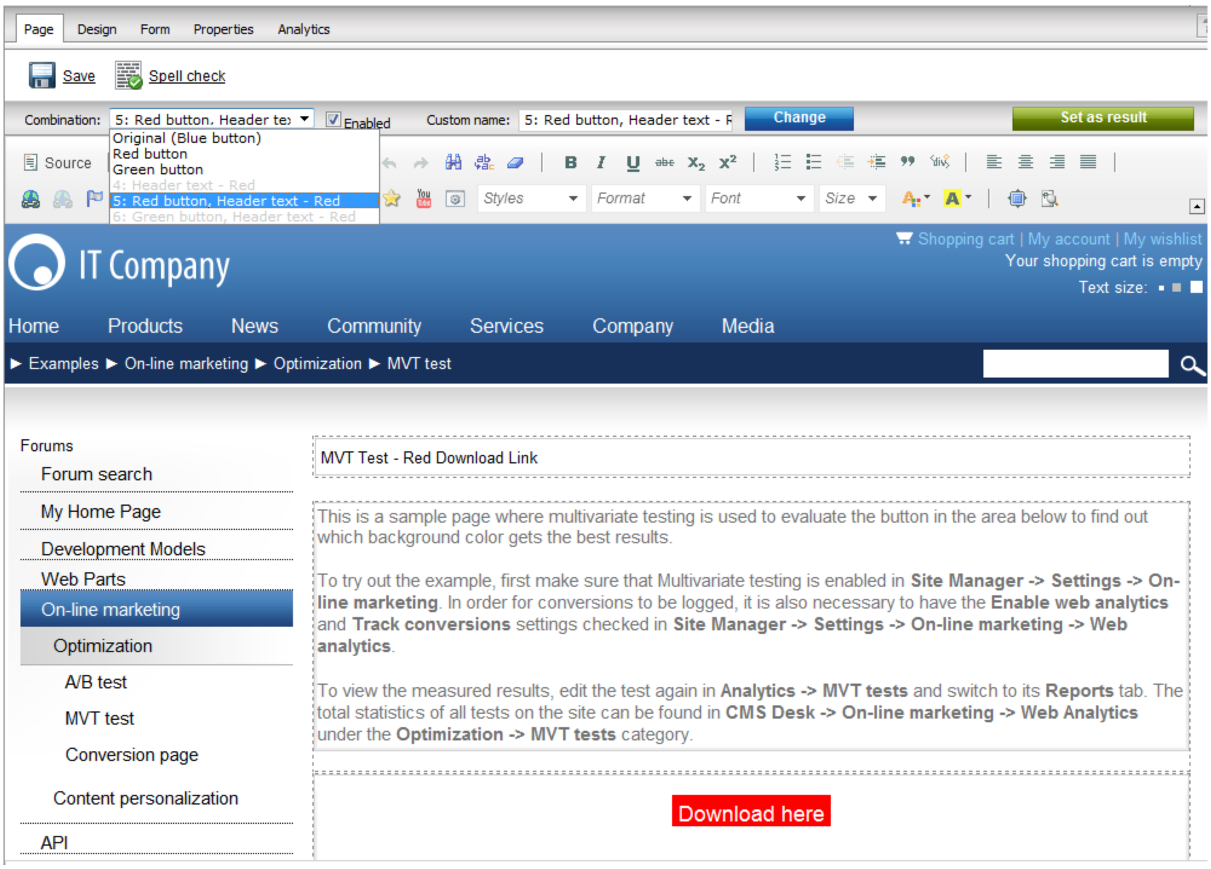


Click **OK**.

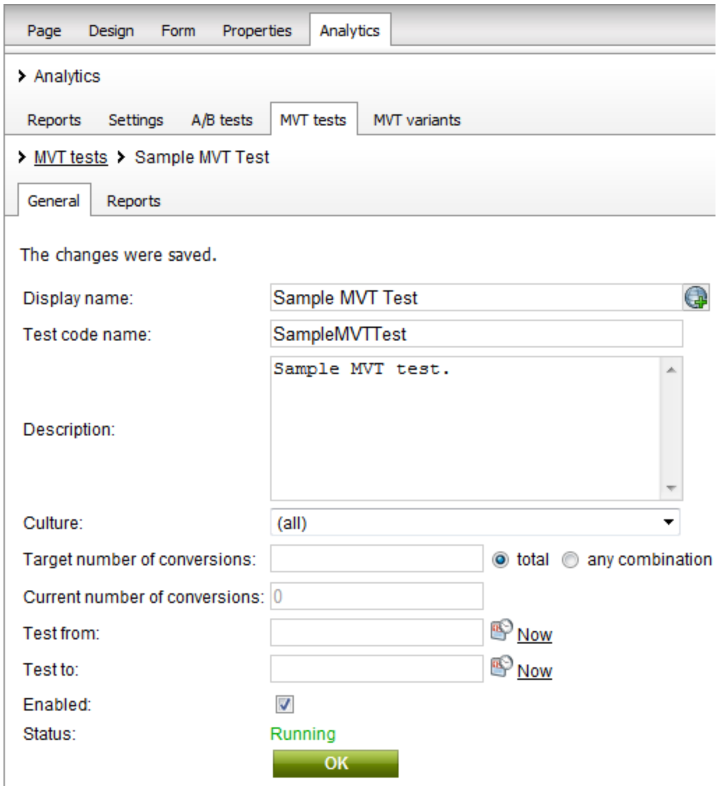
4. A configuration dialog will appear for the Editable text web part used to display the page's header. Enter *Red link header variant* into the **Web part title** property. Setting an appropriate title is recommended, since this allows you to identify which variant you are working with on the Design tab straight from the description in the web part header. You can then confirm the dialog by clicking **OK** again and the variant will be added to the page.

5. Now that there are two different variants of the page's header text, the number of possible combinations will double to a total of six. Switch to the **Page** tab and make sure that one of the new combinations that contains the *Header text - Red* variant is selected. The editable region at the top of the page will be empty, so type in the following text: *MVT Test - Red Download Link*. Click **Save**.

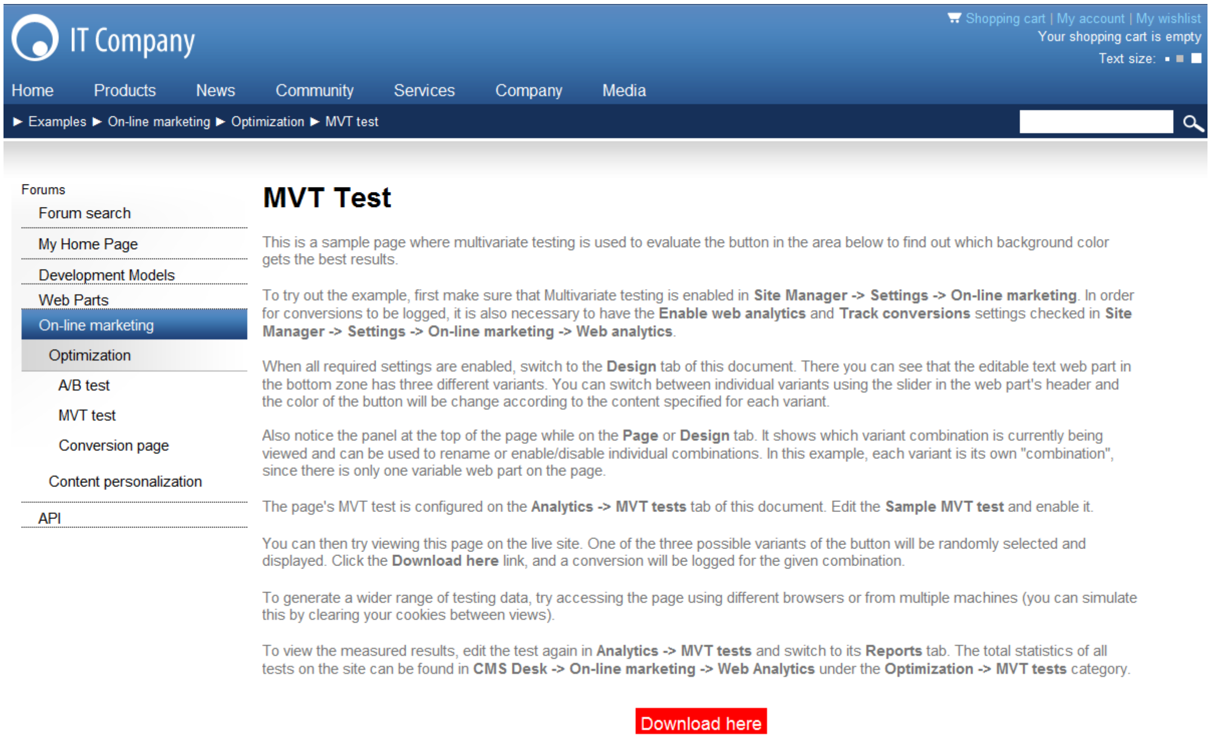
6. In this sample testing scenario, the second variant of the page's header is only intended for the red version of the download link, so we will disable the combinations where the header text does not match the link color. Select the fourth combination using the panel and uncheck the **Enabled** box next to the drop-down list. Repeat this for the sixth combination. This ensures that versions of the page where the new "red" header is combined with a blue or green link will not be included in the MVT test, which leaves only four active combinations: all of the variants with the original header and a new combination that contains the red link together with the second header.



7. The testing combinations are now prepared and you can start the MVT test. Switch to the **Analytics -> MVT tests** tab of the document, edit () the **Sample MVT Test**, check the **Enabled** box on the **General** tab and click **OK**.



8. Now try viewing the page on the live site. To access it, sign out of CMS Desk, open the **Home** page of the Corporate Site, scroll down to the bottom and click the **Examples** link under the **Where to learn more?** section. Then, use the menu on the left to navigate to **On-line marketing -> Optimization -> MVT test**. When you view the tested page, one of the possible combinations will be randomly selected and displayed.



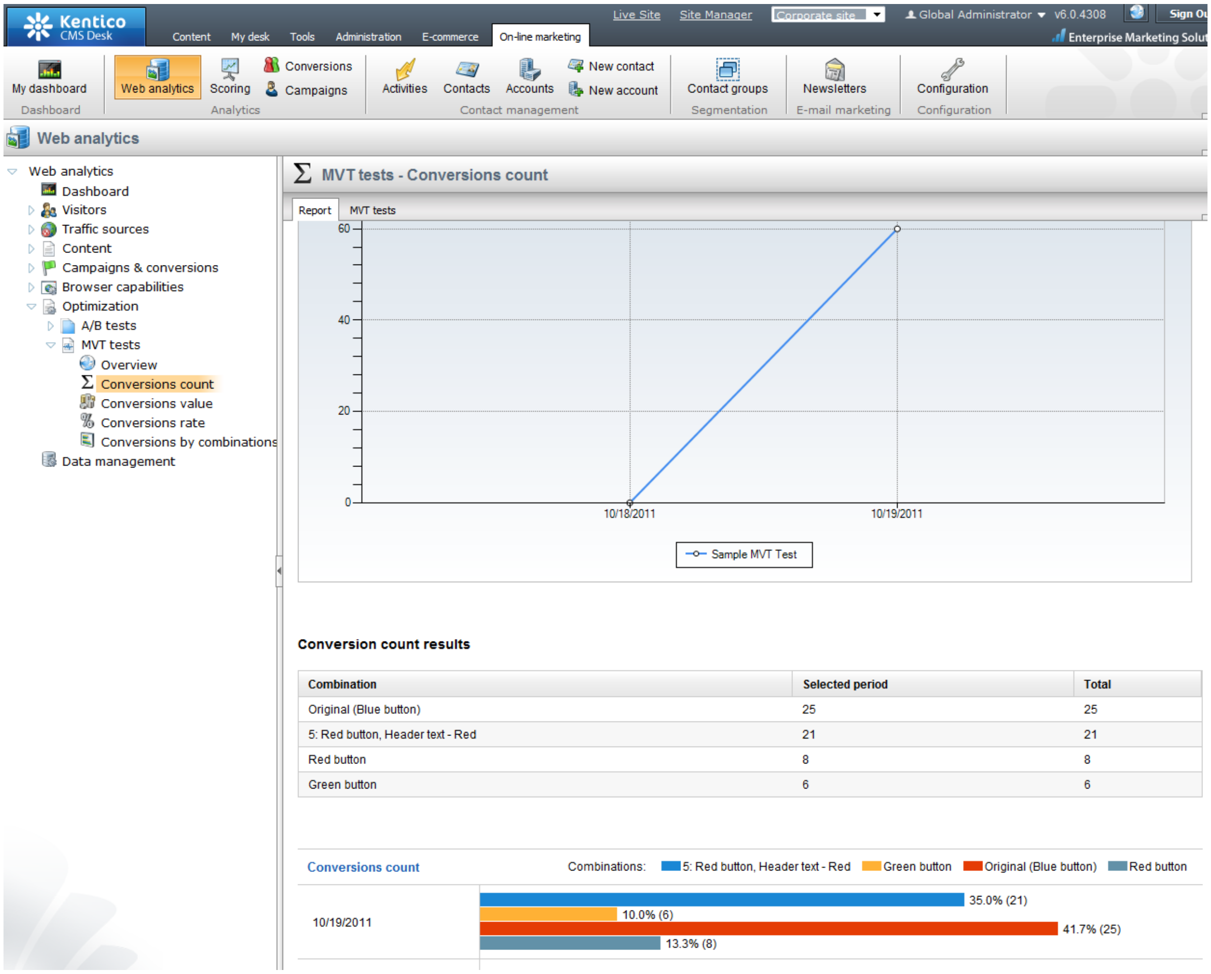
9. Click on the **Download here** link, and a conversion will be logged for the given combination by the sample **Conversion page**. If you return to the MVT test page, the same combination that was assigned during the first visit will always be displayed.

10. Now open the same page in a different browser. Again, one of the combinations will automatically be chosen and assigned. Please note that it may be the same combination as before, since the selection process is entirely random. You can simulate visits from multiple users by clearing the cookies from your browser between views of the tested page. Try to use this approach to log several conversions for each combination.

Please continue reading in the Analyzing MVT test results topic to see how you can view the results measured for the sample test performed in the previous steps.

9.3.4 Analyzing MVT test results

The data gathered during the course of MVT testing may be viewed in reports that display various types of conversion metrics. To access the main interface containing these reports, go to **CMS Desk -> On-line marketing -> Web analytics** and expand the **Optimization -> MVT tests** category.



Here you can check the results of the MVT testing actions performed according to the instructions in the Example: Sample MVT test topic. Because of the way the analytics log is processed, you may have to wait approximately one minute before conversion hits are reflected in the reports.

MVT testing reports use the same basic format as all other web analytics reports, so please refer to Web analytics -> Using web analytics reports for details about the actions that can be used to filter or otherwise manage the displayed data.

The **Conversions** drop-down list may be used to select which conversion's statistics should be displayed. Please note that the system logs *all* conversion hits generated by visitors who have passed through a page where an MVT test is running. If there are many conversions defined on your website, only those that can somehow be affected by the differences between the tested content combinations will have relevant data.

Data in the reports is represented by two possible types of graphs. The line charts show the progress of a certain conversion statistic over time and display combined data for all of the tested combinations. The bar graphs contain details for individual units of time according to the selected report type (hours, days, months etc.). You may also view the conversion data in a table located between the graphs. Each row in the table shows the data logged for a specific combination, both for the time period currently displayed by the report and the entire duration of the test.

The following reports are available for MVT testing:

Overview	This report can be used to view the progress of the primary metrics measured for the site's MVT tests from a single location.
Conversions count	Displays the number of conversion hits logged for the selected MVT test during the specified time interval. In the bar graph, the number of conversion hits is divided into categories that represent individual content combinations. This allows you to compare the tested combinations and determine which one generated the most conversions (in absolute terms).
Conversions value	Displays the sum of the conversion values logged for the selected MVT test during the specified time interval. In the bar graph, the conversion values are divided into categories that represent individual content combinations, which allows you to determine which one generated the highest total conversion value. This way you can easily evaluate the results of the MVT test when using weighted conversions that have a different level of importance.
Conversions rate	Used to indicate how many visitors who access the tested page perform a conversion. The conversion rate is calculated as the amount of logged conversion hits divided by the total number of visitors on the given page. If you select the *(all)* option from the **Conversions** drop-down list, then the rate will be measured for all possible conversions, i.e. as the percentage of visitors who generated at least one conversion hit of any type. The conversion rate in the bar graph is displayed for individual content combinations. This allows you to compare the tested combinations and determine which one encouraged the highest share of its visitors to perform a conversion.
Conversions by combinations	Displays details about the number of conversion hits logged for individual content combinations defined on the page associated with the selected MVT test. You can select the combination that you wish to evaluate from the **Combinations** drop-down list. If the MVT test is used for multiple culture versions of the page, you can also specify the culture. The hits logged for the chosen combination are divided into categories that match individual conversions. This allows you to easily measure which conversions are performed most commonly by visitors assigned to the selected content combination.

Reports for individual MVT tests

These reports can also be viewed when working with the tested page in **CMS Desk ->**

Content -> Edit by editing the given MVT test in **Analytics -> MVT tests** and switching to its **Reports** tab.

The same options are available as described for the web analytics interface, but statistics are only displayed for the currently edited MVT test.

Part X

Content personalization

10 Content personalization

10.1 Content personalization overview

Content personalization is an on-line marketing feature that can significantly increase the flexibility of your website. Personalization allows you to create pages that display different content depending on the circumstances in which they are viewed. For example, you can custom-build pages that offer special content for different types of visitors or dynamically change according to previous actions performed on the website by users.

Personalization is applied through the basic components that form the content of pages, which includes:

- Web parts
- Entire web part zones
- Widgets added into page editor zones

The first necessary step is to define different versions of these objects called *variants*. The details of how personalization variants can be created and configured for objects are described in Managing personalization variants.

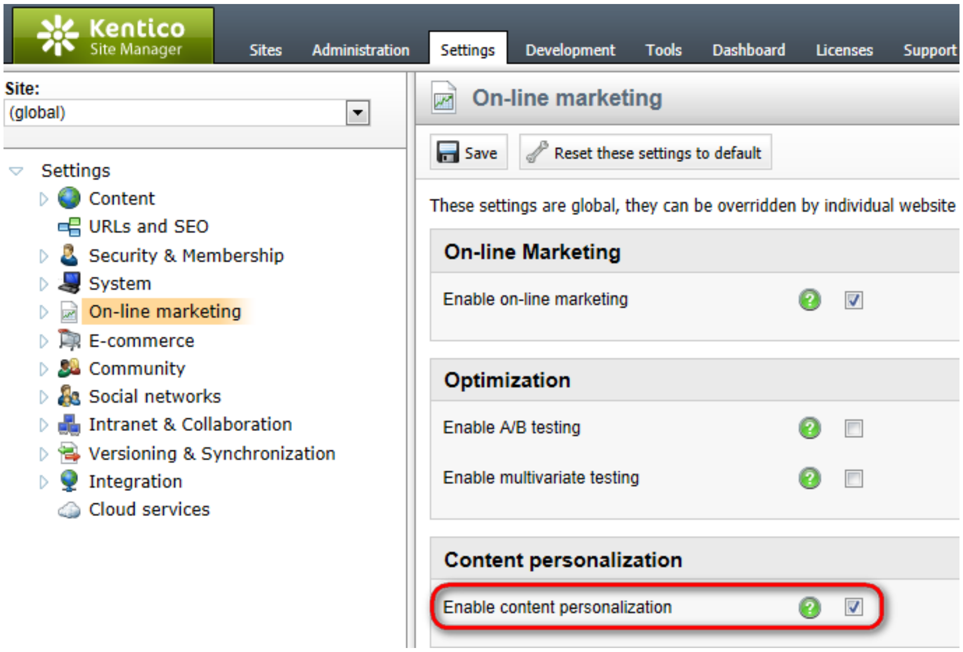
Each variant is dedicated to a particular scenario, which is specified through a macro-based condition. The condition determines in which situations the variant should be displayed. When a visitor selects the page on the live site, the variants of all personalized objects will have their conditions dynamically resolved according to the current context and the appropriate variant will be displayed for each object.

Some of the most effective content personalization can be achieved by using the data provided by other on-line marketing features, such as Contact activities, Contact scoring or Campaigns. You can find an example of this type of scenario in the Example: Creating a personalized page topic, which demonstrates how to perform the personalization process step-by-step.

10.2 Managing personalization variants

To start using content personalization on your website:

1. Go to **Site Manager -> Settings -> On-line marketing**.
2. Select your **Site** (or enable content personalization globally).
3. Check **Enable content personalization**.
4. Click **Save**.



Users with the appropriate [permissions] can now create personalization variants of page components. When visitors view personalized pages on the live website, the system processes the variants and displays the corresponding content.

If you disable content personalization at a later time:

- Existing personalization variants on your website will not be deleted. However, the system does not process the conditions of variants on personalized pages and always displays the default content on the live website.
- Users will no longer be able to manage personalization variants or define new ones.

Managing variants

Once content personalization is enabled, you can create variants of page components through the **CMS Desk -> Content -> Edit** interface. There are three types of components that you can personalize:

- Web parts
- Web part zones
- Editor widgets

Multivariate testing and Content personalization

It is *not* possible to create personalization variants of web parts, zones or widgets that already have included in a Multivariate test.

Variant overview

To access a list of all content personalization variants defined on a given page:

1. Select the document in the content tree in **CMS Desk -> Content -> Edit**.
2. Open the document's **Properties -> Variants** tab.

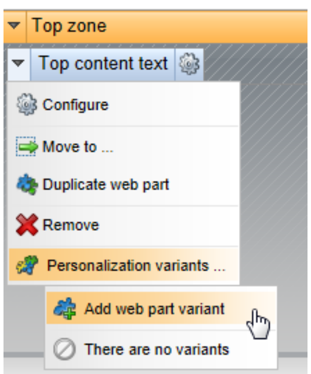
The list includes the variants of all three components types.

Personalizing web parts

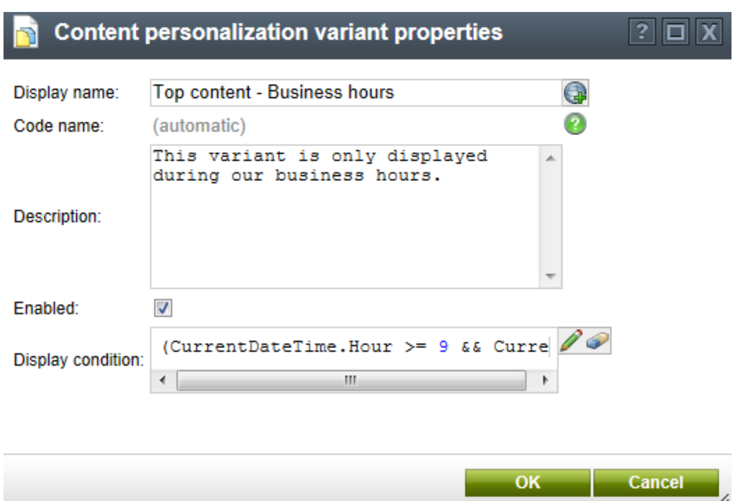
Content personalization of web parts allows you to create different versions of web parts with different property settings. To add personalization variants for web parts:

1. Edit the page containing the web part in CMS Desk on the **Design** tab.

2. Right-click the header of the web part (or click the icon), hover over **Personalization variants** and click **Add web part variant**.



3. Fill in the display condition and other settings of the personalization variant.

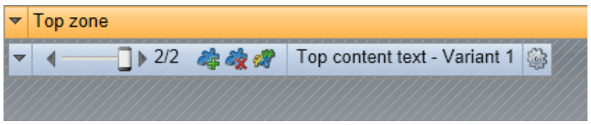


4. Click **OK**.

- o The standard web part configuration dialog opens — the variant is simply another instance of the original web part.
- o By default, the variant has the same values in its properties as are set for the original, but you can change them as required.

5. Click **OK** once you have configured the variant's web part properties as required.

After you create the first variant, the Design tab shows a personalization slider in the header of the given web part. You can use the slider to switch between the web part's variants.

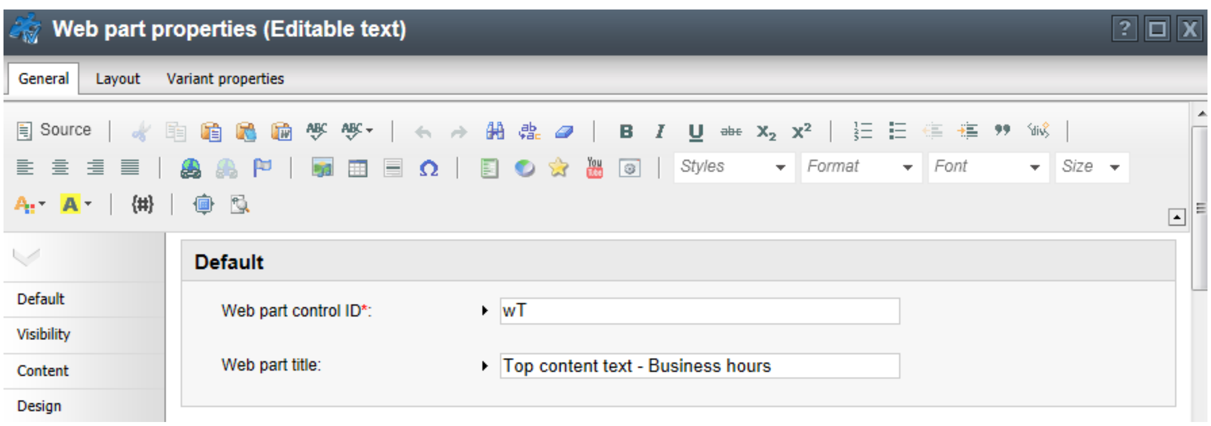


The buttons on the side of the slider allow you to:

- Add new variants
- Remove the currently selected variant
- Open a list of all variants defined for the given web part

You can **Configure** ( ) the properties of the web part variant currently selected in the slider:

- The configuration dialog has an additional **Variant properties** tab, where you can modify the settings of the personalization variant.
- You can specify a unique Web part layout for each variant on the **Layout** tab.



Configuring a web part personalization variant

The system stores the variants of personalized web parts on the given document's page template, so the variants are present on all pages that use the same template.

Note: If you delete ( ) a web part, all of its variants are removed along with it.

Personalizing web part zones

If you need to define personalized content that uses completely different web parts for each variant, create variants of entire web part zones. For each zone variant, you can:

- Add or remove child web parts

- Configure the properties of the web part zone
- Configure the properties of the child web parts

Each zone variant may contain any type or amount of web parts, regardless of what is placed inside other variants or the original zone.

Note: You cannot personalize zones that already contain personalized web parts. The same also applies in the opposite direction: You cannot add variants for individual web parts inside personalized zones. Moreover, personalization is only supported for standard web part zones and cannot be done for widget zones.

To add personalization variants for web part zones:

1. Edit the page in CMS Desk on the **Design** tab.

2. Right-click the header of the web part zone (or click the ▾ icon), hover over **Personalization variants** and click **Add zone variant**.

3. Fill in the display condition and other settings of the personalization variant.

4. Click **OK**.
   - The zone configuration dialog opens for the variant.
   - By default, the variant has the same values in its properties as are set for the original zone.

5. Click **OK** once you have configured the zone variant's properties as required.

When you create a new zone variant, the system automatically copies the content of the original zone. You do not have to rebuild the zone from scratch if you only need to make small modifications. The Design tab shows a personalization slider in the header of the given zone. You can use the slider to switch between the zone's variants. Use the buttons on the slider to manage the variants.

Tip: You can edit the zone variant currently selected on the slider by double-clicking the header of the zone.

The system stores the variants of personalized zones on the given document's page template, so the variants are present on all pages that use the same template.

Personalizing widgets

If you are an editor without access to the Design tab, you can leverage content personalization on pages with editor widget zones. You can create different variants of widgets, each with different property settings and content.

To add personalization variants for editor widgets:

1. Edit the page in CMS Desk on the **Page** tab.

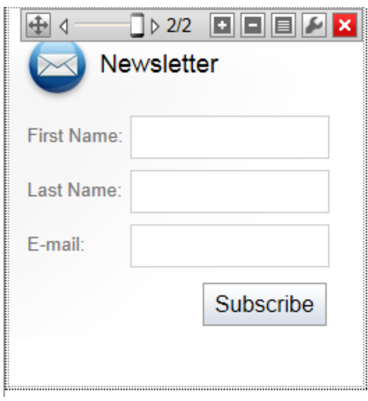
2. Hover over the widget and click **Add content personalization variant** () in the pop-up menu.

3. Fill in the display condition and other settings of the personalization variant.

4. Click **OK**.
   - The widget configuration dialog opens — the variant is simply another instance of the original widget.
   - By default, the variant has the same values in its properties as are set for the original, but you can change them as required.

5. Click **OK** once you have configured the widget variant's properties as required.

When you create a variant for a widget, a slider becomes available in the pop-up menu. You can use the slider to switch between the widget's variants.



The personalization menu offers the following buttons:

- **Add new variant** - creates a new variant for the given widget. If the page has a multivariate test, you need to select which type of variant to create.
- **Remove variant** - removes the variant currently selected on the slider (not available for the original widget).
- **Variant list** - opens a dialog showing a list of all variants defined for the widget. You can also manage the variants.

You can **Configure** () the properties of the widget variant currently selected in the slider. The configuration dialog has an additional **Variant properties** tab, where you can modify the settings of the personalization variant.

Note

- Removing a widget () also deletes all of its variants.
- Using the **Reset to default** action provided by a Widget actions web part removes all widget variants from the page.

The system categorizes editor widgets as page content, so widget variants are bound to specific documents (they are not included on the page template).

Viewing variants

You can use the sliders of personalized page components to switch between individual variants (including the original). The page displays the content of the currently selected variants on the **Design** tab, **Page** tab and in **Preview** mode.

The system only resolves the display conditions of personalization variants on the actual live site. This allows you to check how the page looks with different active variants, without having to fulfill the required conditions. Simply set the sliders to the appropriate variants.

When viewing the content of variants while cycling through sliders on the **Design** tab, make sure that the **Display web part content** checkbox in the **Edit** mode header is checked.

Editing the content of personalized pages

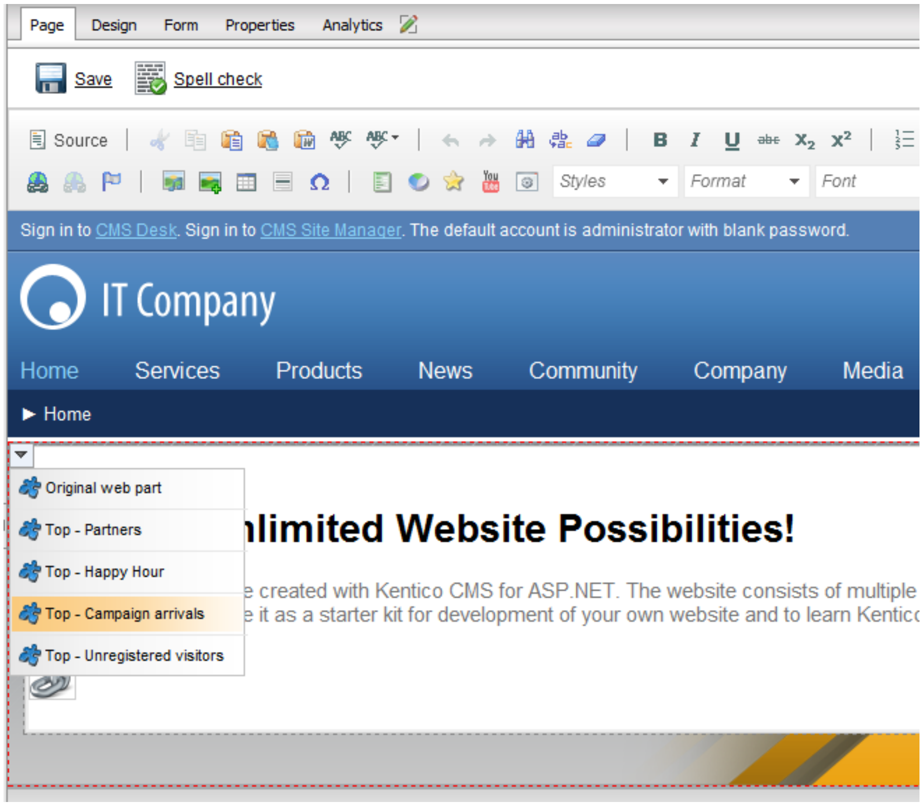
On pages (documents) with personalized web parts that provide editable regions (Editable text or Editable image), you can enter unique content for each variant.

1. View the document on the **Page** tab in CMS Desk.

2. Hover over the personalized section of the page.
   - A menu icon (▾) appears.

3. Click ▾ and select the variant that you wish to edit or view.

The system reloads the page with the content of the chosen variant. You can edit and save the variant's content as required.



Switching between the variants of an editable text region on the Page tab

Personalization variant settings

The following properties are available when creating or configuring personalization variants:

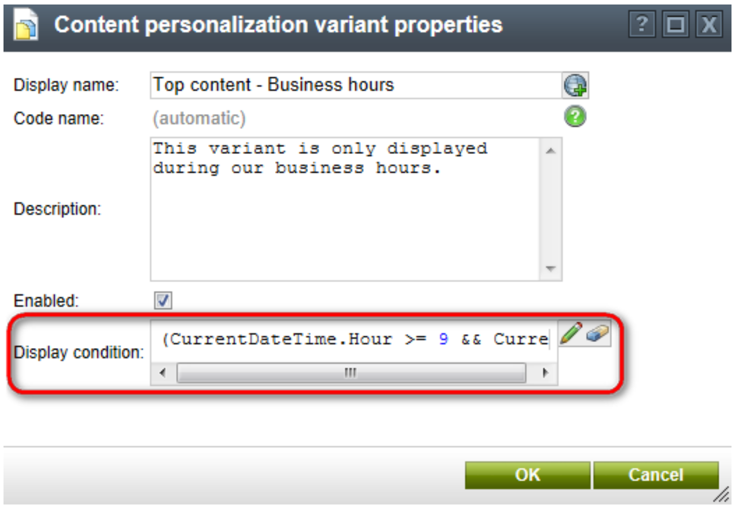
Display name	The name displayed in lists of content personalization variants in the administration interface.
Code name	Serves as a unique identifier of the variant.
Description	To make the variant easier to use and maintain, you can add an explanation about the scenario for which the variant is intended, describe the differences from the original component, etc.
Enabled	Indicates if the variant is considered as a possible content option. When evaluating the variant conditions of a personalized component, the system skips disabled variants (even if the requirements set by the condition are fulfilled).
Display condition	Enter the condition that must be fulfilled for the page to display the variant.

Setting display conditions for variants

When setting up personalized pages, the most important part of the process is to properly define the conditions that indicate when the system displays individual variants. By utilizing macro expressions, you can write conditions for virtually any type of scenario according to your specific requirements. Keep in mind that the result of a condition's expression must be a logical (boolean) value.

For more information about the available macro options, refer to Internals and API -> On-line marketing macros.

Specify the condition of each variant in its **Display condition** property.

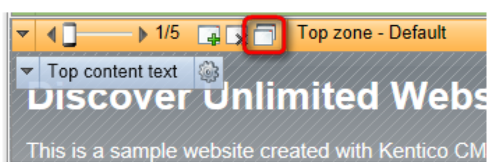


- Click edit ( ) to use the Macro condition editor, which allows non-technical users to create conditions based on predefined macro rules.
- The **Clear condition** ( ) action removes the current content of the condition field.

Setting the priority of personalization variants

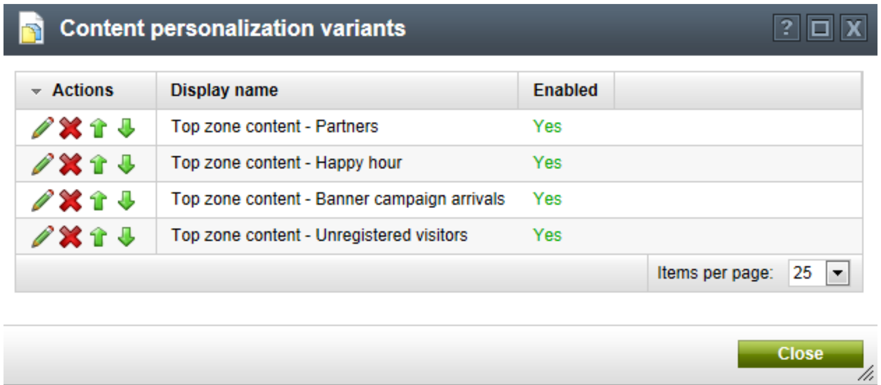
On the live site, the page displays only one variant for each personalized object. The system processes variants in the order of their priority and selects the first **enabled** variant whose condition is fulfilled in the given context. The page displays the original object in cases where the conditions of all variants are resolved as *false*.

By default, the priority of variants depends on the order in which they were created. You can see the current order on the personalization slider. Apart from the original object, which is always first, variants with a higher priority can be found further on the left of the slider.



To change the priority of a component's variants:

1. Click the **Variant list** button on the object's personalization slider.
   - A dialog containing a list of the variants opens.



2. Reorganize the variants as necessary through the **Up** ( ) or **Down** ( ) actions.

The system now evaluates the conditions of the component's variants according to the new order, starting from the top of the list.

10.3 Example: Creating a personalized page

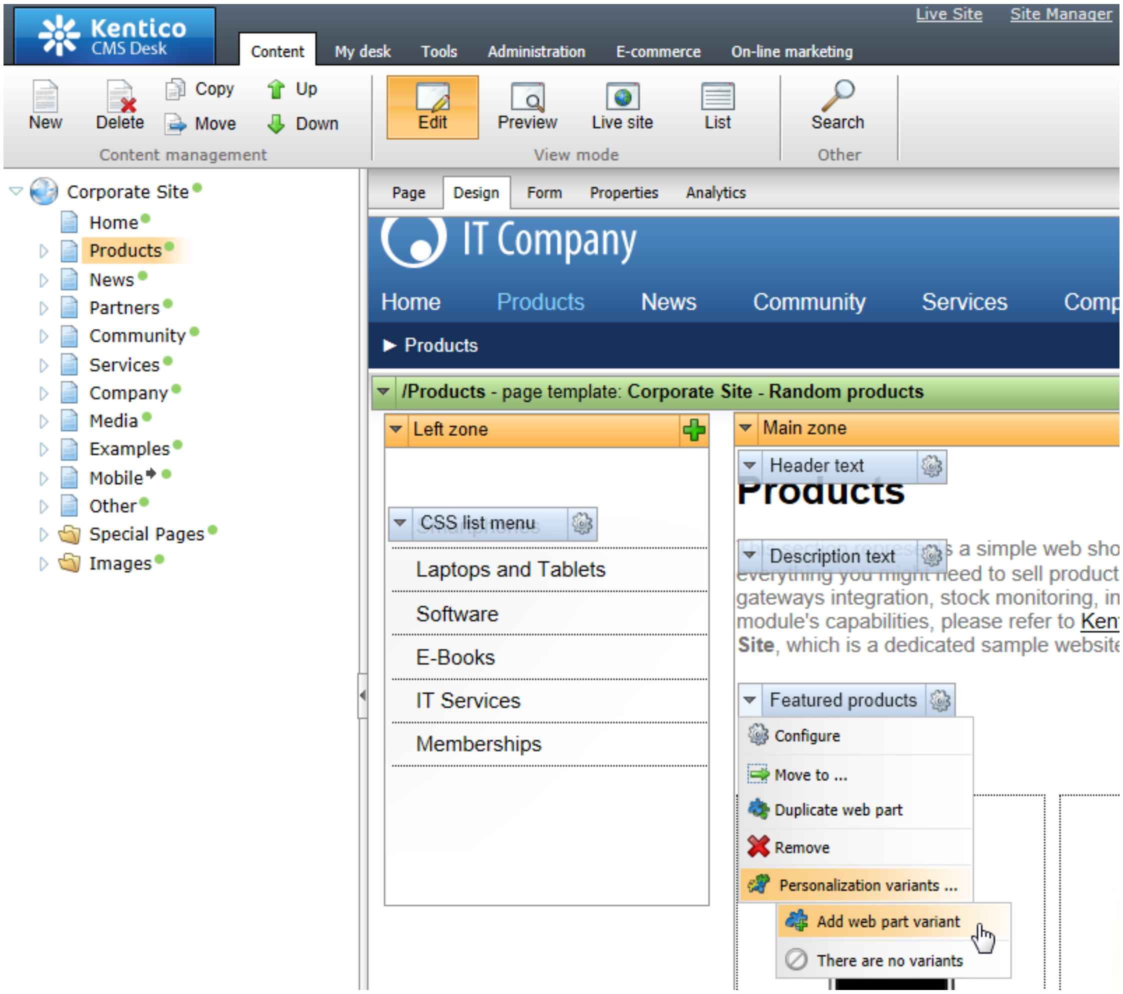
The example on this page describes how to create an advanced personalized page step-by-step. The personalization scenario leverages Contact management features in order to display different content for each visitor according to the actions they previously performed on the website. Specifically, the example demonstrates how to modify the default *Products* page on the sample Corporate Site so that it displays a targeted list of products based on the keywords that visitors enter into the website's search.

Creating the first product list variant

1. Open the Corporate Site in **CMS Desk** and select the **Products** document from the content tree.

2. Switch to the **Design** tab in **Edit** mode, where you can see the zones and web parts that make up the content of the page.

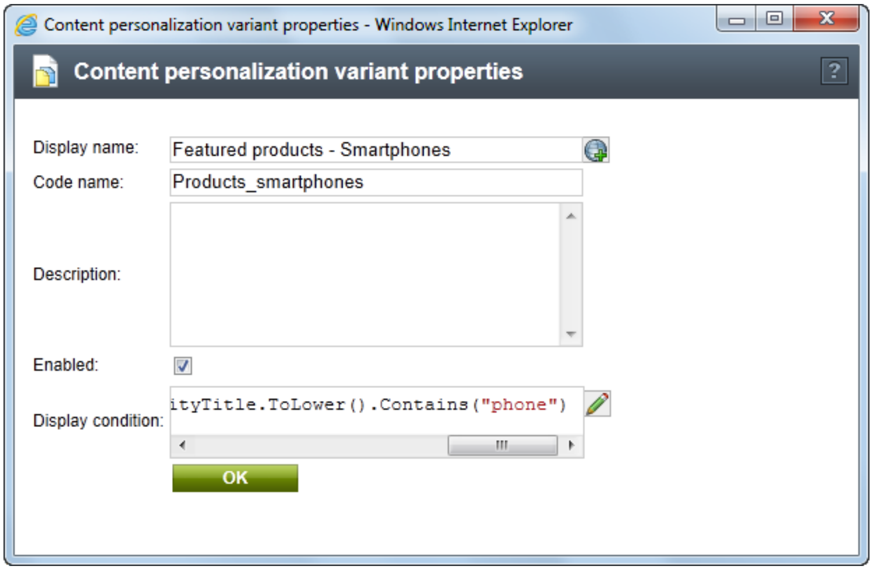
3. Right-click the header of the **Random products** web part located in the **Main zone** (its default title on this page is *Featured products*), hover over the **Personalization variants** option and click **Add web part variant**.



4. Enter the following values for the variant's properties:

- **Display name**: Featured products - Smartphones
- **Enabled**: Yes (leave it checked)
- **Display condition**:

```
OnlineMarketingContext.CurrentContact.LastActivityOfType("internalsearch")
.ActivityTitle.ToLower().Contains("phone")
```



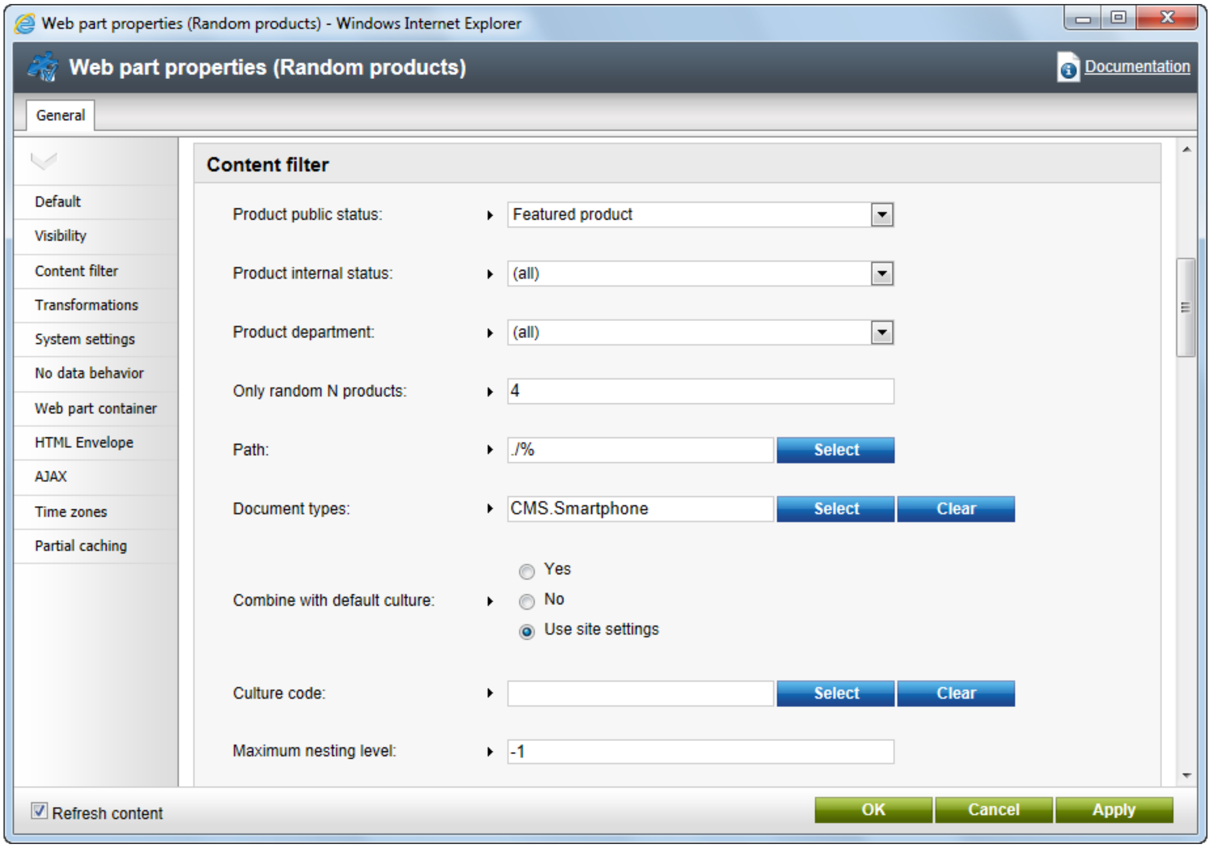
This condition ensures that the page displays the variant only if the latest keyword searched by the current user contains the word *phone*, such as *smartphone*, *iPhone* etc. The condition is composed of several macro methods and expressions:

- The **LastActivityOfType** method retrieves the current contact's most recently logged Activity of the *Internal search* type.
- The **ActivityTitle** property loads the title of the logged activity, which stores the exact text of the search phrase.
- The **ToLower** method converts the title to lower case (to make sure that the condition correctly recognizes search keywords with capital letters).
- The **Contains** method checks if the second parameter (i.e. the word *phone*) is located somewhere within the text returns the result.

5. Click **OK**.
   - The configuration dialog for the variant's web part properties appears.
   - The variant automatically loads the property values from the original web part.

6. Change the following properties:

- **Default -> Web part title**: *Featured products - Smartphones* (allows you to identify which variant you are working with on the Design tab)
- **Content filter -> Product public status**: (all)
- **Content filter -> Document types**: *CMS.Smartphone* (ensures that the web part only loads and displays smartphone products)



7. Click **OK**.

Creating the second product list variant

1. Add another variant to the *Featured products* web part by clicking **Add content personalization variant** ( ) next to the slider.

2. Set the following properties for the variant:

- **Display name**: Featured products - MS Windows
- **Enabled**: Yes (leave it checked)

3. Click edit ( ) next to the **Display condition** field to open the macro condition editor.

4. On the **Designer** tab, use the **Add expression** ( ) button to create three expressions.

5. Copy the following macro code into the text areas on the left:

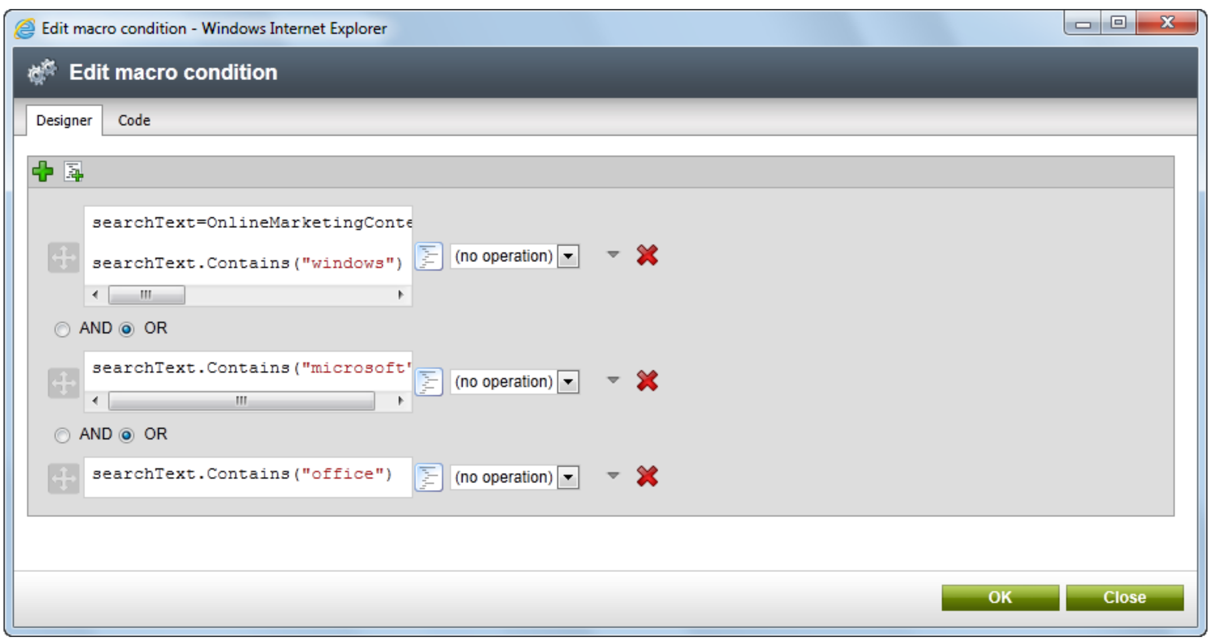
```
searchText=OnlineMarketingContext.CurrentContact.LastActivityOfType
("internalsearch").ActivityTitle.ToLower();

searchText.Contains("windows")
```

```
searchText.Contains("microsoft")
```

```
searchText.Contains("office")
```

- Select the **OR** radio button between the expressions.
- The **Contains** function already returns a logical value, so choose the *(no operation)* option in the selector for each expression.

This condition is composed of multiple parts, so the macro only loads the latest search keyword once (just like the first variant) and stores the value in a variable named **searchText**. This avoids unnecessary repetition and improves performance. The condition checks the variable for three different words. Because you selected the *OR* option between individual expressions, the condition is fulfilled if the system finds at least one of the keywords in the search text.



6. Click **OK** to close the macro condition dialog.
   - The condition editor automatically generates the macro output and places it into the **Display condition** field.

7. Click **OK** to confirm the variant's properties.

8. Make the following changes to the web part's properties:

- **Default -> Web part title**: Featured products - MS Windows
- **Content filter -> Product public status**: (all)
- **Content filter -> WHERE condition**: *SKUManufacturerID = 6* (the web part variant only loads products that have *Microsoft Corporation* set as their manufacturer)

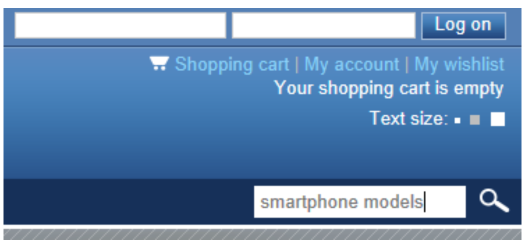
9. Click **OK** again to create the second variant.

There are now three different versions of the product list on the page.

Result

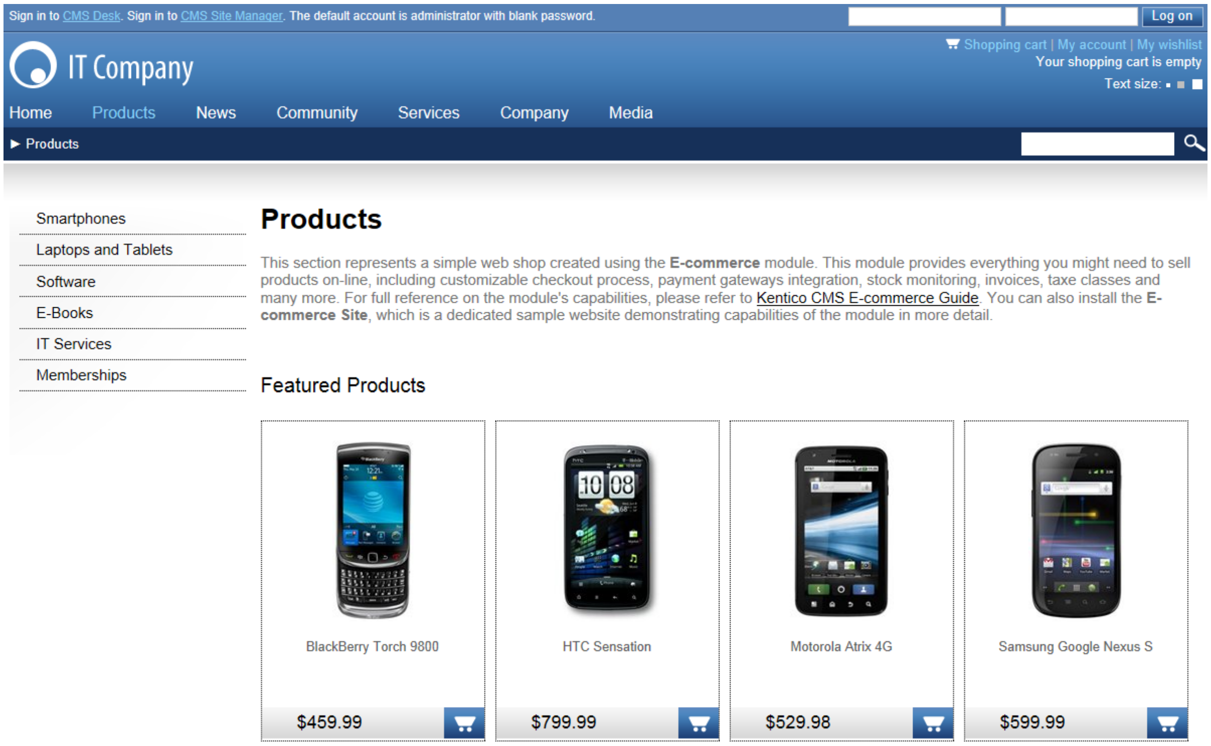
To try out how the personalization works:

1. Sign out of CMS Desk to view the live site.
2. Enter *phone* or any other text containing the word into the search bar on any of the website's pages (it is located in the header under the navigation menu).



3. Select the **Products** page from the menu.

The page only displays smartphone products in the **Featured products** section.



You can test other possible search expressions and check how the content of the page changes. If the searched text does not match the conditions of either one of the created variants, the page shows the default list containing a random selection of featured products. If you log on to the site as a different registered user (e.g. user name *Andy* with a blank password), this changes the current contact and the system displays the default version of the page until you search for an expression that matches one of the personalization variant conditions.

The result of the example is a personalized page that dynamically offers products that the visitor has shown interest in (by searching for related terms), rather than a random selection of products from the entire catalog. When setting up this type of personalization on your own website, you can create any number of variants with appropriate conditions according to your specific requirements and product catalog.

Part XI

E-mail marketing

11 E-mail marketing

11.1 Overview

E-mail marketing can be one of the most effective ways to attract new customers to your website or keep in touch with existing clients. In Kentico CMS, you have the option to prepare and send out mass e-mails via newsletters, which ensure delivery to the addresses of all subscribers. Subscriber management can either be done by importing a mailing list manually or by allowing visitors to opt-in individually. Additionally, you may include any existing users registered on your site, or even subscribe entire Contact groups to a newsletter.

For detailed information about the functionality of the newsletter module and its configuration options, please see the Modules -> Newsletters chapter of the Kentico CMS Developer's Guide.

When using newsletters to carry out an e-mail marketing campaign, it is important to determine its overall effectiveness and optimize the issues according to the results. This can be achieved by tracking the sent e-mails and monitoring the reactions of recipients. For this purpose, you may utilize several features that provide feedback and statistics about how successful your newsletters are and which actions subscribers take when reading issues, including the following:

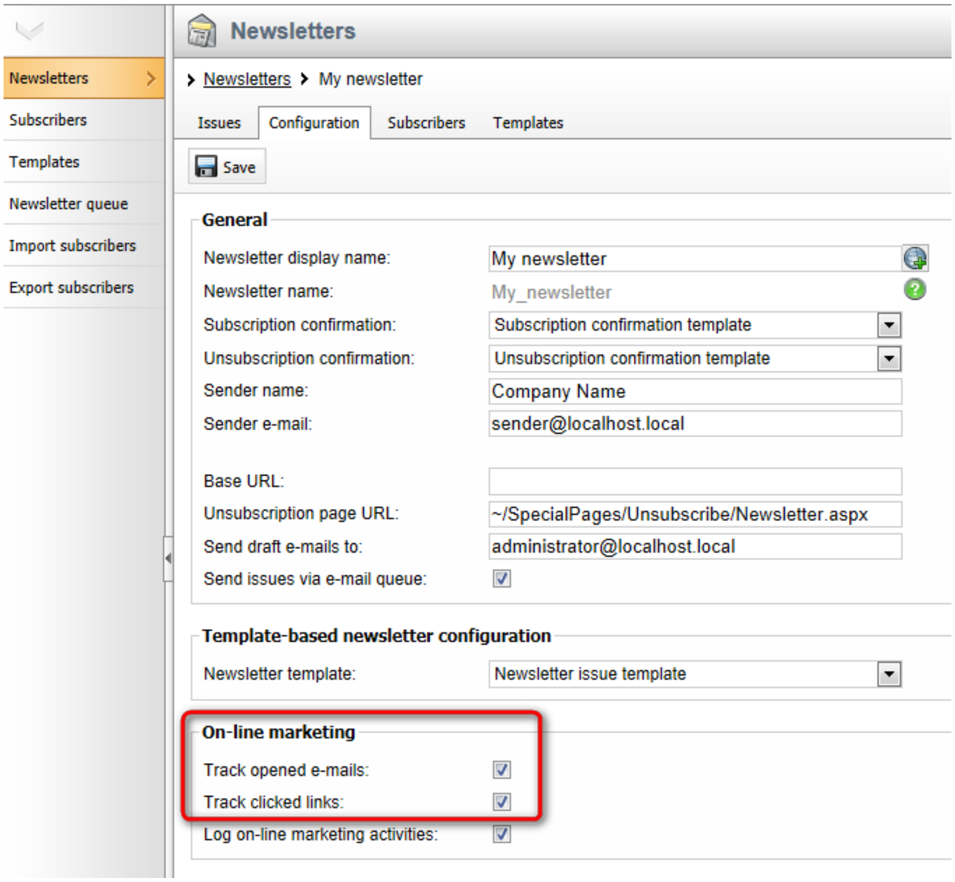
- E-mail tracking - measures how many newsletter issues were actually opened by subscribers and the clickthrough rate of any links placed into the content of the e-mails.
- Bounced e-mail monitoring - allows you to identify and block subscribers with invalid e-mail addresses to which newsletter issues cannot be delivered.
- A/B testing - provides a way to create several different versions of each newsletter issue and evaluate them based on the e-mail tracking statistics measured for a test group of subscribers. This allows you to send the most successful variant to the remainder of the subscribers.

Please note that the on-line marketing features of newsletters are only available if the **Enable on-line marketing** setting is enabled for the given website in **Site Manager -> Settings -> On-line marketing**.

11.2 E-mail tracking

Two types of tracking can be used for the e-mails that deliver newsletter issues to subscribers. This functionality may be enabled for individual newsletters through the **CMS Desk -> On-line marketing -> Newsletters** interface by editing the selected newsletter on the **Configuration** tab:

- **Track opened e-mails** - if enabled, the e-mails used to send out the issues of the newsletter will be tracked and statistics will be kept about the amount of e-mails that are opened by subscribers.
- **Track clicked links** - if enabled, statistics will be kept about the amount of clicks subscribers perform on hyperlinks placed in the e-mail issues of the newsletter.



Please note that tracking cannot be done retroactively for e-mails that have already been sent before the appropriate settings were enabled. Also, since draft e-mails are intended only for testing purposes, they are not included in tracking statistics (only the e-mails sent to actual subscribers are measured).

Logging newsletter actions as activities

You can also decide if you want the actions related to a specific newsletter to be included in the site's on-line marketing Activity statistics.

This is done by setting the **Log newsletter actions as on-line marketing activities** property on the **Configuration** tab.

Newsletter activities include the following types of events:

- Subscription
- Unsubscription
- E-mail opening
- Link clickthrough

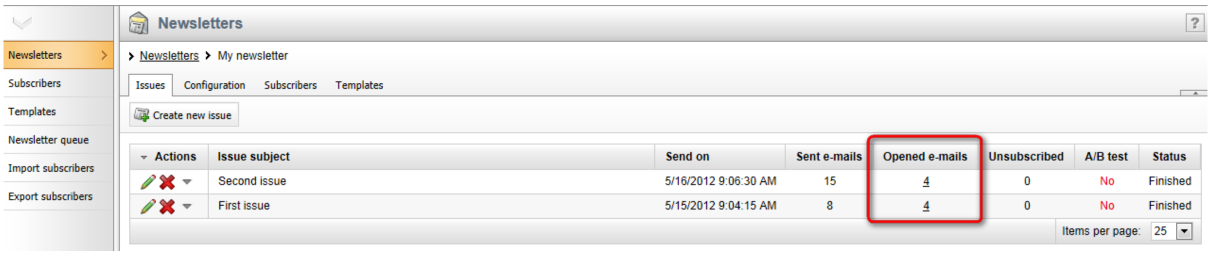
Opened e-mails

This feature allows you to monitor how many newsletter e-mails are actually opened by subscribers.

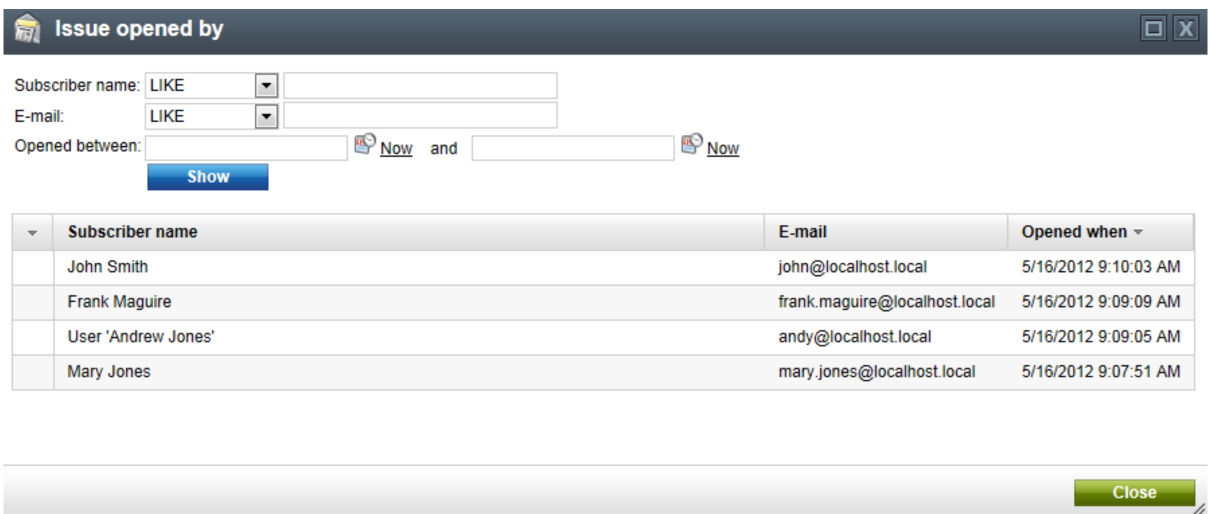
Open rate is one of the key metrics for judging an e-mail campaign's success.

Tracking of this type is achieved by embedding a small, invisible image into the content of e-mails. When this image is requested from the server for the first time, the system marks the e-mail as received and opened for the given subscriber. As a result, mails will only be counted as opened if the e-mail client of the subscriber allows the loading of images. Alternatively, if the recipient clicks on a link contained in the e-mail and **Track clicked links** is enabled for the newsletter, the e-mail will be recognized as opened even if images are blocked. Because of these factors, the indicated number of opened e-mails may be slightly lower than the actual amount.

If tracking is enabled, the statistics of newsletter issues can be viewed when editing a specific newsletter on the **Issues** tab. The amount of e-mails that were received and opened by subscribers can be compared side-by-side with the total number of sent e-mails.



If the number is greater than zero, it can be clicked to display a dialog containing a detailed list of all subscribers who opened the e-mail.

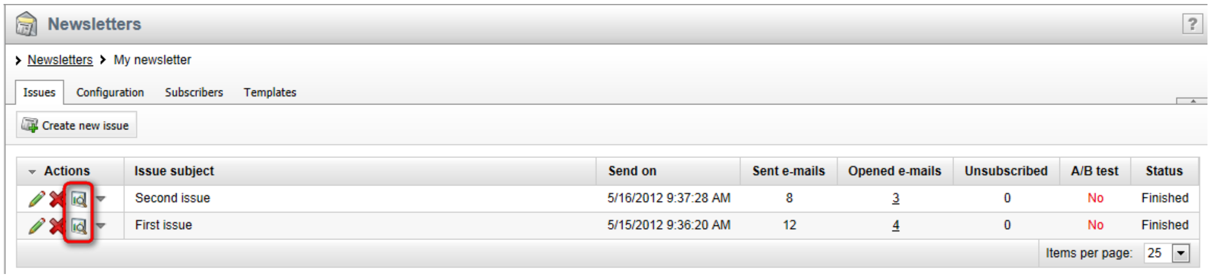


The subscribers can be filtered according to their name, e-mail address or the time interval during which they opened the e-mail.

Clicked links

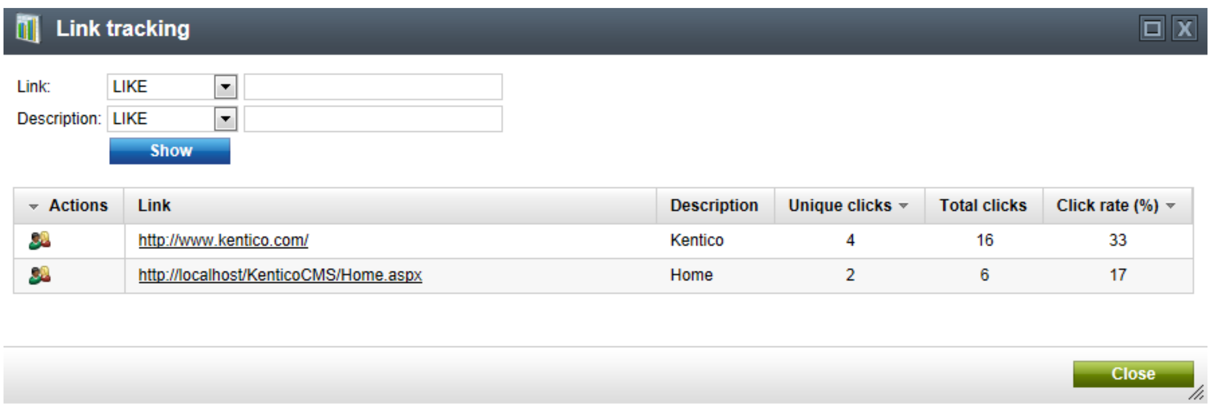
Another way to measure the effectiveness of an e-mail campaign is to track which links are clicked by the recipients and how many times, also known as the *clickthrough rate*. If this type of tracking is enabled, all links in newsletter issues are converted to tracking links when they are sent out. This applies to both template-based and dynamic newsletters. When a tracking link is clicked by a subscriber, the system stores a record of the event.

Link statistics of individual newsletter issues can be viewed when editing a specific newsletter on the **Issues** tab. The **View tracking statistics** ( ) action is available for all issues that contain tracked hyperlinks.

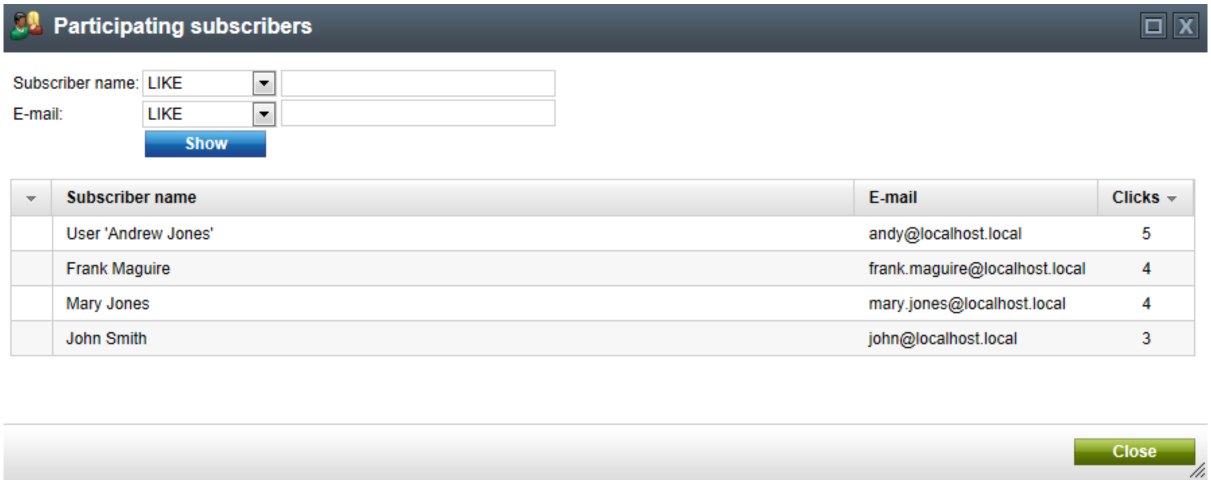


When clicked, this action opens a modal dialog that lists all links in the given issue and displays the following information about them:

- **Unique clicks** - shows how many different subscribers clicked on the link.
- **Total clicks** - indicates how many times the link was clicked, including multiple clicks from the same user.
- **Click rate (%)** - displays the clickthrough rate of the link as a percentage. Calculated as the number of unique clicks divided by the total amount of sent e-mails.



Further details can be displayed for each link by using the **View participating subscribers** ( ) action. The action opens another dialog containing the names and addresses of the subscribers that used the given link and their respective click count.



Excluding links from tracking

Tracking may be manually disabled for any link in a newsletter issue by editing the source and inserting the **tracking="false"** attribute into the **<a>** tag of the given hyperlink, for example:

Kentico CMS

When editing the content of a newsletter issue, you can easily switch to its source using the **Source** button on the toolbar of the WYSIWYG editor.

In addition, please keep in mind that newsletter unsubscription links generated by the *{%UnsubscribeLink%}* expression are not tracked by this feature (unsubscription statistics are kept for each newsletter by default). Links to local anchors within the content of an e-mail are also excluded.

11.3 Monitoring bounced e-mails

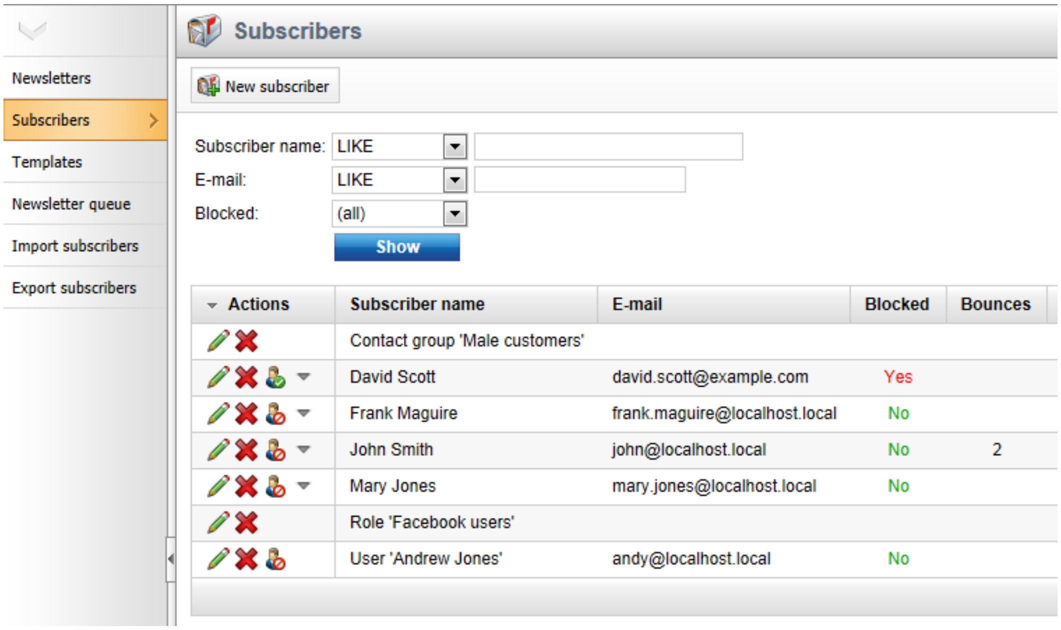
When an e-mail cannot be delivered successfully for some reason, an automatic reply informing about the problem is returned (bounced) back to the sender. Tracking bounced mails allows the system to identify addresses that do not correctly receive newsletter issues. Removing invalid addresses from your mailing lists saves bandwidth and improves the accuracy of your subscription statistics, leading to a better delivery rate.

Once this feature is enabled and properly configured (please see the *Configuration* section below for more information), the mailbox where the bounced e-mails are stored is periodically checked. All contained e-mails are analyzed and the bounce counter of subscribers is increased as necessary. After a bounced e-mail is processed, it is deleted from the mailbox. This functionality is ensured by the **Check bounced e-mails** scheduled task, which is executed once per hour by default.

If the number of bounces received from a subscriber reaches a specified limit, the system will automatically block the corresponding address from receiving any further newsletter issues.

Bounce statistics can be viewed in various parts of the **CMS Desk -> On-line marketing ->**

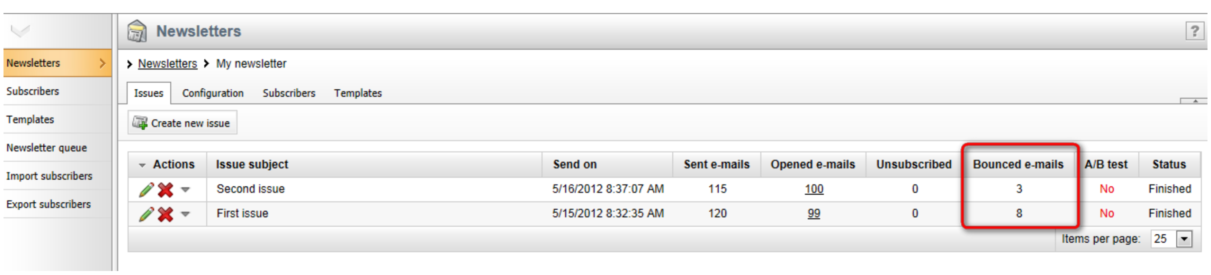
Newsletters interface. On the main **Subscribers** tab, the amount of bounces associated with individual subscribers is displayed.



The **Blocked** field in the filter above the list may be used to display only blocked or active subscribers. The **Block** () or **Unblock** () actions allow you to manually change the status of subscribers. When a subscriber is unblocked, their bounce counter is reset to zero. To perform these actions for individual subscribers assigned through a role or contact group, edit () the given role or contact group subscriber and switch to the **Users** or **Contacts** tab respectively.

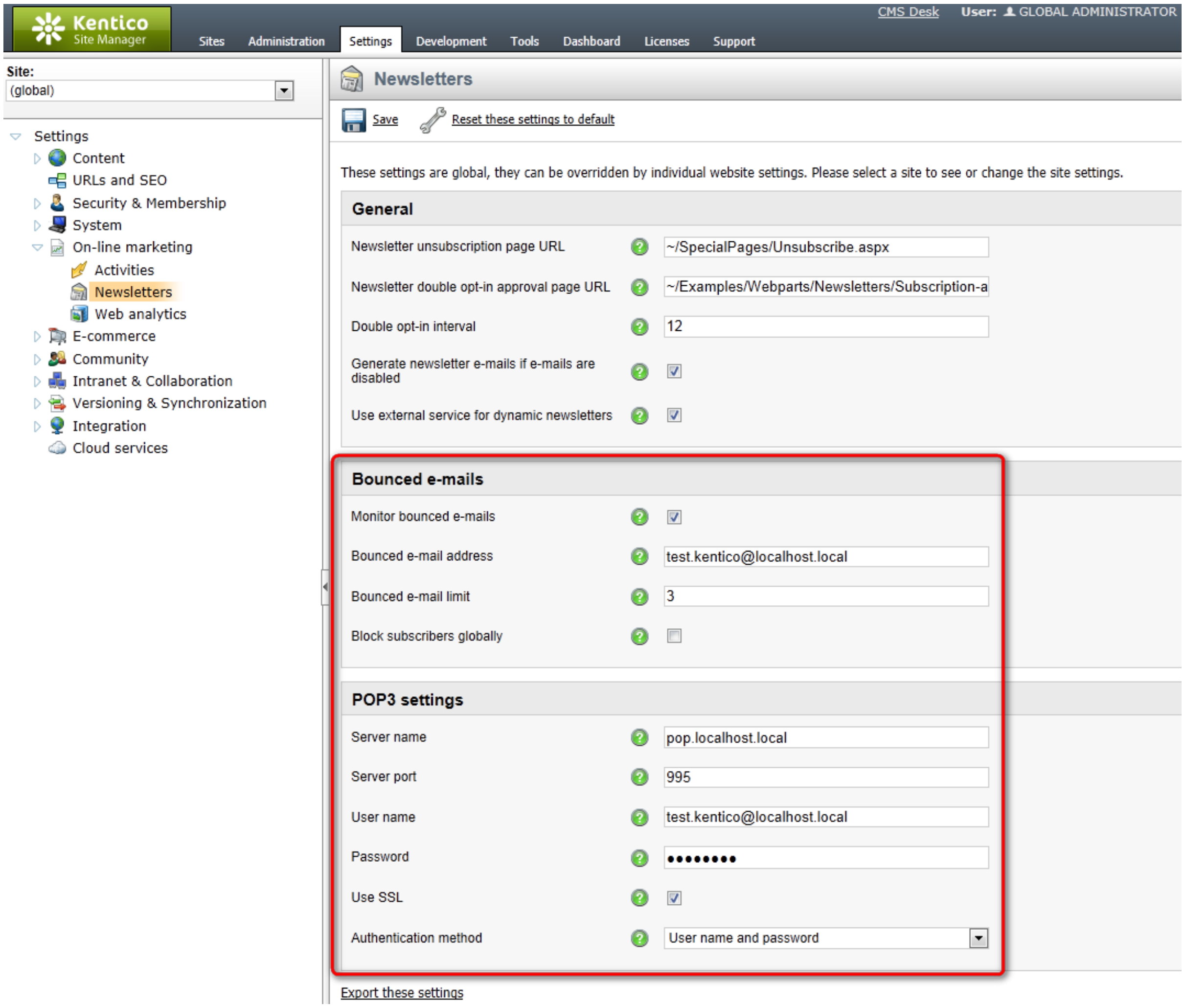
The same options are also available when viewing the subscribers of a specific newsletter in **Newsletters -> edit () newsletter -> Subscribers**. The bounce count of a subscriber is shared for all newsletters on the given site.

On the **Issues** tab of a newsletter, the total amount of bounces is shown for each issue. This way, you can view the amount of sent e-mails and compare it with the number of bounces.



Configuration

To enable this feature, go to **Site Manager -> Settings -> On-line marketing -> Newsletters** and check the **Monitor bounced e-mails** box. It is also necessary to properly fill in the remaining settings in the **Bounced e-mails** and **POP3 settings** categories.



Keep in mind that it is possible to enter different values for each site to override the global settings (you can use the **Site** selector above the settings tree for this purpose). The following options are available:

Bounced e-mails

- **Monitor bounced e-mails** - indicates if bounced e-mails should be counted for newsletter subscribers.
- **Bounced e-mail address** - sets the e-mail address to which bounced e-mails are sent when the delivery of a newsletter issue to a subscriber fails. If set, this address is used in the *From* field of newsletter issue e-mails.
- **Bounced e-mail limit** - sets the amount of bounced e-mails that can be recorded for a subscriber before the system blocks them from receiving further newsletter issues. This limit is set for all newsletters under the selected site. If *0* is entered, subscribers will never be blocked automatically, but their bounced e-mail count will still be tracked and displayed in the *CMS Desk -> On-line marketing -> Newsletters* interface.
- **Block subscribers globally** - if checked, bounces will be added to all subscribers that have the same e-mail address. This is applied across all sites in the system. Please note that this setting does not ensure consistency between the bounce counts of all subscribers with a shared address, only that new bounces will be added to all of them.

POP3 settings

- **Server name** - sets the domain name or IP address of the mail server where the bounced e-mails should be stored. The POP3 protocol is used to check the server and monitor bounced e-mails.
- **Server port** - specifies the number of the port used to connect to the mail server.
- **User name** - sets the user name used for authentication against the mail server.
- **Password** - sets the password used for authentication against the mail server.
- **Use SSL** - indicates whether the connection to the mail server should be secured using SSL.
- **Authentication method** - selects the authentication method used for the connection to the mail server. Options include basic user name and password authentication and several challenge-response mechanisms. The *Auto* option uses APOP if the server supports it and plain text user name and password authentication otherwise.

Some mail servers may be configured to store e-mails even after they are downloaded, which causes bounces to be counted multiple times (every time the mailbox is checked), so please adjust the settings of the e-mail server if you experience issues of this type.

Monitoring bounced e-mails under medium trust

If your website is hosted in a medium trust environment, the bounced e-mails feature will not be functional by default.

To be able to check bounced e-mails, the newsletter module makes use of a component that creates outgoing network connections via POP3, a standard e-mail protocol for receiving maildrops from an e-mail server. This component requires the **SocketPermission** for its operation, otherwise it fails when attempting to connect to the server. This permission is denied for applications under medium trust.

If you cannot raise the trust level yourself or create a custom security policy that includes this permission, the only solution is to attempt to convince your hosting administrators to grant the *SocketPermission* to your application. If you are unable to do so, the bounced e-mail monitoring feature will unfortunately not be available.

11.4 A/B testing

When preparing a newsletter issue for marketing purposes, it can be difficult to predict exactly how subscribers will react to various important elements, such as the e-mail subject or the primary call to action. In some cases, you may also wish to find out which option out of several alternatives will have the most positive effect. Optimizing your newsletter issues via A/B testing provides a possible solution to these problems.

It allows you to create multiple different versions of each issue referred to as *variants*. These variants are then mailed out to a *test group* of subscribers, typically a relatively small percentage of the newsletter's full mailing list. This way, the best version of the issue can be identified based on the tracking statistics measured for the test group, and then sent to the remainder of the subscribers. The winning variant can either be selected automatically by the system according to specified criteria or manually after evaluating the results of the test.

Please note that A/B testing works best for newsletters that have a large number of subscribers. With a small testing group, the results may be heavily affected by random factors and will not be statistically significant.

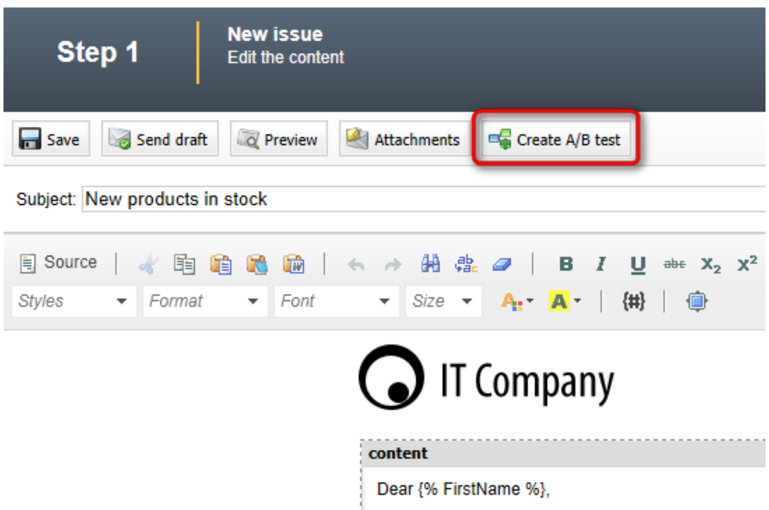
Prerequisites

A/B testing variants of newsletter issues are evaluated based on the actions performed by recipients, so both types of E-mail tracking need to be enabled for the given newsletter on its **Configuration** tab (**Track opened e-mails** and **Track clicked links**). This also requires the **Enable on-line marketing** setting to be enabled in **Site Manager -> Settings -> On-line marketing**.

Additionally, A/B testing is only supported for template-based (static) newsletters.

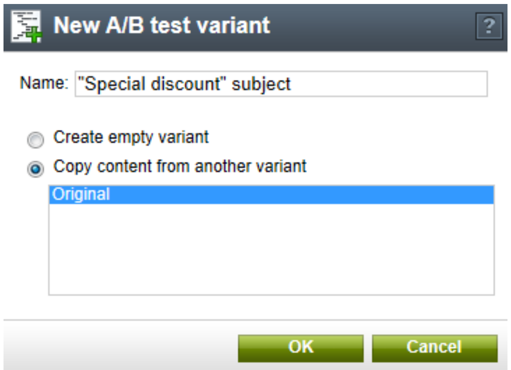
Creating A/B tests for newsletter issues

If the conditions described in the section above are met, A/B tests can be defined through the **CMS Desk -> On-line marketing -> Newsletters** interface either directly when creating a new newsletter issue via the wizard, or when editing an existing issue on the **Content** tab. In the case of new issues, it is necessary to **Save** the content before the advanced actions become available. To add the first testing variant for the issue, click the **Create A/B test** button offered among the header actions.



This opens a dialog where the new variant can be defined. Start by filling in the **Name**, which will be used to identify the variant while working with the A/B test. You can then choose one of two possible options that determine the initial content of the variant:

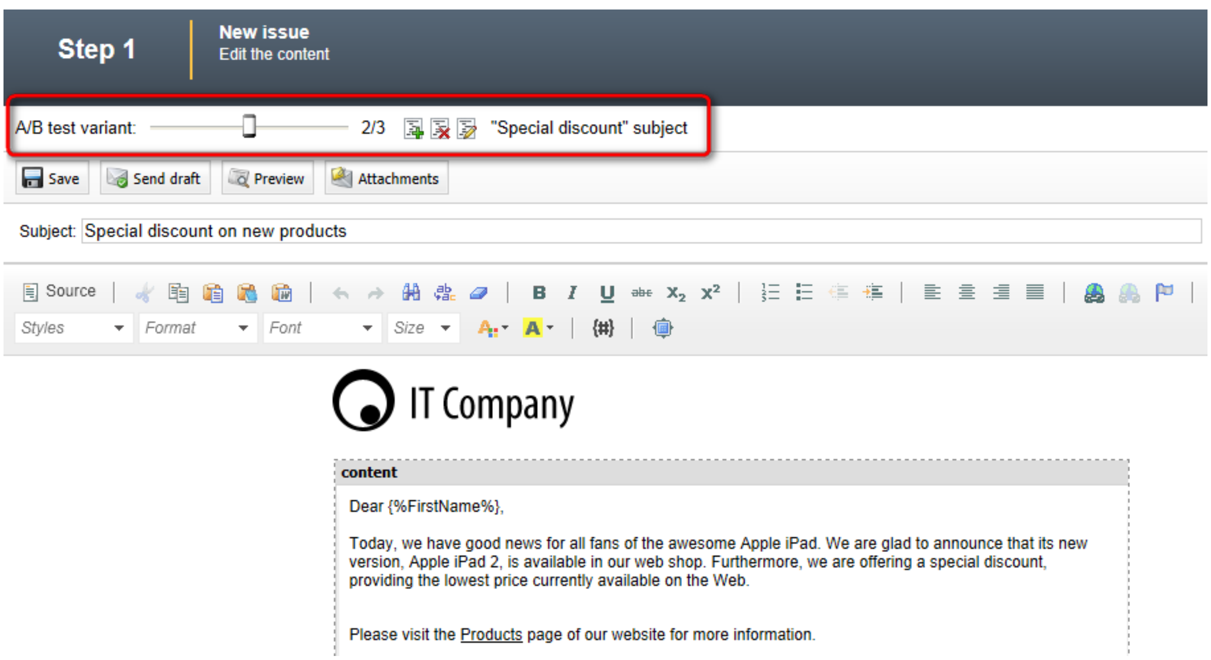
- **Create empty variant** - select this option if you wish to create the issue variant from scratch. The variant will use the main template set for the newsletter on the *Configuration* tab and the content of its editable regions will be empty.
- **Copy content from another variant** - if selected, the content of an existing variant (or the original issue) will be used as a starting point that you can modify as required. Choose the source from the list of issue variants in the list below. The variant will use the same template as the source issue and all editable region content will also be copied. This option makes it easy to create variants for testing small changes, such as a different e-mail subject or text headline.



Now that the issue has an A/B testing variant defined, a slider will be displayed at the top of the content editing page. You can use it to switch between individual variants, including the original issue. The name of the currently selected variant is shown next to the slider. You may manage the A/B test variants using the following buttons:

- **Add variant** - creates another variant of the issue. You can add any number of variants.
- **Remove variant** - deletes the variant currently selected through the slider.
- **Edit properties** - allows you to change the name of the currently selected variant. If required, you may also rename the *original* issue.

The settings and content of the currently selected variant can be modified just like when editing a standard issue. Each variant may have a different subject, issue template, editable region content etc. This allows you to test any variables that you need.

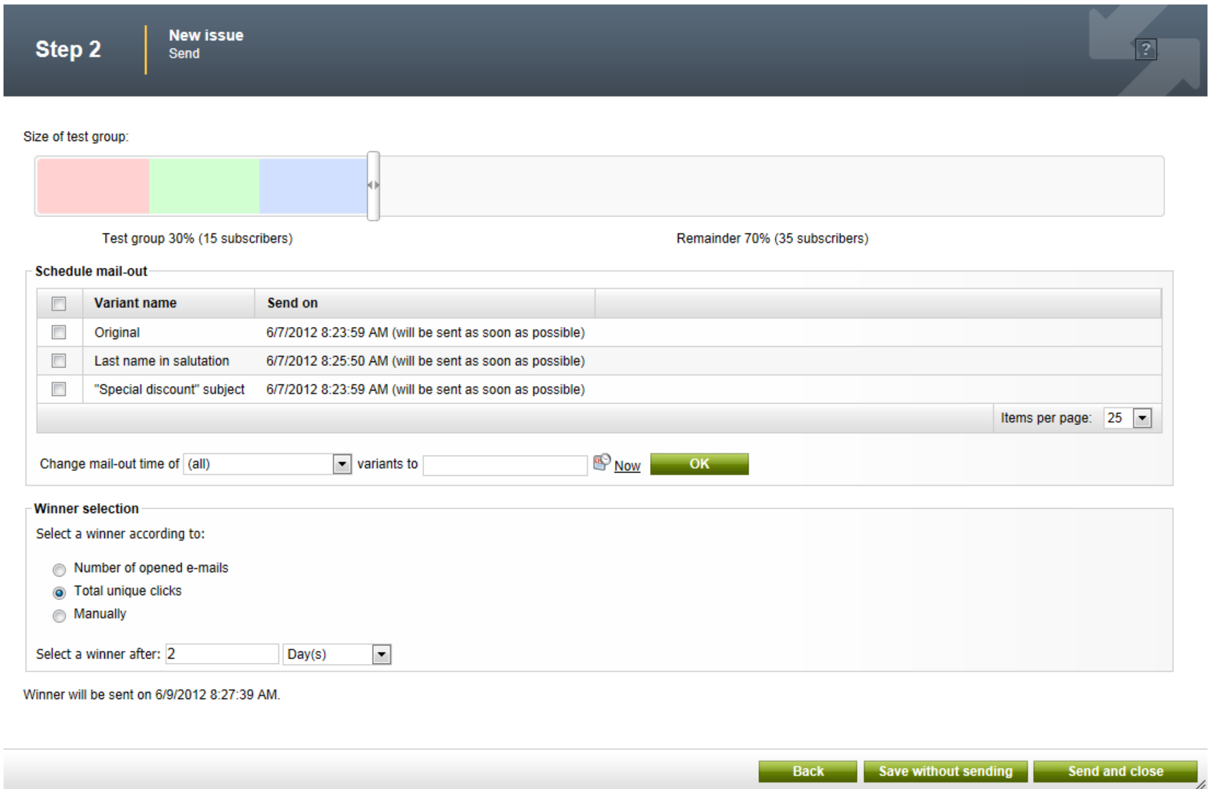


Sending A/B test issues

When all of the issue's variants are defined as required, the next step is to configure and schedule how the test should be sent out and evaluated. This can be done either in the send step of the new issue wizard or when editing an existing issue on the **Send** tab.

The size of the subscriber test group can be defined using the slider in the upper part of the page. By moving the slider's handle, you can increase or decrease the number of subscribers that will receive the variants of the newsletter during the testing phase. The test group is automatically balanced so that each variant is sent to the same amount of subscribers. Because of this, the overall test group size will always be a multiple of the total number of variants created for the issue.

The remaining subscribers who are not part of the test group will receive the variant that achieves the best results (i.e. the winner) after the testing process is complete.



Using a full test group

It is even possible to set up a scenario where the test group includes 100% of all subscribers. In this case, the A/B test simply provides a way to evenly distribute different versions of the issue between the subscribers and the selection of the winner is only done for statistical purposes.

In the **Schedule mail-out** section below the slider, you can specify when individual issue variants should be sent to the subscribers from the corresponding portion of the test group. To schedule the mail-out, enter the required date and time into the field below the list (you may use the **Calendar** selector or the **Now** link) and then click **OK**. This can either be done for *all* variants or only those selected through the checkboxes in the list. If the mail-out time is the same for multiple variants, they will be sent in sequence with approximately 1 minute intervals between individual variants.

The configuration made in the **Winner selection** section determines how the winning variant of the A/B test will be chosen. You can select one of the following options:

- **Number of opened e-mails** - the system will automatically choose the variant with the highest number of opened e-mails as the winner. This type of testing focuses on optimizing the first impression of the newsletter, i.e. the subject of the e-mails and the sender name or address, not the actual content.
- **Total unique clicks** - the winner will be chosen automatically according to the amount of link clicks measured for each variant. Each link placed in the issue's content will only be counted once per subscriber, even when clicked multiple times. This option is recommended if the primary goal of your newsletter is to encourage subscribers to follow the links provided in the issues.
- **Manually** - the winner of the A/B test will not be selected automatically. Instead, the author of the issue (or other authorized users) can monitor the results of the test and choose the winning variant manually at any time.

When using an automatic selection option (one of the first two), it is also necessary to enter the duration of the testing period through the **Select a winner after** settings below. This way, you can specify how long the system should wait after the last variant is sent out before it chooses a winner and mails it to the remaining subscribers.

Once everything is configured as required, you can confirm that the variants should be sent according to their mail-out scheduling time by clicking the **Send** button (or **Send and close** in the new issue wizard). If you only wish to save the configuration of the A/B test without actually starting the mail-out, use the **Save** (**Save without sending**) button instead.

Evaluating A/B tests

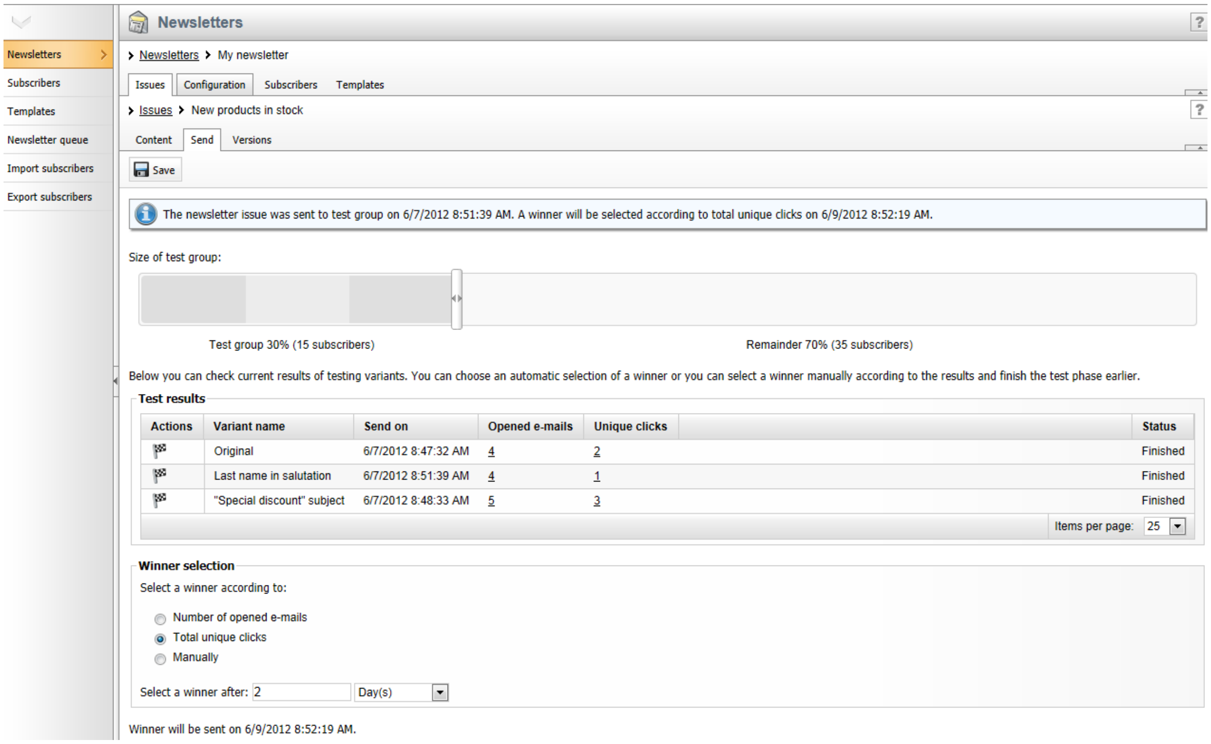
The testing phase begins after the first variant is sent out. If you need to make any changes to the configuration of the A/B test or the content of its variants, you can do so by editing the issue.

On the **Content** tab, you may modify the variants that have not yet been mailed, but the slider actions will be disabled. This means that it is no longer possible to add, remove or rename variants.

If you switch to the **Send** tab, the test group slider will now be locked. However, you can view the e-mail tracking data measured for individual variants in the **Test results** section. The current tracking statistics are shown for each variant, specifically the number of *opened e-mails* and amount of *unique link clicks* performed by subscribers. By clicking on these numbers, you can open a dialog with the details of the corresponding statistic for the given variant. It is also possible to reschedule the sending of variants that have not been mailed yet using the selector and date-time field below the list.

The **Select as winner** action allows you to manually choose a winner (even when using automatic selection). It opens a confirmation dialog where you can schedule when the winning issue variant should be sent to the remaining subscribers. If you specify a date in the future, you will still have the option of choosing a different winner during the interval before the mail-out.

The winner selection criteria may also be changed at any point while the testing is still in progress.

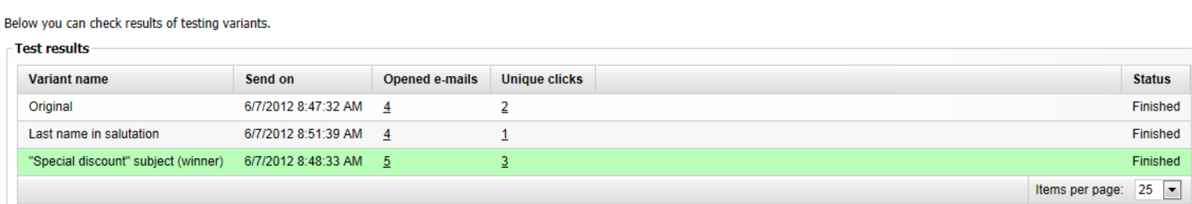


Special cases with tied results

If a draw occurs at the end of the testing phase (i.e. the top value in the tested statistic is achieved by multiple issue variants), the selection of the winner will be postponed and evaluated again after one hour.

In certain situations, you may need to choose the winner manually even when using automatic selection, e.g. if you are testing the number of opened e-mails and all subscribers in the test group view the received issue.

Once the test is concluded and the winner is decided, the given variant is highlighted by a green background. At this point, the winning issue is mailed out to the remaining subscribers who were not included in the test group and no further actions are possible except for viewing the statistics of the variants.



The overall statistics of the A/B tested issue, including the e-mails used to deliver the winning variant to the subscribers outside of the test group, can be monitored in the usual way on the **Issues** tab of the newsletter. When viewing the opened e-mail or clickthrough data in the detail dialogs, you may use the additional **Variants** filter to display either the total *(all)* values for the entire issue, or only those of specific variants. The statistics of the winning variant include both the corresponding portion of the test group and the remainder of the subscribers who received it after the completion of the testing phase.

Part XII

Security

12 Security

12.1 Overview

The security options of the on-line marketing features described in this guide can be configured in the same way as for all Kentico CMS modules. Access to various types of data and the ability to perform specific actions is only allowed for users who belong to roles that have the appropriate Permissions for individual on-line marketing modules.

Additionally, the user interface is composed of separate elements and each user may only view those parts that are included in the UI personalization profile configured for their roles.

Default accounts and roles for on-line marketing

Security restrictions do not apply to users who are designated as global administrators, since such accounts override any permission requirements and automatically have access to all UI elements.

You may also assign users to the **Marketing manager** default role, which is imported along with the sample Corporate Site. By default, this role may perform any on-line marketing actions, since it has all the necessary permissions and UI elements pre-assigned.

You can find permissions for the following modules described in this chapter:

- Web analytics
- Contact management
- Scoring
- A/B and MVT testing
- Content personalization
- E-mail marketing
- On-line marketing

12.2 Permissions

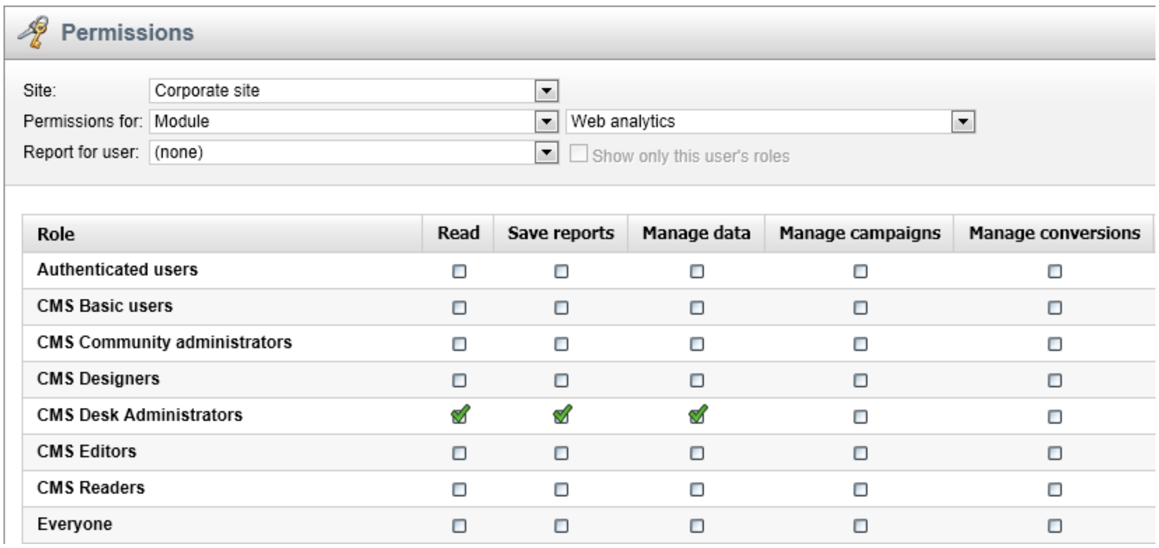
Setting up the authorization options for on-line marketing features can be done by assigning the appropriate permissions to your site's roles. This can be done in the **Site Manager/CMS Desk -> Administration -> Permissions** interface. First, you need to select the appropriate *Module* and then fill in its permission matrix as needed.

Web analytics (including conversions and campaigns)

The following permissions can be given for the *Web analytics* module:

- **Read** - allows members of the specified roles to view web analytics reports and other parts of the *CMS Desk -> Tools -> Web analytics* interface. It is also required to access the analytics reports anywhere else in the UI, e.g. in *CMS Desk -> Content -> Analytics -> Reports*.

- **Save reports** - allows members of the specified roles to save web analytics reports. The saved reports can be viewed in the *CMS Desk -> Tools -> Reporting* interface.
- **Manage data** - allows members of the specified roles to manage the data logged for various statistics (i.e. delete or generate sample data for statistics).
- **Manage campaigns** - allows members of the specified roles to create or delete campaign tracking objects and edit their properties, including goals.
- **Manage conversions** - allows members of the specified roles to create or delete conversions and edit their properties.



Contact management

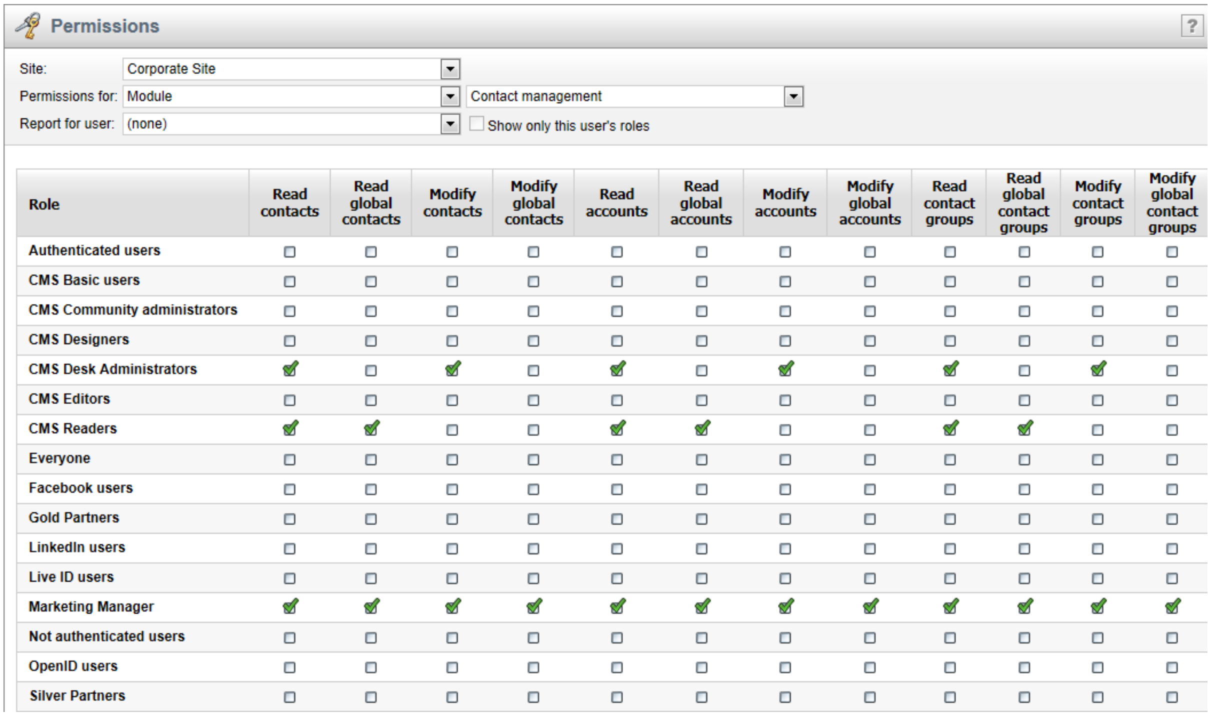
There are four basic types of permissions for the *Contact management* module:

- **Read** - allows members of the specified roles to access the appropriate part of the on-line marketing interface and view any data there.
- **Read global** - allows members of the specified roles to view global objects of the appropriate type. Please note that global data is only available in CMS Desk if this is allowed by the site's settings (*Site Manager -> Settings -> On-line marketing -> Contact management -> Global data & merging*).
- **Modify** - allows members of the specified roles to create, edit or delete objects of the given type.
- **Modify global** - allows members of the specified roles to create, edit or delete global objects of the given type in CMS Desk.

Specific permissions of these types are then available for the following contact management objects:

- **Contacts** - these permissions allow users to view or manage contacts and their settings.
- **Accounts** - affect the viewing or management of accounts.
- **Contact groups** - affect the viewing or management of contact groups.
- **Configuration** - these permissions affect the viewing or management of account statuses, contact statuses and contact roles.
- **Activities** - allow users to view or manage the activity log in CMS Desk. There are no global permissions for activities, since activities are always logged for a specific site.

Please note that the *Read global* and *Modify global* type permissions may only be configured by global administrators.



Permissions

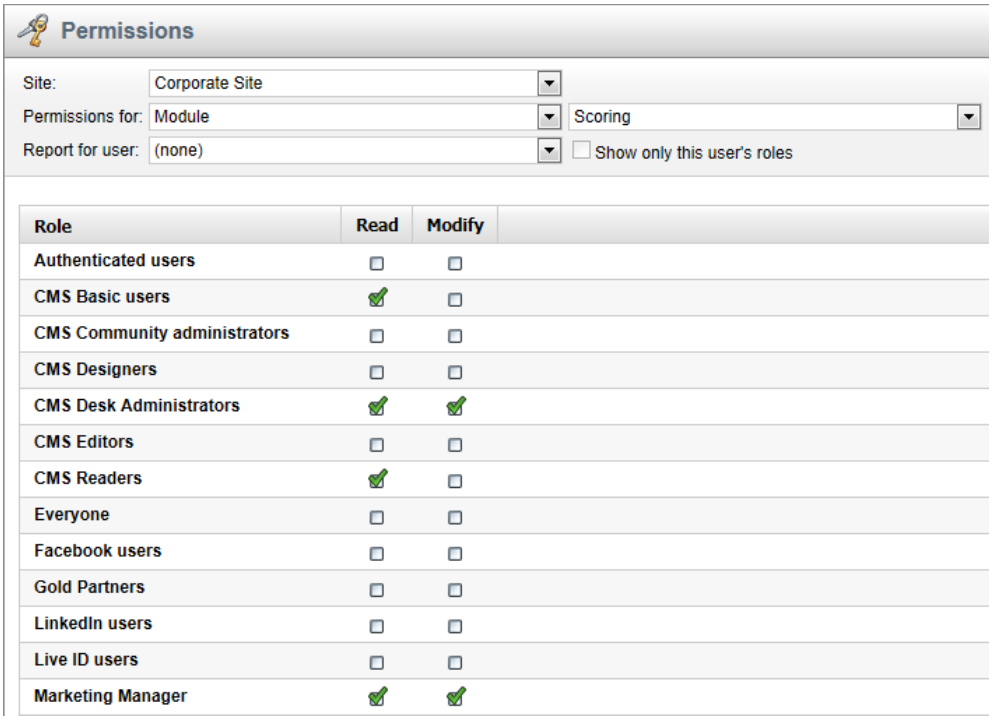
Site: Corporate Site
Permissions for: Module | Contact management
Report for user: (none) | Show only this user's roles

Role	Read contacts	Read global contacts	Modify contacts	Modify global contacts	Read accounts	Read global accounts	Modify accounts	Modify global accounts	Read contact groups	Read global contact groups	Modify contact groups	Modify global contact groups
Authenticated users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
CMS Basic users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
CMS Community administrators	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
CMS Designers	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
CMS Desk Administrators	✓	☐	✓	☐	✓	☐	✓	☐	✓	☐	✓	☐
CMS Editors	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
CMS Readers	✓	✓	☐	☐	✓	✓	☐	☐	✓	✓	☐	☐
Everyone	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Facebook users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Gold Partners	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
LinkedIn users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Live ID users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Marketing Manager	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Not authenticated users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
OpenID users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Silver Partners	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Scoring

The security options of scoring are determined by the permissions of the *Scoring* module. The following can be assigned:

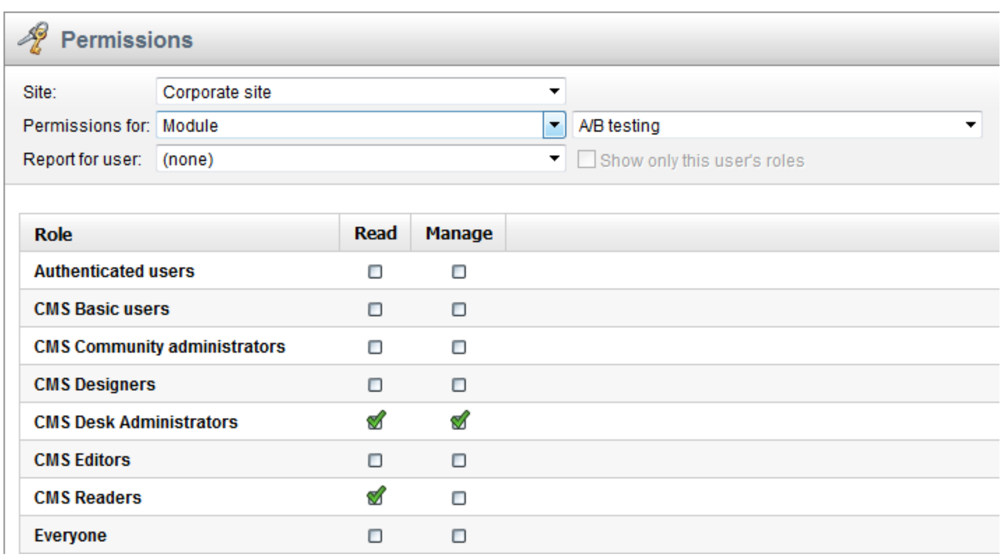
- **Read** - allows members of the selected roles to view the settings of scores and their rules, as well as the score points of individual contacts. This applies to the *CMS Desk -> On-line marketing -> Scoring* interface.
- **Manage** - allows members of the selected roles to create, edit or delete scores and their rules. This also includes the recalculate action.



A/B and MVT testing

To configure the permissions for optimization testing, select the *A/B testing* or *MVT testing* module respectively. The following two permissions can be assigned for either one of the modules:

- **Read** - allows members of the selected roles to view all parts of the A/B or MVT testing management interface and the corresponding reports.
- **Manage** - allows members of the selected roles to create, edit and delete tests and manage their variants. In this case of A/B testing, this means page variants. For MVT testing, this affects management of object (web part, zone or widget) variants on pages that have a test defined.



Additionally, permissions for the *Web analytics* module (as described above) are required to access A/B or MVT testing reports in the web analytics interface and in **CMS Desk -> Content -> Analytics -> A/B tests / MVT tests -> Reports**.

Editing A/B testing page variant documents

Because every page variant is represented by a document in the content tree, the standard document permissions apply.

All permissions that can be configured for the *Content* module are checked, such as for creating, modifying or deleting documents. Also, the **Design web site** permission for the *Design* module is needed to edit page variants on the **Design** tab of **CMS Desk.**

Managing MVT object variants

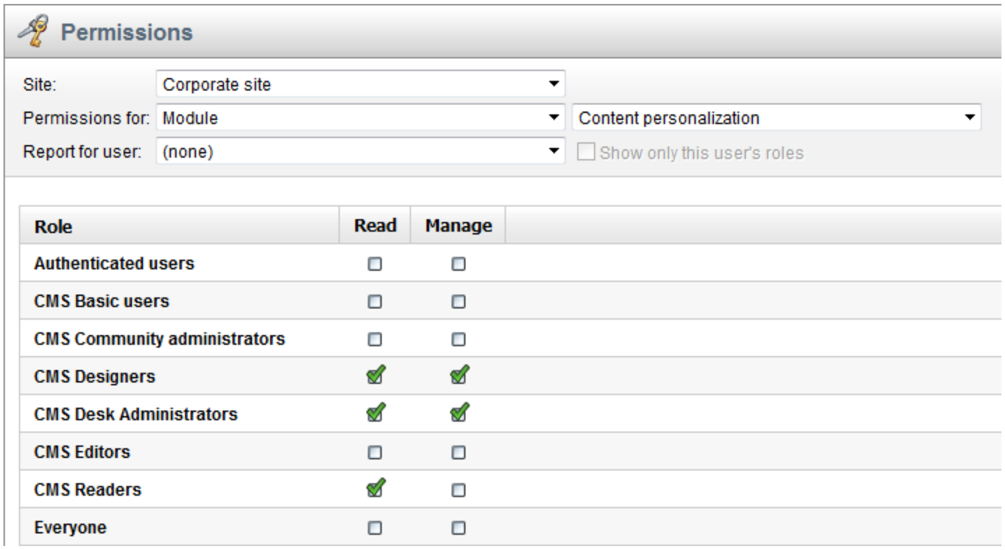
The **Design web site** permission for the *Design* module is also needed for users to be able to manage the variants of web parts and zones on the **Design** tab of **CMS Desk**.

To work with variants of editor widgets on the **Page** tab, the **Modify** permission for the *Content* module is required. Also, the security settings defined for specific widgets are checked, as described in the Development -> Widgets -> Security chapter of the Developer's Guide.

Content personalization

To configure the permissions for content personalization, select the *Content personalization* module. The following can be assigned:

- **Read** - allows members of the selected roles to view the content of personalization variants, their properties and variant lists in the *CMS Desk* administration interface. No special permissions are required to view personalized content on the live site.
- **Manage** - allows members of the selected roles to create, edit and delete personalization variants of objects.

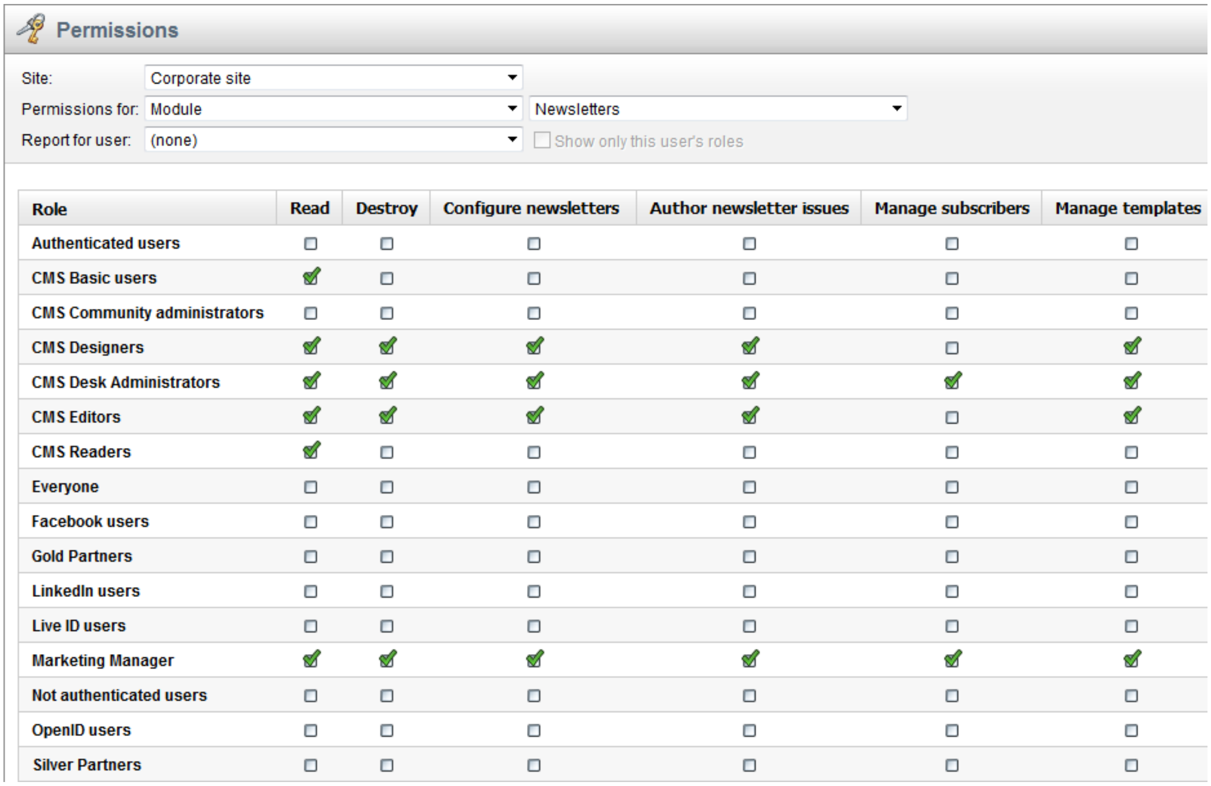


Additionally, the **Design web site** permission for the *Design* module is needed for users to be able to manage the variants of web parts and zones on the **Design** tab of **CMS Desk**. To work with variants of editor widgets on the **Page** tab, the **Modify** permission for the *Content* module is required. Also, the security settings defined for specific widgets are checked.

E-mail marketing

E-mail marketing actions require the appropriate permissions for the *Newsletters* module:

- **Read** - members of the selected roles are allowed to view all data in the *CMS Desk -> Tools -> Newsletters* interface.
- **Destroy** - allows members of the role to delete the version history of newsletter objects.
- **Configure newsletters** - members of the selected roles are allowed to configure newsletter settings.
- **Author newsletter issues** - members of the selected roles are allowed to create and edit newsletter issues.
- **Manage subscribers** - members of the selected roles are allowed to add and remove newsletter subscribers.
- **Manage templates** - members of the selected roles are allowed to create, edit and delete newsletter templates.


Permissions

Site: Corporate site
Permissions for: Module | Newsletters
Report for user: (none) | Show only this user's roles

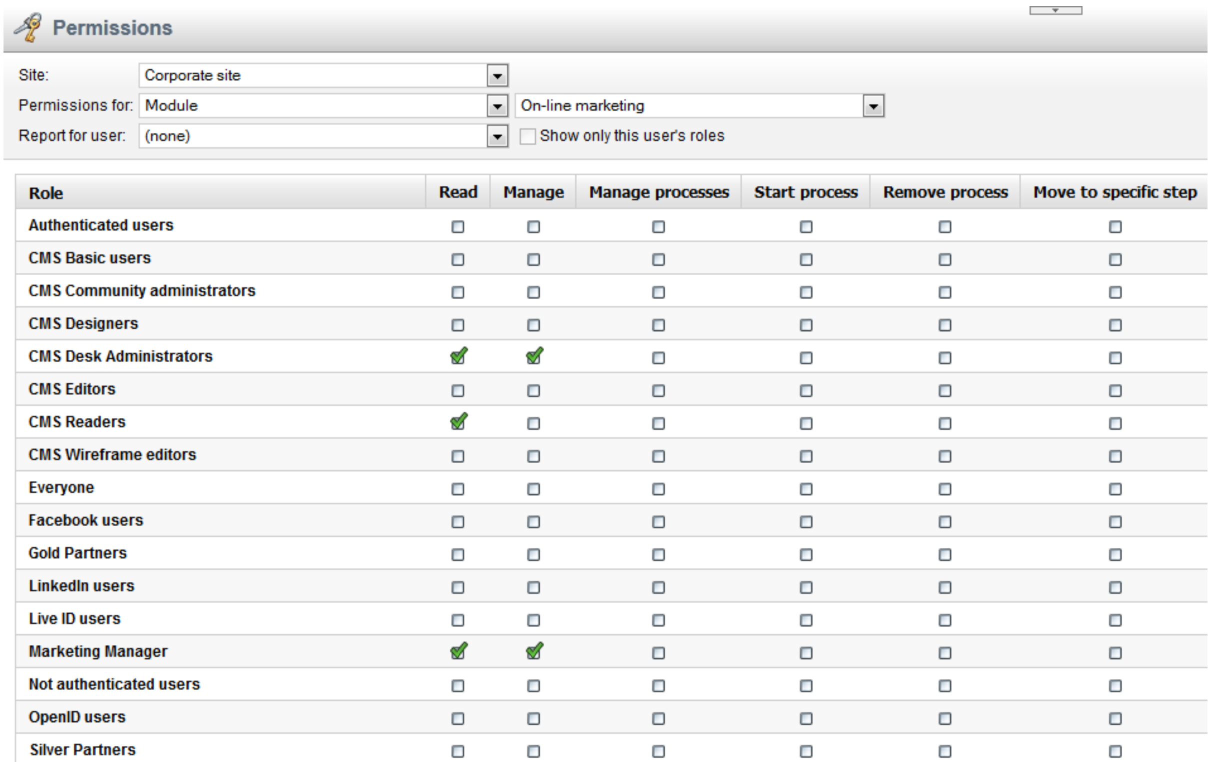
Role	Read	Destroy	Configure newsletters	Author newsletter issues	Manage subscribers	Manage templates
Authenticated users	☐	☐	☐	☐	☐	☐
CMS Basic users	✓	☐	☐	☐	☐	☐
CMS Community administrators	☐	☐	☐	☐	☐	☐
CMS Designers	✓	✓	✓	✓	☐	✓
CMS Desk Administrators	✓	✓	✓	✓	✓	✓
CMS Editors	✓	✓	✓	✓	☐	✓
CMS Readers	✓	☐	☐	☐	☐	☐
Everyone	☐	☐	☐	☐	☐	☐
Facebook users	☐	☐	☐	☐	☐	☐
Gold Partners	☐	☐	☐	☐	☐	☐
LinkedIn users	☐	☐	☐	☐	☐	☐
Live ID users	☐	☐	☐	☐	☐	☐
Marketing Manager	✓	✓	✓	✓	✓	✓
Not authenticated users	☐	☐	☐	☐	☐	☐
OpenID users	☐	☐	☐	☐	☐	☐
Silver Partners	☐	☐	☐	☐	☐	☐

Additionally, the **Read contact groups** permission for the *Contact management* module is required to be able to subscribe contact groups to a newsletter.

On-line marketing

You can modify the following permissions for the On-line marketing module:

- **Read** - members of the selected roles are allowed to view all data in the *CMS Desk -> On-line marketing*.
- **Manage** - members of the selected roles are allowed to manage the data in the *CMS Desk -> On-line marketing*.
- **Manage process** - members of the selected roles are allowed to manage automation processes. Except for *running processes*, *removing processes* and *moving to specific step*. They can, however, move process instances to previous and next steps regardless of step security settings.
- **Start process** - members of the selected roles are allowed to start automation processes for contacts and contact groups.
- **Remove process** - members of the selected roles are allowed to remove contacts from instances of automation processes.
- **Move to specific step** - members of the selected roles are allowed to move an instance of a process to a specific step regardless of step security settings.



Role	Read	Manage	Manage processes	Start process	Remove process	Move to specific step
Authenticated users	☐	☐	☐	☐	☐	☐
CMS Basic users	☐	☐	☐	☐	☐	☐
CMS Community administrators	☐	☐	☐	☐	☐	☐
CMS Designers	☐	☐	☐	☐	☐	☐
CMS Desk Administrators	✓	✓	☐	☐	☐	☐
CMS Editors	☐	☐	☐	☐	☐	☐
CMS Readers	✓	☐	☐	☐	☐	☐
CMS Wireframe editors	☐	☐	☐	☐	☐	☐
Everyone	☐	☐	☐	☐	☐	☐
Facebook users	☐	☐	☐	☐	☐	☐
Gold Partners	☐	☐	☐	☐	☐	☐
LinkedIn users	☐	☐	☐	☐	☐	☐
Live ID users	☐	☐	☐	☐	☐	☐
Marketing Manager	✓	✓	☐	☐	☐	☐
Not authenticated users	☐	☐	☐	☐	☐	☐
OpenID users	☐	☐	☐	☐	☐	☐
Silver Partners	☐	☐	☐	☐	☐	☐

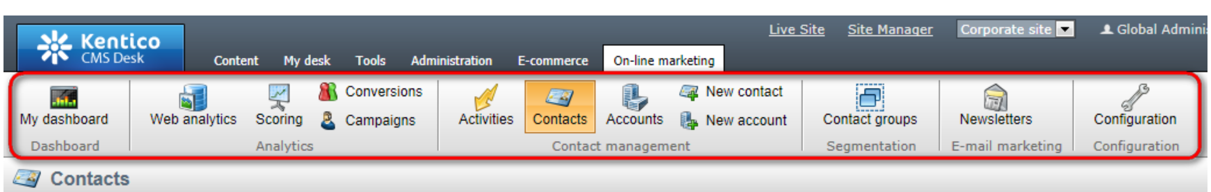
Note that for a user to be able to modify the contacts within an automation process, you need to grant him a **Read contacts** permission for the *Contact management* module as well. Use **manage process** permission to allow user to modify whole process, including moving contacts from step to step regardless of step security settings.

12.3 UI personalization

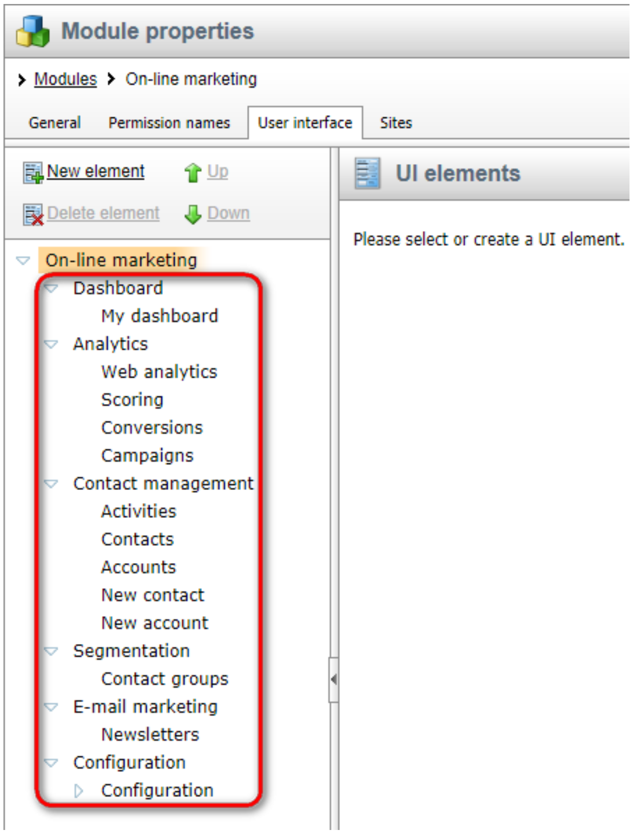
UI personalization can be applied to **CMS Desk only**. Site Manager cannot be personalized as it is typically used by administrators and developers who need the full rather than a simplified UI. Further in the topic you will thus learn how to personalize the UI of On-line marketing, specifically the **Ribbon** and Configuration tab, in CMS Desk.

Ribbon

You can hide individual items or entire item groups in the top ribbon menu of On-line marketing.

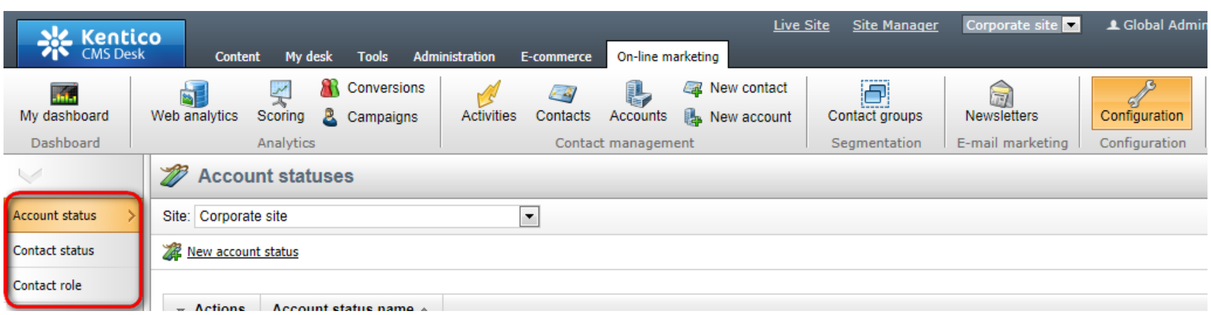


In the screenshot below, you can see the UI elements of the **Online marketing** module. Their names match the names of the ribbon menu items on the screenshot above.

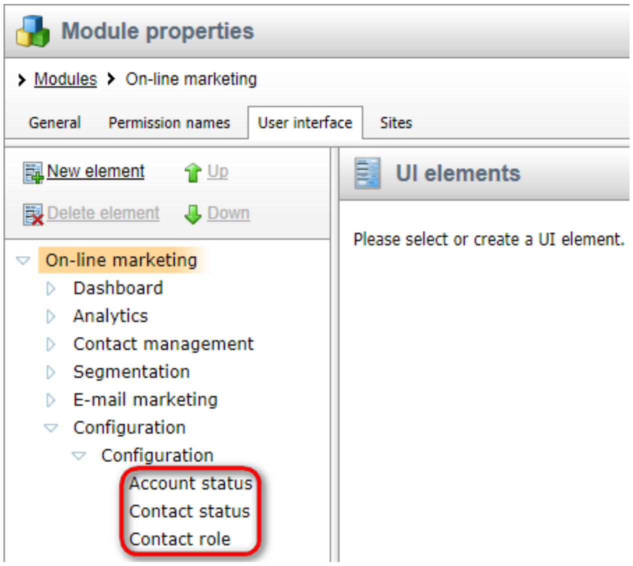


Configuration tab

You can also hide the tabs of the **Configuration** section of On-line marketing.



In the screenshot below, you can see the UI elements of the respective section. Their names match the names of the tabs on the screenshot above.



Detailed information on UI personalization in Kentico CMS can be found in the Membership, permissions and security -> UI personalization chapter in the Development section of the Developer's Guide.

Part XIII

Integrating 3rd party modules

13 Integrating 3rd party modules

13.1 Data.com

13.1.1 Overview

Important!

If you wish to use the Data.com integration, you need to apply hotfix 7.0.44 or newer.

Data.com integration can help keep your Kentico CMS contacts and accounts up-to-date. Data.com allows you to access a community-managed business directory that is updated on a daily basis. You can look up your current contacts and match them with Data.com contacts. In the same way, you can look up the accounts that you have in Kentico CMS and compare them with Data.com company profiles.

The system provides two ways to update your on-line marketing data using Data.com:

- **Manually** - users can look for matching contacts or company profiles directly in the contact or account editing interface.
- **Automatically** - you can set up marketing automation processes that automatically update your contacts according to matching Data.com contacts. You only need to add the *Update from Data.com* action step to your automation processes.

By combining these two methods, you can ensure that your contact and account data is up-to-date and accurate.

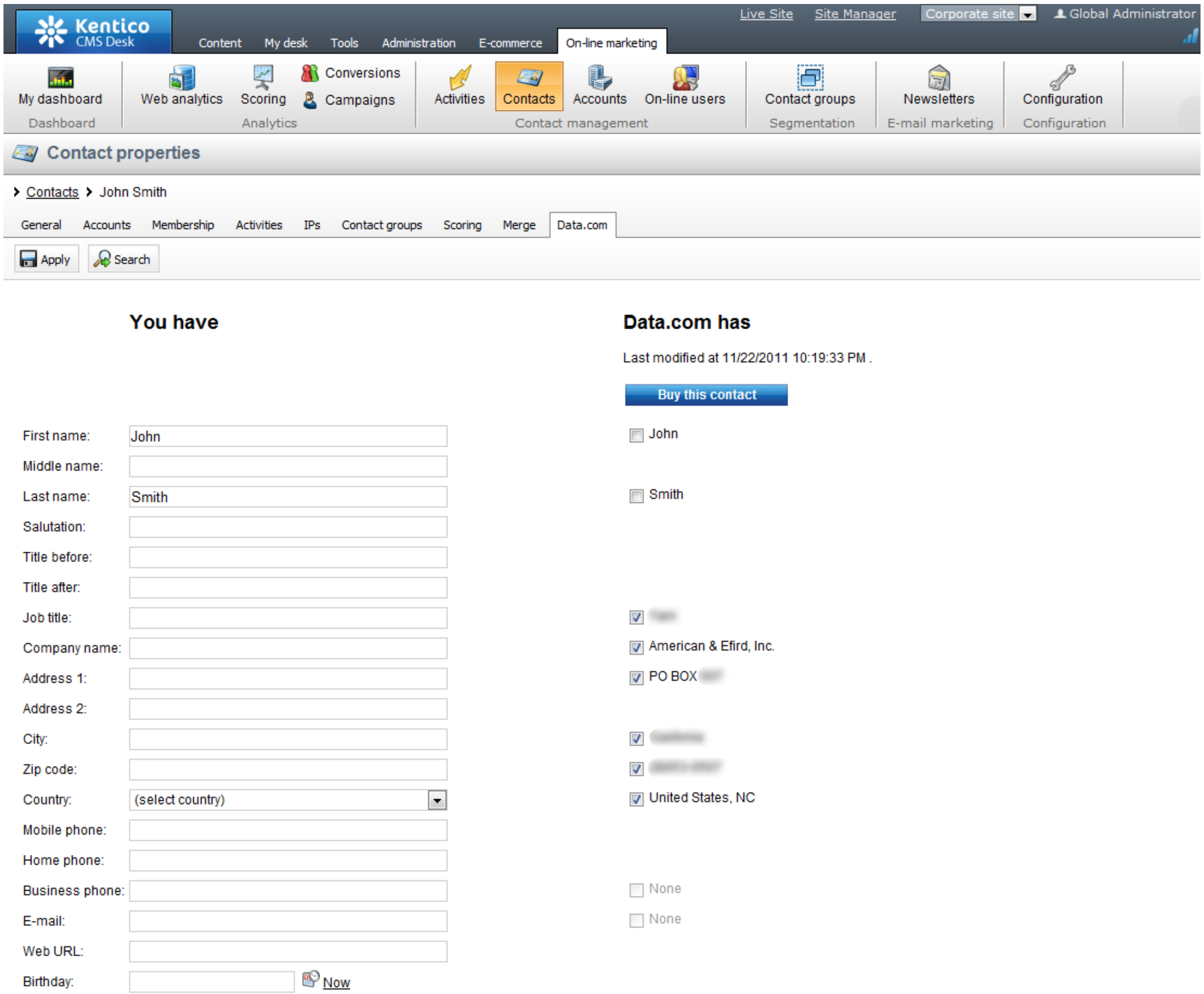
Data.com authentication

Kentico EMS requires a valid Data.com account to communicate with the Data.com service. The system requests Data.com authentication credentials for:

- **Users** - users need to enter the credentials of a Data.com account when accessing the Data.com interface for the first time. The system stores the credentials in the settings of the user account, so the duration of the Data.com authentication is unlimited.
- **Automation steps** - each **Update from Data.com** step in an automation process uses its own Data.com account. You can fill in the credentials when creating or editing the steps.

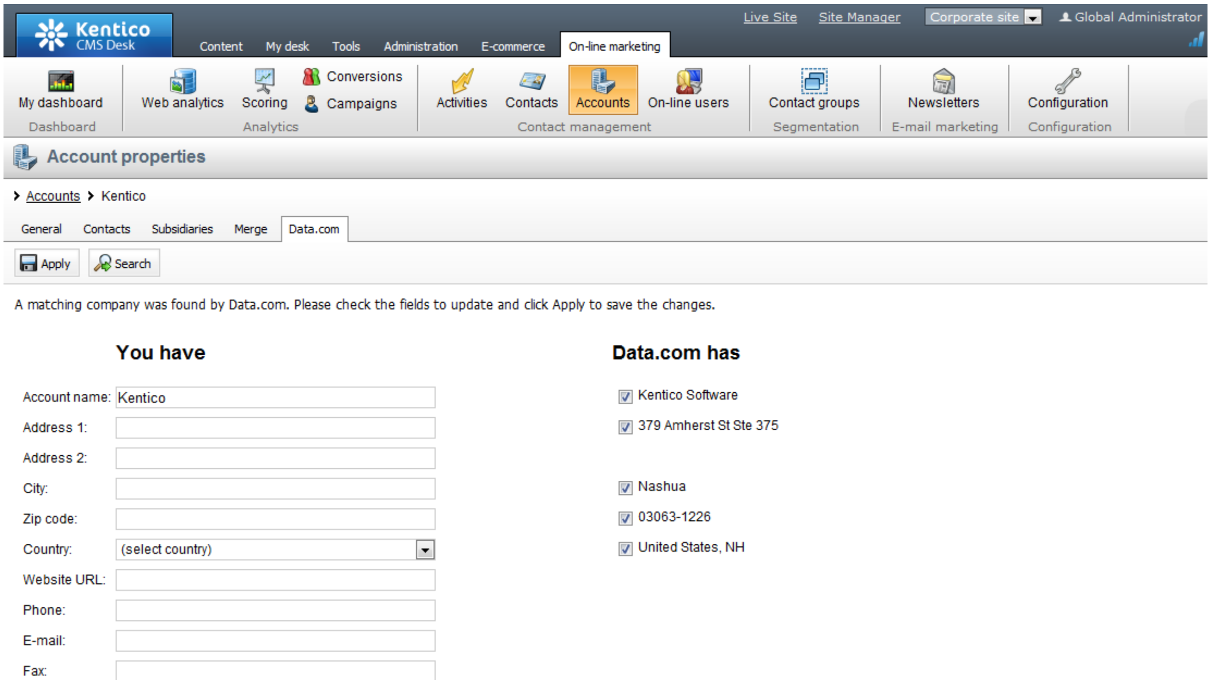
To search and compare contacts, go to one of the following locations in the user interface:

- **CMS Desk -> On-line marketing -> Contacts -> Edit () contact -> Data.com**
- **Site Manager -> Tools -> Contact Management -> Contacts -> Edit () contact -> Data.com**



To compare accounts, navigate to one of the following locations:

- **CMS Desk -> On-line marketing -> Accounts -> Edit ( ) account -> Data.com**
- **Site Manager -> Tools -> Contact Management -> Accounts -> Edit ( ) account -> Data.com**



13.1.2 Mapping Data.com fields

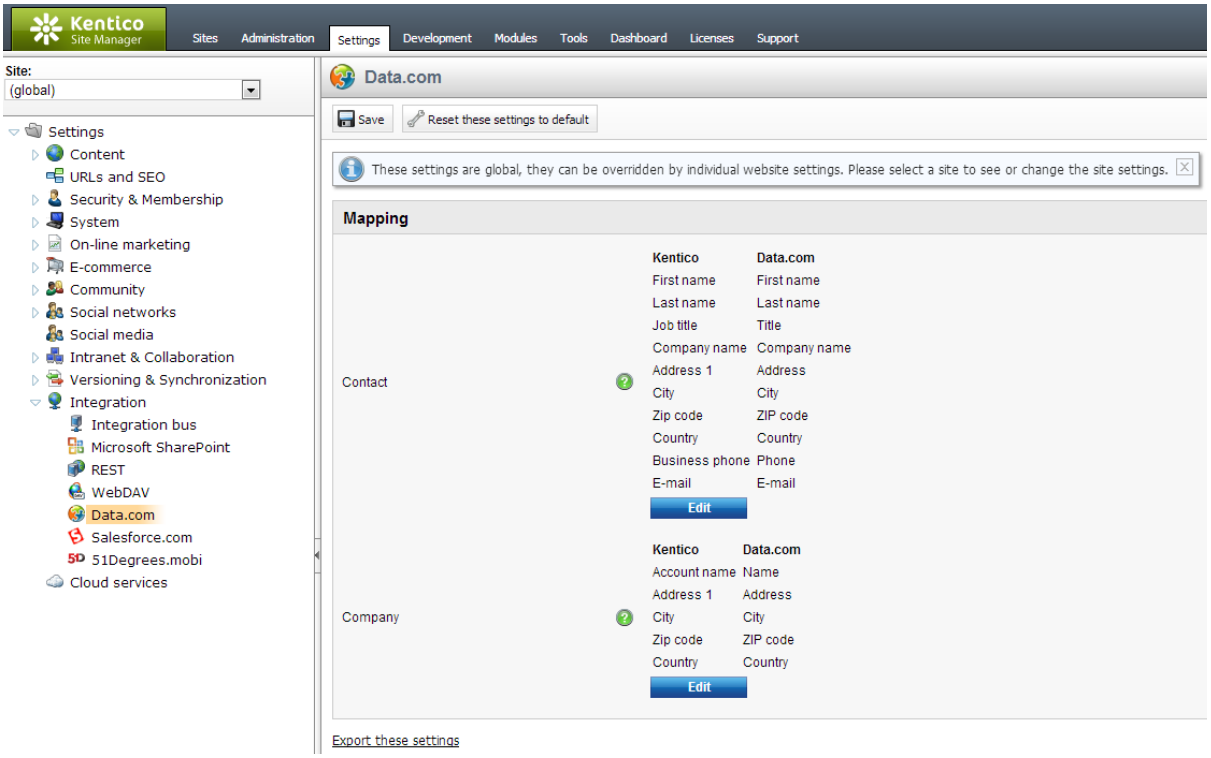
Important!

If you wish to use the Data.com integration, you need to apply hotfix 7.0.44 or newer.

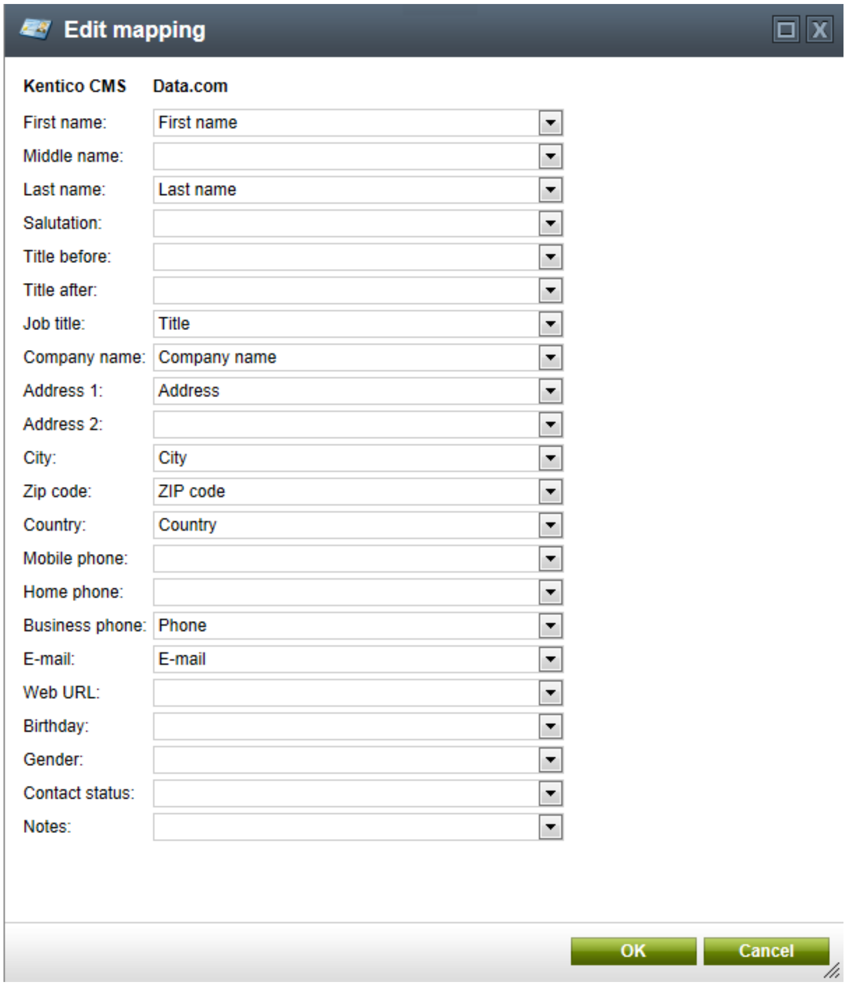
You can configure how the Data.com integration matches data fields between:

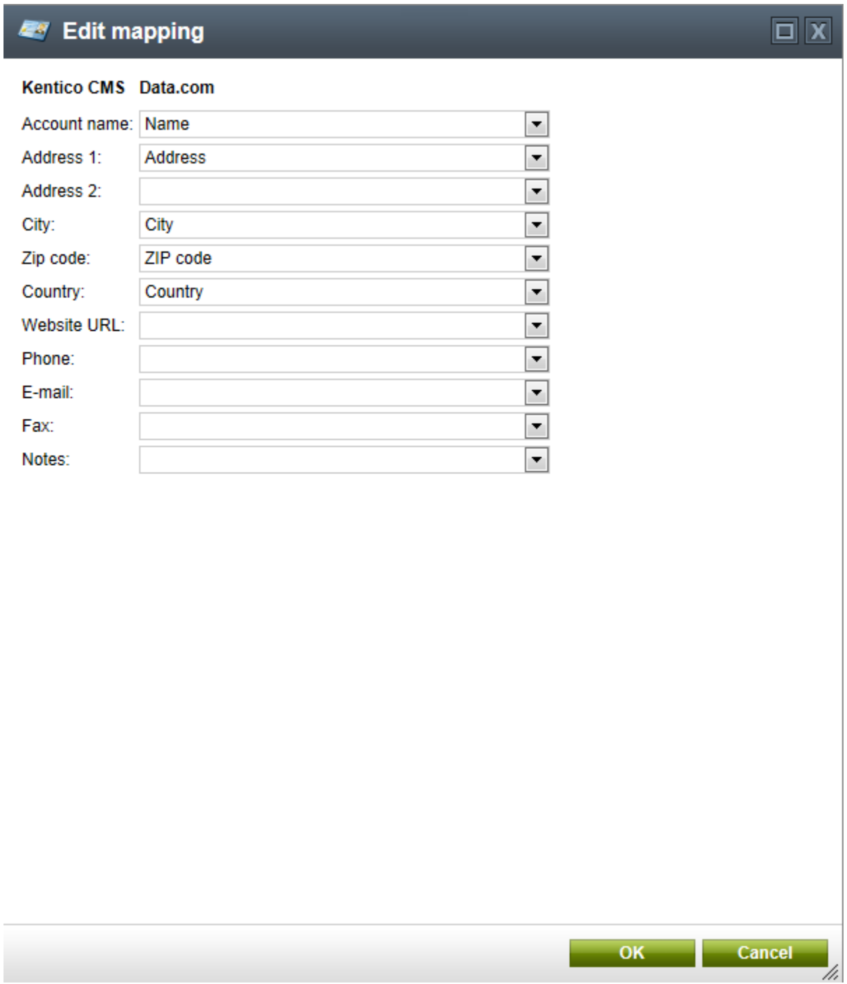
- Kentico CMS and Data.com contacts
- Accounts and Data.com company profiles

To access the mapping settings, go to **Site Manager -> Settings -> Integration -> Data.com**.



To modify the default field mappings, click **Edit** at the bottom of the **Contact** or **Company** sections.


Edit mapping
Kentico CMS
Data.com
First name:
First name
Middle name:
Last name:
Last name
Salutation:
Title before:
Title after:
Job title:
Title
Company name:
Company name
Address 1:
Address
Address 2:
City:
City
Zip code:
ZIP code
Country:
Country
Mobile phone:
Home phone:
Business phone:
Phone
E-mail:
E-mail
Web URL:
Birthday:
Gender:
Contact status:
Notes:
OK
Cancel



You can find more information about managing contacts in the Contact management chapter.

Once you have configured the Data.com integration, you can search for contacts or accounts.

13.1.3 Searching for contacts

Important!

If you wish to use the Data.com integration, you need to apply hotfix 7.0.44 or newer.

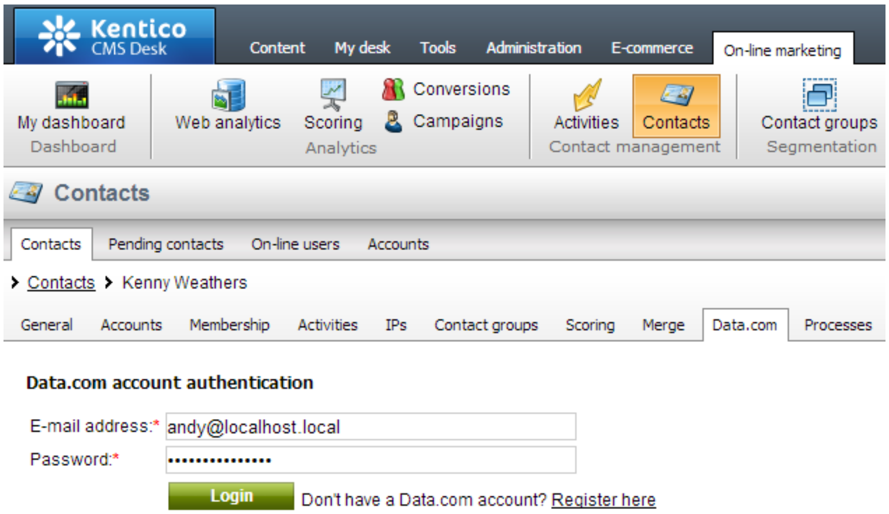
You can look up your Kentico CMS contacts in the Data.com business directory and update their information.

1. Go to **CMS Desk -> On-line marketing -> Contacts**.
2. Edit ( ) a specific contact.
3. Open the **Data.com** tab.

Entering the Data.com credentials

When opening the Data.com tab for the first time, you need to enter the credentials of a valid Data.com account. The system uses the account to communicate with the Data.com service.

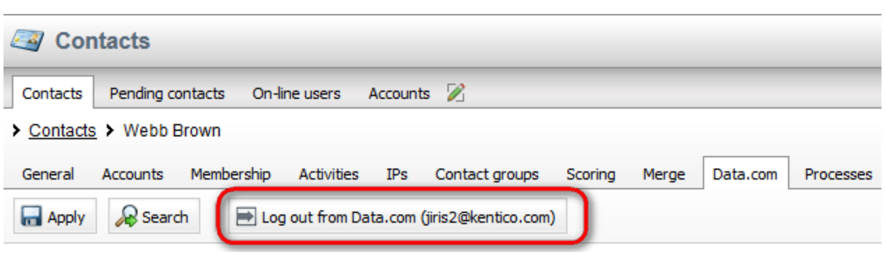
1. Fill in the **E-mail address** and **Password**.
2. Click **Login**.



The duration of the Data.com authentication is unlimited. The system stores the Data.com credentials in the settings of your user account (the password is encrypted). You can now access the **Data.com** tab without authenticating for both contacts and accounts.

If you need to change your Data.com account:

1. Open the **Data.com** tab (either in the contact or account interface).
2. Click **Log out from Data.com**.
3. Fill in the authentication form with the e-mail address and password of the new Data.com account.
4. Click **Login**.



Searching for contacts

Once your user account has valid Data.com credentials, the **Data.com** tab automatically searches the Data.com directory for the given contact. The integration tool makes its best effort to find the correct contact or at least narrow the search down as much as possible, saving you time compared with doing a manual search.

Note

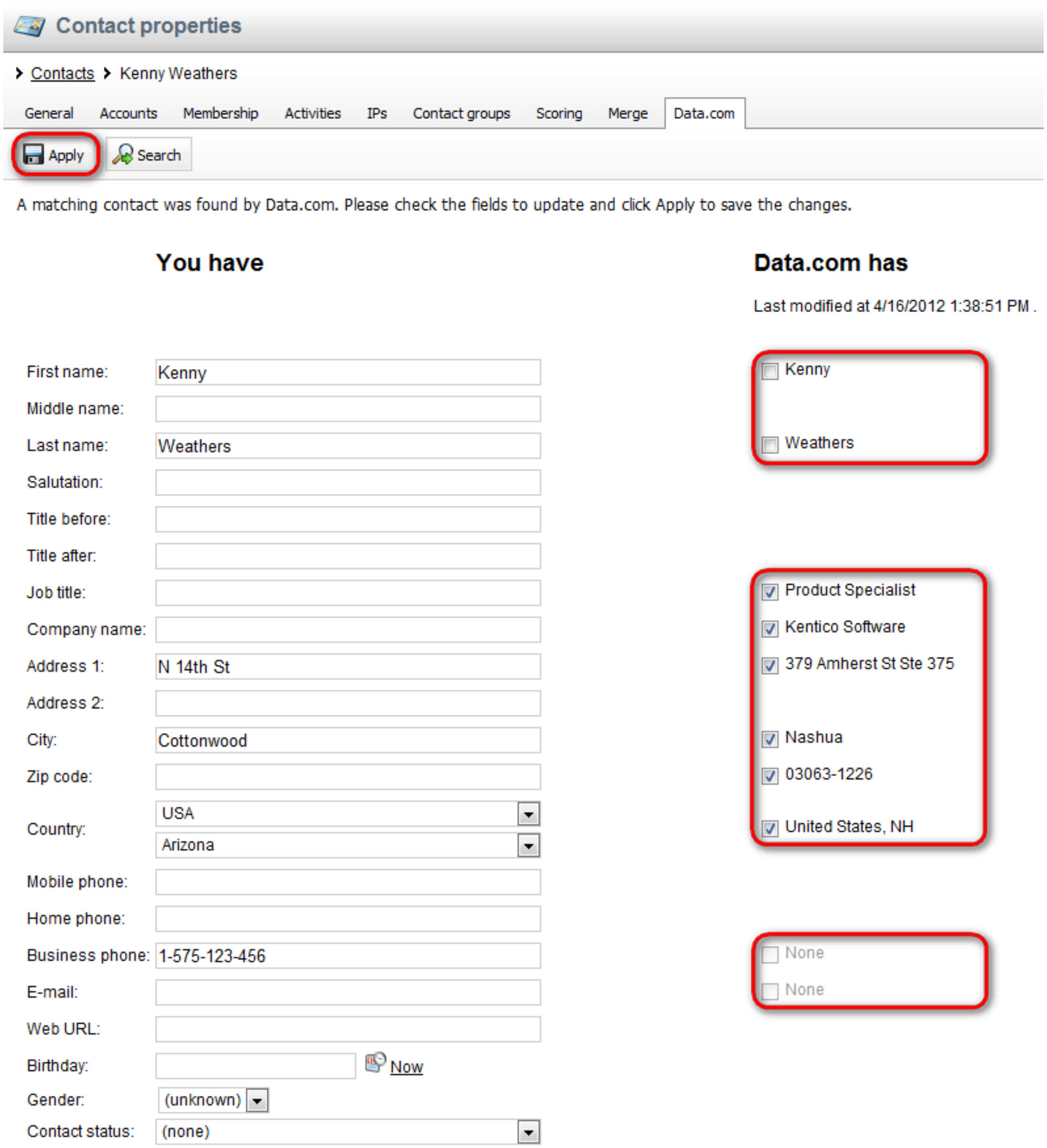
The Data.com search uses the following contact attributes:

- **First name**
- **Last name**
- **Company**
- **E-mail**

There are several possible results of the search.

An exact match is found for the contact

When the system finds an exact match for the selected contact, a comparison page opens. You can view the data of the Kentico CMS contact side-by-side with the information from the Data.com directory.



Note

If you can't see the **Business phone** and **E-mail** values of the found contact, then your Data.com account doesn't own the given contact. You can buy the contact to view the values.

Updating contact information

To update the contact's data using the Data.com information:

1. Check the boxes next to the fields that you want to update.
2. Click **Apply**.

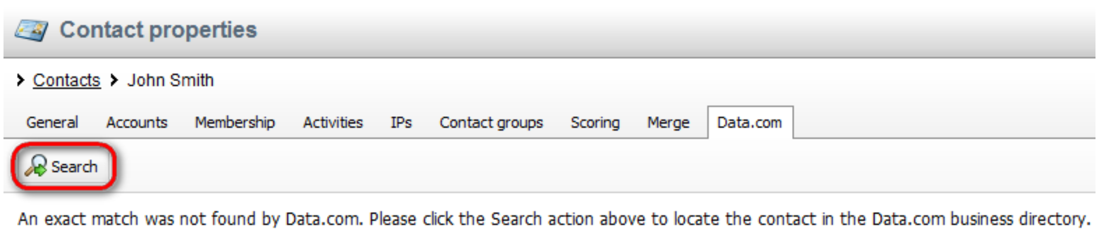
The updated data is immediately visible in the **You have** column. You can also switch to the **General** tab to view the contact's full information.

Too many matches are found

If the contact's search criteria are too general, the query can yield more than one result in the Data.com directory. For example, if you search for a contact with a commonly used name.

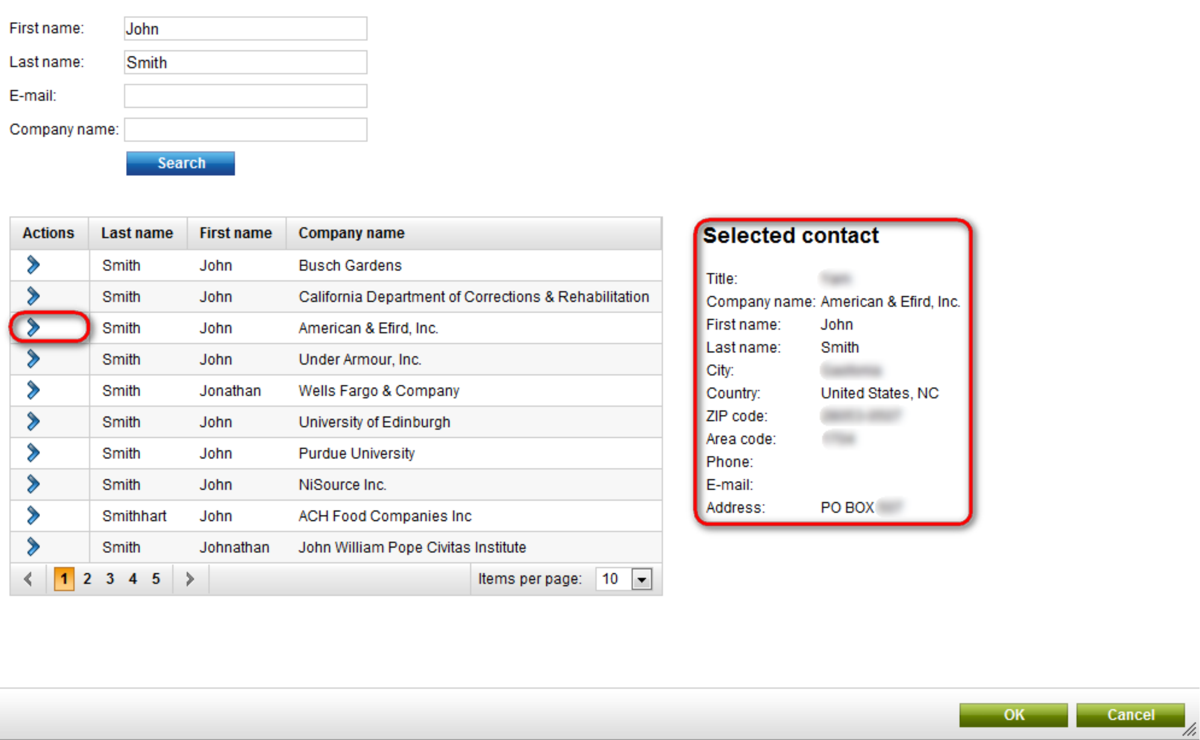
In these cases, you need to manually find the matching contact.

1. Click **Search**.



2. (Optional) Revise the search criteria in the filter dialog and click **Search**.

3. **Select** ( ) the listed contacts to view their information.



4. Click **OK** once you have selected the correct contact.

You can now update the contact's information.

No matches are found

If your contact does not exist in the Data.com directory, the search does not return any results. You can click **Search**, manually revise the search criteria and attempt to find the contact yourself.

13.1.4 Buying contacts

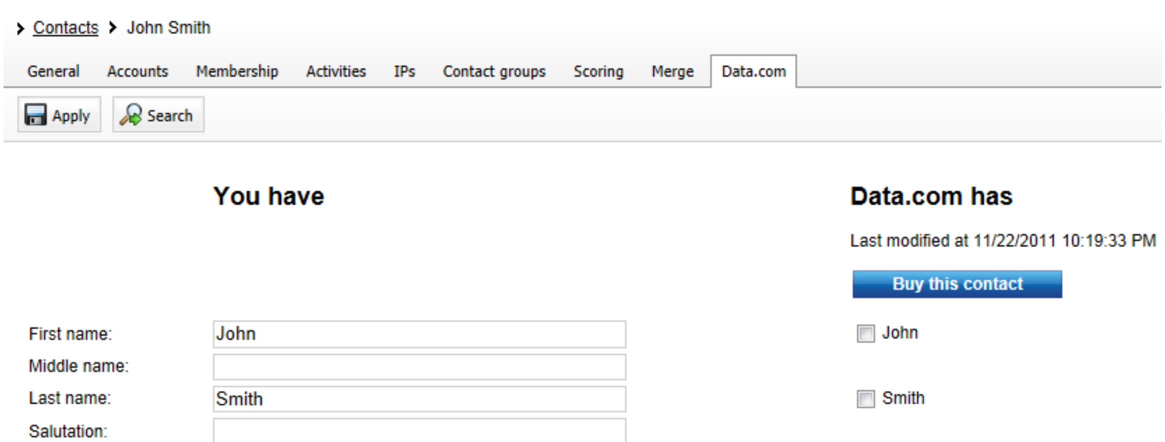
> **Important!**
>
> If you wish to use the Data.com integration, you need to apply hotfix 7.0.44 or newer.

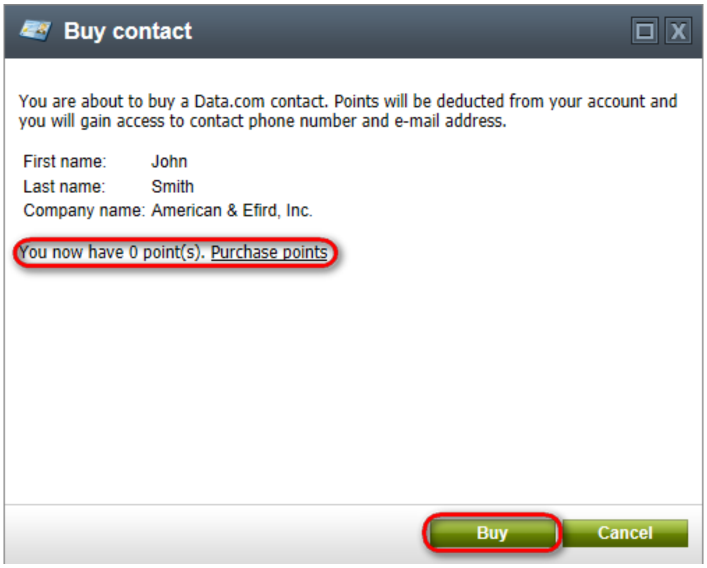
Data.com integration allows you to buy contacts from Data.com. If you own a Data.com contact, you can access additional information — the **Business phone** and **E-mail** fields.

The system always buys contacts using the Data.com account associated with your Kentico CMS user account — identified by the credentials that you entered when accessing the **Data.com** tab for the first time. See Searching for contacts for more information. When you buy a contact, it becomes available for all users who share the same Data.com account.

Use the following steps to buy contacts while searching for contact information on the **Data.com** tab:

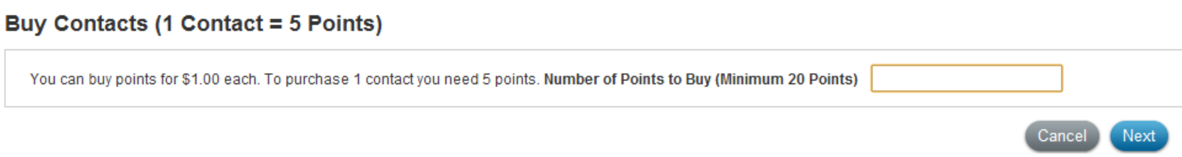
1. Click **Buy this contact**.
    - A confirmation dialog opens.





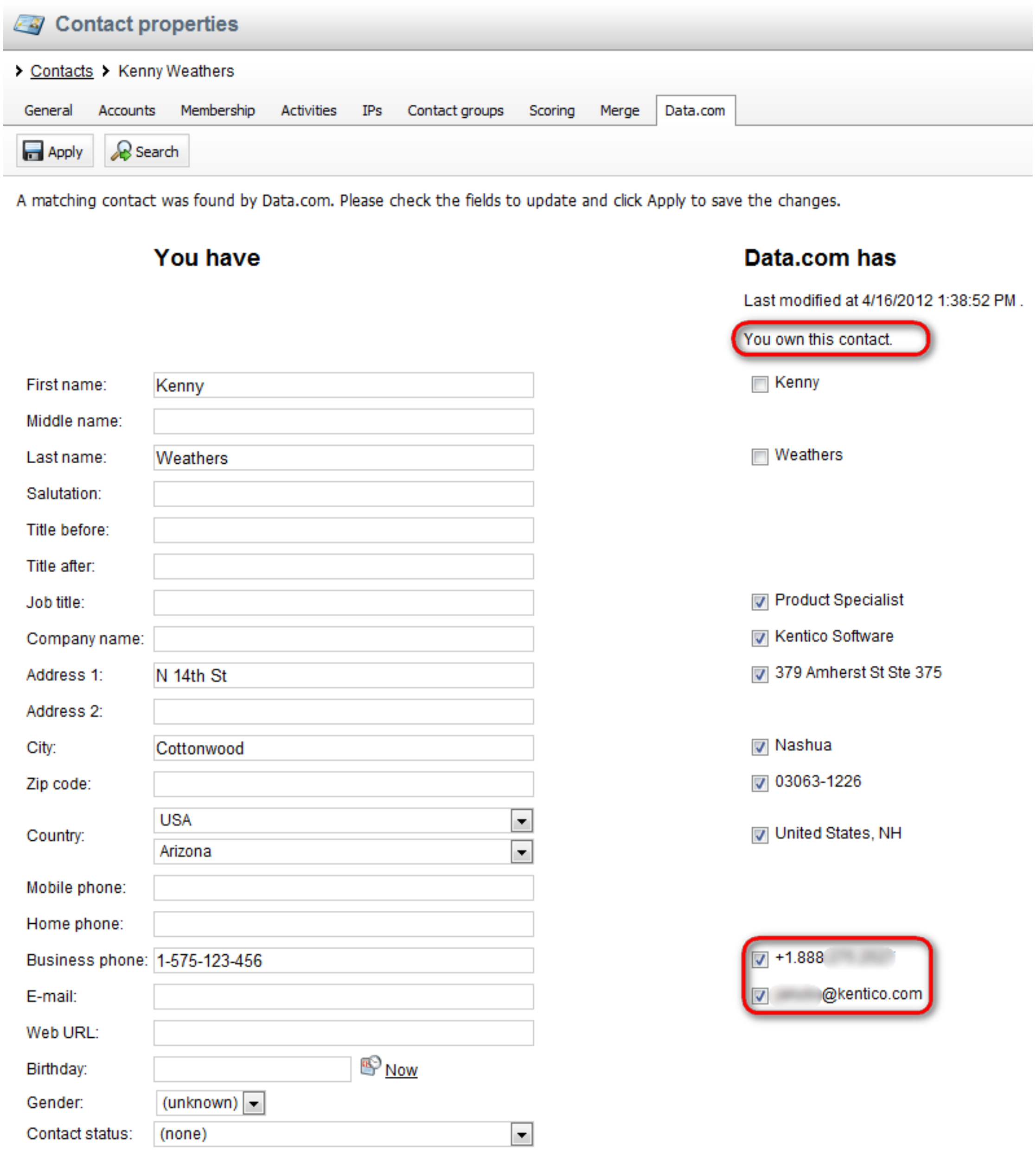
2. If you don't have enough points to **Buy** the contact right away:

a. Click Purchase points. The Data.com website opens.
b. Log in using your Data.com credentials.
c. Specify the amount of points you would like to purchase.



3. When you have enough points to purchase the contact, click **Buy** to complete the process.

Once you own a contact in the Data.com business directory, the **Business phone** and **E-mail** attribute values become visible in the **Data.com has** column.



Click **Apply** button to update the Kentico CMS contact with the values that are marked by the checkboxes.

13.1.5 Customizing the Data.com integration

If you wish to customize any aspect of the Data.com integration, you need to request your own API access token from Data.com.

Important!

Modifying or extending the default functionality without using your own Data.com token is a violation of the licensing terms.

Using a custom Data.com API token

Follow the steps below if you need to work with a custom Data.com access token in your API:

1. Open your web project in Visual Studio.

2. Create a new class:

- In the project's **App_Code** folder (or **Old_App_Code** if the project is installed as a web application) OR
- As part of a custom assembly (Class library)

3. Add a reference to the **CMS.DataCom** namespace:

[C#]

```
using CMS.DataCom;
```

4. Make the class implement the **ITokenProvider** interface.
5. Add the **GetToken()** method and return your Data.com token as a string:

[C#]

```
public class CustomDataComTokenProvider : ITokenProvider
{
    /// <summary>
    /// Gets the token used for Data.com communication.
    /// </summary>
    /// <returns>Token</returns>
    public string GetToken()
    {
        return "Your Data.Com API Token";
    }
}
```

Whenever you call the **DataComHelper.CreateClient** method in your custom code, add an instance of your **ITokenProvider** class as a parameter:

[C#]

```
using CMS.DataCom;

...
```

```
DataComClient client = DataComHelper.CreateClient(new CustomDataComTokenProvider
());
```

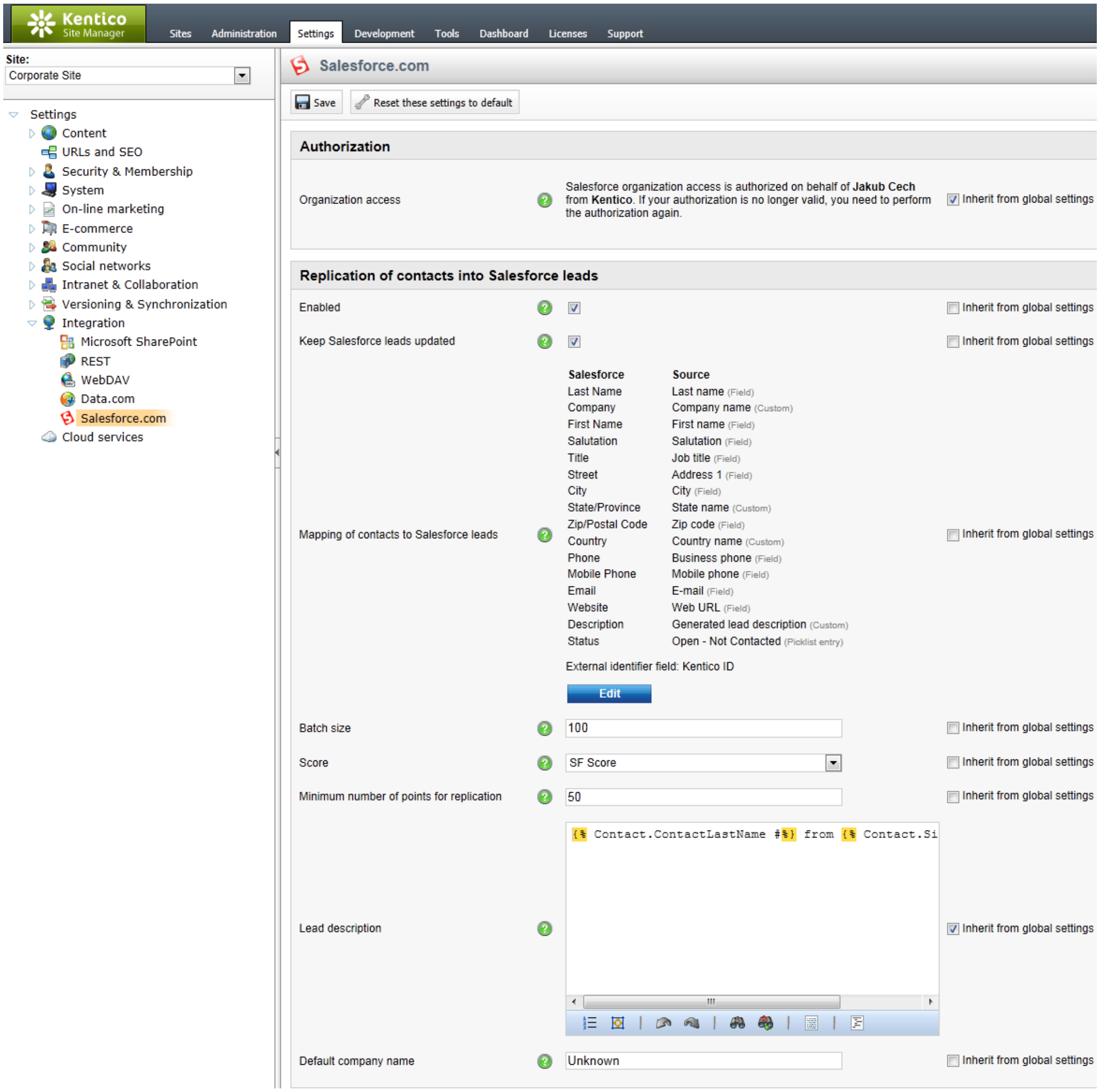
13.2 Salesforce

13.2.1 Overview

The Salesforce integration currently allows you to replicate Kentico CMS contacts into your Salesforce organization as leads. Other types of replication will be added in the future.

The system replicates contacts to Salesforce leads based on score requirements, which you can adjust according to your specific needs. You can then work with the leads within Salesforce itself.

You can configure the replication in **Site Manager -> Settings -> Integration -> Salesforce.com**. To use the replication, you have to link the Kentico CMS application with your Salesforce account. For more information, see Configuring Salesforce integration.



Salesforce integration settings

If you are interested in accessing and modifying your Salesforce organization's data using a programmatic interface implemented into Kentico CMS, refer to the Force.com integration API topic.

Differences between contact management in Kentico CMS and Salesforce

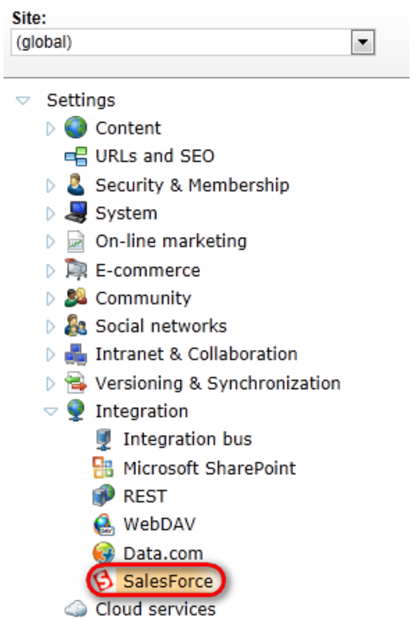
The following table briefly describes some of the differences between contacts in Kentico CMS and leads in Salesforce.

Action	Kentico CMS contacts	Salesforce leads
Deleting contacts	Can be deleted	Only marked for deletion
Merging contacts	Can be merged, use the target's ID	Can be merged

Splitting contacts	Can be split, retrieve their original ID	Can't be split

13.2.2 Configuring Salesforce integration

You can access the Salesforce integration configuration in **Site Manager -> Settings -> Integration -> Salesforce.com**. These settings allow you to authorize access to your Salesforce organization and modify how the system maps the attributes of Kentico CMS contacts to leads in Salesforce.

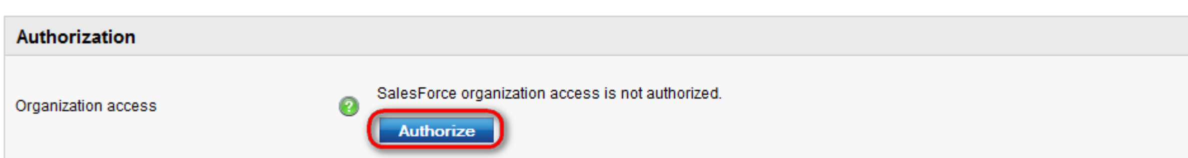


Authorizing access to Salesforce

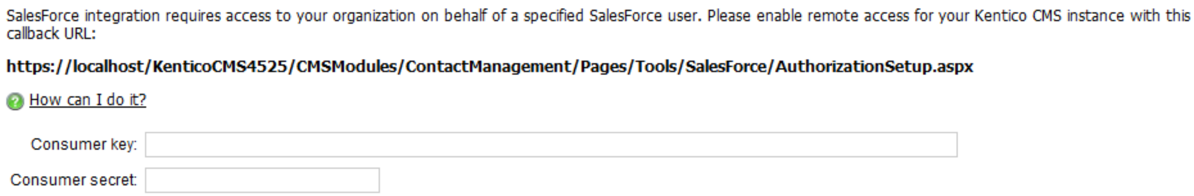
You need to authorize access of Kentico CMS to your organization on behalf of a specific Salesforce user. This process requires you to generate a **Consumer key** and **Consumer secret** in Salesforce itself.

> **Important!**
>
> To be able to configure the authorization, you need to have SSL set up on your web site in IIS. You can find the detailed steps described on the official IIS website.

1. Click **Authorize** next to the **Organization access** setting.

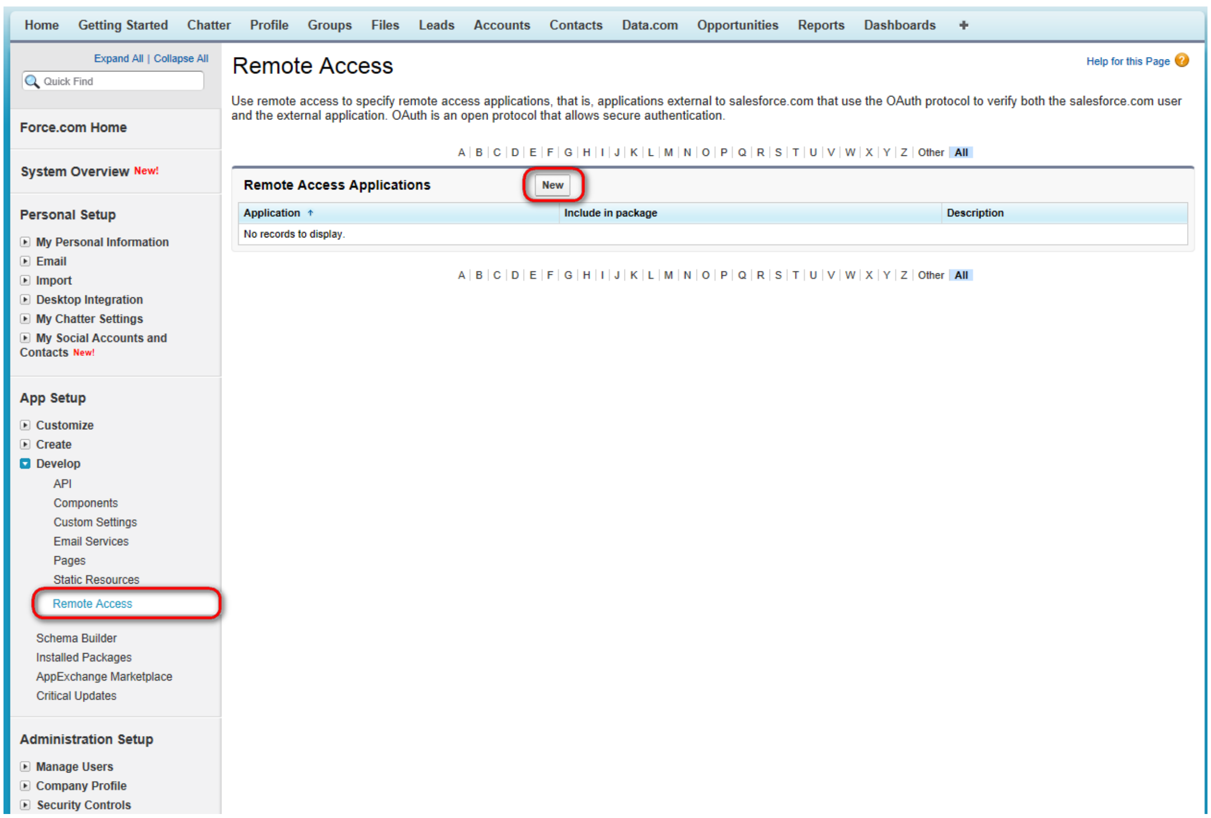


An authorization dialog opens.



2. Log on to Salesforce.com.

3. Create a new remote access application in **App Setup -> Develop -> Remote access**.

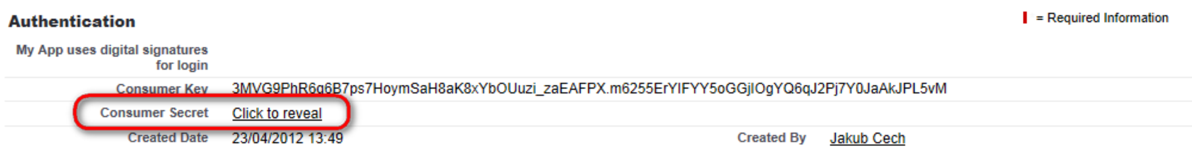


Creating a new remote access application in Salesforce

4. Enter the following values for the remote access application:

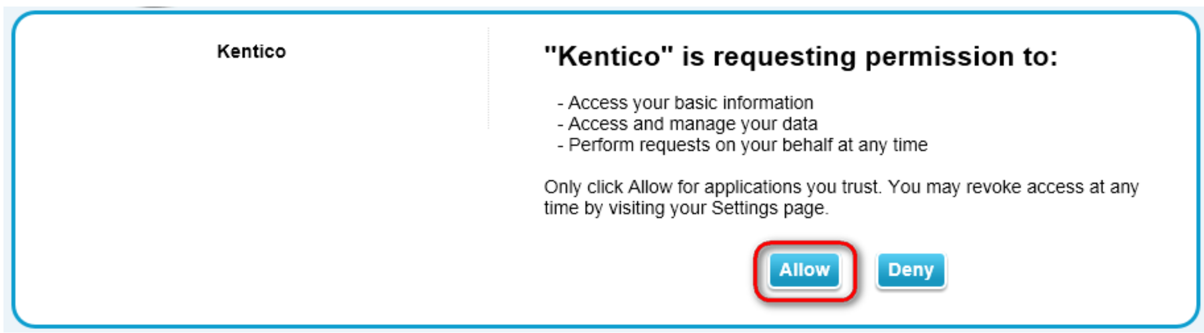
- **Application**: Type a name for the application. This name allows you to identify your application as the source of the request when authorizing access to your organization.
- **Callback URL**: Use the following format: *https://www.example.com/CMSModules/ContactManagement/Pages/Tools/SalesForce/AuthorizationSetup.aspx*
- **Use digital signatures**: Leave unchecked. The Salesforce integration does not use digital signatures for logging in.

5. Save the form and copy the **Consumer Key** and **Consumer Secret** provided by Salesforce for your remote access application.



6. Paste your consumer values into the corresponding fields in the Kentico CMS authorization dialog and click **Authorize**.
   - A Salesforce login screen opens in the current window.

7. Log in and click **Allow** to grant permission to your Kentico CMS application.

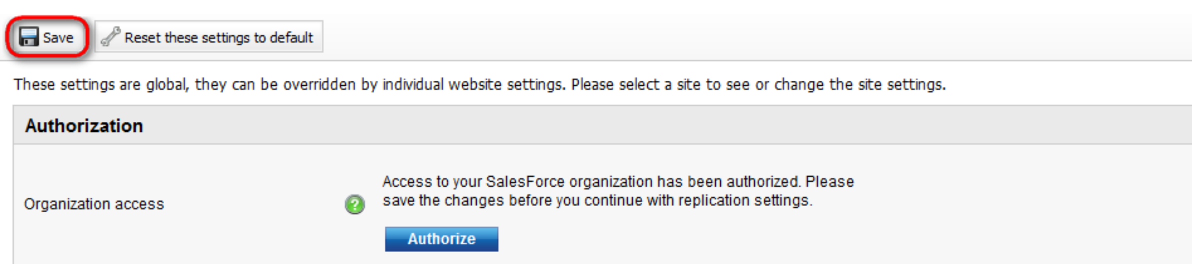


Security

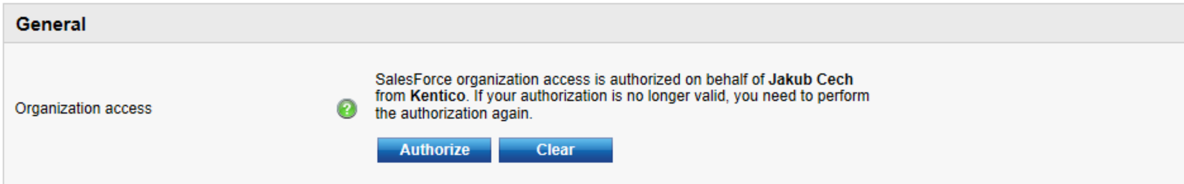
The system uses the OAuth 2.0 protocol to authorize access to Salesforce. Your Salesforce login and password are never disclosed to Kentico CMS. Access tokens are stored in encrypted format and the communication itself is encrypted using SSL.

8. **Save** the settings to finish the authorization process.

The window will close and you will be asked to **Save** the changes to finish the authorization process.



The **Organization access** section now shows the user and company name that the application uses to authorize access to Salesforce.



Successfully configured Organization access

Configuring the replication of contacts

Once the authorization is complete, you need to configure how the system replicates contacts into Salesforce leads. You can edit the settings globally or for specific websites.

Replication of contacts into Salesforce leads	
Enabled	Enables or disables replication of contacts into Salesforce leads.
Keep Salesforce leads updated	If checked, the replication process includes contacts that have already been replicated before. This ensures that the system updates the corresponding Salesforce leads based on the current contact data. If disabled, contacts are only replicated once.
Mapping of contacts to SalesForce leads	Determines how contact fields are mapped to the fields of Salesforce leads. See the Mapping contacts to Salesforce leads section below for details.
Batch size	Applications can only make a limited number of API calls to Salesforce within a 24 hour window. To minimize the number of calls, the replication process handles contacts in batches. Each batch only requires one API call. This setting specifies the maximum number of contacts that the system replicates in a single batch.
Score	Allows you to select the score that determines which contacts are replicated. The system only replicates contacts that reach a certain value in the given score (specified via the **Minimum number of points for replication** setting). If you do not choose a score (*None*), the system replicates all contacts. **Note**: The replication process is always performed separately for each website, so you can only select a score for individual sites, not globally. To learn more about scoring, refer to the Scoring chapter.
Minimum number of points for replication	Specifies the amount of points that contacts must reach in the score selected through the **Score** setting. Once a contact reaches this value, the system marks it for replication as a Salesforce lead.
Lead description	Defines a custom description for replicated contacts. To map the description to a specific field of Salesforce leads, select the **Generated**

	lead description source in the field mappings.
	
	You can insert the values of contact fields into the description using macro expressions.
	
	The default description adds the **Last name** of the contact that is being replicated and the name of the current site:
	
	`{% Contact.ContactLastName %} from {% CurrentSite.SiteName`
	
	For example, to use the **Business phone** value instead of the last name, enter the following expression:
	
	`{% Contact.ContactBusinessPhone %} from {% CurrentSite.Sit`

Default company name	All Salesforce leads require a company name value. This setting allows you to specify a default company name, which the replication process uses for contacts who are not associated with any company. The system attempts to retrieve the company name value from sources in Kentico CMS in the following order: 1. The value of the contact's **Company name** field 2. The name of the account in which the contact is listed as a *primary contact* 3. The account that lists the contact as a *secondary contact* 4. The first account that contains the contact as a regular contact 5. The value of the **Default company name** setting (if none of the above steps are successful)

Mapping contact fields to Salesforce lead fields

The replication process transfers data from Kentico CMS contacts to Salesforce leads based on field mapping settings.

Note: You need to authorize Organization access before you can adjust the field mappings.

Preparing the Salesforce mapping identifier field

Kentico CMS stores the bindings between Salesforce leads and contacts in a custom Salesforce field. You need to create a dedicated field for your Salesforce leads for this purpose.

1. Log on to Salesforce.com.

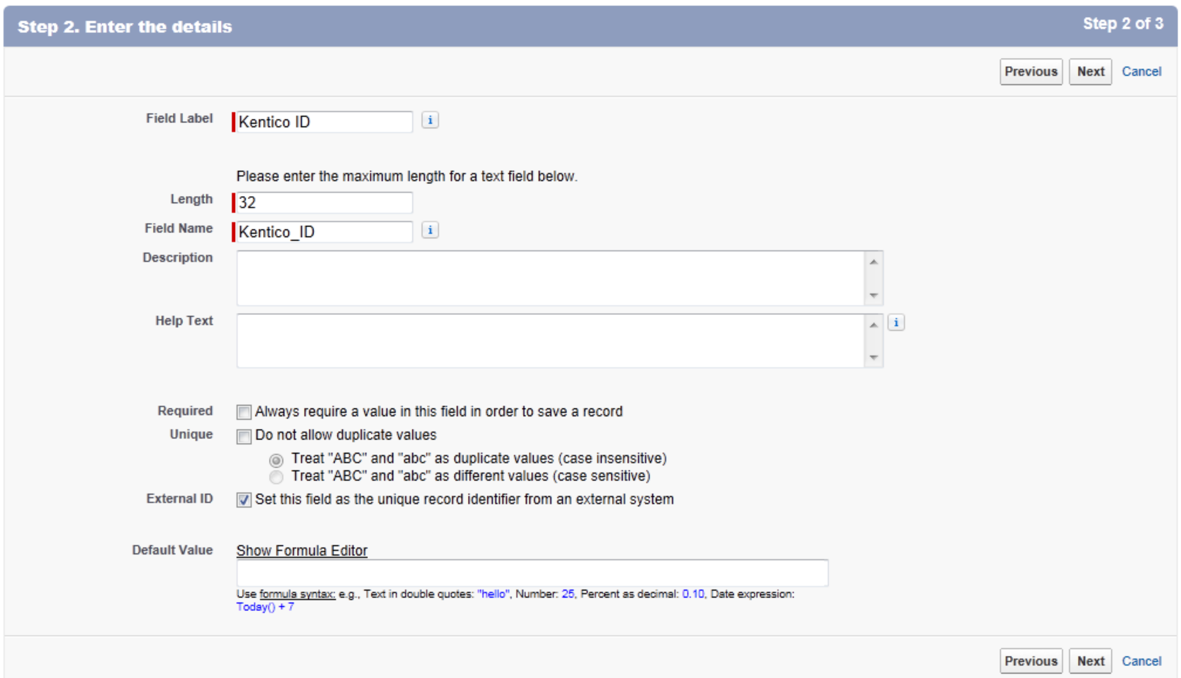
2. Navigate to **App Setup -> Customize -> Leads -> Fields**.

3. Click **New** in the **Lead Custom Fields & Relationships** section.

4. Choose **Text** as the field's data type and click **Next**.

5. In the **Enter the details** step, fill in the following values:

- **Field Label**: type any label for the field, for example: *Kentico ID*
- **Length**: 32
- **Field Name**: type any name for the field, for example: *Kentico_ID*
- **External ID**: check the box



Adding a custom field to Salesforce leads

6. Go to **Site Manager -> Settings -> Integration -> Salesforce.com** in Kentico CMS and click **Edit** below the **Mapping of contacts to Salesforce leads** section.

7. Select your new Salesforce lead field as the **External identifier field**.

8. Click **OK** and **Save** the settings.

You can now configure the mappings between the fields of Kentico CMS contacts and Salesforce leads.

Setting up the field mappings

To access the field mapping dialog, go to **Site Manager -> Settings -> Integration -> Salesforce.com** and click **Edit** below the **Mapping of contacts to Salesforce leads** section.

The mapping dialog offers a list of all standard and custom fields defined for your Salesforce leads. You can select sources for the fields from the following sections:

- **Field** - the fields of Kentico CMS contacts. Only fields containing relevant data are available (internal system fields of contacts are hidden).
- **Custom** - related values that cannot be loaded directly from contact fields, including the following:
  - *Company name* - ensures that each contact has a company name value. See the description of the Default company name setting for details about the process.
  - *Generated lead description* - loads the value of the website's *Lead description* setting.
  - *Country* - provides the name of the contact's country as a text value.
  - *State name* - provides the name of the contact's state as a text value.
- **Picklist entry** - allows you to select from the values predefined in Salesforce for picklist type fields.

You can only select contact fields that fit the data type of the target Salesforce field. The following table shows which types of Kentico CMS fields are supported by the available Salesforce data types:

Salesforce Data type	Supported Kentico CMS field types
Checkbox	Boolean (Yes/No)
Currency	*Replication does not support the Currency data type at this time.*
Date	Date and time
Date/Time	Date and time
Email	Text, Long text
Number	Decimal number, Integer number, Long integer number
Percent	Decimal number, Integer number, Long integer number
Phone	Text, Long text
Picklist	Text, Long text
Picklist (Multi-Select)	Text, Long text
Text	Text, Long text
Text Area	Text, Long text
Text (Encrypted)	Text, Long text
URL	Text, Long text

Note

The system may modify values during the replication process according to the parameters of the target fields in Salesforce.

- **Text values** - the replication process may shorten long text values (strings) to fit the maximum length of the target field.

- **Decimal numbers** - the fractional part of decimal numbers may be truncated based on the data type settings of the target field. For example, the number *2.45397* could be shortened to *2.45*.

- **Large numbers** - replication of large numbers may result in an error if the target field has an insufficient length. Such errors block the related contact from replication until either the contact's values are updated, or the field mappings change.

Once you configure all field mappings as required, you can start Running the Salesforce replication process.

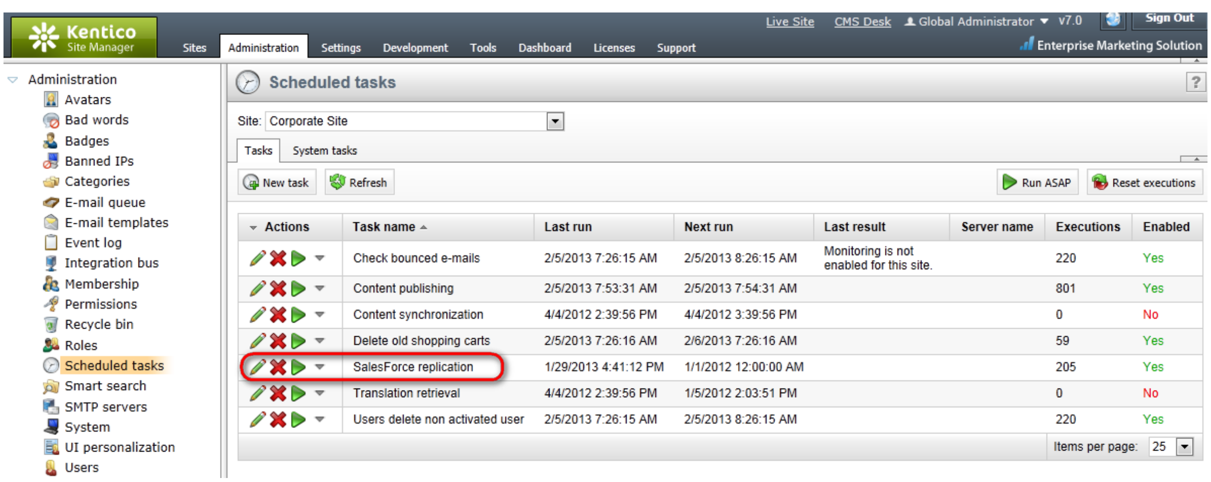
13.2.3 Running the Salesforce replication process

The system replicates contacts to Salesforce leads using the **SalesForce replication** scheduled task. This task runs separately for each site in Kentico CMS, by default once per hour.

Prerequisite: Before the system can run the replication task, you need to configure the website for access to your Salesforce organization and the related replication settings. See: Configuring Salesforce integration

You can configure the Salesforce replication task according to your own requirements, for example to change when and how often the replication takes place.

1. Go to **Administration -> Scheduled tasks** (in CMS Desk or Site Manager).
   - In Site Manager, you need to select a specific **Site**.

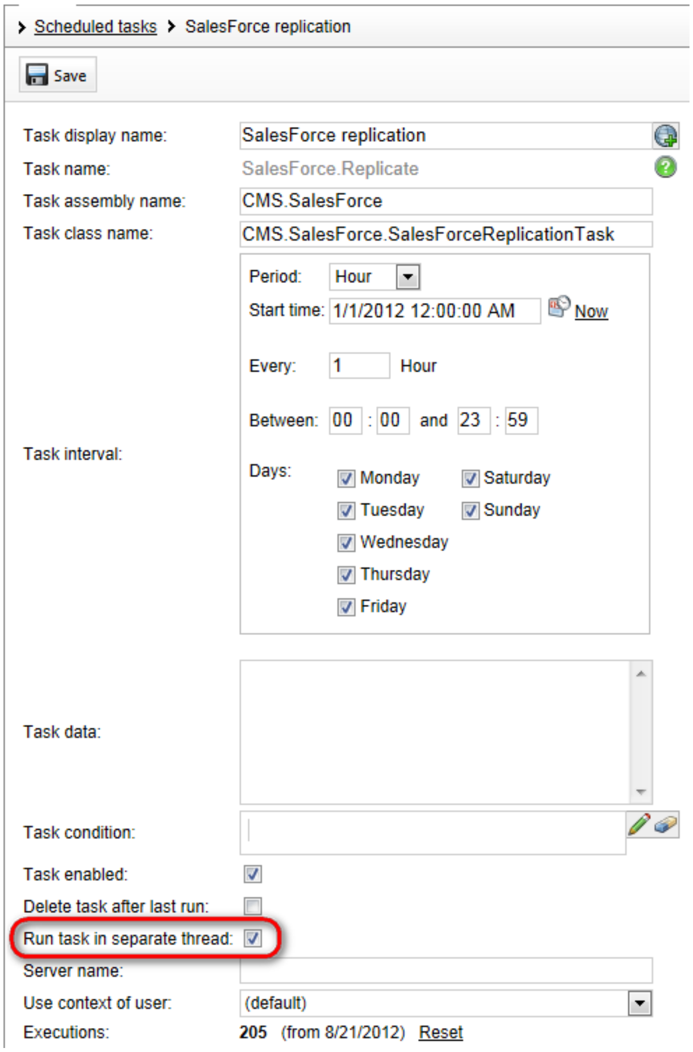


The SalesForce replication scheduled task

2. Click **Edit** () next to the **SalesForce replication** task.

3. Set the properties of the task as required.
   - Use the **Task interval** section to schedule the time and frequency of the Salesforce replication process.
   - We recommend keeping the **Run task in separate thread option** enabled to maintain optimal

replication performance.



Scheduled task configuration dialog

4. Click **Save**.

The system performs contact replication for the given website based on the settings of the task.

Tip: You can **Execute** () the Salesforce replication task manually at any time from the list of scheduled task.

Checking the replication status of your contacts

You can view which contacts in Kentico CMS have or have not been replicated to Salesforce leads.

1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.
2. Click **Display advanced filter**.
3. Select one of the options on the second **Show** row in the filter:
   - *Replicated into Salesforce leads*
   - *Not replicated into Salesforce leads*

4. Click **Search**.

The list now only shows contacts that match the selected replication status.

Replication process specifics

When executed, the **SalesForce replication** scheduled task attempts to replicate all contacts that meet the scoring requirements set for the given website.

The application can only make a limited number of API calls to Salesforce within a 24 hour window, so the replication process handles contacts in batches. Each batch is processed using one API call. You can specify the maximum number of contacts that the system replicates in a single batch through the **Batch size** setting in **Site Manager -> Settings -> Integration -> Salesforce.com**.

Note: The replication process runs separately for each site in the system. As a result, each site also has its own batch counter.

To view how many API requests your company has made and how close you are to the limit, refer to **User Menu -> Setup -> System Overview -> API Usage** in the Salesforce interface.

Replication error handling

Errors can occur during the replication of contacts, for example if a numeric value is too large for the target Salesforce field. In these cases, the process skips the related contact and continues with the replication.

You can find records of all replication errors in **Administration -> Event log**.

The system flags all contacts that caused an error and blocks them from subsequent replication attempts.

The replication block is automatically removed for individual contacts when one of their fields is updated, or globally if the **Mapping of contacts to Salesforce leads** settings change (see the Configuration section for details).

13.2.4 Example: Replicating a contact into a lead

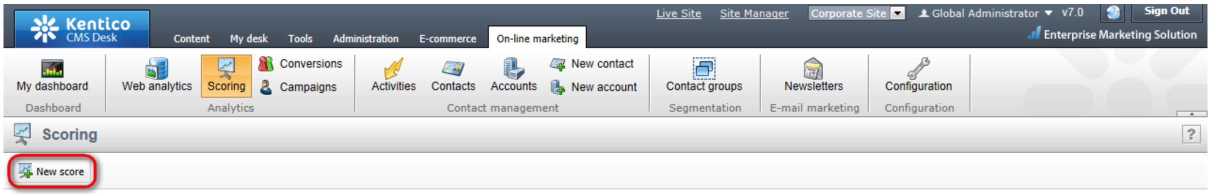
The following example demonstrates how the system replicates a sample contact into a Salesforce lead.

Important: The site must be authorized for Salesforce access.

Creating the replication score

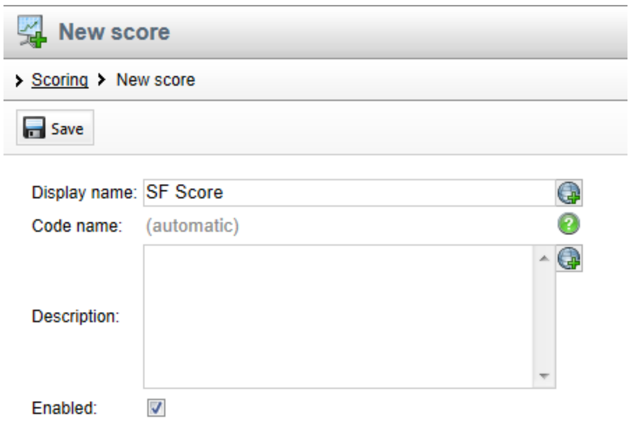
First you need to define a score to determine which contacts the system will replicate.

1. Go to **CMS Desk -> On-line marketing -> Scoring**.



2. Click **New score**.

3. Type *SF Score* as the score's **Display name** and click **Save**.

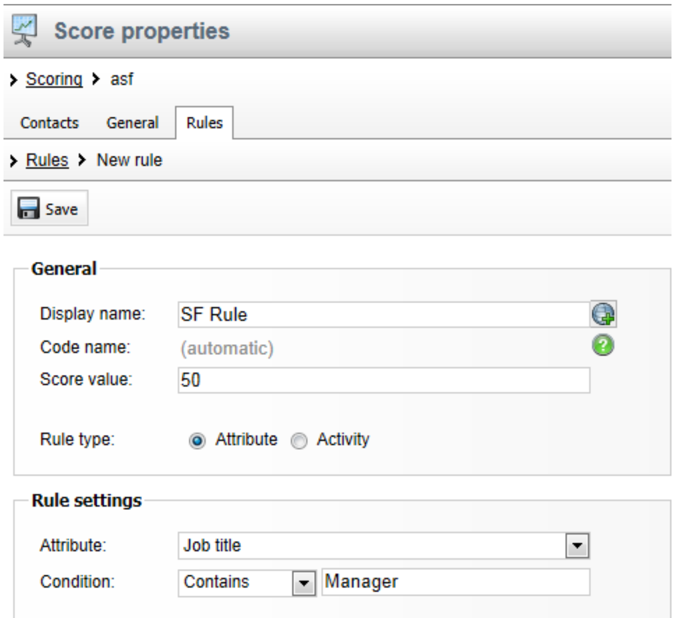


4. Switch to the **Rules** tab of the new score and click **New rule**.

5. Set the following properties for the rule:

- **Display name**: SF Rule
- **Score value**: 50
- **Rule type**: Attribute

- **Attribute**: Job title
- **Condition**: *Contains* - Manager



6. Click **Save**.

This *SF Score* grants 50 points to all contacts who have the word *Manager* in their job title.

Creating a new contact

Now you need to create a contact that fulfills the rule of the previously defined score. On live deployments, the system automatically creates and maintains contacts automatically for the website's visitors, but you can add a contact manually to try out the Salesforce replication.

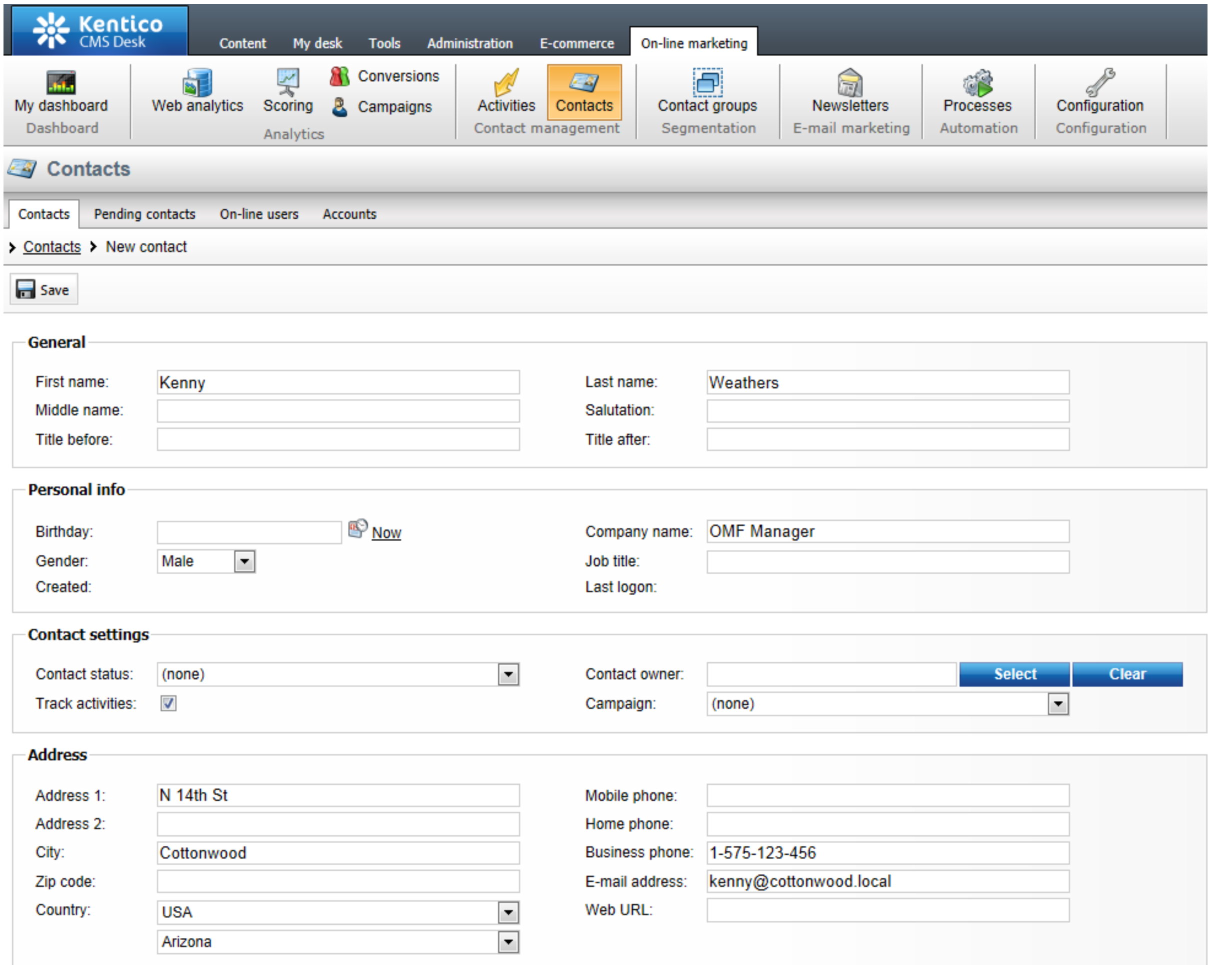
1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.

2. Click **New contact** and fill in the following attributes:

- **First name**: Kenny
- **Last name**: Weathers

- **Job title**: OMF Manager

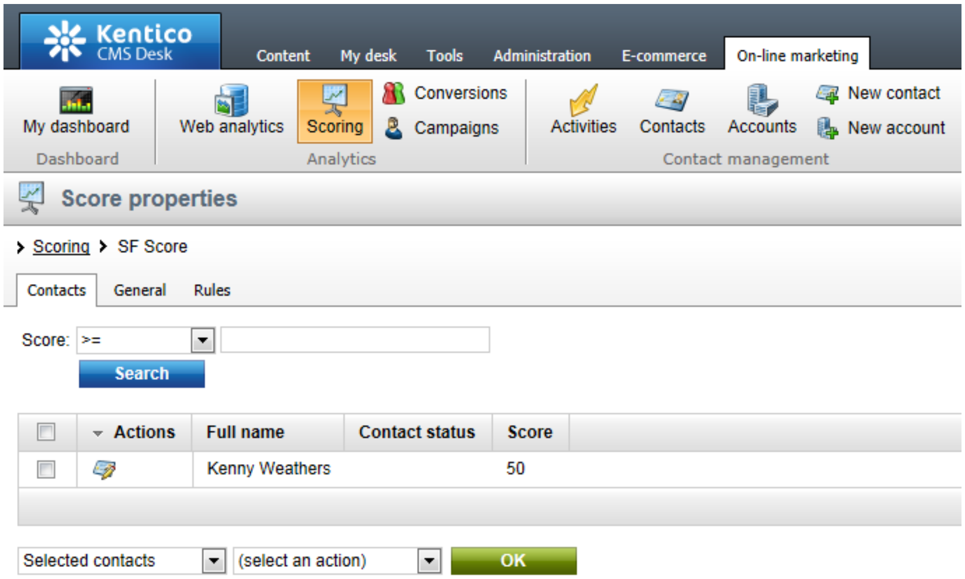
- **Address 1**: N 14th St
- **City**: Cottonwood
- **Country:** USA
- **State**: Arizona
- **Business phone**: 1-575-123-456
- **E-mail**: kenny@cottonwood.local



3. Click **Save**.

4. Go back to **CMS Desk -> On-line marketing -> Scoring** and **Edit** ( ) the *SF score*.

On the **Contacts** tab, you can see that your new contact has 50 points in the replication score. The system awarded the points because the contact contains the word *Manager* in the **Job title** field.

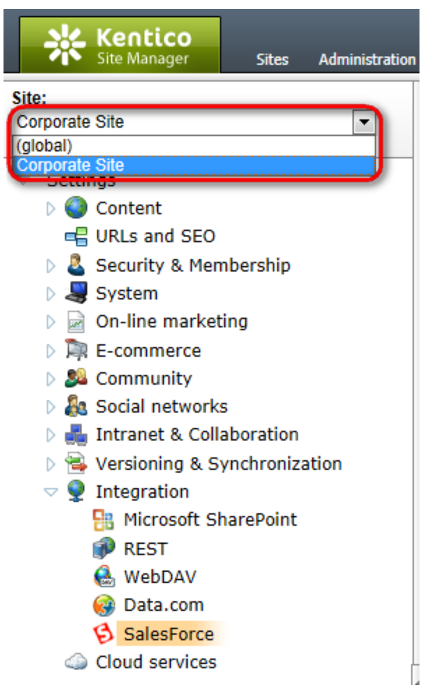


Replicating the contact to Salesforce

You now need to configure the website's Salesforce replication settings.

1. Go to **Site manager -> Settings -> Integration -> Salesforce**.

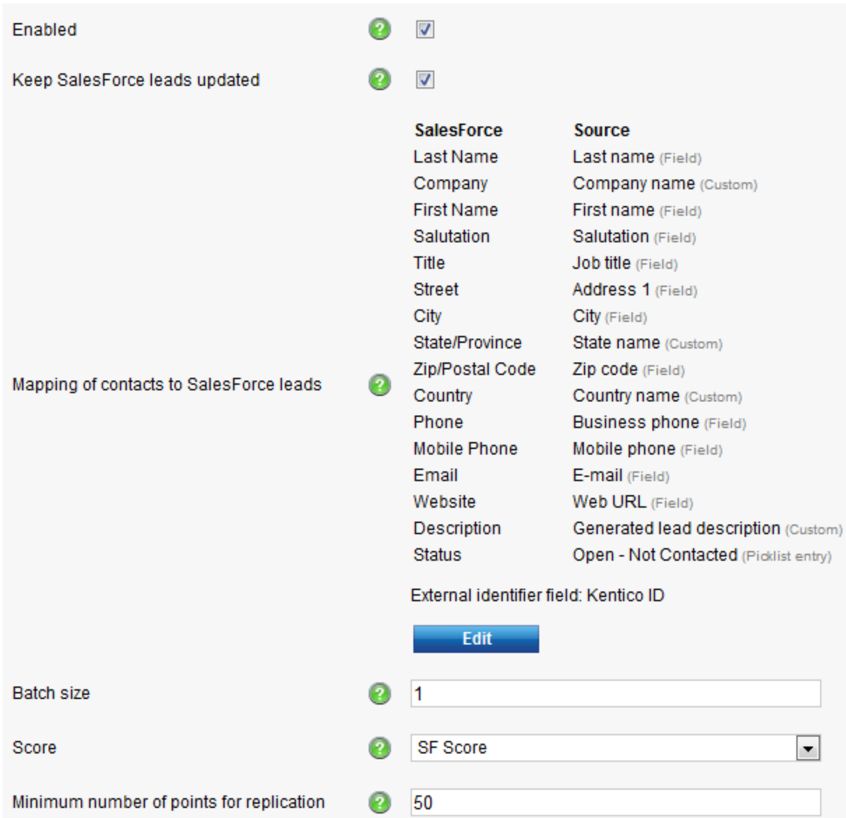
2. Choose the site where you created the score.



3. Make sure that the field mapping settings match your requirements.

4. Specify the replication score requirements through the following settings:

- **Score**: Select your replication score (*SF Score*)
- **Minimum number of points for replication**: 50

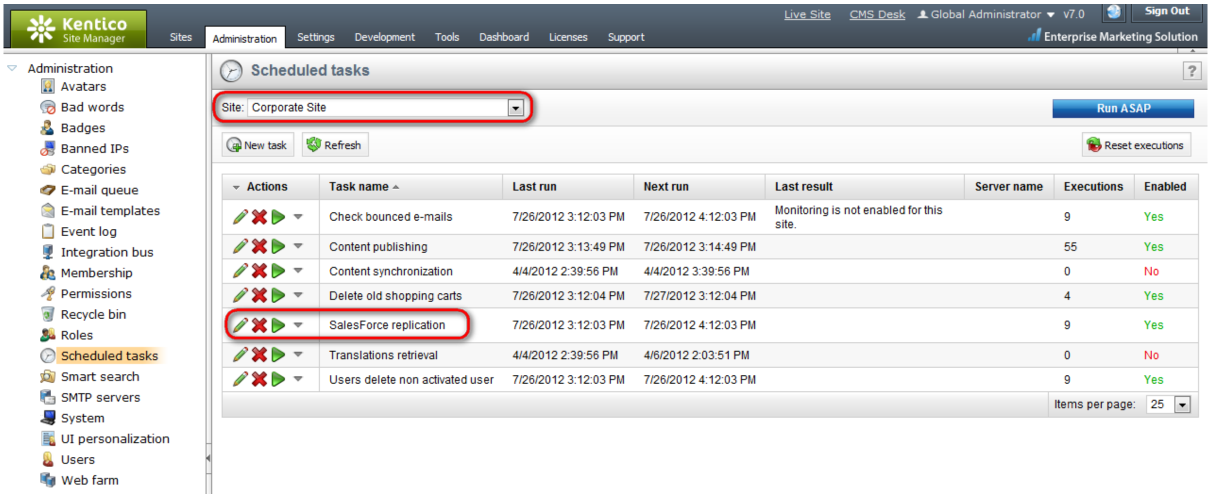


5. **Save** the settings.

You now have a contact on the website that fulfills the replication criteria. By default, the system replicates contacts once every hour. You can however run the replication manually at any time.

Starting the replication process manually:

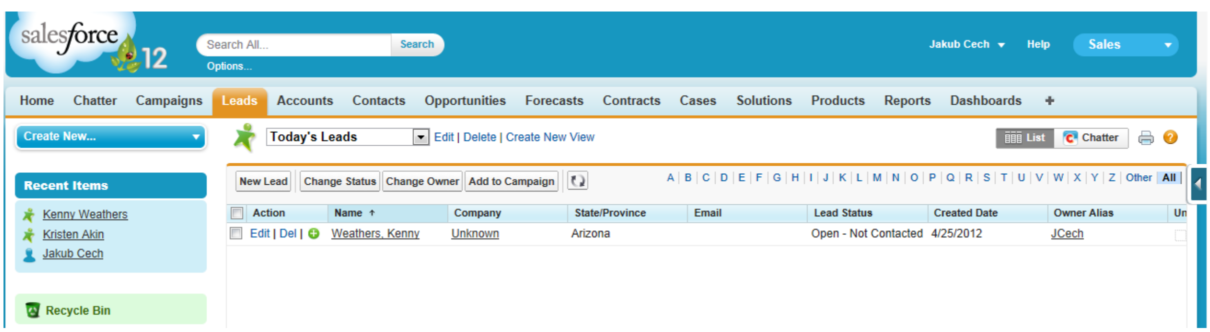
1. Go to **Site manager -> Administration -> Scheduled tasks**.
2. Select the site where you configured Salesforce replication.
3. Click **Execute** ( ) next to the **Salesforce replication** task.



Result

After several minutes at most, your contact should appear as a lead in Salesforce. To check whether the replication was successful, log in to Salesforce using your account, navigate to **Leads** and choose **Today's Leads**.

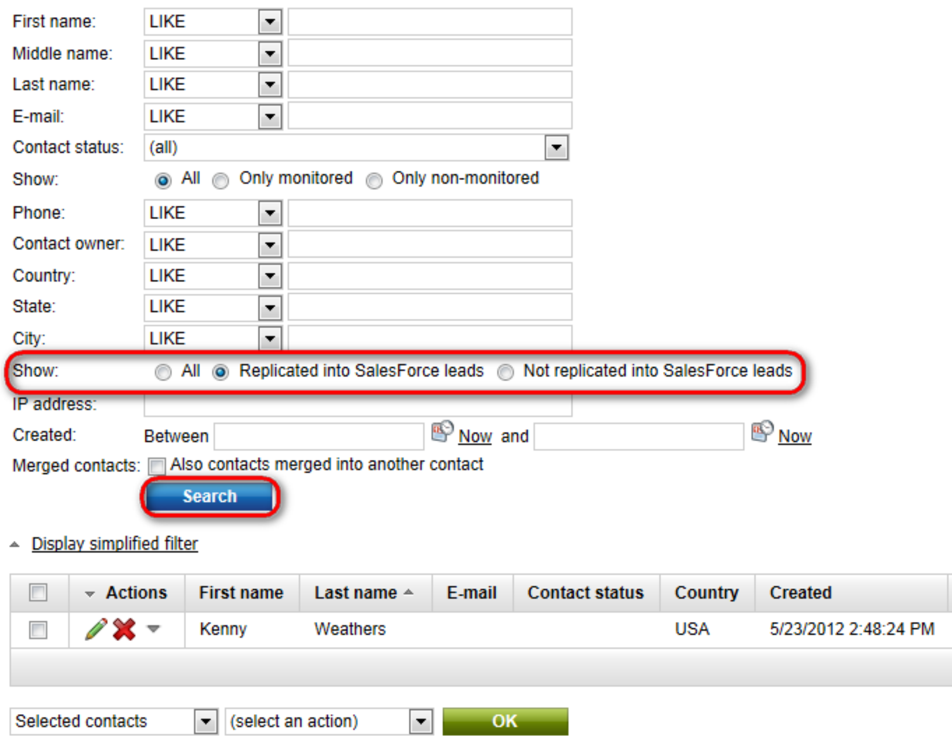
You should see the lead replicated from the Kentico CMS contact. You can now work with the lead as with any other Salesforce lead.



You can also check the replication status of the contact in Kentico CMS:

1. Navigate to **CMS Desk -> On-line marketing -> Contacts**.
2. Click **Display advanced filter**.
3. Select **Replicated into Salesforce** leads in the second **Show** row.
4. Click **Search**.

The list now shows only contacts that have been successfully replicated to Salesforce.


First name:
LIKE
Middle name:
Last name:
E-mail:
Contact status:
(all)
Show:
All
Only monitored
Only non-monitored
Phone:
Contact owner:
Country:
State:
City:
Show:
Replicated into SalesForce leads
Not replicated into SalesForce leads
IP address:
Created:
Between
Now
and
Merged contacts:
Also contacts merged into another contact
Search
Display simplified filter
Actions
First name
Last name
E-mail
Contact status
Country
Created
Kenny
Weathers
USA
5/23/2012 2:48:24 PM
Selected contacts
(select an action)
OK

Part XIV

Internals and API

14 Internals and API

14.1 Overview

The content of this chapter is mainly intended for developers who wish to customize the behaviour of the on-line marketing features, or for those interested in the internal workings. The information is divided into the following sub-chapters, which are dedicated to the corresponding modules:

- Web analytics (including campaigns and conversions)
- Contact management
- A/B testing
- MVT testing
- Content personalization
- Newsletters (E-mail marketing)

Each section contains an overview of the database structure for the given module and a list of classes that you can use when working with the API. If you are not familiar with database table data management through *Info* and *Provider* classes, we strongly recommend that you read the Database table API topic in the Kentico CMS Developer's Guide before you continue. For detailed API documentation, such as a list of all methods from the module classes, please refer to Kentico CMS API Reference.

The on-line marketing features are also supported by the macro engine, which can be extremely useful when creating dynamic website content. You can learn more in the On-line marketing macros topic.

14.2 Web analytics

14.2.1 Database tables

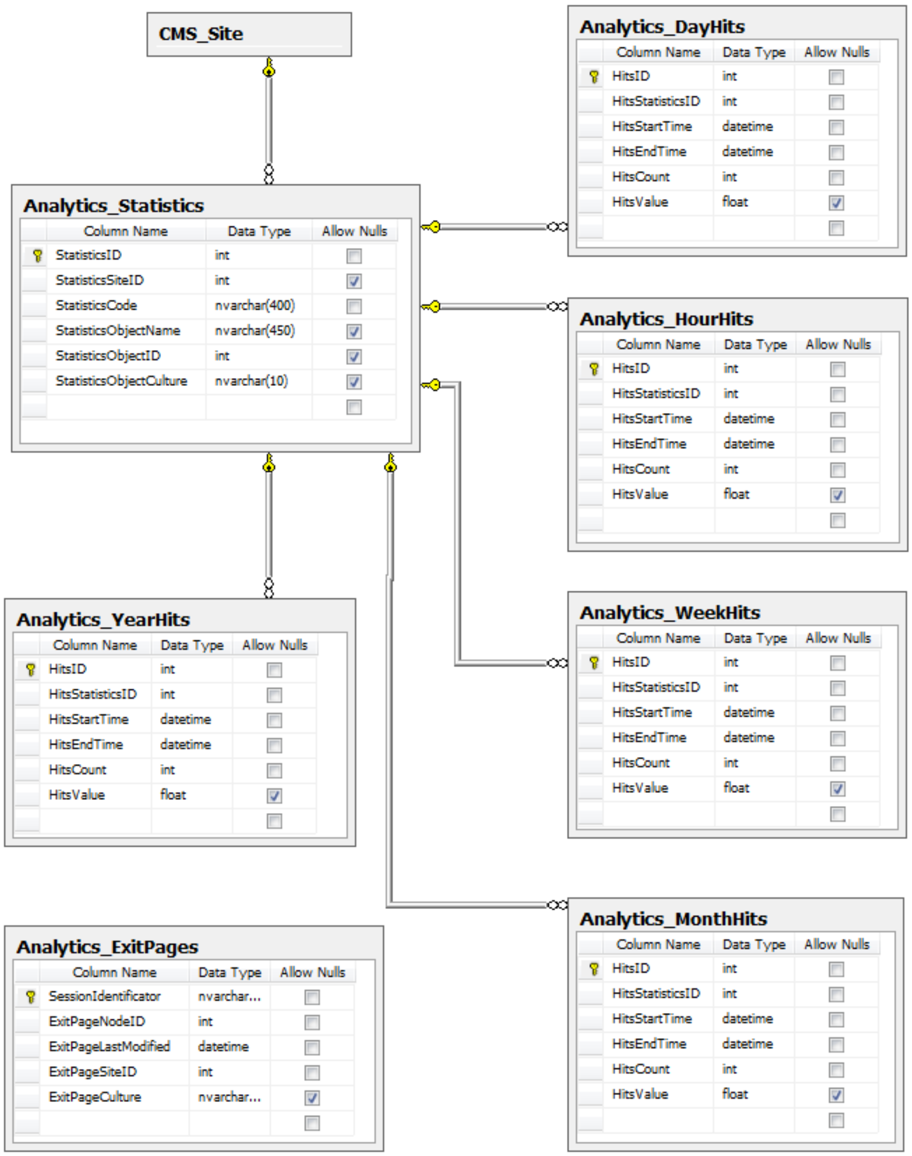
There are six important database tables used by the web analytics module to keep track of statistics and their values. The **Analytics_Statistics** table stores records that represent the statistics of a tracked event within a certain context, i.e. related to a specific object, site and culture.

Five other tables are used to store the exact number of hits for the statistics in the Analytics_Statistics table:

- **Analytics_HourHits**
- **Analytics_DayHits**
- **Analytics_WeekHits**
- **Analytics_MonthHits**
- **Analytics_YearHits**

When a hit for a tracked statistic occurs, it is logged into all of these tables. The difference between them is in the unit of time used to separate hits into individual records. For example, a record in the **Analytics_HoursHits** table would contain the number of hits that were logged for a given statistic during one hour, while a single record in **Analytics_MonthHits** would count all hits that occurred over an entire month.

The **Analytics_ExitPages** table is used to temporarily store exit page candidates for the *Top exit pages* statistic. The latest candidate recorded for a visitor when their session expires is then stored as the final exit page.



Additionally, the tables listed below are used to store campaign and conversion tracking objects:

- **Analytics_Campaign** - contains records representing campaigns and their settings.
- **Analytics_Conversion** - contains records representing conversions.
- **Analytics_ConversionCampaign** - stores relationships between campaigns and conversions. Each entry in this table indicates that a conversion should be included in the statistics of a certain campaign. Only campaigns that are restricted to a limited set of conversions have these relationships.

Campaign and conversion statistic storage

The data logged for campaigns and conversions is stored like all other web analytics statistics. The main records are kept in the **Analytics_Statistics** table and the corresponding amount of hits for individual units of time are saved in the appropriate **Hits** table.

Page view statistics for campaigns always use the *campaign* **StatisticsCode** with the name of the given campaign stored in the **StatisticsObjectName** column.

The following code names are used for conversion statistics:

- **conversion** - general statistic used to store the overall conversion records. This statistic is always logged when a conversion is performed on the website.
- **campconversion;<campaign name>** - logged when a conversion is performed by a user associated with the particular campaign.
- **abconversion;<A/B test code name>;<Variant code name>** - logged when a conversion is performed by a user who viewed the given page variant of an A/B test.
- **mvtconversion;<MVT test code name>;<Combination name> -** logged when a conversion is performed by a user who viewed the given content combination on a page with a defined multivariate test.

All types of conversions use the **StatisticsObjectName** column to store the code name of the logged conversion.

14.2.2 API classes

The following classes are used to manage the web analytics module. They can be found in the **CMS.WebAnalytics** namespace.

Analytics_Statistics table API:

- **StatisticsInfo** - represents the statistics of a certain event within a specific context.
- **StatisticsInfoProvider** - provides management functionality for statistic records.

Analytics_<time interval>Hits table API:

- **HitsInfo** - represents the hits of a statistic during a specific time interval.
- **HitsInfoProvider** - provides management functionality for statistic hits.

Analytics_Campaign table API:

- **CampaignInfo** - represents one campaign tracking object.
- **CampaignInfoProvider** - provides management functionality for campaigns.

Analytics_Conversion table API:

- **ConversionInfo** - represents one conversion tracking object.
- **ConversionInfoProvider** - provides management functionality for conversions.

Analytics_ConversionCampaign table API:

- **ConversionCampaignInfo** - represents a relationship between a campaign and a conversion.

- **ConversionCampaignInfoProvider** - provides management functionality for campaign-conversion relationships.

Other classes:

- **AnalyticsHelper** - provides general web analytics functionality and data.
- **HitLogProvider** - contains methods used to create the analytics log files for statistics.
- **HitLogProcessor** - this class implements the scheduled task used to periodically process the analytics log, which transfers the information to the database.

14.2.3 Logging conversions using the API

If you need to track a type of action that is not supported by default (as described in Logging actions as conversions), you can write your own custom code and use the API to log conversions. This allows you to monitor any type of activity performed by users on your website and view the results using the various conversion reports available in the web analytics interface.

To log a conversion via the API, you can use code similar to the following:

[C#]

```
using CMS.WebAnalytics;
using CMS.CMSHelper;

...

string siteName = CMSContext.CurrentSiteName;
string aliasPath = CMSContext.CurrentAliasPath;

// Checks that web analytics and conversion tracking are enabled in the site's
settings.
// Also confirms that the current IP address, alias path and URL extension are not
excluded from web analytics tracking.
if (AnalyticsHelper.IsLoggingEnabled(siteName, aliasPath)
    && AnalyticsHelper.TrackConversionsEnabled(siteName))
{
    // Logs the conversion according to the specified parameters.
    HitLogProvider.LogConversions(siteName, CMSContext.PreferredCultureCode,
ConversionName, 0, 1, ConversionValue);
}
```

There are several possible ways to include this type of code in your website's functionality. When tracking activity on a specific page, you may use a custom user control or web part to ensure that the code is executed as required. If you wish to log actions that may occur anywhere on the website, you may utilize global event handlers.

As shown above, conversions can be logged using the **HitLogProvider** class from the **CMS.WebAnalytics** namespace, specifically the following method:

LogConversions(string siteName, string culture, string objectName, int objectId, int count, double value)

- **siteName** - sets the code name of the site for which the conversion should be logged.
- **culture** - sets the culture code under which the conversion should be logged.
- **objectName -** used to specify the code name of the conversion that should be logged.
- **objectId** - used to specify the ID of the conversion. This parameter may be set to *0* if a valid code name is passed via the *objectName*.
- **count** - sets the amount of conversion hits that should logged. This parameter is optional and the default value (*1*) is used if it is not specified.
- **value** - specifies the value that will be logged for the conversion.

In addition to logging a general conversion, this method checks if the current user has passed through a page with a running A/B or Multivariate test, or has arrived on the website through a Campaign. If this is the case, then the conversion is also automatically logged within the appropriate context and included in the statistics of the given test or campaign.

14.3 Contact management

14.3.1 Database tables

The following database tables are used to store contact management and scoring data:

- **OM_Contact -** contains records representing contacts and their settings.
- **OM_ContactStatus** - contains records representing possible contact statuses.
- **OM_ContactRole** - stores the possible roles that contacts can have within accounts.
- **OM_Membership** - this table is used to bind users, subscribers and customers to contacts. Each entry has a member type, which determines whether the related object is a user, subscriber or customer. The given object is then connected through its ID (*RelatedID*). Please note that the relationships with the *CMS_User*, *Newsletter_Subscriber* and *COM_Customer* tables are not indicated in the diagram below.
- **OM_IP -** used to store the IP addresses associated with contacts.
- **OM_UserAgent -** stores the browser user agents of contacts.

- **OM_Account -** contains records representing accounts and their settings.
- **OM_AccountContact -** stores relationships between contacts and accounts. Each record indicates that a contact belongs to a specific account. If the contact has a certain role within the account, the information is also stored in this relationship.
- **OM_AccountStatus** - contains records representing possible account statuses.

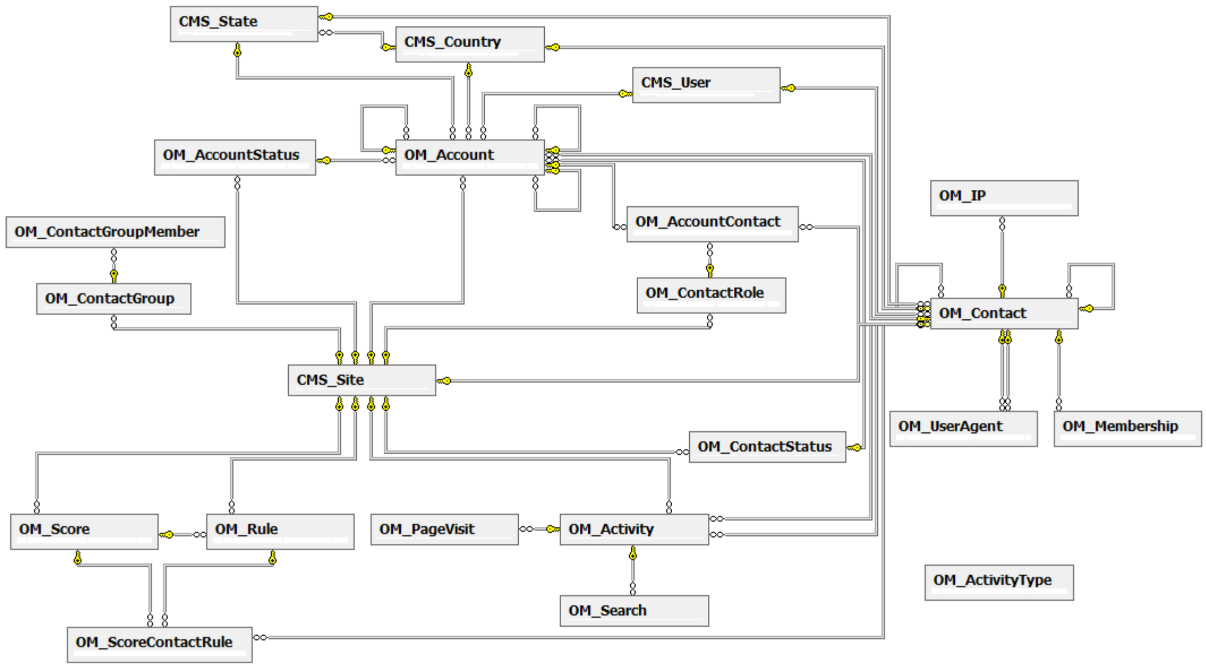
- **OM_ContactGroup** - contains records representing contact groups.
- **OM_ContactGroupMember** - this table is used to bind contacts and accounts to contact groups. Each entry has a *member type*, which determines whether the related object is a contact or account. The given object is then connected through its ID (*ContactGroupMemberRelatedID*). Please note that the relationships with the *OM_Contact* and *OM_Account* tables are not indicated in the diagram below.

- **OM_Activity** - contains records representing the logged activities.
- **OM_ActivityType** - contains records of all possible activity types.
- **OM_PageVisit** - used to store the additional details, which are logged along with *Page visit* activities.
- **OM_Search** - used to store additional details for *Search* activities.

- **OM_Score** - contains records representing and their configuration.
- **OM_Rule** - contains records representing scoring rules.

- **OM_ScoreContactRule -** contains relationships between contacts, scores and rules. Each record stores the value that a specific contact has according to a certain scoring rule. The total score points of a contact are calculated as the sum of all values for the rules that belong to the given score.

The diagram below shows the relationships between these tables. For a full-size diagram including table columns, please click here.



14.3.2 API classes

The following contact management classes can be used in the API. Unless specified otherwise, they may be found in the **CMS.OnlineMarketing** namespace.

OM_Contact table:

- **ContactInfo** - represents one contact object.
- **ContactInfoProvider** - provides management functionality for contacts.

OM_ContactStatus table:

- **ContactStatusInfo** - represents one contact status.
- **ContactStatusInfoProvider** - provides management functionality for contact statuses.

OM_ContactRole table:

- **ContactRoleInfo** - represents one contact role.
- **ContactRoleInfoProvider** - provides management functionality for contact roles.

OM_Membership table:

- **MembershipInfo** - represents a binding object between a contact and a user, subscriber or customer.
- **MembershipInfoProvider** - provides management functionality for contact binding objects.

OM_IP table:

- **IPInfo** - represents one IP address associated with a contact.
- **IPInfoProvider** - provides management functionality for contact IP addresses.

OM_UserAgent table:

- **UserAgentInfo** - represents one browser user agent object associated with a contact.
- **UserAgentInfoProvider** - provides management functionality for contact user agents.

OM_Account table:

- **AccountInfo** - represents one account object.
- **AccountInfoProvider** - provides management functionality for accounts.

OM_AccountContact table:

- **AccountContactInfo** - represents a relationship between a contact and an account.
- **AccountContactInfoProvider** - provides management functionality for account-contact relationships.

OM_AccountStatus table:

- **AccountStatusInfo** - represents one account status.
- **AccountStatusInfoProvider** - provides management functionality for account statuses.

OM_ContactGroup table:

- **ContactGroupInfo** - represents one contact group object.
- **ContactGroupInfoProvider** - provides management functionality for contact groups.

OM_ContactGroupMember table:

- **ContactGroupMemberInfo** - represents a binding object between a contact group and contact or account.
- **ContactGroupMemberInfoProvider** - provides management functionality for contact group binding objects.

OM_Activity table:

- **ActivityInfo** - represents one activity object.
- **ActivityInfoProvider** - provides management functionality for activities.

OM_ActivityType table:

- **CMS.WebAnalytics.ActivityTypeInfo** - represents an activity type object.

- **CMS.WebAnalytics.ActivityTypeInfoProvider** - provides management functionality for activity types.

OM_PageVisit table:

- **PageVisitInfo** - represents an object used to store additional data for *Page visit* activities.
- **PageVisitInfoProvider** - provides management functionality for *PageVisitInfo* objects.

OM_Search table:

- **SearchInfo** - represents an object used to store additional data for *Search* activities.
- **SearchInfoProvider** - provides management functionality for *SearchInfo* objects.

OM_Score table:

- **ScoreInfo** - represents one score object.
- **ScoreInfoProvider** - provides management functionality for scores.

OM_Rule table:

- **RuleInfo** - represents one scoring rule.
- **RuleInfoProvider** - provides management functionality for scoring rules.

OM_ScoreContactRule table:

- **ContactScoreListInfo** - represents a binding object between a contact, score and specific scoring rule.

Other classes:

- **CMS.WebAnalytics.ActivityLogProvider** - contains methods used to create activity log files.
- **CMS.WebAnalytics.ActivityLogProcessor** - this class implements the scheduled task used to periodically process the activities log, which transfers the information to the database.

14.3.3 API examples

14.3.3.1 Overview

The topics listed below contain examples that demonstrate how the contact management API can be used:

- Managing contacts
- Managing contact groups
- Managing contact roles
- Managing contact statuses
- Managing accounts
- Managing account statuses
- Managing activities

Please note

All of the API examples can be simulated in the API examples user interface. Please refer to the API examples chapter in the Developer's Guide for more details.

If you would like to check the code of the examples, please refer to ***<web project folder>\CMSAPIExamples\Code\OnlineMarketing\ContactManagement\Default.aspx.cs***.

The contact management API examples use the following namespaces:

```
using System;
using System.Data;

using CMS.GlobalHelper;
using CMS.CMSHelper;
using CMS.SiteProvider;
using CMS.DataEngine;
using CMS.OnlineMarketing;
using CMS.SettingsProvider;
```

14.3.3.2 Managing contacts

The following example creates a new contact.

```
private bool CreateContact()
{
    // Create new contact object
    ContactInfo newContact = new ContactInfo()
        {
            ContactLastName = "My New Contact",
            ContactFirstName = "My New Firstname",
            ContactSiteID = CMSContext.CurrentSiteID,
            ContactIsAnonymous = true
        };

    // Save the contact
    ContactInfoProvider.SetContactInfo(newContact);

    return true;
}
```

The following example gets and updates the contact created by the example above.

```
private bool GetAndUpdateContact()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
```

```
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Update a property
        contact.ContactLastName = contact.ContactLastName.ToLower();

        // Save the changes
        ContactInfoProvider.SetContactInfo(contact);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple contacts specified by a WHERE condition.

```
private bool GetAndBulkUpdateContacts()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        foreach (ContactInfo contact in contacts)
        {
            // Update a property of each contact
            contact.ContactLastName = contact.ContactLastName.ToUpper();

            // And save them
            ContactInfoProvider.SetContactInfo(contact);
        }

        return true;
    }

    return false;
}
```

The following example adds a contact status to the contact created by the first example in this topic.

```
private bool AddContactStatusToContact()
{
```

```
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    // Get the contact status
    ContactStatusInfo contactStatus =
ContactStatusInfoProvider.GetContactStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (contactStatus != null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // If relationship doesn't already exist
        if (contact.ContactStatusID != contactStatus.ContactStatusID)
        {
            // Add contact status to contact
            contact.ContactStatusID = contactStatus.ContactStatusID;

            // Save the changes
            ContactInfoProvider.SetContactInfo(contact);

            return true;
        }
    }

    return false;
}
```

The following example removes the contact status added by the example above from the contact.

```
private bool RemoveContactStatusFromContact()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    // Get the contact status
    ContactStatusInfo contactStatus =
ContactStatusInfoProvider.GetContactStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (contactStatus != null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // If relationship exists
        if (contact.ContactStatusID == contactStatus.ContactStatusID)
```

```
        {
            // Remove the status
            contact.ContactStatusID = 0;

            // Save the changes
            ContactInfoProvider.SetContactInfo(contact);

            return true;
        }
    }

    return false;
}
```

The following example adds a membership (user) to the contact created by the first example in this topic.

```
private bool AddMembership()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Set relationship to user
        CMS.OnlineMarketing.MembershipInfoProvider.SetRelationship(
            CMSContext.CurrentUser.UserID,
            MemberTypeEnum.CmsUser,
            contact.ContactID,
            contact.ContactID,
            false);

        return true;
    }

    return false;
}
```

The following example removes the membership (user) added by the previous example from the contact.

```
private bool RemoveMembership()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
```

```
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get the membership
        CMS.OnlineMarketing.MembershipInfo membership =
CMS.OnlineMarketing.MembershipInfoProvider.GetMembershipInfo(contact.ContactID,
contact.ContactID, CMSContext.CurrentUser.UserID, MemberTypeEnum.CmsUser);

        // Delete the membership
        CMS.OnlineMarketing.MembershipInfoProvider.DeleteRelationship
(membership.MembershipID);

        return (membership != null);
    }

    return false;
}
```

The following example adds an IP address to the contact created by the first example in this topic.

```
private bool AddIPAddress()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Create new IP address
        IPInfo newIP = new IPInfo()
            {
                IPAddress = "127.0.0.1",
                IPOriginalContactID = contact.ContactID,
                IPActiveContactID = contact.ContactID
            };

        // Save the IP info
        IPInfoProvider.SetIPInfo(newIP);

        return true;
    }

    return false;
}
```

The following example removes the IP address added by the previous example from the contact.

```
private bool RemoveIPAddress()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get contact's IP
        where = String.Format("IPOriginalContactID = '{0}' AND IPAddress = '{1}'",
contact.ContactID, "127.0.0.1");
        InfoDataSet<IPInfo> deleteIPs = IPInfoProvider.GetIps(where, null, 1,
"IPID");

        if (!DataHelper.DataSourceIsEmpty(deleteIPs))
        {
            // Delete IP
            IPInfoProvider.DeleteIPInfo(deleteIPs.First<IPInfo>());

            return true;
        }
    }

    return false;
}
```

The following example adds user agent information to the contact created by the first example in this topic.

```
private bool AddUserAgent()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Create new agent info
        UserAgentInfo agentInfo = new UserAgentInfo()
            {
                UserAgentActiveContactID = contact.ContactID,
```

```
                UserAgentOriginalContactID = contact.ContactID,
                UserAgentString = "My User Agent"
            };

        // Save the agent info
        UserAgentInfoProvider.SetUserAgentInfo(agentInfo);

        return true;
    }

    return false;
}
```

The following example removes the user agent information added by the previous example from the contact.

```
private bool RemoveUserAgent()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get the user agent info
        where = String.Format("UserAgentOriginalContactID = '{0}' AND
UserAgentString = '{1}'", contact.ContactID, "My User Agent");
        InfoDataSet<UserAgentInfo> deleteAgents =
UserAgentInfoProvider.GetUserAgents(where, null, 1, null);

        if (!DataHelper.DataSourceIsEmpty(deleteAgents))
        {
            // Delete the user agent info
            UserAgentInfoProvider.DeleteUserAgentInfo
(deleteAgents.First<UserAgentInfo>());

            return true;
        }
    }

    return false;
}
```

The following example deletes the contact created by the first example in this topic.

```
private bool DeleteContact()
```

```
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Delete the contact
        ContactInfoProvider.DeleteContactInfo(contact);

        return true;
    }

    return false;
}
```

14.3.3.3 Managing contact groups

The following example creates a new contact group.

```
private bool CreateContactGroup()
{
    // Create new contact group object
    ContactGroupInfo newGroup = new ContactGroupInfo()
        {
            ContactGroupDisplayName = "My new group",
            ContactGroupName = "MyNewGroup",
            ContactGroupSiteID = CMSContext.CurrentSiteID,
            ContactGroupDynamicCondition = "{%Contact.ContactLastName.Contains
(\"My new\")%}"
        };

    // Save the contact group to database
    ContactGroupInfoProvider.SetContactGroupInfo(newGroup);

    return true;
}
```

The following example gets and updates the contact group created by the example above.

```
private bool GetAndUpdateContactGroup()
{
    // Get the contact group
    ContactGroupInfo updateGroup = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);
    if (updateGroup != null)
```

```
    {
        // Update contact group's properties
        updateGroup.ContactGroupDisplayName =
updateGroup.ContactGroupDisplayName.ToLower();

        // Save the contact group
        ContactGroupInfoProvider.SetContactGroupInfo(updateGroup);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple contact groups specified by a WHERE condition.

```
private bool GetAndBulkUpdateContactGroups()
{
    // Get the contact groups
    string where = "ContactGroupName LIKE N'MyNewGroup%'";
    InfoDataSet<ContactGroupInfo> groups =
ContactGroupInfoProvider.GetContactGroups(where, null);

    if (!DataHelper.DataSourceIsEmpty(groups))
    {
        foreach (ContactGroupInfo group in groups)
        {
            // Update a property
            group.ContactGroupDisplayName = group.ContactGroupDisplayName.ToUpper
();

            // Save the contact group
            ContactGroupInfoProvider.SetContactGroupInfo(group);
        }

        return true;
    }

    return false;
}
```

The following example adds a contact to the contact group created by the first example in this topic.

```
private bool AddContactToGroup()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);
```

```
    // Get the contact group
    ContactGroupInfo group = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (group != null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Create the contact - contactgroup relationship
        ContactGroupMemberInfo newContactGroupMember = new ContactGroupMemberInfo
()
            {
                ContactGroupMemberContactGroupID = group.ContactGroupID,
                ContactGroupMemberType = ContactGroupMemberTypeEnum.Contact,
                ContactGroupMemberRelatedID = contact.ContactID,
                ContactGroupMemberFromManual = true
            };

        // Save the contact group
        ContactGroupMemberInfoProvider.SetContactGroupMemberInfo
(newContactGroupMember);

        return true;
    }

    return false;
}
```

The following example removes the contact added by the previous example from the group.

```
private bool RemoveContactFromGroup()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    // Get the contact group
    ContactGroupInfo group = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (group != null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get the contact - contactgroup relationship
        ContactGroupMemberInfo deleteContactGroupMember =
ContactGroupMemberInfoProvider.GetContactGroupMemberInfoByData
(group.ContactGroupID, contact.ContactID, ContactGroupMemberTypeEnum.Contact);
```

```
        if (deleteContactGroupMember != null)
        {
            // Delete the info
            ContactGroupMemberInfoProvider.DeleteContactGroupMemberInfo
(deleteContactGroupMember);

            return true;
        }
    }

    return false;
}
```

The following example adds an account to the contact group created by the first example in this topic.

```
private bool AddAccountToGroup()
{
    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);

    // Get the contact group
    ContactGroupInfo group = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);

    if ((account != null) && (group != null))
    {
        // Create new account - contact group relationship
        ContactGroupMemberInfo newContactGroupMember = new ContactGroupMemberInfo
()
            {
                ContactGroupMemberContactGroupID = group.ContactGroupID,
                ContactGroupMemberType = ContactGroupMemberTypeEnum.Account,
                ContactGroupMemberRelatedID = account.AccountID
            };

        // Save the object
        ContactGroupMemberInfoProvider.SetContactGroupMemberInfo
(newContactGroupMember);

        return true;
    }

    return false;
}
```

The following example removes the account added by the previous example from the group.

```
private bool RemoveAccountFromGroup()
{
```

```
    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);

    // Get the contact group
    ContactGroupInfo group = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);

    if ((account != null) && (group != null))
    {
        // Get the account - contactgroup relationship
        ContactGroupMemberInfo deleteContactGroupMember =
ContactGroupMemberInfoProvider.GetContactGroupMemberInfoByData
(group.ContactGroupID, account.AccountID, ContactGroupMemberTypeEnum.Account);

        if (deleteContactGroupMember != null)
        {
            // Delete the info
            ContactGroupMemberInfoProvider.DeleteContactGroupMemberInfo
(deleteContactGroupMember);

            return true;
        }
    }

    return false;
}
```

The following example deletes the contact group created by the first example in this topic.

```
private bool DeleteContactGroup()
{
    // Get the contact group
    ContactGroupInfo deleteGroup = ContactGroupInfoProvider.GetContactGroupInfo
("MyNewGroup", CMSContext.CurrentSiteName);

    if (deleteGroup != null)
    {
        // Delete the contact group
        ContactGroupInfoProvider.DeleteContactGroupInfo(deleteGroup);

        return true;
    }

    return false;
}
```

14.3.3.4 Managing contact roles

The following example creates a contact role.

```
private bool CreateContactRole()
{
    // Create new contact role object
    ContactRoleInfo newRole = new ContactRoleInfo()
        {
            ContactRoleDisplayName = "My new role",
            ContactRoleName = "MyNewRole",
            ContactRoleSiteID = CMSContext.CurrentSiteID
        };

    // Save the contact role
    ContactRoleInfoProvider.SetContactRoleInfo(newRole);

    return true;
}
```

The following example gets and updates the contact role created by the example above.

```
private bool GetAndUpdateContactRole()
{
    // Get the contact role
    ContactRoleInfo updateRole = ContactRoleInfoProvider.GetContactRoleInfo
("MyNewRole", CMSContext.CurrentSiteName);
    if (updateRole != null)
    {
        // Update a property
        updateRole.ContactRoleDisplayName =
updateRole.ContactRoleDisplayName.ToLower();

        // Save the changes
        ContactRoleInfoProvider.SetContactRoleInfo(updateRole);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple contact roles specified by a WHERE condition.

```
private bool GetAndBulkUpdateContactRoles()
{
    // Get the contact roles dataset
    string where = "ContactRoleName LIKE N'MyNewRole%'";
    InfoDataSet<ContactRoleInfo> roles = ContactRoleInfoProvider.GetContactRoles
(where, null);

    if (!DataHelper.DataSourceIsEmpty(roles))
    {
        foreach (ContactRoleInfo role in roles)
        {
```

```
                // Update the properties
                role.ContactRoleDisplayName = role.ContactRoleDisplayName.ToUpper();

                // Save the changes
                ContactRoleInfoProvider.SetContactRoleInfo(role);
            }

            return true;
        }

        return false;
    }
```

The following example deletes the contact role created by the first example in this topic.

```
private bool DeleteContactRole()
{
    // Get the contact role
    ContactRoleInfo deleteRole = ContactRoleInfoProvider.GetContactRoleInfo
("MyNewRole", CMSContext.CurrentSiteName);

    if (deleteRole != null)
    {
        // Delete the contact role
        ContactRoleInfoProvider.DeleteContactRoleInfo(deleteRole);

        return true;
    }

    return false;
}
```

14.3.3.5 Managing contact statuses

The following example creates a contact status.

```
private bool CreateContactStatus()
{
    // Create new contact status object
    ContactStatusInfo newStatus = new ContactStatusInfo()
        {
            ContactStatusDisplayName = "My new status",
            ContactStatusName = "MyNewStatus",
            ContactStatusSiteID = CMSContext.CurrentSiteID
        };

    // Save the contact status
    ContactStatusInfoProvider.SetContactStatusInfo(newStatus);

    return true;
```

```
}
```

The following example gets and updates the contact status created by the example above.

```
private bool GetAndUpdateContactStatus()
{
    // Get the contact status
    ContactStatusInfo updateStatus =
ContactStatusInfoProvider.GetContactStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);
    if (updateStatus != null)
    {
        // Update a property
        updateStatus.ContactStatusDisplayName =
updateStatus.ContactStatusDisplayName.ToLower();

        // Save the changes
        ContactStatusInfoProvider.SetContactStatusInfo(updateStatus);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple contact statuses specified by a WHERE condition.

```
private bool GetAndBulkUpdateContactStatuses()
{
    // Get the contact statuses dataset
    string where = "ContactStatusName LIKE N'MyNewStatus%'";
    InfoDataSet<ContactStatusInfo> statuses =
ContactStatusInfoProvider.GetContactStatuses(where, null);

    if (!DataHelper.DataSourceIsEmpty(statuses))
    {
        foreach (ContactStatusInfo contactStatus in statuses)
        {
            // Update a property
            contactStatus.ContactStatusDisplayName =
contactStatus.ContactStatusDisplayName.ToUpper();

            // Save the changes
            ContactStatusInfoProvider.SetContactStatusInfo(contactStatus);
        }

        return true;
    }
```

```
    return false;
}
```

The following example deletes the contact status created by the first example in this topic.

```
private bool DeleteContactStatus()
{
    // Get the contact status
    ContactStatusInfo deleteStatus =
ContactStatusInfoProvider.GetContactStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if (deleteStatus != null)
    {
        // Delete the contact status
        ContactStatusInfoProvider.DeleteContactStatusInfo(deleteStatus);

        return true;
    }

    return false;
}
```

14.3.3.6 Managing accounts

The following example creates a new account.

```
private bool CreateAccount()
{
    // Create new account object
    AccountInfo newAccount = new AccountInfo()
        {
            AccountName = "My New Account",
            AccountSiteID = CMSContext.CurrentSiteID
        };

    // Save the account
    AccountInfoProvider.SetAccountInfo(newAccount);

    return true;
}
```

The following example gets and updates the account created by the example above.

```
private bool GetAndUpdateAccount()
{
    // Get the account
    AccountInfo updateAccount = AccountInfoProvider.GetAccountInfo("My New
```

```
Account", CMSContext.CurrentSiteName);

    if (updateAccount != null)
    {
        // Update a property
        updateAccount.AccountName = updateAccount.AccountName.ToLower();

        // And save it
        AccountInfoProvider.SetAccountInfo(updateAccount);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple accounts specified by a WHERE condition.

```
private bool GetAndBulkUpdateAccounts()
{
    // Get dataset of accounts
    string where = "AccountName LIKE N'My New Account%'";
    InfoDataSet<AccountInfo> accounts = AccountInfoProvider.GetAccounts(where,
null);

    if (!DataHelper.DataSourceIsEmpty(accounts))
    {
        foreach (AccountInfo account in accounts)
        {
            // Update each one's property
            account.AccountName = account.AccountName.ToUpper();

            // And save it
            AccountInfoProvider.SetAccountInfo(account);
        }

        return true;
    }

    return false;
}
```

The following example adds an account status to the account created by the first example in this topic.

```
private bool AddAccountStatusToAccount()
{
    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);
```

```
    // Get the account status
    AccountStatusInfo accountStatus =
AccountStatusInfoProvider.GetAccountStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if ((account != null) && (accountStatus != null))
    {
        // Check that account doesn't have this status
        if (account.AccountStatusID != accountStatus.AccountStatusID)
        {
            // Set new status
            account.AccountStatusID = accountStatus.AccountStatusID;

            // Save changes to the object
            AccountInfoProvider.SetAccountInfo(account);

            return true;
        }
    }

    return false;
}
```

The following example removes the account status added by the previous example from the account.

```
private bool RemoveAccountStatusFromAccount()
{
    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);

    // Get the account status
    AccountStatusInfo accountStatus =
AccountStatusInfoProvider.GetAccountStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if ((account != null) && (accountStatus != null))
    {
        // Check if account has this status set
        if (account.AccountStatusID == accountStatus.AccountStatusID)
        {
            // Remove the status from account
            account.AccountStatusID = 0;

            // Save the object
            AccountInfoProvider.SetAccountInfo(account);

            return true;
        }
    }

    return false;
```

```
}
```

The following example adds a contact to the account created by the first example in this topic.

```
private bool AddContactToAccount()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);

    // Get the role
    ContactRoleInfo role = ContactRoleInfoProvider.GetContactRoleInfo("MyNewRole",
CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (account != null) && (role !=
null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Create new account - contact relationship
        AccountContactInfo accountContact = new AccountContactInfo()
            {
                AccountID = account.AccountID,
                ContactID = contact.ContactID,
                ContactRoleID = role.ContactRoleID
            };

        // And save it
        AccountContactInfoProvider.SetAccountContactInfo(accountContact);

        return true;
    }

    return false;
}
```

The following example removes the contact added by the previous example from the account.

```
private bool RemoveContactFromAccount()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);
```

```
    // Get the account
    AccountInfo account = AccountInfoProvider.GetAccountInfo("My New Account",
CMSContext.CurrentSiteName);

    if (!DataHelper.DataSourceIsEmpty(contacts) && (account != null))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Find account - contact relationship
        AccountContactInfo accountContact =
AccountContactInfoProvider.GetAccountContactInfo(account.AccountID,
contact.ContactID);

        if (accountContact != null)
        {
            // Delete the object
            AccountContactInfoProvider.DeleteAccountContactInfo(accountContact);

            return true;
        }
    }

    return false;
}
```

The following example deletes the account created by the first example in this topic.

```
private bool DeleteAccount()
{
    // Get the account
    AccountInfo deleteAccount = AccountInfoProvider.GetAccountInfo("My New
Account", CMSContext.CurrentSiteName);

    if (deleteAccount != null)
    {
        // Delete the account
        AccountInfoProvider.DeleteAccountInfo(deleteAccount);

        return true;
    }

    return false;
}
```

14.3.3.7 Managing account statuses

The following example creates an account status.

```
private bool CreateAccountStatus()
```

```
{
    // Create new account status object
    AccountStatusInfo newStatus = new AccountStatusInfo()
        {
            AccountStatusDisplayName = "My new status",
            AccountStatusName = "MyNewStatus",
            AccountStatusSiteID = CMSContext.CurrentSiteID
        };

    // Save the account status
    AccountStatusInfoProvider.SetAccountStatusInfo(newStatus);

    return true;
}
```

The following example gets and updates the account status created by the example above.

```
private bool GetAndUpdateAccountStatus()
{
    // Get the account status
    AccountStatusInfo updateStatus =
AccountStatusInfoProvider.GetAccountStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);
    if (updateStatus != null)
    {
        // Update a property
        updateStatus.AccountStatusDisplayName =
updateStatus.AccountStatusDisplayName.ToLower();

        // Save the changes
        AccountStatusInfoProvider.SetAccountStatusInfo(updateStatus);

        return true;
    }

    return false;
}
```

The following example gets and bulk updates multiple account statuses specified by a WHERE condition.

```
private bool GetAndBulkUpdateAccountStatuses()
{
    // Get the account status dataset
    string where = "AccountStatusName LIKE N'MyNewStatus%'";
    InfoDataSet<AccountStatusInfo> statuses =
AccountStatusInfoProvider.GetAccountStatuses(where, null);

    if (!DataHelper.DataSourceIsEmpty(statuses))
    {
```

```
        foreach (AccountStatusInfo accountStatus in statuses)
        {
            // Update a property
            accountStatus.AccountStatusDisplayName =
accountStatus.AccountStatusDisplayName.ToUpper();

            // Save the changes
            AccountStatusInfoProvider.SetAccountStatusInfo(accountStatus);
        }

        return true;
    }

    return false;
}
```

The following example deletes the account status created by the first example in this topic.

```
private bool DeleteAccountStatus()
{
    // Get the account status
    AccountStatusInfo deleteStatus =
AccountStatusInfoProvider.GetAccountStatusInfo("MyNewStatus",
CMSContext.CurrentSiteName);

    if (deleteStatus != null)
    {
        // Delete the account status
        AccountStatusInfoProvider.DeleteAccountStatusInfo(deleteStatus);

        return true;
    }

    return false;
}
```

14.3.3.8 Managing activities

The following example logs an activity.

```
private bool CreateActivity()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
```

```
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get an activity type
        ActivityTypeInfo activityType = ActivityTypeInfoProvider.GetActivityTypes
(null, null, 1, null).First<ActivityTypeInfo>();

        // Create new activity object
        ActivityInfo newActivity = new ActivityInfo()
            {
                ActivityType = activityType.ActivityTypeName,
                ActivityTitle = "My new activity",
                ActivitySiteID = CMSContext.CurrentSiteID,
                ActivityOriginalContactID = contact.ContactID,
                ActivityActiveContactID = contact.ContactID
            };

        // Save the activity
        ActivityInfoProvider.SetActivityInfo(newActivity);

        return true;
    }

    return false;
}
```

The following example gets and updates a the activity logged by the example above.

```
private bool GetAndUpdateActivity()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get all activities associated with user
        where = String.Format("ActivityActiveContactID = '{0}'",
contact.ContactID);
        InfoDataSet<ActivityInfo> updateActivities =
ActivityInfoProvider.GetActivities(where, null);

        if (!DataHelper.DataSourceIsEmpty(updateActivities))
        {
            // Get just the first activity
            ActivityInfo activity = updateActivities.First<ActivityInfo>();

            // Update the activity
            activity.ActivityTitle = activity.ActivityTitle.ToLower();
```

```
            // Save the activity
            ActivityInfoProvider.SetActivityInfo(activity);

            return true;
        }
    }

    return false;
}
```

The following example gets and bulk updates multiple activities specified by a WHERE condition.

```
private bool GetAndBulkUpdateActivities()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get all activities associated with contact
        where = String.Format("ActivityActiveContactID = '{0}'",
contact.ContactID);
        InfoDataSet<ActivityInfo> updateActivities =
ActivityInfoProvider.GetActivities(where, null);

        if (!DataHelper.DataSourceIsEmpty(updateActivities))
        {

            foreach (ActivityInfo activity in updateActivities)
            {
                // Update activity content
                activity.ActivityTitle = activity.ActivityTitle.ToLower();

                // Save the activity
                ActivityInfoProvider.SetActivityInfo(activity);
            }

            return true;
        }
    }

    return false;
}
```

The following example deletes the activity logged by the first example in this topic.

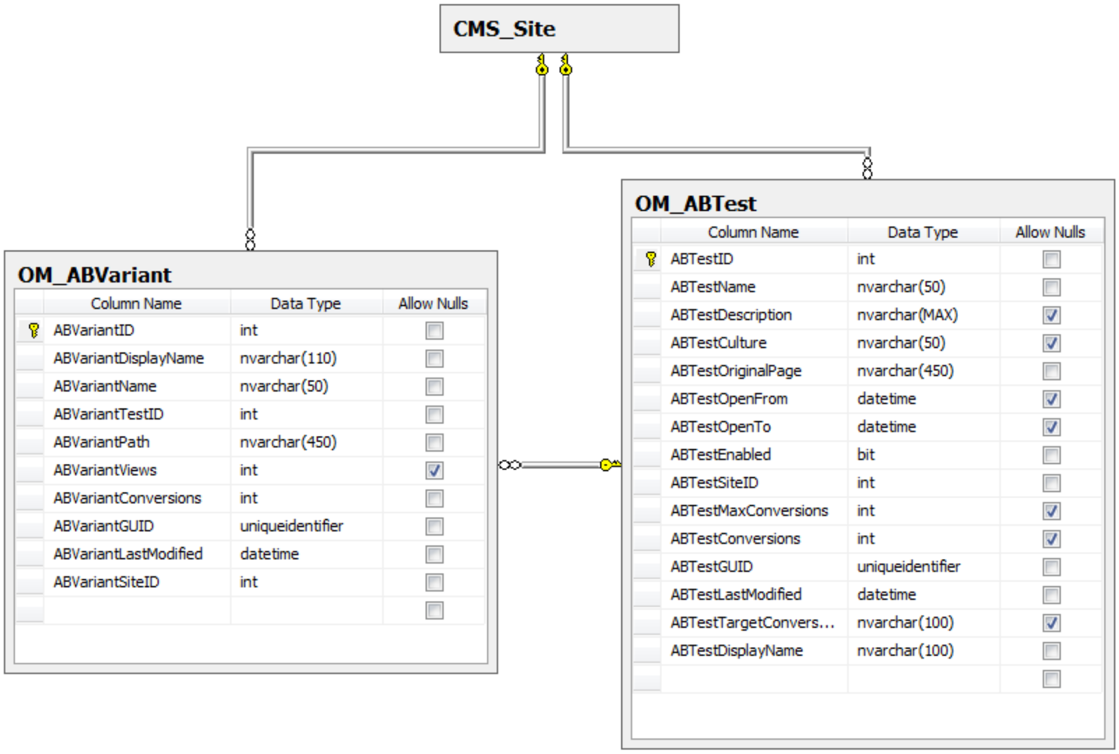
```
private bool DeleteActivity()
{
    // Get dataset of contacts
    string where = "ContactLastName LIKE N'My New Contact%'";
    InfoDataSet<ContactInfo> contacts = ContactInfoProvider.GetContacts(where,
null, 1, null);

    if (!DataHelper.DataSourceIsEmpty(contacts))
    {
        // Get the contact from dataset
        ContactInfo contact = contacts.First<ContactInfo>();

        // Get all activities associated with contact
        where = String.Format("ActivityOriginalContactID = '{0}'",
contact.ContactID);
        InfoDataSet<ActivityInfo> activities = ActivityInfoProvider.GetActivities
(where, null);

        if (!DataHelper.DataSourceIsEmpty(activities))
        {
            foreach (ActivityInfo activity in activities)
            {
                // Delete the object
                ActivityInfoProvider.DeleteActivityInfo(activity);
            }

            return true;
        }
    }

    return false;
}
```

14.4 A/B testing

14.4.1 Database tables

The following database tables are used to store A/B testing data:

- **OM_ABTest** - contains records representing A/B tests and their settings.
- **OM_ABVariant** - contains records representing individual A/B testing page variants. Each variant record contains the path to the corresponding document in the content tree, which will be displayed when the given variant is selected during the test.



The results of A/B tests (i.e. conversion statistics) are stored among the data of the web analytics module. Please see Web analytics -> Database tables to learn more.

14.4.2 API classes

The following classes are used to manage the A/B testing module. They may be found in the **CMS.OnlineMarketing** namespace.

OM_ABTest table API:

- **ABTestInfo** - represents one A/B test object.
- **ABTestInfoProvider** - provides management functionality for A/B tests.

OM_ABVariant table API:

- **ABVariantInfo** - represents one A/B testing page variant.
- **ABVariantInfoProvider** - provides management functionality for page variants.

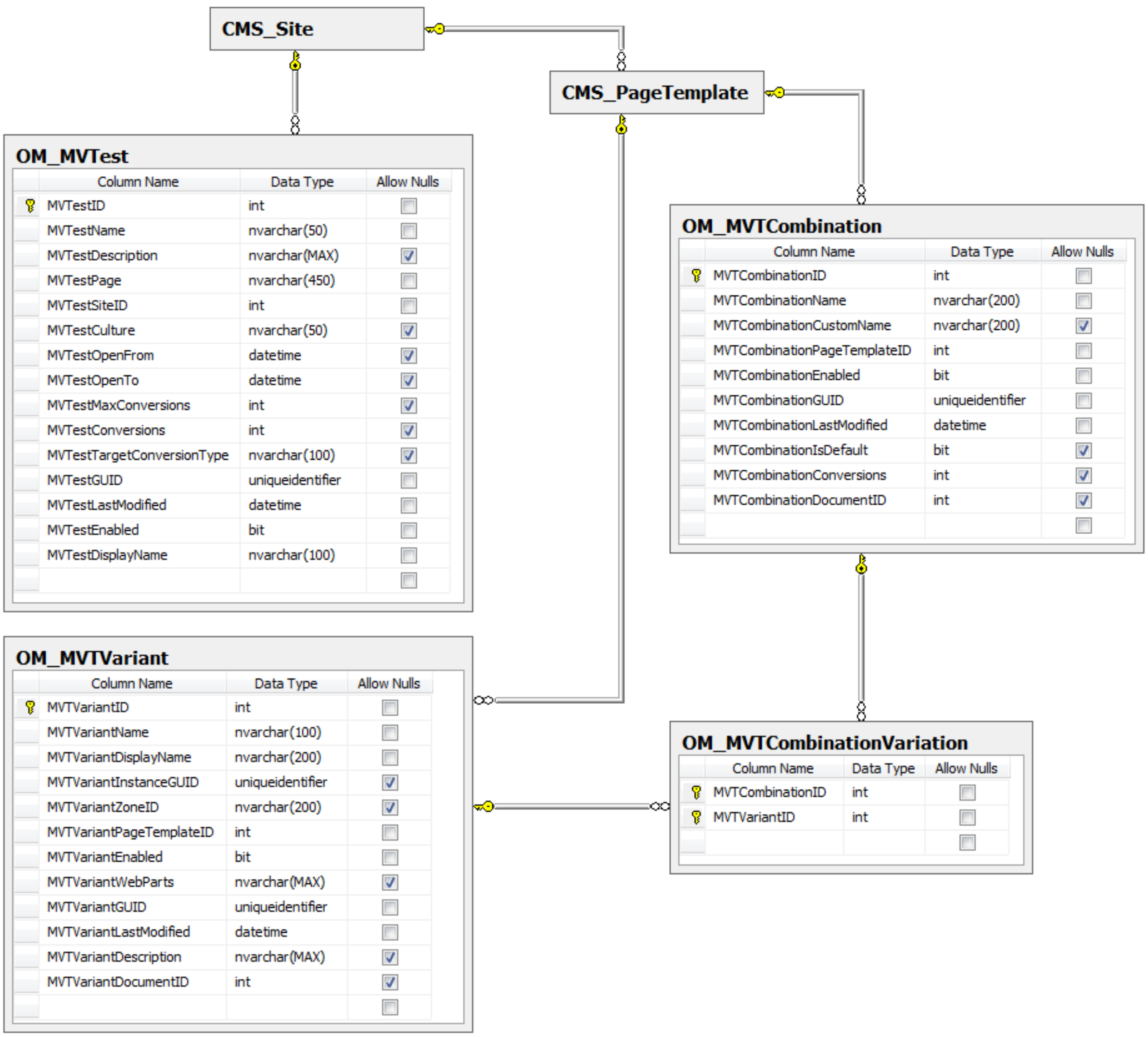
14.5 MVT testing

14.5.1 Database tables

The following database tables are used to store multivariate testing data:

- **OM_MVTest** - contains records representing MVT tests and their settings. Each test record contains the path to the corresponding document in the content tree. When the test is running, possible MVT combinations will be loaded through their connection to the given document's page template or content.

- **OM_MVTVariant -** contains records representing MVT testing variants. As you can see, variants are not linked to specific MVT tests. Instead, they are either stored on the page template used by the given document (in the case of web parts and zones) or as part of the document's content (editor widget variants).
- **OM_MVTCombination** - this table is used to store MVT combinations and their configuration.
- **OM_MVTCombinationVariation -** stores relationships between MVT variants and combinations. Each entry in this table indicates that a variant is included in a specific combination.



The results of MVT tests (i.e. conversion statistics) are stored among the data of the web analytics module. Please see [Web analytics -> Database tables] to learn more.

14.5.2 API classes

The following classes are used to manage the MVT testing module. They may be found in the **CMS.OnlineMarketing** namespace.

OM_MVTest table API:

- **MVTestInfo** - represents one MVT test object.
- **MVTestInfoProvider** - provides management functionality for MVT tests.

OM_MVTVariant table API:

- **MVTVariantInfo** - represents one MVT variant.
- **MVTVariantInfoProvider** - provides management functionality for MVT variants.

OM_MVTCombination table API:

- **MVTCombinationInfo** - represents one MVT combination.
- **MVTCombinationInfoProvider** - provides management functionality for combinations.

OM_MVTCombinationVariation table API:

- **MVTCombinationVariationInfo** - represents a relationship between an MVT variant and a combination.
- **MVTCombinationVariationInfoProvider** - provides management functionality for variant-combination relationships.

14.6 Content personalization

14.6.1 Database tables

Content personalization data is stored in a single database table:

- **OM_PersonalizationVariant** - contains records representing personalization variants of web parts, zones and widgets. Web part and zone variants are bound to a specific page template. Widget variants are additionally connected to individual documents.

OM_PersonalizationVariant

Column Name	Data Type	Allow Nulls
VariantID	int	☐
VariantEnabled	bit	☐
VariantName	nvarchar(200)	☐
VariantDisplayName	nvarchar(200)	☐
VariantInstanceGUID	uniqueidentifier	☑
VariantZoneID	nvarchar(200)	☑
VariantPageTemplateID	int	☐
VariantWebParts	nvarchar(MAX)	☐
VariantPosition	int	☑
VariantGUID	uniqueidentifier	☐
VariantLastModified	datetime	☐
VariantDescription	nvarchar(MAX)	☑
VariantDocumentID	int	☑
VariantDisplayCondition	nvarchar(MAX)	☐
		☐

All variant records contain the ID of the zone where they are located. Web part and widget variants additionally have bindings to the specific web part/widget instance for which they are defined.

14.6.2 API classes

The following classes are used to manage content personalization. They may be found in the **CMS.OnlineMarketing** namespace.

OM_PersonalizationVariant table API:

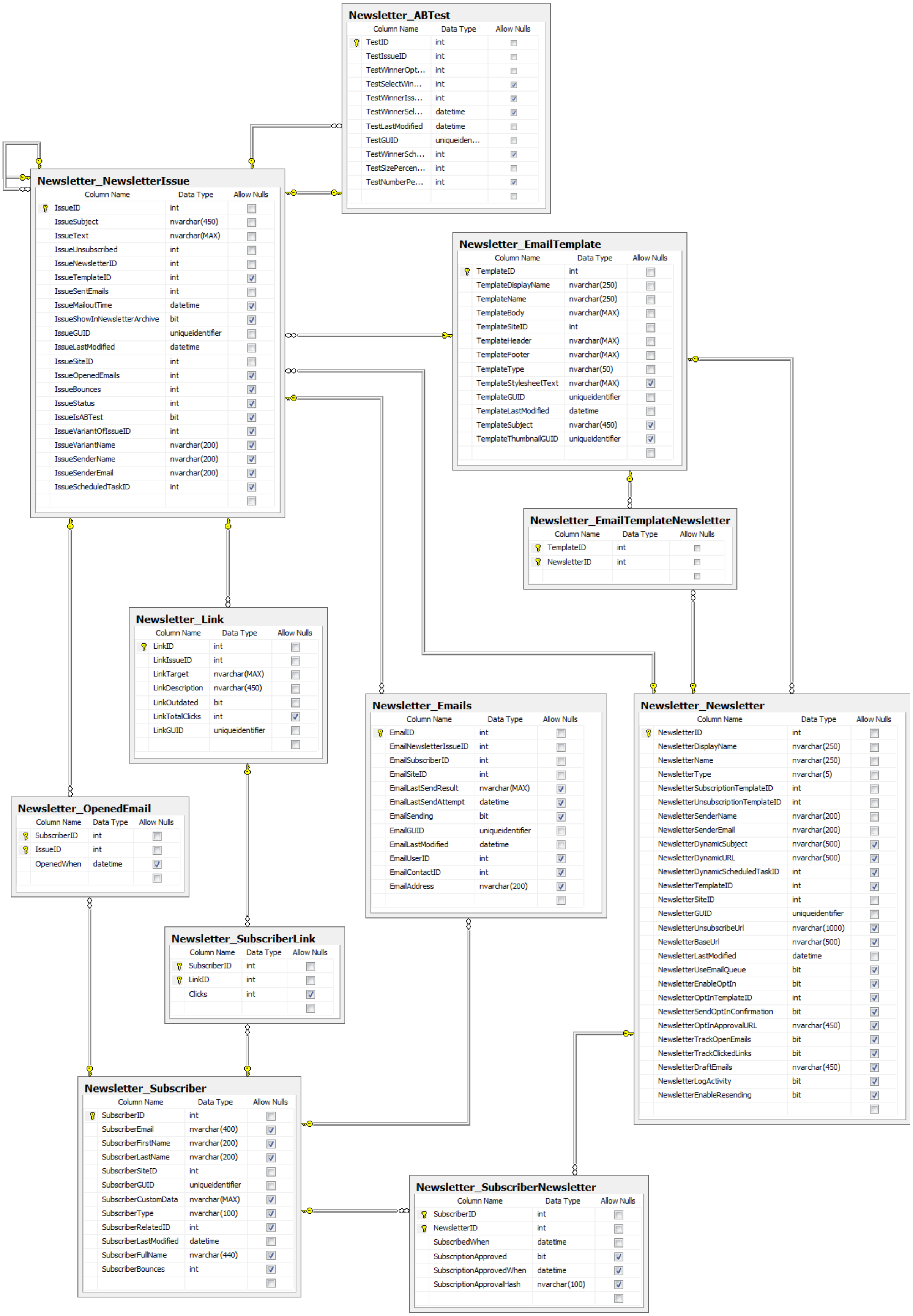
- **ContentPersonalizationVariantInfo** - represents one personalization variant.
- **ContentPersonalizationVariantInfoProvider** - provides management functionality for content personalization variants.

14.7 Newsletters

14.7.1 Database tables

The following database tables are used for the newsletters module:

- **Newsletter_Newsletter** - contains records representing newsletters and their configuration.
- **Newsletter_NewsletterIssue** - used to store individual newsletter issues, including their content.
- **Newsletter_Subscriber** - contains records representing newsletter subscribers.
- **Newsletter_SubscriberNewsletter** - stores relationships between subscribers and newsletters. This table also contains data that determines whether the subscriber is approved for the given newsletter.
- **Newsletter_EmailTemplate** - used to store the templates on which newsletter issues and notification messages are based (all types).
- **Newsletter_EmailTemplateNewsletter** - stores relationships between newsletter issue templates and static newsletters. Each record indicates that a certain template may be used for the issues of a specific newsletter.
- **Newsletter_Emails** - this table stores newsletter e-mails that could not be sent out successfully (i.e. the newsletter queue). This allows them to be re-sent at a later time.
- **Newsletter_OpenedEmail** - stores relationships between subscribers and newsletter issues. Each record indicates that a subscriber has opened a specific issue.
- **Newsletter_Link** - contains records representing links inside the content of newsletter issues.
- **Newsletter_SubscriberLink** - stores relationships between subscribers and newsletter links. Each record indicates that a subscriber has clicked on a specific link.
- **Newsletter_ABTest** - stores records that contain the A/B testing configuration of newsletter issues. Individual A/B test variants of issues are saved as separate items in the *Newsletter_NewsletterIssue* table.


Newsletter_ABTest
Column Name Data Type Allow Nulls
TestID int
TestIssueID int
TestWinnerOpt... int
TestSelectWin... int
TestWinnerIss... int
TestWinnerSel... datetime
TestLastModified datetime
TestGUID uniqueiden...
TestWinnerSch... int
TestSizePercen... int
TestNumberPe... int
Newsletter_NewsletterIssue
Column Name Data Type Allow Nulls
IssueID int
IssueSubject nvarchar(450)
IssueText nvarchar(MAX)
IssueUnsubscribed int
IssueNewsletterID int
IssueTemplateID int
IssueSentEmails int
IssueMailoutTime datetime
IssueShowInNewsletterArchive bit
IssueGUID uniqueidentifier
IssueLastModified datetime
IssueSiteID int
IssueOpenedEmails int
IssueBounces int
IssueStatus int
IssueIsABTest bit
IssueVariantOfIssueID int
IssueVariantName nvarchar(200)
IssueSenderName nvarchar(200)
IssueSenderEmail nvarchar(200)
IssueScheduledTaskID int
Newsletter_EmailTemplate
Column Name Data Type Allow Nulls
TemplateID int
TemplateDisplayName nvarchar(250)
TemplateName nvarchar(250)
TemplateBody nvarchar(MAX)
TemplateSiteID int
TemplateHeader nvarchar(MAX)
TemplateFooter nvarchar(MAX)
TemplateType nvarchar(50)
TemplateStylesheetText nvarchar(MAX)
TemplateGUID uniqueidentifier
TemplateLastModified datetime
TemplateSubject nvarchar(450)
TemplateThumbnailGUID uniqueidentifier
Newsletter_EmailTemplateNewsletter
Column Name Data Type Allow Nulls
TemplateID int
NewsletterID int
Newsletter_Link
Column Name Data Type Allow Nulls
LinkID int
LinkIssueID int
LinkTarget nvarchar(MAX)
LinkDescription nvarchar(450)
LinkOutdated bit
LinkTotalClicks int
LinkGUID uniqueidentifier
Newsletter_Emails
Column Name Data Type Allow Nulls
EmailID int
EmailNewsletterIssueID int
EmailSubscriberID int
EmailSiteID int
EmailLastSendResult nvarchar(MAX)
EmailLastSendAttempt datetime
EmailSending bit
EmailGUID uniqueidentifier
EmailLastModified datetime
EmailUserID int
EmailContactID int
EmailAddress nvarchar(200)
Newsletter_Newsletter
Column Name Data Type Allow Nulls
NewsletterID int
NewsletterDisplayName nvarchar(250)
NewsletterName nvarchar(250)
NewsletterType nvarchar(5)
NewsletterSubscriptionTemplateID int
NewsletterUnsubscriptionTemplateID int
NewsletterSenderName nvarchar(200)
NewsletterSenderEmail nvarchar(200)
NewsletterDynamicSubject nvarchar(500)
NewsletterDynamicURL nvarchar(500)
NewsletterDynamicScheduledTaskID int
NewsletterTemplateID int
NewsletterSiteID int
NewsletterGUID uniqueidentifier
NewsletterUnsubscribeUrl nvarchar(1000)
NewsletterBaseUrl nvarchar(500)
NewsletterLastModified datetime
NewsletterUseEmailQueue bit
NewsletterEnableOptIn bit
NewsletterOptInTemplateID int
NewsletterSendOptInConfirmation bit
NewsletterOptInApprovalURL nvarchar(450)
NewsletterTrackOpenEmails bit
NewsletterTrackClickedLinks bit
NewsletterDraftEmails nvarchar(450)
NewsletterLogActivity bit
NewsletterEnableResending bit
Newsletter_OpenedEmail
Column Name Data Type Allow Nulls
SubscriberID int
IssueID int
OpenedWhen datetime
Newsletter_SubscriberLink
Column Name Data Type Allow Nulls
SubscriberID int
LinkID int
Clicks int
Newsletter_Subscriber
Column Name Data Type Allow Nulls
SubscriberID int
SubscriberEmail nvarchar(400)
SubscriberFirstName nvarchar(200)
SubscriberLastName nvarchar(200)
SubscriberSiteID int
SubscriberGUID uniqueidentifier
SubscriberCustomData nvarchar(MAX)
SubscriberType nvarchar(100)
SubscriberRelatedID int
SubscriberLastModified datetime
SubscriberFullName nvarchar(440)
SubscriberBounces int
Newsletter_SubscriberNewsletter
Column Name Data Type Allow Nulls
SubscriberID int
NewsletterID int
SubscribedWhen datetime
SubscriptionApproved bit
SubscriptionApprovedWhen datetime
SubscriptionApprovalHash nvarchar(100)

14.7.2 API classes

The following classes are used to manage the newsletters module. They are located under the **CMS.Newsletter** namespace.

Newsletter_Newsletter table API:

- **NewsletterInfo** - represents one newsletter object.
- **NewsletterInfoProvider** - provides management functionality for newsletters.

Newsletter_NewsletterIssue table API:

- **IssueInfo** - represents one newsletter issue.
- **IssueInfoProvider** - provides management functionality for newsletter issues.

Newsletter_Subscriber table API:

- **SubscriberInfo** - represents one newsletter subscriber.
- **SubscriberInfoProvider** - provides management functionality for subscribers.

Newsletter_SubscriberNewsletter table API:

- **SubscriberNewsletterInfo** - represents a relationship between a subscriber and a newsletter.
- **SubscriberNewsletterInfoProvider** - provides management functionality for subscriber-newsletter relationships.

Newsletter_EmailTemplate table API:

- **EmailTemplateInfo** - represents one newsletter e-mail template.
- **EmailTemplateInfoProvider** - provides management functionality for newsletter templates.

Newsletter_EmailTemplateNewsletter table API:

- **EmailTemplateNewsletterInfo** - represents a relationship between a newsletter and an issue template.
- **EmailTemplateNewsletterInfoProvider** - provides management functionality for newsletter-template relationships.

Newsletter_Emails table API:

- **EmailQueueItem** - represents an issue e-mail in the newsletter e-mail queue.
- **EmailQueueManager** - provides management functionality for the newsletter queue.

Newsletter_OpenedEmail table API:

- **OpenedEmailInfo** - represents a relationship which indicates that a subscriber has opened a newsletter issue.
- **OpenedEmailInfoProvider** - provides management functionality for subscriber-issue relationships.

Newsletter_Link table API:

- **LinkInfo** - represents a link in a newsletter.
- **LinkInfoProvider** - provides management functionality for newsletter links.

Newsletter_SubscriberLink table API:

- **SubscriberLinkInfo** - represents a relationship between a subscriber and a newsletter link (i.e. the amount of clicks that the given subscriber performed, if any).
- **SubscriberLinkInfoProvider** - provides management functionality for subscriber-link relationships.

Newsletter_ABTest table API:

- **ABTestInfo** - represents the A/B test configuration of a newsletter issue.
- **ABTestInfoProvider** - provides management functionality for newsletter issue A/B tests.

14.8 Force.com

14.8.1 Force.com integration API

In addition to the built-in replication process that automatically converts Kentico contacts into Salesforce leads, Kentico also provides integration with the Force.com API. The integration allows you to develop custom logic for manipulating data inside your Salesforce organization (query, create, update and delete).

This page introduces the Force.com integration API and presents basic examples to get you started.

The following content assumes you already have knowledge of the Force.com API and Salesforce Object Query Language (SOQL). SOQL is the object language developed for querying data on the Force.com platform. Refer to the official Force.com documentation for more information on the Force.com API and SOQL.

Kentico integrates the API using the Partner version of the web service, which is weakly typed. This means that the integration API does not use classes such as Lead or Company. Only general classes are available: *Object* and *Field*. For example, to create a new contact in your Salesforce organization, you need to create a new Object and set its properties so that it represents a contact. The flexibility of this form of integration reduces the reliance on the Force.com data model.

Note: There are several basic naming differences between the default Force.com API and the integration API in Kentico CMS:

Salesforce	Kentico CMS
Object	Entity
Object Type	Model
Object Field	Entity Attribute
Object Field Type	Entity Attribute Model

Salesforce API requirements and limitations

You can only use the integration API if your Salesforce organization has the API feature enabled. This feature is enabled by default for the following editions:

- Unlimited
- Enterprise
- Developer
- Some Professional Edition organizations may have the API enabled

Force.com also enforces a limit on the number of API calls your organization can make during a 24 hour window. If your organization exceeds the limit, all calls result in an error until the number of calls over the last 24 hours drops below the limit. The maximum number of allowed calls for your organization is determined by the edition you are using.

Reference requirement

The Salesforce integration API requires a reference to the **CMS.SalesForce** namespace.

Add the following *using* directive to all files that access the integration API:

```
using CMS.SalesForce;
```

Establishing a session with the Salesforce organization

Before you can start working with the data, you need to establish a session with your Salesforce organization.

1. Use the following code to create a session for your Salesforce organization based on your website's Salesforce integration settings.
   - Enter the code name of your Kentico CMS site in place of *"MyCompanySite"*.

```
// Provides a Salesforce organization session
ISessionProvider sessionProvider = new ConfigurationSessionProvider { SiteName =
"MyCompanySite" };
Session session = sessionProvider.CreateSession();
```

2. Create a client to access the organization's data using the session.

```
// Creates a client to access Salesforce organization's data
SalesForceClient client = new SalesForceClient(session);
```

You can then use the client to perform specific operations.

Note: All examples in the sections below use the **client** variable for this purpose.

Querying data

1. Describe the entity you will be working with.
   - The following examples work with Salesforce *Contact* entities, but you can use the same approach for other Salesforce objects such as Leads or Accounts.

```
// Describes the Salesforce Contact entity
EntityModel model = client.DescribeEntity("Contact");

// Displays basic information about the entity's attributes
foreach (EntityAttributeModel attributeModel in model.AttributeModels)
{
    Console.WriteLine("{0} is an attribute labeled {1} of type {2}",
attributeModel.Name, attributeModel.Label, attributeModel.Type);
}
```

2. Load the attributes of the entity using SOQL.

```
// Executes a query using SOQL
SelectEntitiesResult result = client.SelectEntities("SELECT Id, Name, MobilePhone
FROM Contact", model);

// Displays how many contacts were found
Console.WriteLine("{0} contacts were found", result.TotalEntityCount);

// Displays basic information about each of the contacts
foreach (Entity contact in result.Entities)
{
    Console.WriteLine("Contact {0} with name {1} and phone number {2}",
contact.Id, contact["Name"], contact["MobilePhone"]);
}
```

Creating objects

1. Create a new contact.

```
// Describes the Contact entity
EntityModel model = client.DescribeEntity("Contact");

// Creates a new contact entity
Entity customContact = model.CreateEntity();

customContact["LastName"] = "Jones";
customContact["Description"] = "Always has his wallet on him";

Entity[] customContacts = new Entity[] { customContact };
```

```
CreateEntityResult[] results = client.CreateEntities(customContacts);
```

2. Use the following code to check whether the creation was successful or not.

```
// Checks if the contact was successfully created in Salesforce
foreach (CreateEntityResult createResult in results)
{
    if (createResult.IsSuccess)
    {
        Console.WriteLine("A contact with id {0} was inserted.",
createResult.EntityId);
    }
    else
    {
        Console.WriteLine("A contact could not be inserted.");
        foreach (Error error in createResult.Errors)
        {
            Console.WriteLine("An error {0} has occurred: {1}", error.StatusCode,
error.Message);
        }
    }
}
```

Updating existing objects

To update an object, you first need to either prepare a new entity, or load an existing one using SOQL.

Custom Salesforce field required

The following examples use a custom external ID attribute for Salesforce contacts, so you first need to define the custom field in Salesforce.

1. Log in to Salesforce.
2. Navigate to **App Setup -> Customize -> Contacts -> Fields**.
3. Click **New** in the **Contact Custom Fields & Relationships** section.
4. Choose **Text** as the data type and click **Next**.
5. In the **Enter the details** step, fill in the following values:

- **Field label**: KenticoContactID
- **Length**: 32
- **Field Name**: KenticoContactID
- **External ID**: enabled

1. Update a contact and assign an external ID value.

```
// Describes the Contact entity
```

```
EntityModel model = client.DescribeEntity("Contact");

// Updates an existing contact
Entity updateContact = model.CreateEntity();

updateContact["Description"] = "Always has his satchel on him";
updateContact["KenticoContactID"] = "D900172AABCE11E1BC6924C56188709B";

Entity[] updateContacts = new Entity[] { updateContact };
UpdateEntityResult[] updateResults = client.UpdateEntities(updateContacts);
```

2. Use the following code to check whether the update was successful or not.

```
// Checks if the contact was successfully updated
foreach (UpdateEntityResult updateResult in updateResults)
{
    if (updateResult.IsSuccess)
    {
        Console.WriteLine("The contact with id {0} was updated.",
updateResult.EntityId);
    }
    else
    {
        Console.WriteLine("The contact could not be updated.");
        foreach (Error error in createResult.Errors)
        {
            Console.WriteLine("An error {0} has occurred: {1}", error.StatusCode,
error.Message);
        }
    }
}
```

Upserting objects

The *Upsert* operation uses an external ID to identify already existing objects and then decides whether to *update* the object or *create* a new one:

- If the external ID does not exist, a new record is *created*.
- When the external ID matches an existing record, the given record is *updated*.
- If multiple external ID matches are found, the upsert operation reports an error.

Note: It is generally recommended to use upsert instead of directly creating entities if you plan on specifying an External ID attribute. This allows you to avoid creating duplicate records.

1. Create a new entity using the *upsert* call.
    - If you leave the value of *KenticoContactID* the same as in the previous example, the existing contact will be *updated*.

```
// Describes the Contact entity
```

```
EntityModel model = client.DescribeEntity("Contact");

// Creates a new contact or updates an existing one
Entity upsertContact = model.CreateEntity();

upsertContact["Description"] = "Is a professor";
upsertContact["KenticoContactID"] = "D900172AABCE11E1BC6924C56188709B";

Entity[] upsertContacts = new Entity[] { upsertContact };
UpsertEntityResult[] upsertResults = client.UpsertEntities(upsertContacts,
"KenticoContactID");
```

2. Use the following code to check for the results of the upsert call.

```
// Checks the results of the upsert call
foreach (UpsertEntityResult upsertResult in upsertResults)
{
    if (upsertResult.IsSuccess)
    {
        if (upsertResult.IsUpdate)
        {
            Console.WriteLine("The contact with id {0} was updated.",
upsertResult.EntityId);
        }
        else
        {
            Console.WriteLine("A new contact with id {0} was inserted.",
upsertResult.EntityId);
        }
    }
    else
    {
        Console.WriteLine("The contact could not be created nor updated.");
        foreach (Error error in upsertResult.Errors)
        {
            Console.WriteLine("An error {0} has occurred: {1}", error.StatusCode,
error.Message);
        }
    }
}
```

Deleting objects

The delete call takes an array of entity IDs as a parameter. The ID is generated by Salesforce and is unique for each object.

1. Use an upsert call to create several new contacts that you will later delete.

```
// Describes the Contact entity
EntityModel model = client.DescribeEntity("Contact");
```

```
// Creates new contacts
Entity customContact1 = model.CreateEntity();
Entity customContact2 = model.CreateEntity();
Entity customContact3 = model.CreateEntity();

customContact1["LastName"] = "Drake";
customContact1["Description"] = "Is a hunter";
customContact1["KenticoContactID"] = "6C2L16B8W0ZXSU5SYPYRVE08QSKOR7F6";
customContact2["LastName"] = "Croft";
customContact2["Description"] = "Is an archeologist";
customContact2["KenticoContactID"] = "N5XX8Z42ISTVYGPC98AH81T1RHVOSESR";
customContact3["LastName"] = "Solo";
customContact3["Description"] = "Is a pilot";
customContact3["KenticoContactID"] = "N34VY7D5K677GKG8JOGDIA9UHVBHHVSL";

Entity[] newCustomContacts = new Entity[] { customContact1, customContact2,
customContact3 };
UpsertEntityResult[] createResults = client.UpsertEntities(newCustomContacts,
"KenticoContactID");
```

2. Use the following example to check whether the creation was successful or not.

```
// Checks the results of the upsert call
foreach (UpsertEntityResult upsertResult in createResults)
{
    if (upsertResult.IsSuccess)
    {
        if (upsertResult.IsUpdate)
        {
            Console.WriteLine("The contact with id {0} was updated.",
upsertResult.EntityId);
        }
        else
        {
            Console.WriteLine("A new contact with id {0} was inserted.",
upsertResult.EntityId);
        }
    }
    else
    {
        Console.WriteLine("The contact could not be created nor updated.");
        foreach (Error error in upsertResult.Errors)
        {
            Console.WriteLine("An error {0} has occurred: {1}", error.StatusCode,
error.Message);
        }
    }
}
```

3. Select the new contacts using a SOQL query and list their IDs.

Note: Force.com returns an 18 character version of object IDs in API calls.

```
// Executes a query using SOQL
SelectEntitiesResult queryResult = client.SelectEntities("SELECT Id, Name FROM
Contact WHERE description LIKE 'Is a%'", model);

// Displays how many contacts have been found
Console.WriteLine("{0} contacts were found", queryResult.TotalEntityCount);

// Displays basic information about each of the contacts
foreach (Entity contact in queryResult.Entities)
{
    Console.WriteLine("Contact {0} with name {1}", contact.Id, contact["Name"]);
}
```

4. Delete all of the contacts returned by the query.
   - Identify the contacts using an array of ID values.
   - Make sure you fill in the IDs returned by the previous selection query.

```
// Deletes contacts
String[] contactIDs = new String[] { "a0BA0000000L2ZCMA0", "a0BA0000000L2ZCMA1",
"a0BA0000000L2ZCMA2" };
DeleteEntityResult[] deleteResults = client.DeleteEntities(contactIDs);
```

5. Check whether the deletion was successful.

```
// Checks if the contact was successfully deleted
foreach (DeleteEntityResult deleteResult in deleteResults)
{
    if (deleteResult.IsSuccess)
    {
        Console.WriteLine("A contact with id {0} was deleted.",
deleteResult.EntityId);
    }
    else
    {
        Console.WriteLine("A contact could not be deleted.");
        foreach (Error error in deleteResult.Errors)
        {
            Console.WriteLine("An error {0} has occurred: {1}", error.StatusCode,
error.Message);
        }
    }
}
```

Loading deleted objects

After you delete an object in Salesforce, its *IsDeleted* attribute is set to *True*. You can't select deleted objects unless you enable the IncludeDeleted option for the query.

1. Change the client options so that queries include deleted objects.

```
// Changes the client settings
client.Options.IncludeDeleted = true;
```

2. Select the deleted contacts using a SOQL query.

```
// Describes the Contact entity
EntityModel model = client.DescribeEntity("Contact");

// Executes a query using SOQL
SelectEntitiesResult includeDeletedQueryResult = client.SelectEntities("SELECT Id,
Name FROM Contact WHERE description LIKE 'Is a%'", model);

// Displays how many contacts were found
Console.WriteLine("{0} contacts were found",
includeDeletedQueryResult.TotalEntityCount);

// Displays the basic information of each contact
foreach (Entity contact in includeDeletedQueryResult.Entities)
{
    Console.WriteLine("Contact {0} with name {1}", contact.Id, contact["Name"]);
}

// Reverts the client settings to ignore deleted contacts again
client.Options.IncludeDeleted = false;
```

Call options

You can adjust Salesforce client calls through additional options. The options are similar to the SOAP headers available in the Force.com API.

Call option	Type	Description
`TransactionEnabled`	Bool	Specifies whether the command operations run in a transaction.
`AttributeTruncationEnabled`	Bool	Specifies whether truncation is allowed for string values.
`FeedTrackingEnabled`	Bool	Specifies whether the changes made by the call are tracked in feeds.
`MruUpdateEnabled`	Bool	Specifies whether the call updates the list of most recently used items in Salesforce.
`IncludeDeleted`	Bool	Specifies whether SOQL queries include deleted entries.
`ClientName`	String	String identifier for the client.
`DefaultNamespace`	String	String that identifies a developer namespace prefix.

CultureName	String	Specifies the language of the returned labels.
BatchSize	Int	Specifies the maximum number of entities returned by one query call.

14.9 On-line marketing macros

You can use macro expressions to dynamically load values related to on-line marketing features. Properly using these expressions allows you to set up flexible content and behavior for your website. With on-line marketing, macros are required when building conditions for:

- Content personalization variants
- Dynamic Contact groups
- Marketing automation triggers and process steps

You can also use on-line marketing macros in all other parts of Kentico CMS where macros are supported.

Note: All expressions described below are *context macros*, which means they must be enclosed in *{% %}* parentheses when entered into text or other general fields. You do not need to use macro parentheses inside macro-specific fields, such as condition editors, where the system automatically ensures that the content is processed as a macro. For more information about possible macro options and syntax in general, please refer to the Development -> Macro expressions chapter in the Kentico CMS Developer's Guide.

Contact management and Scoring data

The fundamental part of most on-line marketing macros is the loading of a specific contact or its data. You may use the following expressions for this purpose:

- **OnlineMarketingContext.CurrentContact** - returns a *ContactInfo* object representing the current contact, i.e. the visitor who is viewing the website when the macro is resolved.
- **OnlineMarketingContext.CurrentContactID** - returns the ID of the current contact.

- **Contact** - you may use this expression when writing conditions for dynamic contact groups. It allows you to work with a general *ContactInfo* object. When the contact group is rebuilt, all contacts under the corresponding site (or from all sites in the system if the group is global) are processed and checked whether they fulfill the specified condition (e.g. *Contact.ContactGender == 1*).

ContactInfo objects serve as a way to access the values set for the given contact (e.g. *OnlineMarketingContext.CurrentContact.ContactLastName*). In addition to standard contact fields, you can also use the following advanced properties to get data related to the contact:

- **Accounts** - *InfoObjectCollection* containing all accounts (*AccountInfo* objects) to which the given contact is assigned.
- **LastActivity** - *ActivityInfo* object representing the most recently logged activity for the given contact.
- **Orders** - *InfoObjectCollection* containing all e-commerce orders (*OrderInfo* objects) made by the contact on the current site.
- **PurchasedProducts** - *InfoObjectCollection* containing all products (*SKUInfo* objects) purchased by the contact across all sites in the system.
- **Wishlist** - *InfoObjectCollection* containing all products (*SKUInfo* objects) added to the wishlist of the

given contact (across all sites in the system).

There are many ways how an *InfoObjectCollection* can be further processed. For example:

`OnlineMarketingContext.CurrentContact.Accounts.Exists(AccountName == "CompanyAccount")`
This sample macro checks whether the current contact belongs to an account named *CompanyAccount* and returns the result as a boolean value.

`OnlineMarketingContext.CurrentContact.Orders.Exists(OrderTotalPrice > 500)`
Checks if the current contact has made an order with a total value greater than 500 of the given currency.

Please see the Available macro methods topic in the Developer's Guide for a comprehensive list of the possible options.

On-line marketing macro methods

You may also call the following methods inside macro expressions, which are specifically designed to help retrieve on-line marketing data:

LastActivityOfType(Object contact, *Object activityType*)
Returns an *ActivityInfo* object representing the activity most recently logged for the specified contact.
- contact - this parameter must contain the appropriate *ContactInfo* object of the contact whose activity should be loaded.
- *activityType* - optional parameter that may be used to get the last activity of a specific type. The value must be entered as a string matching the code name of the appropriate activity type.

Example:

```
LastActivityOfType(OnlineMarketingContext.CurrentContact, "purchasedproduct")
```

or:

```
OnlineMarketingContext.CurrentContact.LastActivityOfType("purchasedproduct")
```

FirstActivityOfType(Object contact, *Object activityType*)
Returns an *ActivityInfo* object representing the first activity logged for the specified contact.
- contact - this parameter must contain the appropriate *ContactInfo* object of the contact whose activity should be loaded.
- *activityType* - optional parameter that may be used to get the first activity of a specific type. The value must be entered as a string matching the code name of the appropriate activity type.

```
OnlineMarketingContext.CurrentContact.FirstActivityOfType
("productaddedtowishlist")
```

IsInContactGroup(Object contact, Object contactGroupName)
Checks if a contact belongs to a specific contact group and returns the result as a *boolean* value (true or false).

- contact - this parameter is used to specify the *ContactInfo* object representing the contact that will be checked.
- contactGroupName - must contain the code name of the given contact group. You can enter the value as a string.

```
OnlineMarketingContext.CurrentContact.IsInContactGroup("MaleCustomers")
```

GetScore(Object contact, Object scoreName)
This method returns the total number of score points that a given contact has in a specified score (as an *integer*).
- contact - used to specify the *ContactInfo* object representing the contact whose score points should be loaded.
- scoreName - must contain the code name of the given score. You can enter the value as a string.

```
OnlineMarketingContext.CurrentContact.GetScore("InterestedInSmartphones")
```

GetEmailDomain(Object email)
This method may be used to read an e-mail address and return only its domain part (all characters after the "@" sign).
- email - this parameter is used to specify the e-mail address from which the domain should be taken. You can enter the address as a string.

```
OnlineMarketingContext.CurrentContact.ContactEmail.GetEmailDomain()
```

Scoring - Notification e-mail template macros

It is also possible to place macros into the text of the **Scoring - Notification e-mail** e-mail template, so you can dynamically load the appropriate values. This template is used for the automatic messages that inform administrators or marketers that a contact has reached a certain amount of score points. There are several special macros available for this purpose.

The **{% ScoreValue %}** expression returns the current number of score points for the given contact. Additionally, you can access the following related objects and their properties (e.g. *{% Score.ScoreDisplayName %}* etc.):

- **{% Score %}** - *ScoreInfo* object representing the given score.
- **{% Contact %}** - *ContactInfo* object of the contact that has reached the designated number of score points.

Campaign, A/B testing and multivariate testing context data

If you wish to check whether the current visitor has arrived on the website as a result of a campaign, or passed through a page with an A/B or multivariate test, you can read the information from the appropriate browser cookie. The following macro expressions may be used to get the values of the given cookies:

- **Cookies.Campaign** - returns the name of the campaign assigned to the visitor.

- **Cookies.CMSAB<A/B test code name> -** returns the code name of the page variant assigned to the visitor by the specified A/B test.
- **Cookies.CMSMVT<MVT test code name> -** returns the internal code name of the combination assigned to the visitor by the specified MVT test. Please note that this is not the custom name set for the combination, but an identifier in format *Combination_<order number>*, e.g. *Combination_003*.

Index

- W -