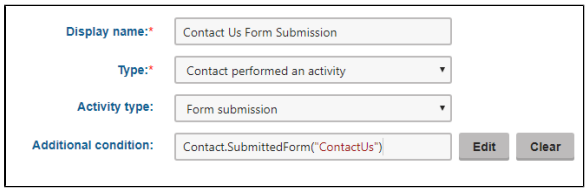
You can influence trigger processing when Marketing automation processes start by implementing custom handlers for the **ProcessTrigger** event. For example, you can use the handler to penetrate activity data into the trigger macro resolver. This then allows you to create macro conditions that start a Marketing automation process, for example, based on the data submitted by a contact in a column of an online form.

Example

The following example demonstrates how to add online form submit activity data into a marketing automation trigger macro resolver.

You have a Marketing automation process that subscribes contacts to a newsletter.

A triggers starts the process whenever a user submits the *ContactUs* form, as displayed in the following image.



Now you can improve the trigger to make it start the process only when users submit the form with the "example.com" domain in the email field.

1. Open your Kentico web project in Visual Studio (using the **WebSite.sln** or **WebApp.sln** file).
2. Create a custom module class.
    - Either add the class into a custom project within the Kentico solution (recommended) or directly into the Kentico web project (into a custom folder under the **CMSApp** project for *web application* installations, into the **App_Code** folder for *web site* installations).
3. Override the module's **OnInit** method and assign a handler to the **AutomationEvents.ProcessTrigger.Before** event.

```
using CMS;
using CMS.DataEngine;
using CMS.Automation;
using CMS.OnlineForms;
using CMS.Activities;
using CMS.MacroEngine;

// Registers the custom module into the system
[assembly: RegisterModule(typeof(ProcessTriggerHandlerModule))]

public class ProcessTriggerHandlerModule : Module
{
        // Module class constructor, the system registers the module under the
name "ProcessTriggerHandlers"
        public ProcessTriggerHandlerModule()
                : base("ProcessTriggerHandlers")
        {
        }

        // Contains initialization code that is executed when the application
starts
        protected override void OnInit()
        {
                base.OnInit();

                // Assigns a handler to the AutomationEvents.ProcessTrigger.
Before event
                AutomationEvents.ProcessTrigger.Before += ProcessTrigger_Before;
        }
}
```

4. Define the handler method (inside the custom module class):

```
void ProcessTrigger_Before(object sender, AutomationProcessTriggerEventArgs e)
{
    // Only applies to activity-based triggers
    if (e.TriggerInfo.TriggerTargetObjectType == ActivityTypeInfo.OBJECT_TYPE)
    {
                foreach (TriggerOptions option in e.Options)
                {
                        // Gets the related activity from the macro resolver
                        MacroResolver resolver = option.Resolver;
                ActivityInfo activity = resolver.GetNamedSourceData("Activity")
as ActivityInfo;

            // Checks if the activity represents the "form submit" action
                        if (activity != null && activity.ActivityType.Equals
("bizformsubmit"))
            {
                int formId = activity.ActivityItemID;
                int itemId = activity.ActivityItemDetailID;

                // Gets the form
                BizFormInfo form = BizFormInfoProvider.GetBizFormInfo(formId);

                if (form != null)
                {
                    // Gets the submitted form data
                    string className = DataClassInfoProvider.GetClassName(form.
FormClassID);
                    BizFormItem item = BizFormItemProvider.GetItem(itemId,
className);
                    if (item != null)
                    {
                        // Makes FormData available in the trigger condition
evaluation
                        resolver.SetNamedSourceData("FormData", item);
                    }
                }
            }
        }
    }
}
```

5. Save the class. You can now use *FormData* in the trigger condition evaluation.
6. Change the trigger condition to:

   FormData["Email"].EndsWith("@example.com")

7. Submit the form using "@example.com" in the E-mail field.

The trigger automatically starts the process.