Create and design forms CRM 2015

In Microsoft Dynamics CRM, forms provide the user interface that people will use to interact with the data they need to do their work. It is important that the forms people use are designed to allow them to find or enter the information they need efficiently.

This topic introduces how forms vary by groups of entities, the different types of forms available, and how you can control access to forms. For information about the elements and properties of forms

Types of forms

The following table describes the types of forms in Microsoft Dynamics CRM:

Form Type	Description
Main	Used in the web application, CRM for Outlook and CRM for tablets. These forms provide the main user interface for interacting with entity data. More information: Design considerations for main forms
Mobile	Used for the Microsoft Dynamics CRM for phones pages. This simplified form is designed to be used for mobile devices. The mobile forms for updated entities are not changed. More information: Create and edit mobile forms
Quick Create	Used in the web application, CRM for Outlook and CRM for tablets. For updated entities, these forms provide a basic form optimized for creating new records. More information: Create and edit quick create forms
Quick View	Used in the web application, CRM for Outlook and CRM for tablets. For updated entities, these forms appear within the main form to display additional data for a record that is referenced by a lookup field in the form. More information: Create and edit quick view forms

Use the form editor

This topic explains how to access the form editor, the features it contains, the form elements you can edit, and the properties of those elements.

1. Open the form editor
2. Form editor user interface
3. Form properties
4. Visibility options
5. Tab properties
6. Section properties
7. Common field properties
8. Special field properties
9. Sub-grid properties
10. Quick view control properties
11. Web resource properties
12. IFRAME properties
13. Notes control
14. Configure Bing maps
15. Edit navigation
16. Timer control
17. Configure event handlers
18. Open the form editor

You can access the form editor through the command bar or the ribbon, depending on the entity. Both of these methods will open the form in the context of the default solution. If you create any new solution components in the process of editing the form, for example web resources, the names of the components will use the solution publisher customization prefix for the default solution and these components will only be included in the default solution. If you want any new solution components to be included in a specific unmanaged solution, you should open the form editor through that unmanaged solution.

To access the form editor through the command bar

---

1. View a record for one of the Updated Entities, for example, open an account record.
2. If there are multiple main forms for the entity, verify that the form is the one you want to edit. If it is not, use the form selector to choose the form you want to edit.
3. In the command bar, choose the **More Commands** button ⋯.
4. Choose **Form**.

This will open the form editor for this form using the default solution.

To access the form editor through the ribbon

---

1. View a record for one of the Entities using classic forms.

2. If there are multiple main forms for the entity, verify that the form is the one you want to edit. If it is not, use the form selector to choose the form you want to edit.
3. On the **Customize** tab of the ribbon, in the **Design** group, select **Form**.

This will open the form editor for this form using the default solution.

To access the form editor through the default solution

---

1. Choose **Settings** > **Customizations**.
2. Choose **Customize the System** to open the default solution.
3. Under **Components**, expand **Entities**, and then the entity you want, and select **Forms**.
4. In the list of forms, select the form you want to edit.

To access the form editor for an unmanaged solution

---

1. Choose **Settings** > **Customizations**.
2. Choose **Solutions**.
3. Double-click the unmanaged solution you want to work with.
4. Locate the entity with the form you want to edit. If the entity isn't there, you'll need to add it.

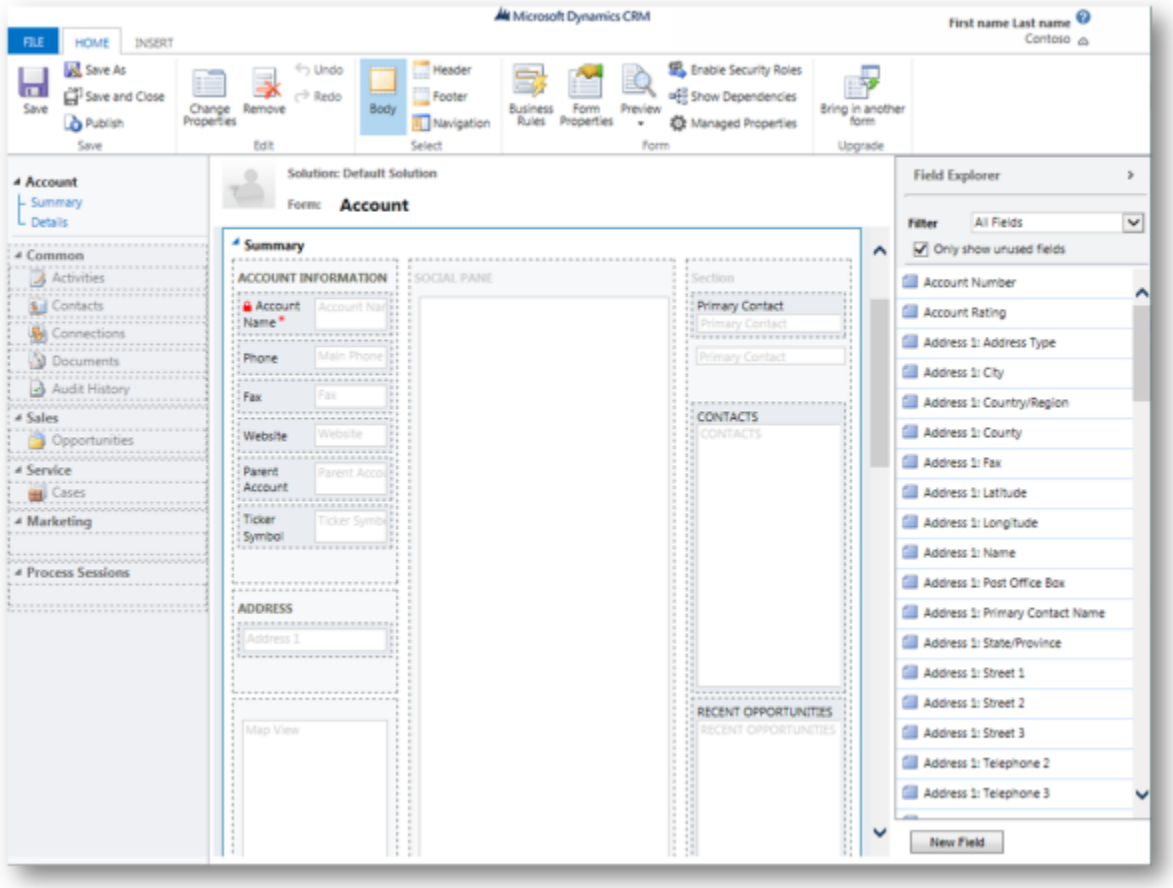
Add an entity to an unmanaged solution

---

    1. Select the **Entities** node and, in the toolbar above the list, choose **Add Existing**.
    2. In the **Select Solution Components** dialog box, with the **Component Type** selector set to **Entity**, select the entity you want to add and choose **OK**.
    3. If the **Missing Required Components** dialog box appears, you can choose **No, do not include required components** if you don't intend to export this unmanaged solution to another organization. If you choose not to include missing required components at this time, you can add them later. You'll receive notification again if you export this solution in the future.
5. In the solution explorer expand the entity with the form you want to edit and select **Forms**.
6. In the list of forms, double-click the form you want to edit.

Form editor user interface

The form editor displays commands in two ribbon tabs: **Home** and **Insert**. For details about the commands available there, see Home tab and Insert tab.



The body of the form editor is divided into three areas: **Navigation**, **Body**, and **Explorer**.

Navigation area

Located on the left side, use the navigation area to control access to related entities or to add links to web resources or URLs to be displayed in the main pane of the form. To edit navigation you must first select the **Navigation** command in the **Select** group of the **Home** tab.

Forms for Entities using classic forms provide a navigation experience that is visually similar to what you see in the navigation area. Forms for Updated Entities provide navigation options through the navigation bar, but use the same data to control what navigation options are available. More information: Edit Navigation

Body area

Located in the center, use the body area to control the layout of the form. You can select and drag form elements to position them. Double-clicking on an element will open the properties for the element.

- To add a field, select it from the **Field Explorer** and drag it into a section.
- To add an element that is not a field, select where you want to place it and use the appropriate command from the **Insert** tab add it.
- To remove an element, select it and use the **Remove** command in the **Edit** group of the **Home** tab.
- To edit the **Header** or **Footer** for the form you must first select the corresponding command in the **Select** group of the **Home** tab.

Explorer area

Located on the right side, the content of the explorer area depends on the context.

When you select **Body**, **Header**, or **Footer** in the **Select** group of the **Home** tab, you'll see the **Field Explorer**. Use the **Field Explorer** to drag fields you want to display into a section in the form or within the header or footer. You can include the same field multiple times in a form. Use the **New Field** button as a shortcut to create a new field.

When you select **Navigation** in the **Select** group of the **Home** tab you'll see the **Relationship Explorer**. Drag any of the relationships into one of the groups within the navigation area. You cannot add the same relationship twice. Relationships are available based on how they are configured. If you configure a relationship to not display, it won't display in the **Relationship Explorer**. For information about how to configure default display options for relationships, see Navigation Pane Item for Primary Entity.

You can use the **New 1:N** and **New N:N buttons** as a shortcut to add new entity relationships.

Home tab

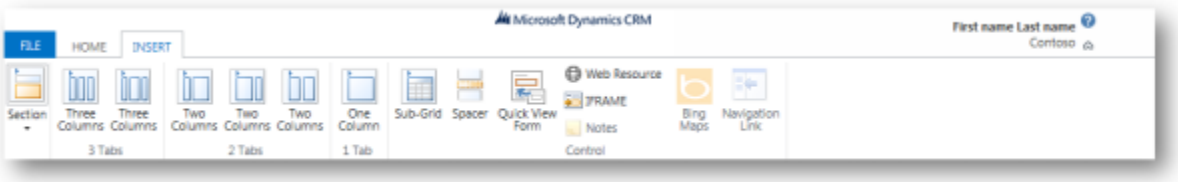
The **Home** tab displays the commands in the following table.

Group	Command	Description
	Save (Ctrl+S)	Save the form.
	Save As	Create a copy of this form with a different name.
	Save and Close	Save the form and close the form editor.
	Publish	Publish the form. More information: Publishing customizations

	Change properties	Change properties of the selected item in the body. See the following sections depending on the selected item: • Tab Properties • Section properties • common Field properties • Special field properties • Sub-grid properties • Quick view controlm properties • Web resource properties • IFRAME properties • Notes control • Configure Bing maps
	Remove	Remove the selected item.
	Undo (Ctrl+Z)	Undo the previous action.
	Redo (Ctrl+Y)	Redo the previous action.
	Body	Edit the main body of the form.
	Header	Edit the form header.
	Footer	Edit the form footer.
	Navigation	Edit the form navigation. More information: Edit Navigation
	Business Rules	View, Edit, or Create new Business Rules with the Business Rules explorer. More information: Create and edit business rules
	Form Properties	More information: Form Properties

	Preview	Preview how the form will look after it is published. The options are: • **Create Form**: How the form will appear before a record is saved. • **Update Form**: How a form for an existing record will appear. • **Read-Only Form**: How the form will appear for people who have only read access to a record. Scripts in the form can be tested but certain scenarios, like checking data values in the **OnLoad** event, can't be tested because the preview form doesn't contain data.
	Enable Security Roles	Use this to set which security roles will have access to the forms. More information: Control access to forms **Important** If you create a new form only the System Administrator and System Customizer security roles will have access to the form. You must assign access to other security roles before people in your organization can use it.
	Show Dependencies	See which solution components depend on this form and which solution components are required by this form. More information: Solution dependencies
	Managed Properties	The only managed property is **Customizable**. Setting this to false means the form won't be customizable after you included it in a solution, export that solution as a managed solution, and import that managed solution into a different organization. More information: Managed properties
	Merge Forms	Use this setting to merge a form from a previous version after you upgrade. This will facilitate adopting new form layouts introduced in this version. The form you bring in will be appended to the bottom of the current form. Use this to combine forms while preserving event handers for form scripts.

Insert tab



The **Insert** tab displays the commands in the following table:

Group	Command	Description
	Section	Add a section to a selected tab. You can choose to include a section with one to four columns. More information: Section properties
3 Tabs	**Three Columns**	Insert a three-column tab with equal widths. More information: Tab Properties
	Three Columns	Insert a three-column tab with a wider middle column.
2 Tabs	**Two Columns**	Insert a two-column tab with a wider right column.
	Two Columns	Insert a two-column tab with a wider left column.
	Two Columns	Insert a two-column tab with equal width columns.
1 Tab	**One Column**	Insert a one-column tab.
Control	**Sub-Grid**	Format a sub-grid and insert it into the form. More information: Sub-grid properties
	Spacer	Insert an empty space.
	Quick View Form	Insert a Quick View Form. More information: Quick view control properties
	Web Resource	Insert a web resource. More information: Web resource properties
	IFRAME	Insert an IFRAME. More information: IFRAME properties
	Notes	Insert a control to view activities, posts, and notes. More information: Notes control

	Bing Maps	Insert a control to show maps in the form. More information: Configure Bing maps
	Navigation Link	Insert a navigation link into the navigation area. This command is disabled unless you select the **Navigation** command in the **Select** group of the **Home** tab. More information: Navigation Link Properties
	Timer	Insert a timer control. More information: Timer control

Form properties

The properties of the form are in the following table:

Tab	Property	Description
Events	**Form Libraries**	Manage which JavaScript web resources are available in the form and the order in which they will be loaded.
	Event Handers	Configure which JavaScript functions from the Form Libraries will run for the **OnLoad** and **OnSave** form events and the order in which they'll be run.
Display	**Form Name**	Enter a name that will be meaningful to people. This name will be shown to people when they use the form. If they can use multiple forms configured for the entity they will use this name to differentiate between available forms.
	Description	Enter a description that explains how this form is different from other main forms. This description is only shown in the list of forms for an entity in the solution explorer.
	Page Navigation	You can choose not to show navigation items. In forms for updated entities this means the primary name value for the record currently being viewed will not appear in the navigation bar to allow navigation to associated views. In forms using the classic presentation, the navigation options to choose associated views on the left side of the form will not be shown.
	Image	When an entity has an image field and the entities' **Primary Image** option is set, this setting will enable showing the image field in the header of this form. See Enable or disable options for more information about entity options.
	Display	**Set a Max Width (in pixels)** to limit the width of the form. The default value is 1900.

Parameters	**Parameters**	Each form can be opened with code using a URL. The URL may also contain data that can be passed to the form using a query string that is appended to the URL. Query strings look like this example: ?p_firstName=Jim&p_lastName=Daly As a security measure, forms will not accept any unknown query string parameters. Use this parameters list to specify parameters this form should accept to support code that will pass data to the forms using a query string. The name and type of data will be checked and the form won't open if invalid query string parameters are passed to it. For more information see the topic Open Forms, Views, Dialogs and Reports with a URL in the Microsoft Dynamics CRM SDK.
Non-Event Dependencies	**Dependent Fields**	Each event handler has a similar **Dependent Fields** property so that any fields that are needed by the script can be registered. Anyone who tries to remove the dependent fields will not be able to. Some scripts operate on the form but are not configured in an event handler. Scripts that are initiated from the command bar do not have a place where dependent fields can be registered. This form property provides a place for dependent fields for those scripts to be registered.

Visibility options

Several types of form elements have the option to be shown or hidden by default. Tabs, sections, fields, IFRAMEs, and web resources all provide this option. Using form scripts or business rules the visibility of these elements can be controlled to create a dynamic form to provide a user interface that adapts to conditions in the form.

Note
Hiding form elements is not a recommended way to enforce security. There are several ways people can view all the elements and data in the form when elements are hidden.

The Microsoft Dynamics CRM for Microsoft Office Outlook reading pane presentation does not support form scripts. This presentation will use whatever the default visibility options are set for the form.

Rather than designing forms that depend on scripts to control visibility of options, consider whether a business process flow, a dialog, or switching to a different form may be better suited

to meet your requirements. If you do use scripts, make sure that any element that might be hidden is hidden by default. Only show it with scripts when your logic calls for it. This way it will not be displayed in presentations that do not support scripts.

Tab properties

In the body of the form tabs provide horizontal separation. Tabs have a label that can be displayed. If the label is displayed tabs can be expanded or collapsed to show or hide their content by choosing the label.

Tabs contain up to three columns and the width of each column can be set to a percentage of the total with. When you create a new tab, each column is pre-populated with a section.

The following table shows properties that may be set for tabs in the form.

Tab	Property	Description
Display	**Name**	**Required**: The unique name for the tab that is used when referencing it in scripts. The name can contain only alphanumeric characters and underscores.
	Label	**Required**: The localizable label for the tab visible to users.
	Show the label of this tab on the Form	When the label is displayed people can click it to toggle whether the tab is expanded or collapsed. Choose whether you want to show the label.
	Expand this tab by default	The tab state can toggle between expanded or collapsed using form scripts or by people clicking the label. Choose the default state for the tab.
	Visible by default	Showing the tab is optional and can be controlled using scripts. Choose whether to make the tab visible. More information: Visibility Options
	Lock the tab on the form	This will prevent the tab from accidentally being removed and prevents people from modifying the contents. Removing a tab will not only remove the tab, but also any script event handlers defined for the tab or fields within the tab. Recreating all this work could be a substantial effort. Someone wanting to remove this tab would need to change this setting before removing it.
Formatting	**Layout**	Tabs may have up to three columns. Use these options to set the number of tabs and what percentage of the total width they should fill.

Events	**Form Libraries**	Specify any JavaScript web resources that will be used in the tab **TabStateChange** event handler. See the SDK Form Events Reference : Tab TabStateChange Event topic
	Event Handers	Configure the functions from the libraries that should be called for the tab **TabStateChange** event. More information: Configure Event Handlers

Section properties

A section occupies the space available in a tab column. Sections have a label that can be displayed and a line may be shown below the label.

Sections can have up to 4 columns and includes options for displaying how labels for fields in the section are displayed.

Headers and footers are similar to sections but cannot be removed. If they don't contain anything they will not be shown.

Tab	Property	Description
Display	**Name**	**Required**: The unique name for the section that is used when referencing it in scripts. The name can contain only alphanumeric characters and underscores.
	Label	**Required**: The localizable label for the section visible to users.
	Show the label of this section on the form	Sections are frequently used without labels to control formatting of the fields within them.
	Show a line at top of the section	A line at the top of a section can help break up the form layout.
	Field Label Width	**Required**: Set a value between 50 and 250 to specify the space allowed for field labels. Header and footer elements also have this property.
	Visibility	Showing the section is optional and can be controlled using scripts. More information: Visibility Options

	Lock the section on the form	This will prevent the section from accidentally being removed and prevents people from removing the contents. Removing a section will not only remove the section, but also any fields within it. Someone wanting to remove this section would need to change this setting before removing it.
Formatting **Header and footer components also have this property.**	**Layout**	Specify up to four columns to be in the section.
	Field Label Alignment	Labels for fields within the section can be aligned left, right, or center.
	Field Label Position	Labels for fields within the section can be positions on the side or on top of the fields.

Common field properties

Fields display controls people use to view or edit data in an entity record. Fields can be formatted to occupy up to four columns within a section.

The following table describes properties that all fields have. Certain types of fields have special properties. These are described in Special field properties.

Tab	Property	Description
Display	**Label**	**Required**: By default the label will match the display name of the field. You can override that name for the form by entering a different label here.
	Display label on the form	You can choose not to display the label at all.
	Field is read-only	You can specify that the field is not editable. Using form scripts you can change this to enable or disable editing based on criteria evaluated in the script.
	Lock the field on the form	This will prevent the field from being removed from the form accidentally. This will prevent any configuration you have applied to the field, such as event handlers, from being cleared if the field were removed. To remove this field a customizer would need to clear this setting first.
	Visible by default	Showing the field is optional and can be controlled using scripts. More information: Visibility Options
Formatting	**Select the number of columns the control occupies**	When the section containing the fields has more than one column you can set the field to occupy up to the number of columns that the section has.

Details	**Display Name**, **Name**, and **Description**	These read-only fields are for reference. Choose the **Edit** button for convenient access to the field definition if you want to edit it. Each instance of a field in the form has a name property so that they can be referenced in form scripts, but this name is managed by the application. The first instance of the field is the name of the field specified when it was created. More information: Creating and editing fields For each additional time that a field is included in a form, the name appends a number starting with 1 to the end. So if the field name is 'new_cost', the first instance is 'new_cost', the second is 'new_cost1', and so on for each instance of the field in the form. **Note** The field **Description** value provides tooltip text for the field when people place their cursor over it.
Events	**Form Libraries**	Specify any JavaScript web resources that will be used in the field **OnChange** event handler. See the SDK Form Events Reference : Field OnChange Event
	Event Handlers	Configure the functions from the form libraries that should be called for the field **OnChange** event. More information: Configure Event Handlers
Business Rules	**Business Rules**	View and manage any business rules that reference this field. More information: Create and edit business rules

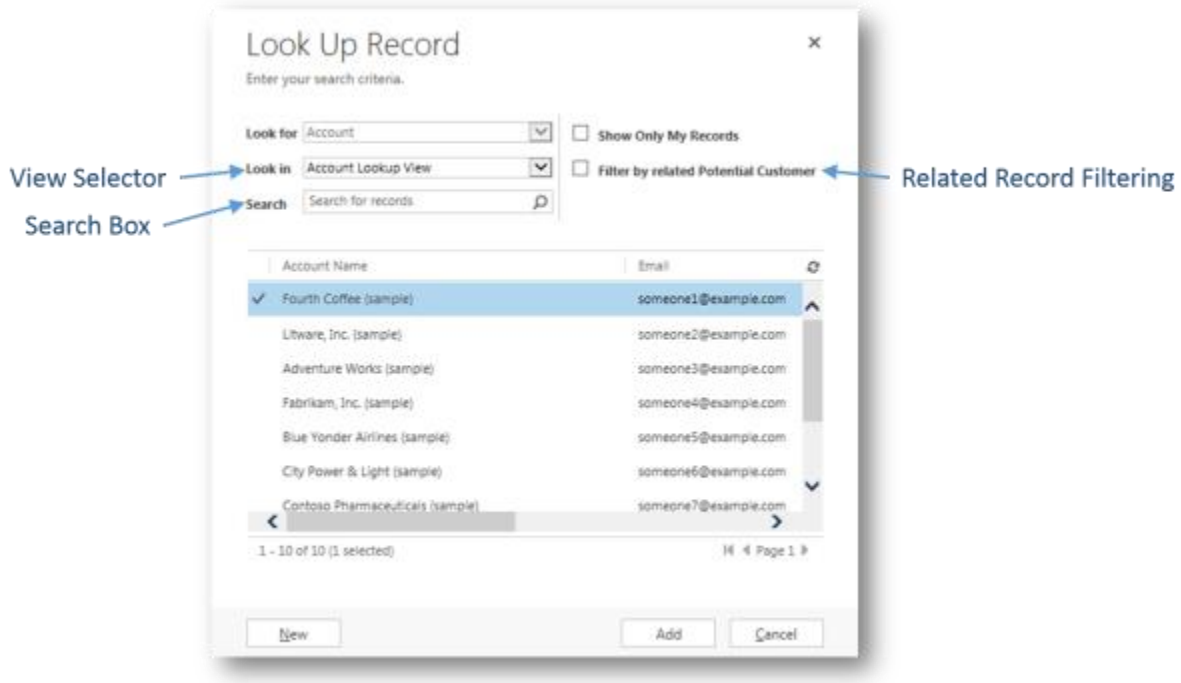
Special field properties

All fields have the properties listed in Common field properties, but certain fields have additional properties.

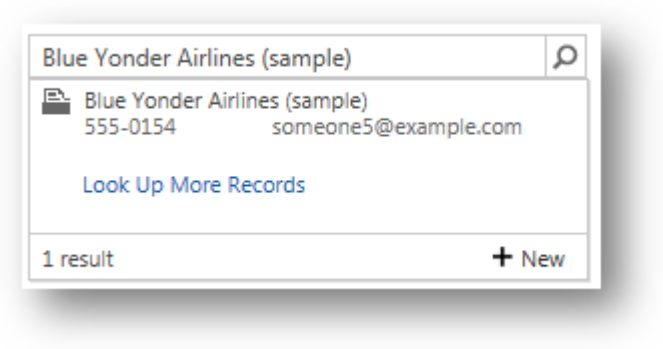
Lookup field properties

On the **Display** tab, lookup fields have some additional properties. Some system fields that look like lookup fields and have similar behaviors are Owner, Customer, PartyList and Regarding lookups. These fields are different from lookups because they allow for setting multiple values or multiple types, or both. These fields have only the first two properties: **Turn off automatic resolutions in the field** and **Disable most recently used items for this field**.

This is an example of the lookup dialog shown when people choose the **Look Up More Records** option when setting the value for a lookup.



Property	Description
Turn off automatic resolutions in the field	Only main forms using the Classic forms support automatic resolution. This can be disabled with this setting.
Disable most recently used items for this field	Only main forms using the Classic forms support most recently used items. This can be disabled with this setting.
Related Records Filtering	When this is enabled the records displayed when someone searches for a record will have additional filtering applied. This helps provide more relevant searches when setting the value of the lookup. You can also allow users to turn off the filter.
Display Search Box in lookup dialog	You can choose not to display the search box in the lookup dialog.

Default View	This view will be used to filter the results of the inline search and specify the default view shown in the lookup dialog if people choose the **Look Up More Records** option. The default view also controls which fields are included in the inline lookup.  For lookups that only allow selection of a single type of entity, the fields displayed in the inline lookup are set to be the first two fields included in the default view. In this example, **Main Phone** and **Email** are the first two columns in the default view configured for an account lookup. For system lookups that allow for multiple types of entities, the first two columns of the entity lookup view are shown.
View Selector	You can choose from three options: • **Off**: Do not allow people to choose a different view. • **Show All Views**; All views are available. • **Show Selected Views**: When you choose this option you can use the Ctrl key and your cursor to choose which views to show. The Lookup view for the entity cannot be de-selected.

Two option field properties

On the formatting tab, two option fields have the following formatting options

- **Two radio buttons**: Two labeled controls with labels. Only one may be selected.
- **Checkbox**: A single checkbox to set the true value, otherwise false.
- **List**: A drop-down list containing both values.

Multiple lines of text field properties

Multiple lines of text and single line of text fields that use the **Text Area** format have a **Row Layout** property. With this property you can specify a value for **Number of Rows** or select **Automatically expand to use available space**.

Sub-grid properties

You can configure a sub-grid to display a list of records or a chart. Select **Show Chart Only** on the **Display** tab to show a chart instead of a list.

Tab	Property	Description
Display	**Name**	**Required**: The unique name for the sub-grid that is used when referencing it in scripts. The name can contain only alphanumeric characters and underscores.
	Label	**Required**: The localizable label for the sub-grid visible to users.
	Display label on the Form	Whether the label should be displayed on the form. This is required if you enable **Display Search Box**.
	Records	Choose from two options: • **Only Related Records**: Sub-grid will display only records related to the current record. • **All Record Types**: Sub-grid will display records filtered only by the default view or, if the view selector is enabled, any views the user chooses. The option you choose will affect the behavior of the show list control. More information: Show list behavior
	Entity	Depending on the option you choose for **Records**, this list displays either: • **Only Related Records**: A list of entities that are related to this entity with the name of the lookup field on that entity which defines the relationship in parentheses. • **All Record Types**: A list of all entities.
	Default View	Choose the view that will be applied by default. If you do not enable any other views using the **View Selector** property. This will be the only view. Use the **Edit** button to open the default view for editing. Use the **New** button to create a new view to use for this sub-grid.

	Display Search Box	Display the search box. When this option is chosen the **Display Label on the Form** option is required.
	Display Index	Only forms using the Classic forms support display index. Select this check box if you want the alphabetical index to be available with the list. This lets you jump to records starting with a particular letter or number.
	View Selector	You have three options: • **Off**: Only the default view can be used. • **Show All Views**: Allow people to choose any view. • **Show Selected Views**: Use the Ctrl key with your cursor to select which of the available views to show.
	Default Chart	Select which chart to show if **Show Chart Only** is selected.
	Show Chart Only	Rather than a list of records a chart will be displayed.
	Display Chart Selection	If **Show Chart Only** is selected, allow people to choose different charts.
Formatting	**Layout**	**Select the number of columns the control occupies**. When the section containing the sub-grid has more than one column you can set the field to occupy up to the number of columns that the section has.
	Row Layout	**Number of Rows** will determine how many records are shown on a page of a sub-grid. If **Automatically expand to use available space** is chosen the form will allow space for two records and will expand the space as the number of records increases. If the number exceeds the **Number of Rows**, people can navigate to additional pages to view the records. If **Automatically expand to use available space** is not chosen the form will provide space for the number of records defined by **Number of Rows** and people can navigate to additional pages to view any additional records.

In forms using the Classic presentation, actions performed on a sub-grid were available in the ribbon. Developers can customize the behavior of these actions or add additional actions by customizing the ribbon.

In forms using the Updated forms actions for sub-grids are placed near the sub-grid, making them easier to access. However the command bar does not allow for custom actions to be added. Developers can edit the ribbon to modify the actions for the remaining three actions: show list, add record, and delete record.

Show list behavior

When displaying a list in forms with the Updated forms, each sub-grid displays the **Open View** button in the top right corner when the entity is also displayed as one of the entities included in the navigation area of the form editor. Choosing this button will open the view. The behavior will change depending on the option chosen for the **Records** property.

When you select **Only Related Records** the view will open using one of the associated views in the same window. To return to the form, use the back button or choose the current record primary name value in the navigation bar.

When you select **All Record Types** the view will open in a new window.

Add record behavior

When displaying a list in forms with the Updated forms, each sub-grid displays the **Add record** button in the top right side of the sub-grid. Choosing this button will allow you to add a record. This behavior will change depending on the option chosen for the **Records** property and if the lookup is for activity records.

When you select **Only Related Records** the default behavior is the behavior to add existing records. People see an in-line lookup to search for an existing record first. This helps prevent creating duplicate records. If they can't find an existing record, they can choose the **New** option. When a new record is created any of the field mappings defined in the relationship will be applied. More information: Mapping entity fields

When you select **All Record Types** the default behavior is to add a new record. The quick create form will be shown if the target entity has one. If not, the default entity main form is shown.

If the sub-grid displays activities, people will first need to choose the type of activity and then they will see the "add new record" behavior.

Delete record behavior

When you select a record in a sub-grid the **Delete** button appears on the right side of the row. The behavior of this delete action is different depending on the type of relationship with the current entity.

When the sub-grid uses a 1:N (one-to-many) relationship, the normal record delete behavior is to show a confirmation dialog before deleting the record.

When the sub-grid uses a N:N (many-to-many) relationship, the record in the relationship (or intersect) entity relating to two records is deleted without a confirmation and the record will no longer be displayed in the sub-grid. But the record that was displayed is not deleted.

Quick view control properties

A quick view control displays data from a record that is selected in a lookup on the form. The data displayed in the control is defined using a quick view form. The data displayed is not editable, but when the primary field is included in the quick view form, it becomes a link to open the related record. More information: Create and edit quick view forms

Property	Description
Name	**Required**: The unique name for the quick view form that is used when referencing it in scripts.
Label	**Required**: A label to display for the quick view form.
Display label on the Form	Displays the label on the form.
Lookup Field	Choose one of the lookup fields included in the form.
Related entity	This value depends on the **Lookup Field** you choose. It is usually the primary entity for the 1:N entity relationship for the lookup. If the entity includes a **Potential Customer** lookup that can accept either an account or contact, in the **Quick View Form** field you can choose a quick view form for both account and contact by changing this value and then choosing another quick view form.
Quick View Form	If the **Related entity** has any quick view forms you can select them here. Otherwise, choose **New** to create one. Choose **Edit** to change the selected quick view form.

Web resource properties

You can add or edit web resources on a form to make it more appealing or useful to users. Form enabled web resources are images, HTML files, or Silverlight controls.

For step-by-step instructions, see Add or edit a form web resource.

Tab	Property	Description

General	**web resource**	**Required**: The image, HTML, or Silverlight web resource that you want.
	Name	**Required**: A unique name for the field. The name can contain only alphanumeric characters and underscores.
	Label	**Required**: A label to display for the web resource.
	Visible by default	Showing the web resource is optional and can be controlled using scripts. More information: Visibility Options
	Custom Parameter	A custom value to pass as the **data** query string parameter. More information: Pass parameters to web resources
	Alternative Text	When an image web resource is displayed, this value will provide tooltip text for people using screen readers.
	Restrict cross-frame scripting, where supported.	When pages exist on different domains you may want to prevent them from accessing the content of your form pages. Web resources are always in the same domain, so this should not be an issue with web resources.
	Pass record object-type code and unique identifiers as parameters	Data about the organization, user, and the record can be passed to the web resource so it can adapt to organization settings. More information: Pass parameters to web resources
Formatting	**Select the number of columns the control occupies**	When the section containing the web resource has more than one column you can set the field to occupy up to the number of columns that the section has.
	Select the number of rows the control occupies	You can control the height of the web resource by specifying a number of rows.
	Automatically expand to use available space	You can allow the web resource height to expand to available space.
	Select the scrolling type for the IFRAME	An HTML web resource is added to the form using an IFRAME. • **As Necessary**: Show scrollbars when the size of the web resource is larger than the available. • **Always**: Always show scrollbars. • **Never**: Never show scrollbars.
	Display border	Display a border around the web resource.

Dependencies	**Dependent fields**	A web resource may interact with fields in the form using script. If a field is removed from the form the script in the web resource may break. Add any fields referenced by scripts in the web resource to the **Dependent fields** so that they cannot be removed accidentally.

Pass parameters to web resources

An HTML or Silverlight web resource can accept parameters to be passed as query string parameters.

Information about the record can be passed by enabling the **Pass record object-type code and unique identifiers as parameters** option. If information is typed into the **Custom Parameter(data)** field it will be passed using the data parameter. The values passed are:

Parameter	Description
data	This parameter is only passed when text is provided for **Custom Parameter(data)**.
orglcid	The Organization default language LCID.
orgname	The name of the organization.
userlcid	The user's preferred language LCID
type	The entity type code. This value can be different for custom entities in different organizations. Use entity type name instead.
typename	The entity type name.
id	The id value of the record. This parameter has no value until the entity record is saved.

Any other parameters are not allowed and the web resource will not open if other parameters are used. If you need to pass multiple values, the data parameter can be overloaded to include more parameters within it

IFRAME properties

You can add IFRAMEs to a form to integrate content from another website within a form.

Note
Microsoft Dynamics CRM forms are not designed to be displayed within IFRAMEs.

Tab	Property	Description
General	**Name**	**Required**: A unique name for the IFRAME. The name can contain only alphanumeric characters and underscores.
	URL	**Required**: The URL for the page to display in the IFRAME.
	Pass record object-type code and unique identifiers as parameters	Data about the organization, user, and the record can be passed to the IFRAME. More information: Pass parameters to IFRAMES
	Label	**Required**: A label to display for the IFRAME.
	Display label on the Form	Whether the label should be displayed.
	Restrict cross-frame scripting, where supported	It is considered a security risk to allow pages from a different web site to interact with the Microsoft Dynamics CRM application using scripts. Use this option to restrict cross frame scripting for pages you do not have control over. More information: Select Whether to Restrict Cross-Frame Scripting
	Visible by default	Showing the IFRAME is optional and can be controlled using scripts. More information: Visibility Options
Formatting	**Select the number of columns the control occupies**	When the section containing the IFRAME has more than one column you can set the field to occupy up to the number of columns that the section has.
	Select the number of rows the control occupies	You can control the height of the IFRAME by specifying a number of rows the control occupies.
	Automatically expand to use available space	Instead of setting the height by a number of rows, you can allow the IFRAME height to expand to available space.
	Select the scrolling type for the IFRAME	You have three options: • **As Necessary**: Show scrollbars when the size of the IFRAME is larger than the available space. • **Always**: Always show scrollbars. • **Never**: Never show scrollbars.
	Display border	Display a border around the IFRAME.

Dependencies	**Dependent fields**	An IFRAME may interact with fields in the form using script. If a field is removed from the form the script in the IFRAME may break. Add any fields referenced by scripts in the IFRAMES to the **Dependent fields** so that they cannot be removed accidentally.

Pass parameters to IFRAMES

Information about the record can be passed by enabling the **Pass record object-type code and unique identifiers as parameters** option. The values passed are:

Parameter	Description
orglcid	The Organization default language LCID.
orgname	The name of the organization.
userlcid	The user's preferred language LCID
type	The entity type code. This value can be different for custom entities in different organizations. Use **typename** instead.
typename	The entity type name.
id	The id value of the record. this parameter has no value until the entity record is saved.

Notes control

In forms for certain system entities using the Updated forms, the notes control provides the ability to access information about **POSTS**, **ACTIVITIES**, and **NOTES**. For custom entities where you have enabled notes and activities, you will only see **NOTES** and **ACTIVITIES**. To include **POSTS** you must enable them for the custom entity.

Enable posts for a custom entity

---

1. Navigate to **Settings** > **Post Configurations** and locate the record for your custom entity.
2. Make sure that **Enable walls for this type of record form** is selected and save the record.
3. In the command bar, select **ACTIVATE**.
4. If you needed to enable walls, you need to publish the entity.

By default, for system entities the notes control is positioned in a social pane section in the center of a three column tab at the top of the form. It can appear in a form just one time. You can move or remove the notes control. To add it back, use the **Notes** button in the **Control** group of the **Insert** tab.

The following table describes the properties for the Notes control.

Tab	Property	Description
Display	**Label**	**Required**: Although the label is not displayed by default, a label is required.
	Display Label on the form	You can choose to display the label.
	Lock the field on the form	This will prevent the notes from being removed from the form accidentally.
	Default tab	Select which tab should be displayed by default. The options are: • **Activities** • **Posts** • **Notes**
Formatting	**Select the number of columns the control occupies**	When the section containing the notes control has more than one column you can set the field to occupy up to the number of columns that the section has.
	Number of Rows	Control the height of the notes control by selecting the number of rows the control occupies.
	Automatically expand to use available space	Instead of setting the height by a number of rows, you can allow the notes control height to expand to available space.

Configure Bing maps

Bing Maps can be displayed in forms for the account, contact, lead, quote, order, invoice, competitor, and system user forms. You can remove the Bing Maps area in the form editor or add it back by using the **Bing Maps** button on the **Insert** tab of the form editor.

To enable Bing Maps the system setting **Show Bing Maps on forms** must be enabled. Microsoft Dynamics CRMon-premise organizations will need to enter a Bing Maps Key and enter it in the system setting **Please enter Bing Maps key**. Obtain a Bing Map key from: https://www.bingmapsportal.com. Microsoft Dynamics CRM Online subscribers do not require a key.

Tab	Property	Description

General	**Label**	**Required**: A label to display for the Bing Maps.
	Display label on the form	Whether the label should be displayed.
	Select an address to use with the Bing Maps control	Choose which address should be used to provide data for the map.
	Visible by default	Showing the Bing maps is optional and can be controlled using scripts. More information: Visibility Options
Formatting	**Select the number of columns the control occupies**	When the section containing the Bing Maps has more than one column you can set the field to occupy up to the number of columns that the section has.
	Select the number of rows the control occupies	You can control the height of the Bing Maps by specifying a number of rows.
	Automatically expand to use available space	You can allow the Bing Maps height to expand to available space.

Edit Navigation

Navigation within the form allows people to view lists of related records. Each entity relationship has properties to control whether it should be shown. More information: Navigation Pane Item for Primary Entity

Any entity relationships that are configured to be displayed can be overridden within the form editor. You can also include navigation links to display web resources or other web sites via form navigation.

For step-by-step instructions, see Add or edit form navigation for related entities

To enable editing navigation you must first select **Navigation** from the **Select** group on the **Home** tab.

In the **Relationship Explorer** you can filter by 1:N (one-to-many) or N:N (many-to-many) relationships, or view all available relationships. The **Only show unused relationships checkbox** is disabled and selected. So you can only add each relationship one time.

To add a relationship from the **Relationship Explorer** just double click it and it will be added below the currently selected relationship in the navigation area. Double-click a relationship in the navigation area and you can change the label on the **Display** tab. On the **Name** tab you can see information about the relationship. Use the **Edit** button to open the definition of the entity.

There are five groups in the navigation area. You can drag them to reposition them and double-click them to change the label, but you can't remove them. These groups will only display when there is something in them. So if you don't want a group to appear, just don't add anything to it.

Use the **Navigation Link** button in the **Control** group of the **Insert** tab to add a link to a web resource or external URL.

Navigation link properties

Navigation links have the following properties:

Property	Description
Name	**Required**: Text to display as a label.
Icon	Use a 32x32 pixel web resource. Use a PNG image with a transparent background is recommended.
Web Resource	Specify a web resource to display in the main pane of the form.
External URL	Specify the URL of a page to display in the main pane of the form.

Timer control

Use a timer control in forms where records need to meet a specific time-based milestone. A timer control shows people how much time is available to complete an action in the resolution of an active record or how much time has passed since the time to complete the action has passed. At a minimum, timer controls must be configured to show success or failure in completing the action. In addition, they can be configured to display warnings when the conditions are approaching failure.

A timer control can be added to a form for any entity, but they are most frequently used for the case entity, especially when linked to fields that track service level agreements. You can add multiple timer controls in the body of a form. You can't add them to the header or footer.

Timer control **Data Source** properties use fields for the entity.

- The **Failure Time Field** uses a date-time field to set the time.
- The three condition fields use one of the **Option Set**, **Two Options**, **Status**, or **Status Reason** fields for the entity.

Timer control properties

The following table describes the properties of a timer control.

Group	Name	Description

Name	Name	**Required**. A unique name for the control.
	Label	**Required**. The label to display for the timer control.
Data Source	Failure Time Field	**Required**. Choose one of the date-time fields for the entity to represent when a milestone should be successfully completed.
	Success Condition	**Required**. Select a field for the entity to evaluate the success of the milestone, then choose which option indicates success.
	Warning Condition	Select a field for the entity to evaluate whether the success of the milestone is at risk so that a warning should be displayed, then choose which option indicates that a warning should be displayed.
	Cancel Condition	Select a field for the entity to evaluate whether the achievement of th milestone should be cancelled, then choose which option indicates that the milestone is cancelled.

Configure event handlers

Form event handlers can be configured for the following areas in a form:

Element	Event	Description
Form	**OnLoad**	Occurs when the form loads.
	OnSave	Occurs when data is saved.
Tab	**TabStateChange**	Occurs when the tab is expanded or collapsed.
Field	**OnChange**	Occurs when data in the field changes and the control loses focus.
IFRAME	**OnReadyStateComplete**	Occurs when the content of an IFRAME loads.

An event handler consists of a reference to a JavaScript web resource and a function defined within that web resource that will execute when the event occurs. Each element can have up to 50 separate event handlers configured.

Important

Configuring an event handler incorrectly can result in script errors that may cause the form to fail to load or function correctly. If you are not the developer of the script, make sure you understand exactly what configuration options the script requires.

Do not configure a script event handler using a library that does not come from a source you trust. Scripts can be used to perform any action a user might perform and a poorly written script can significantly damage the performance of a form.

After you configure an event handler always test it to verify it is working correctly.

To configure an event handler

---

1. In the form editor, select the element with the event you want to configure a handler for.
2. On the Home tab, in the **Edit** group, choose **Change Properties** or simply double-click the element.
3. In the element properties dialog, select the **Events** tab.
4. Expand the **Form Libraries** area. If the library containing the function you want to set as the event handler is not already listed, add the library.
5. **To add a form library to an event handler**
    1. In the **Form Libraries** section of the **Event List**, choose **Add**.
    2. Locate the JavaScript web resource in the list of available web resources. Select it and choose **Add**.

       If the JavaScript web resource you need does not exist, choose **New** to open a new web resource form and create one.

    3. **To create a JavaScript web resource**
        1. In the web resource form set the following properties:

Property	Value
Name	**Required**. Type the name of the web resource.
Display Name	**Required**. Type the name to be displayed in the list of web resources.
Description	Optional. Type a description of the web resource.
Type	**Required**. Select **Script (JScript)**.
Language	Optional. Choose one of the languages available for your organization.

        2. If you have been provided with a script, we highly recommend that you use the **Browse** button to locate the file and upload it.

           Alternatively, you can choose the **Text Editor** button and paste or type the contents of the script in the **Edit Content** dialog.

Note

Because this simple text editor does not provide any features to check the correctness of the script, generally you should always try to use a separate application like Microsoft Visual Studio to edit scripts and then upload them.

        3. Choose **Save** and close the web resource dialog.

4. The web resource you created is now selected in the **Look Up Record** dialog. Choose **Add** to close the dialog.

6. When the script is available in the form you may adjust the order in which the script will be loaded by using the green arrows to move it up or down.
7. In the **Event Handlers** section, select the event you want to set an event handler for.
8. Choose **Add** to open the **Handler Properties** dialog.
9. On the **Details** tab choose the appropriate library and type the name of the function that should be executed for the event.
10. By default the event handler is enabled. Clear the **Enabled** checkbox if you do not want to enable this event.

    Some functions require an execution context to be passed to the function. Select **Pass execution context as the first parameter** if it is required.

    Some functions can accept a set of parameters to control the behavior of a function. If these are required, enter them in the **Comma separated list of parameters that will be passed to the function**.

11. On the **Dependencies** tab, add any fields that the script depends on into the **Dependent Fields** area.
12. Choose **OK** to close the **Handler Properties** dialog.
13. When the event handler is entered you may adjust the order in which the function will be executed relative to any other functions by using the green arrows to move it up or down.
14. Choose **OK** to close the element properties dialog.
15. Choose **Save** to save your changes. Choose **Publish** to publish the form.