MODULE 3: PAGES

Module Overview

In Microsoft Dynamics NAV 2013, you use pages to display, enter, and change data in the client. Pages provide an easy way to create a task-oriented, intuitive, and dynamic user interface. There are different types of pages that let the client present data in different ways. These pages let users interact productively with Microsoft Dynamics NAV 2013 to perform tasks.

Pages can access data from one table at a time, or they can combine information from several tables. A page can also display information that is calculated as the page is displayed, and it can also contain information that is not related to data from any table, such as labels or bitmap pictures.

Objectives

- Explain the concepts of pages and page components.
- Describe Page Designer and Action Designer.
- Create a simple page and add basic controls to the page.
- Provide an overview of different page types and their characteristics.
- Discuss best practices in designing pages.
- Create a Card page, add a container, FastTabs, and fields.
- Create a List page and link it to the Card page.
- Create a main page, a Part page, and link the two pages.

Page Fundamentals

Pages contain properties, triggers, and controls which define the behavior of the page. The page is shown in the client. However, the application logic in page triggers runs on the Microsoft Dynamics NAV Service Tier. This makes Microsoft Dynamics NAV 2013 a secure, scalable, and flexible application.

The following figure shows the components of a page and how they are related.

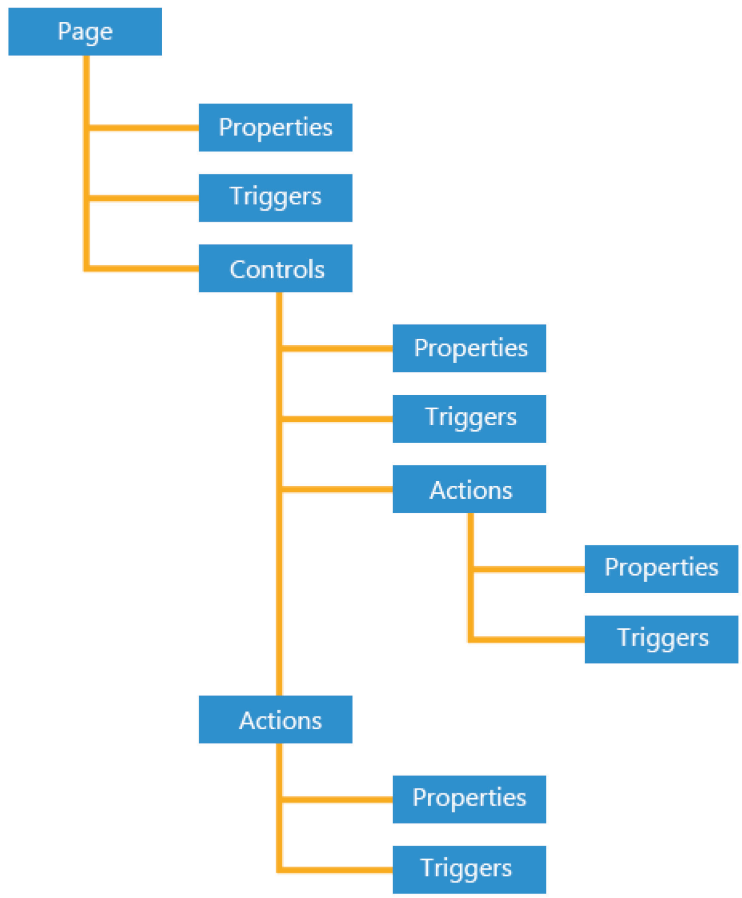


FIGURE 3.1: PAGE DEFINITION

You create and design pages in the Page Designer in Microsoft Dynamics NAV Development Environment.

Properties

Use page properties to define the behavior of the pages. For example, page properties such as InsertAllowed, ModifyAllowed, and DeleteAllowed, specify whether you can use a page only for displaying information, or whether you can insert new records, update, or delete existing records through the page.

Some page properties are common with other objects, such as **ID**, **Name,** or **Caption**, whereas many more properties are specific to pages, such as **PageType**, **CardPageID,** or **SourceTable**.

The following steps show how to open the Properties window for the Customer Card page.

To create a page in Microsoft Dynamics NAV Development Environment, do the following:

1. On the **Tools** menu, click **Object Designer**. The Object Designer opens.
2. Click **Page** to open the **Page** list.
3. Select page **21**, **Customer Card**, and then **Design** to design the Customer Card page. The Page Designer opens.
4. Scroll down to the blank line at the bottom of the page, and then click **View** > **Properties**, or click the **Properties** button on the Toolbar.
5. The **Page - Properties** window opens and shows the properties for the page. Here you can view and change properties for the Customer Card page.

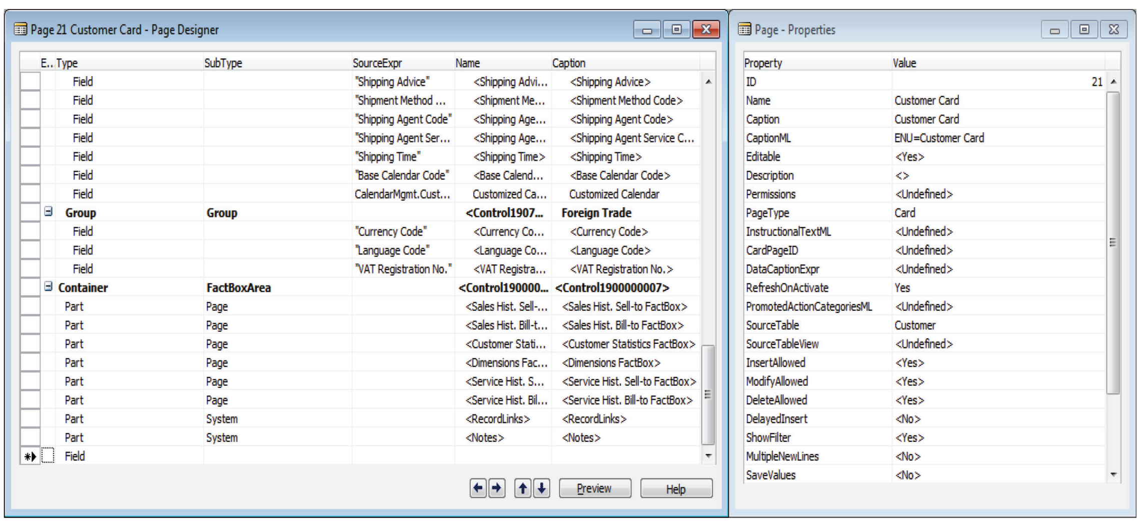


FIGURE 3.2: THE CUSTOMER CARD PAGE AND ITS PROPERTIES

Triggers

Certain predefined events that occur to a page or a control can cause the system to execute a user-defined C/AL function. The event and the function together are called a trigger.

Triggers in a page can be divided into the following three categories:

- Page triggers
- Control triggers
- Action triggers

Examples of Page triggers include OnOpenPage that contains statements that run when the page opens, and OnModifyRecord that contains statements that run before the system accepts changes that the user makes to a record. Triggers in a page are edited in the C/AL Editor which is accessed from the Page Designer.

Controls

Use controls to define the layout of a page. Controls can group or arrange other controls or display information on a page. Controls can display data from the following sources:

- A database table field
- The value of a C/AL expression
- Bitmap pictures
- Static information such as a descriptive text
- Other pages
- Predefined system features, such as **Outlook** or **Record Links**

Depending on the page type and their parent controls, certain controls may not be available all the time. For example, use Cues only within CueGroups on activity pages in role centers. Use FastTabs only on Card, ListPlus, and Document pages.

Control Properties

Control Properties determine the behavior and appearance of the control. For example, you can use page control properties to link a field to a specific field in a table, or to determine what happens when you enter information into a field. Different page controls can have different sets of properties. For example, container, group, and field controls each have their own unique sets of properties.

***Note:** You cannot explicitly specify positioning and sizing of a control. These two characteristics are determined by the client in relationship to other controls present on a page.*

The following steps show how to open the Properties window for one of the controls (**General** FastTab) in the **Customer Card** page.

1. Design page **21**, **Customer Card**, from the **Object Designer**.
2. Select the line of type **Group**, with caption General (the second line from the top in the figure).
3. On the **View** menu, click **Properties**, or click the **Properties** button on the Toolbar.
4. The **Properties** window opens and shows the properties for the control. Here you can view and change properties for the **General** FastTab.

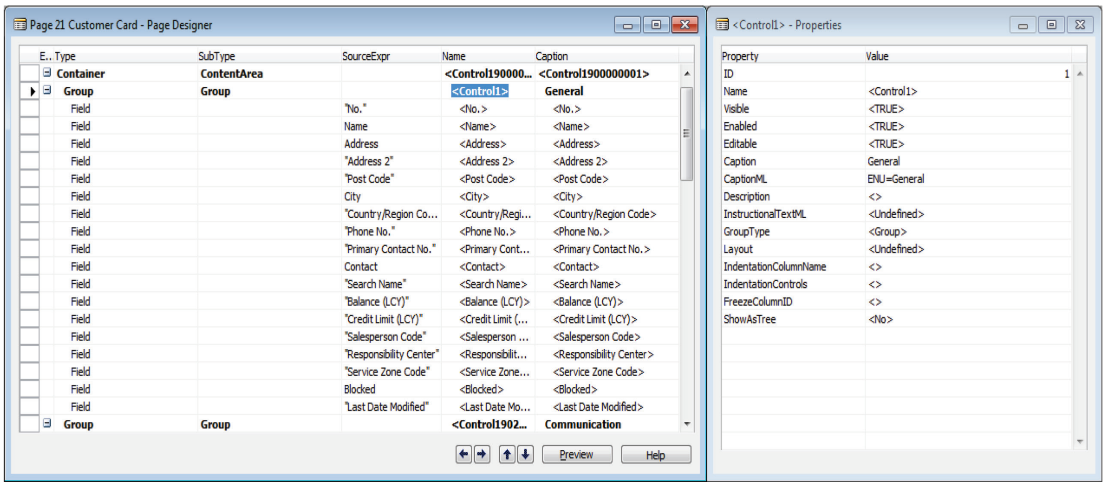


FIGURE 3.3: THE GENERAL FASTTAB PROPERTIES ON THE CUSTOMER CARD PAGE

Expression as Property Value

Several control properties allow for expressions as their value. This enables dynamic appearance of these controls, depending on the value of the expression. For example, in a page, a field's Visible property value is an expression. You can control the field's visibility by assigning an expression that returns TRUE or FALSE to its Visible property. The default value of properties that accept expression is <TRUE> instead of <Yes>.

Note: *Expression properties let you easily control the same behavior of several controls at the same time. By binding the same property, such as **Visible**, to the same Boolean variable, you can toggle visibility of multiple controls from a single line of code. Do not forget to set the **IncludeInDataset** variable property to **Yes**.*

Control Triggers

Only controls of type Field have triggers.

The following steps show how to view the triggers for the **No.** field in the Customer Card page.

1. Design page **21**, **Customer Card**, from the Object Designer.
2. Select the **No.** field (the line that has **SourceExpr** No. - the third line from the top) and then do one of the following:
   - Click View > **C/AL Code**.
   - Click the **C/AL Code** button on the Toolbar.
   - Right-click the **No.** field, and then click **C/AL Code** or press F9.
3. The C/AL Editor opens and displays the triggers for the **No.** field. Here you can write C/AL statements to program certain activities when a specific trigger runs, for example, when the user clicks the **AssistEdit(...)** button in the **No.** field.

Note: *The triggers for the Customer Card page are also shown together with the triggers for the* ***No.*** *field.*

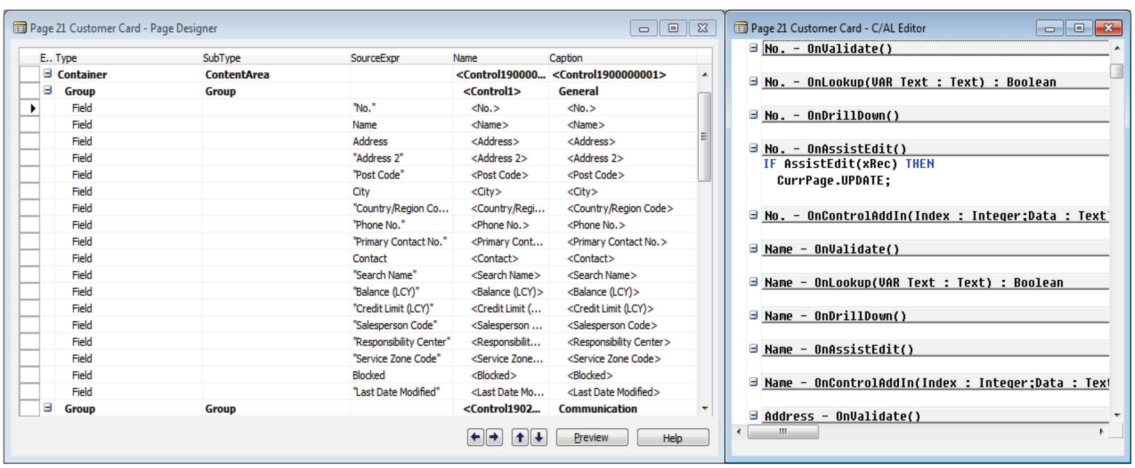


FIGURE 3.4: THE NO. FIELD TRIGGERS AND THE CUSTOMER CARD PAGE TRIGGERS

Bound and Unbound Pages and Controls

A page is frequently related to a table in a Microsoft Dynamics NAV 2013 database. Use a page to display and manage information from that table. The page is now *bound* to the table.

An unbound page is not related to any table. An example of an unbound page is a RoleCenter page that you use as a starting point for any activity in Microsoft Dynamics NAV 2013.

The controls on a page that are bound to a table are usually bound to fields in the same table. Not every control on the page is bound to a table field. Some controls may be bound to expressions or variables, whereas some controls may be not bound at all. Such controls are called *unbound* controls. An example of an unbound control is a FastTab.

The Ribbon

Each page has its own set of actions which show in the ribbon. The ribbon consists of tabs, groups, menus, and actions.

The following figure shows the elements of the ribbon.

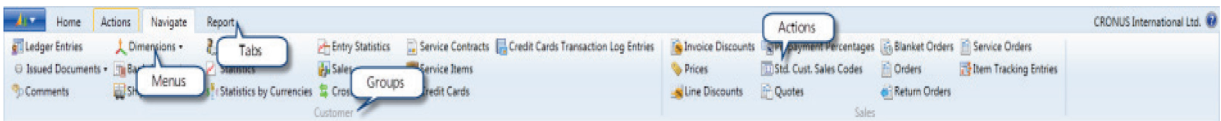


FIGURE 3.5: RIBBON STRUCTURE AND ELEMENTS

By default, the following four tabs are provided: Home, Actions, Navigate, and Report. You can define more tabs while you design the page, and users can define more tabs while they customize the ribbon.

Each tab has one or more groups. Some tabs contain default groups. However, you or the end-user may define other groups. For example, the Home tab has New, Process, and Report groups. The page type may also have Manage, View, and Show Attached groups. However, you can define more groups while you design the page.

The **Home** tab uses its group as containers for the promoted actions. Promoted actions are the most common actions. You can promote actions while you design the page. End-users can demote already promoted actions, or promote actions of their choice.

End-users can fully customize the ribbon, and can add, rearrange, and remove tabs, groups, menus, and actions.

Actions

In the client, buttons that are located in the ribbon are called *actions*. When a user clicks an action, the action starts a specific system-defined activity or runs a trigger.

You create and change actions from the **Action Designer** that is accessed from the **Page Designer**. Actions have their own properties and triggers. These triggers can be accessed from the Action Designer.

Actions can be *page actions* or *control actions*. Page actions are displayed in the ribbon at the top of the page. Most page types support page actions. Control actions are displayed in the control itself, and only the CueGroup control supports control actions.

The following steps show how to open the **Action Designer** and view the properties and triggers of an action in the **Customer Card** page.

1. Design page **21**, **Customer Card**, from the **Object Designer**.
2. On the **View** menu, click **Page Actions**, or press CTRL+SHIFT+F4.
3. The **Action Designer** opens, and shows all the actions in the Customer Card page.

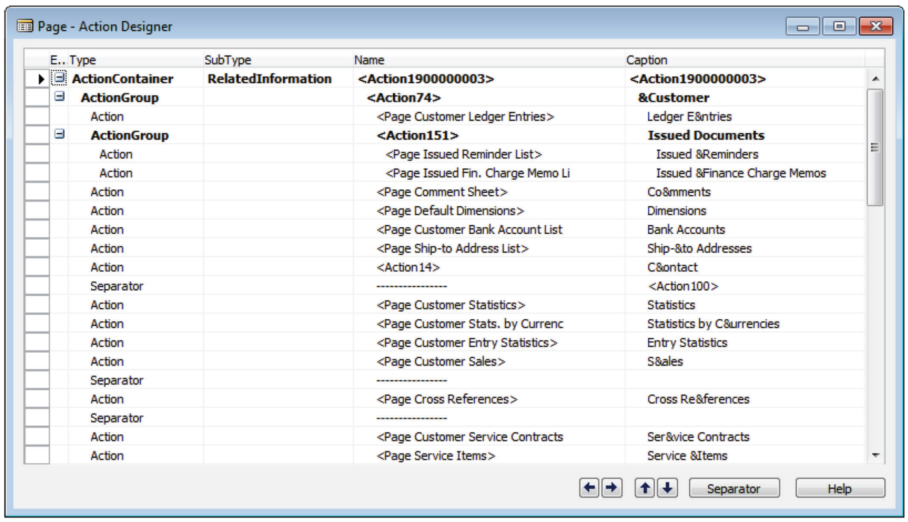


FIGURE 3.6: THE ACTION DESIGNER

4. Select any action (any of the lines), and then click **View** > **Properties**. The **Properties** window opens and shows the properties for the current action.
5. Close the **Properties** window.
6. Select any action, and then click **View** > **C/AL Code**. The **C/AL Editor** opens and shows the triggers for the current action.

Action Properties

Action properties define where and how the client displays the action in the ribbon, and what the action does when selected by a user. For example, changing the **Promoted** property from **No** to **Yes** promotes an action to the **Actions** tab in the ribbon. This makes it more prominent to end-users.

The following properties change the way an action is displayed:

Action Property	Purpose
Caption	Sets the caption of the action that appears in the client.
Promoted	Promotes an action to the Actions tab.
PromotedIsBig	Increases the size of the action icon in the ribbon, both for the promoted and regular actions.
PromotedCategory	Determines the group in the **Actions** tab in the ribbon where the action is displayed. There are three system-defined categories to select from: New, Process, or Report. There are seven user-defined categories that can be configured by setting the PromotedActionCategoriesML page property.
Image	Associates an icon with an Action.

Action properties also define the activity that is performed when users click the action, such as the **RunObject**, **RunPageLink**, and **RunPageMode** properties.

Action Triggers

Unlike page and control triggers, there is only one action trigger: the **OnAction** trigger. This trigger runs when a user clicks the action in the ribbon. You typically define the OnAction trigger when **RunObject** property is undefined.

Note: *You can define both the OnAction trigger and the RunObject property. If you define both, the client runs both the object that is defined in the RunObject property, and the C/AL code that is defined in the OnAction trigger.*

Page Designer

Use **Page Designer** to create new pages and change existing ones. You use Page Designer to define page controls and their relationships. Together they determine the content and the layout of the page. Page Designer also includes the Action Designer that lets you define page actions.

Control Type and Subtype

When you design a page, you specify a hierarchy of page elements and decide which data is displayed by an element. Instead of positioning the controls by specifying the X and Y coordinates, you define the order of the controls. The client determines the exact positioning at runtime. This eliminates pixel-level editing and guarantees a consistent user experience across the application.

The core development activity in page design is specifying the contents of the page in relation to the different Microsoft Dynamics NAV page types, such as List, Card, Document, or other page types.

There are several types of controls that can be added to a page. The following list groups controls into several broad categories.

Type	Purpose
Container	Acts as a placeholder for other controls.
Group	Arranges other controls in a logical manner.
Field	Displays data from a table field, or a value of a variable, constant, or an expression.
Part	Displays another page, a chart, or a system-defined feature, such as **Outlook** or **Record Links**.

Container Control

Every page element is a control with a specific type and subtype. Every page has a container control at the root of its control hierarchy. Following are three subtypes of container:

Subtype	Purpose
ContentArea	Is a topmost element of every page type, except for role centers.
RoleCenterArea	Replaces the ContentArea on pages of type RoleCenter.
FactboxArea	Contains FactBox controls in a page.

Every page must have at least one container control as the topmost element.

Group Control

Use group controls to group several other controls together. Following are the five group subtypes:

Subtype	Purpose
Group	Groups fields into FastTabs in card pages, or groups several controls together.
Repeater	Presents data in a tabular format, such as in list page.
CueGroup	Create Cues, such as in the SO Processor Activities page.

Subtype	Purpose
FixedLayout	Puts fields in a table or matrix-like configuration that has a row and column headings. You can only set up fields on a column-by-column basis. You typically use the FixedLayout control to display statistical data in a FastTab or information in the details section of a Worksheet page. FixedLayout shows captions as a heading to the rows and columns. You cannot specify a heading for each field. Fields cannot span rows and columns. Fields are noneditable and are displayed without a border.
GridLayout	Puts fields in a uniform grid. When you set up a grid layout, you can decide to set up fields in a row-by-row or a column-by-column configuration. With GridLayout, captions are shown to the left or above each field, fields can span rows and columns, and fields appear with borders, exactly like in a FastTab control.

Note: *To learn more about the differences between the FixedLayout and the GridLayout controls, refer to the Microsoft Dynamics NAV Developer and IT Pro Help.*

Field Control

Field control does not have a subtype, and is always used to display a value. The value that is displayed in a field control can come from any of the following sources:

- A database table field
- The value of a C/AL expression
- Bitmap pictures
- Static information, such as a descriptive text

Note: *When entering data in a page, users can press Enter to move between fields. You can control which fields are skipped by setting the* ***QuickEntry*** *property. By default this property is set to TRUE. If you set it to FALSE, this property prevents the field from having focus when a user repeatedly presses Enter.*

Part Control

Use part control to display another page or system data as a part of the current page. Following are the three subtypes of part control:

Subtype	Purpose
Page	Displays another page as a part of a parent page. Examples of page parts are lines in document pages, a FactBox list part, or card part pages. You specify the page to display by using the PagePartID property.
Chart	Displays a chart from the data that is contained in a table, or based on the results of a query. You specify the system part by using the SystemPartID property.
System	Displays a system part, such as **Outlook**, **Notes**, **MyNotes**, or **RecordLinks**. You specify the chart to be displayed by using the ChartPartID property.

Linking Pages and Page Parts

Pages and page parts are usually linked so that the page part shows additional information about the record that is currently shown in the main page. For example, the **Sell-to Customer Sales History** FactBox in the **Customer List** page always displays the sales history for the customer who is currently selected in the list.

Use the following properties to set the link between the page and its page part:

Property	Purpose
SubPageView	Sets a specific table view that is applied to the page part. You can choose the key, the sorting order, and the table filter that are applied to the page part.
SubPageLink	Sets up a link between the main page, and the page part. The link lets you filter the page part by field values on the main page, and to apply additional page filters.

Action Designer

Design page and control actions by using **Action Designer**. Action Designer is very similar to page designer: it groups actions into containers and groups of different subtypes. Access both page and control action designers from the **View** menu. **Page Actions** opens the action designer for the page actions, and **Control Actions** opens the action designer for the currently selected control.

***Note: Control Actions** is only enabled for groups of **CueGroup** subtype, and is disabled for all other control types.*

When a page is displayed, page actions are positioned in a different tab and group in the ribbon, depending on the action container and group to which they belong. Control actions are positioned next to the cues.

The following figure shows how the actions are displayed in a CueGroup control.

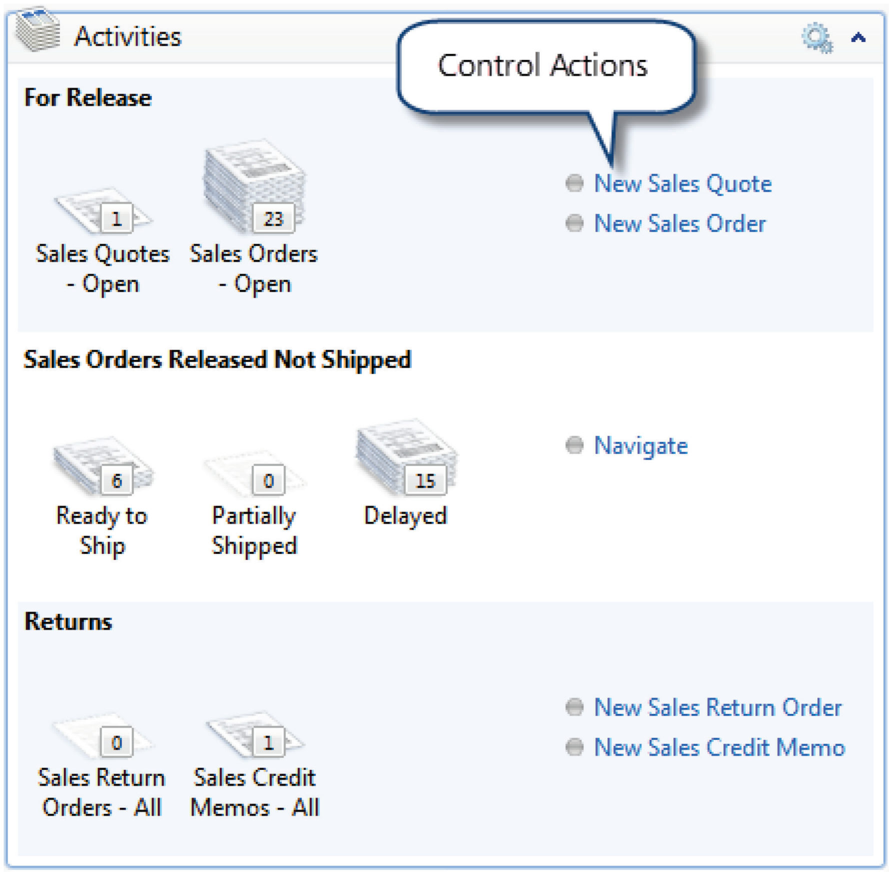


FIGURE 3.7: CUEGROUP CONTROL ACTIONS

Action Type and Subtype

Following are the three types of action in page objects:

- Action Container
- Action
- Action Group

Action Container

Action Container is the top level element in the hierarchy and must always have a subtype. Other action types that are located under it do not have subtypes. Use subtypes to decide the location of the action in the page. Depending on the subtype, an action can be put in the following positions in the ribbon or in the navigation pane:

Subtype	Position	Page Type	Figure
NewDocumentItems	In the **New Document** group, on the **Actions** tab in the ribbon.	All pages	3.7-1
ActionItems	In the **General** group, on the **Actions** tab in the ribbon.	All pages	3.7-2
RelatedInformation	On the **Navigate** tab. Each group in a **RelatedInformation** container is displayed as a separate group of the **Navigate** tab in the ribbon.	All pages	3.8-3
Reports	On the **Report** tab in the ribbon.	All pages	3.9-4
HomeItems	On the **Home** menu in the navigation pane.	Role Centers	3.10-5
ActivityButtons	In separate menus in the navigation pane. For example, **Posted Documents** in the **Sales Order Processor** role center is an activity button.	Role Centers	3.10-6

Actions that belong to **NewDocumentItems** and **ActionItems** action containers are put on the **Actions** tab.

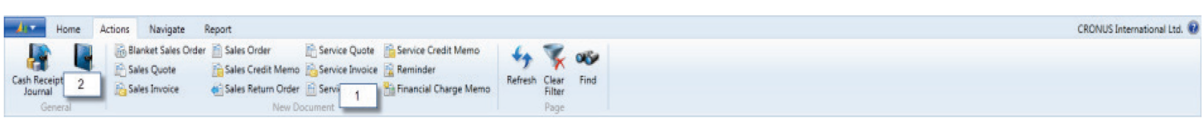


FIGURE 3.8: ACTION PLACEMENT FOR NEW DOCUMENT ITEMS AND ACTION ITEMS

Actions that pertain to **RelatedInformation** action container are put on the **Navigate** tab.

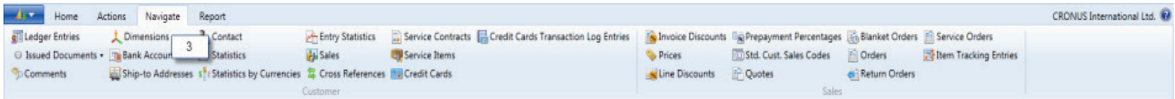


FIGURE 3.9: ACTION PLACEMENT FOR RELATED INFORMATION ACTIONS

Actions that belong to **Reports** action container are put on the **Reports** tab.

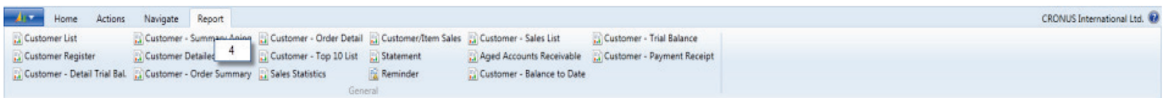


FIGURE 3.10: ACTION PLACEMENT FOR REPORTS ACTIONS

Actions that pertain to **HomeItems** and **ActivityButtons** containers are put in the navigation pane.

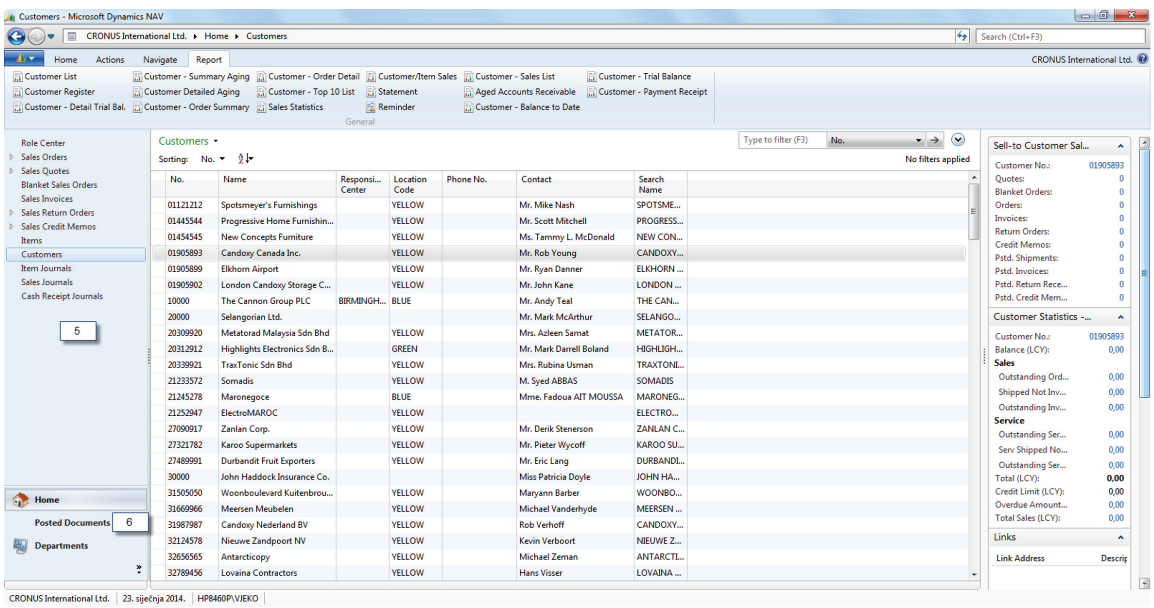


FIGURE 3.11: ACTION PLACEMENT FOR HOME ITEMS AND ACTIVITY BUTTONS ACTIONS

Action containers must be unique in a page. There cannot be two action containers with the same subtype. For example, there can be only one action container with the **Reports** subtype.

Action container must not be indented. Other actions under the action container must be indented one level or more, depending on placement in the hierarchy.

Action Group

Use Action Groups to group several related actions. For example, in **Customer List** page, several actions that show different information about a particular customer (**Ledger Entries**, **Issued Documents**, and **Bank Accounts**) are grouped into a **Customer** action group.

Action

Action represents a command that users can click to perform any of the following actions:

- Show a page
- Run a report or a codeunit
- Start an XMLport
- Run C/AL code that is defined in the action triggers.

Running Objects from Actions

Actions can run other objects or the C/AL code that is defined in the **OnAction** trigger. Most actions run other objects. To control which object is run and change the behavior of the object running action, specify the following action properties:

Property	Purpose
RunObject	Specifies which object runs when a user clicks the action. The value is a combination of the object type and object name. For example, if you want to run the **Sales Order** page, you must enter "Page Sales Order" in the **RunObject** property. ***Note:*** *Actions can run pages, reports, XMLports, and codeunits. When you enter object types, do not enter the full reference, such as "Page Sales Order". Enter only the first letter of the object type, followed by the object ID or the first letters of the object name. For example, entering "P42" or "p sales ord" is acceptable and is replaced automatically with "Page Sales Order".*

Property	Purpose
RunPageView	Sets a specific table view that is applied to the page that the action runs. You can select the key, the sorting order, and the table filter that is applied to the target page. ***Note:*** *The user cannot change the table filter that you specify with the* ***RunPageView****. The user can add more filters, but cannot take away, or even see, the filters defined in this property.*
RunPageLink	Sets up a link between the source page and the target page. The source page contains the action being run; the target page is run by the action. The link lets you filter the target page by field values of the source page and to apply additional page filters. ***Note:*** *In the run time environment, the link is translated into the table filter that is applied to the target page. The user can change the filters set by the* ***RunPageLink*** *property without limitations.*
RunPageMode	Specifies the mode in which the target page runs. The mode can be one of the following types: • **View**: The linked record is shown, and the page is noneditable. • **Edit**: The linked record is shown, and the page is editable. • **Create**: No record is shown, and the page creates a new record as soon as the user starts entering data. ***Note:*** *Users can always change the mode of the page, by clicking the corresponding action from the ribbon.*

Preview Mode

In the Microsoft Dynamics NAV Development Environment, you can preview a page directly from **Page Designer** to view how the page will look in the client. Unlike running a page from Object Designer, previewing a page does not start the client. This means that you can keep working in the development environment.

To preview a page, click **View** > **Preview**. This displays the **Page Preview** window.

The **Page Preview** window shows how the page looks in the client.

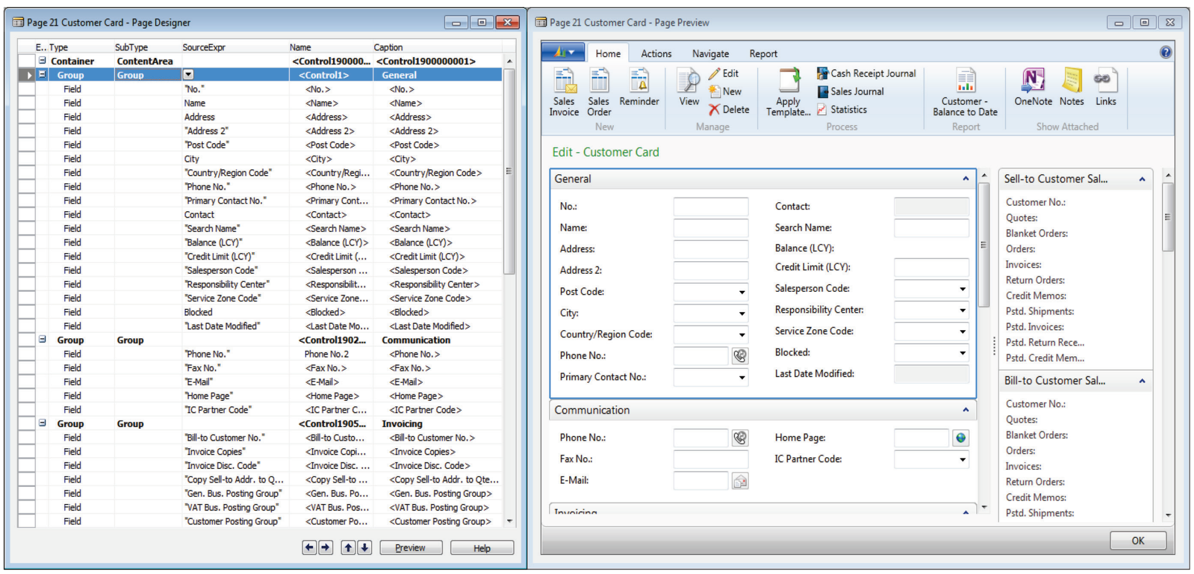


FIGURE 3.12: PAGE PREVIEW WINDOW

The **Page Preview** window includes the following features:

- The **Page Preview** window is updated automatically as you make changes in **Page Designer** and **Action Designer**. When you design a page, you can have the page preview open next to **Page Designer** so that you can see changes as you make them.
- The **Page Preview** window is interactive. You can use the **Page Preview** window to identify and learn which controls in **Page Designer** and **Action Designer** correspond to different elements on the rendered page, such as FastTabs, FactBoxes, fields, and actions.
- You do not need to save the page before viewing it in the **Page Preview** window.

Use the **Page Preview** window to quickly access objects on the page. This includes controls and actions. When you select a control in the **Page Preview** window, the same control is selected automatically in the **Page Designer**. When you select an action in the **Page Preview** window, the **Action Designer** window opens, and the same action is selected automatically in the **Action Designer**.

When you select a control or an action in the **Page Designer** or **Action Designer**, a blue frame is shown around the same control or action in the **Page Preview** window.

If you right-click a control or an action in the **Page Preview** window, a popup menu opens that lets you choose from the following actions:

- To open the **Properties** window for the selected object, click **Properties**.
- To open the **C/AL Editor** window for the selected object, click **C/AL Code**. This opens the C/AL Editor and focuses the triggers for the selected control or action.
- To open the **Page Editor** for a page part, such as FactBox or Lines part in a document, click **Go To Definition**.

Demonstration: Explore the Page Designer

The following demonstration shows how to create a simple page, add basic controls and actions to the page, and how to use their properties.

Demonstration Steps

1. Create a blank page.
   a. In Object Designer, click **Page**.
   b. In the pages list, click **New**. The **New Page** window opens.
   c. Click **OK** to create a blank page. The **Page Designer** window opens.
   d. Click **File** > **Save** to save the page. The **Save As** dialog box opens.
   e. In the **ID** field, enter “90001”.
   f. In the **Name** field, enter ”Custom Page”, then verify that the **Compiled** check box is selected
   g. Click **OK** to compile and save the page.
   h. Close **Page Designer**.
   i. In the Object Designer, locate and select the page 90001, **Custom Page**, and then click **Run**. This opens the Custom Page in the client for Windows.
   j. Close the **Custom Page** window in the client.

This is how the page 90001, **Custom Page**, looks like now.

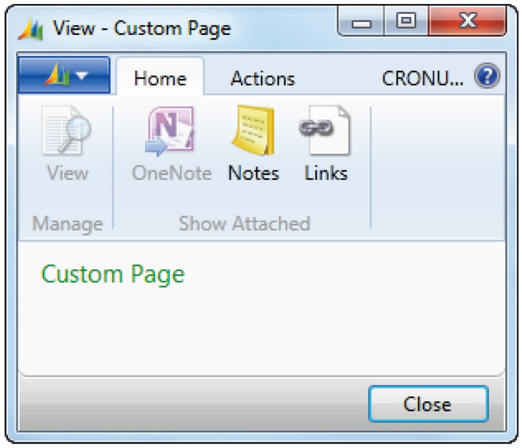


FIGURE 3.13: PAGE 90001, CUSTOM PAGE

2. Select the card page type and specify **Item** as the source table.
   a. In **Object Designer**, make sure that page 90001, **Custom Page** is selected.
   b. Click **Design** to open the **Page Designer**.
   c. On the **View** menu, click **Properties**, to open the Properties window for the page.
   d. Set the following properties:

Property	Value
PageType	Card
SourceTable	Item

   e. Close the **Properties** window.
   f. Compile, save and close the page.
   g. Run page 90001, **Custom Page**.

After you set the **SourceTable** property, the page is bound, and shows the **No**. and **Name** field values of the first **Item** in the database.

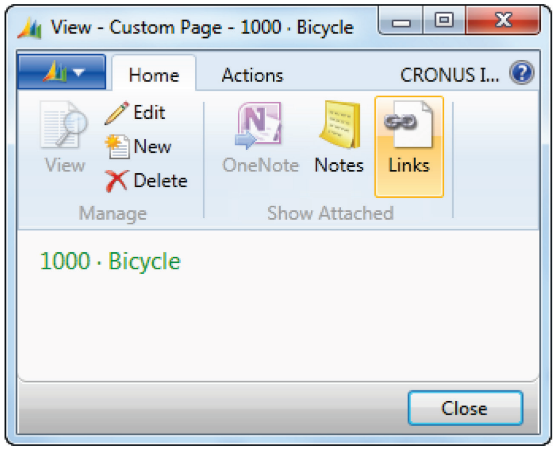


FIGURE 3.14: CUSTOM PAGE AFTER YOU SET THE SOURCETABLE PROPERTY

h. Close the **Custom Page** window in the client.

3. Add a container to the page, and start the preview mode.

   a. In **Object Designer**, select page 90001, **Custom Page**, and then click **Design** to open the **Page Designer**.

   b. Type the following on the first line of the **Page Designer**:

Type	SubType	Name
Container	ContentArea	Custom Page

   c. On the **View** menu, click **Preview** to open the **Page Preview** window for the page 90001, **Custom Page**.

The **Page Preview** window shows how the page is rendered in the client.

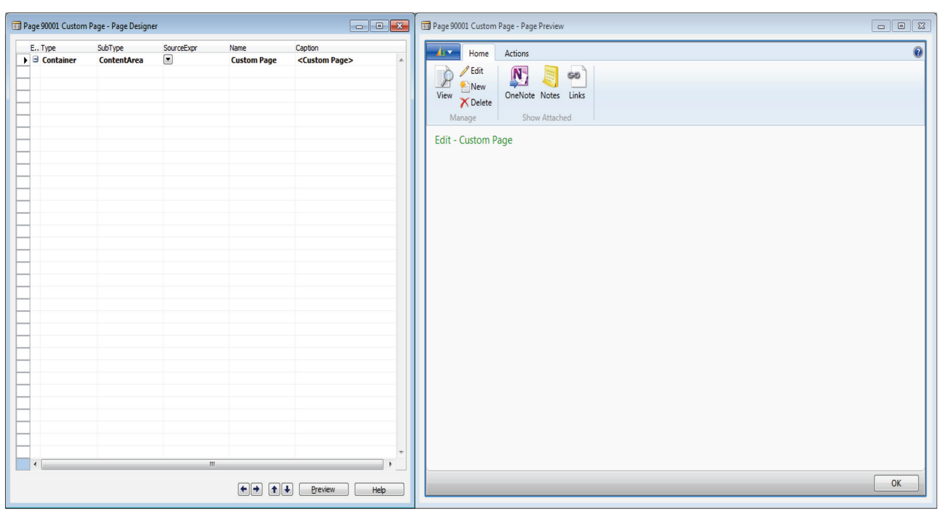


FIGURE 3.15: PAGE DESIGNER WITH THE PAGE PREVIEW WINDOW

4. Add the group control for the **General** FastTab.
    a. In the **Page Designer** window, go to the next blank line. The line automatically indents.
    b. Type the following:

Type	SubType	Name
Group	Group	General

The **Page Preview** window updates to show the **General** FastTab.

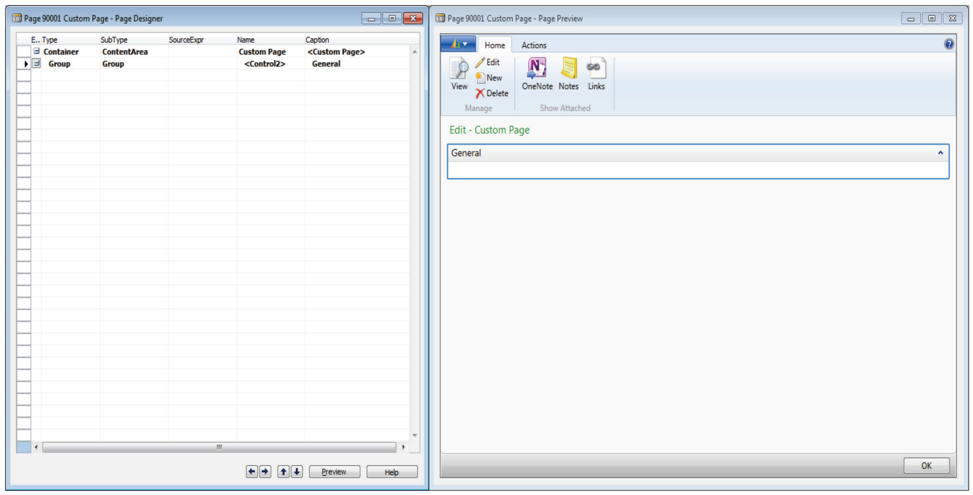


FIGURE 3.16: PAGE PREVIEW WITH THE GENERAL FASTTAB

5. To the **General** FastTab, add field controls for **No.**, **Description**, **Costing Method**, **Profit %**, and **Picture** fields.
    a. In **Page Designer**, select the next blank line. The line automatically indents.
    b. Click **View** > **Field Menu** to display the **Field Menu** window.
    c. In the **Field Menu** window, select the following fields while holding down the Ctrl key:
        - No.
        - Description
        - Costing Method
        - Profit %
        - Picture.
    d. In the **Field Menu**, click **OK**.
    e. Click **Yes** In the dialog box that asks you if you want to add fields to the page.
    f. The fields are added and indented under the **General** group.

The **Page Preview** window updates to include the field controls.

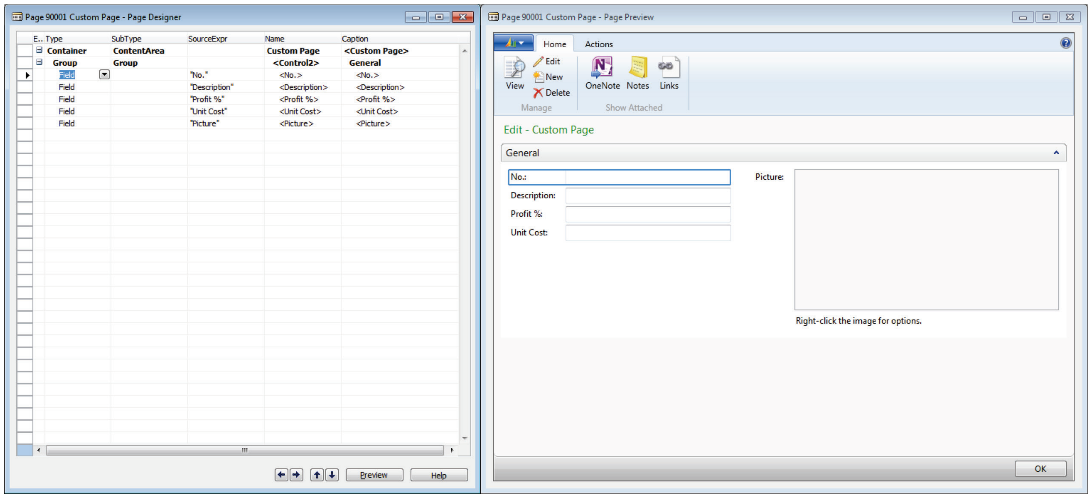


FIGURE 3.17: PAGE PREVIEW WITH FIELDS IN THE GENERAL FASTTAB

6. Set the properties on the **Profit %** field control to show the progress indicator.
   a. In **Page Preview** window, right-click the **Profit %** field.
   b. In the pop-up menu, click **Properties** to open the **Properties** window for the **Profit %** field.
   c. Set the **ExtendedDataType** property value to **Ratio**.
   d. Save, compile, and close the page.
   e. Run page 90001, **Custom Page**.

Custom Page opens in the client for Windows, and shows the values for the first **Item** in the database.

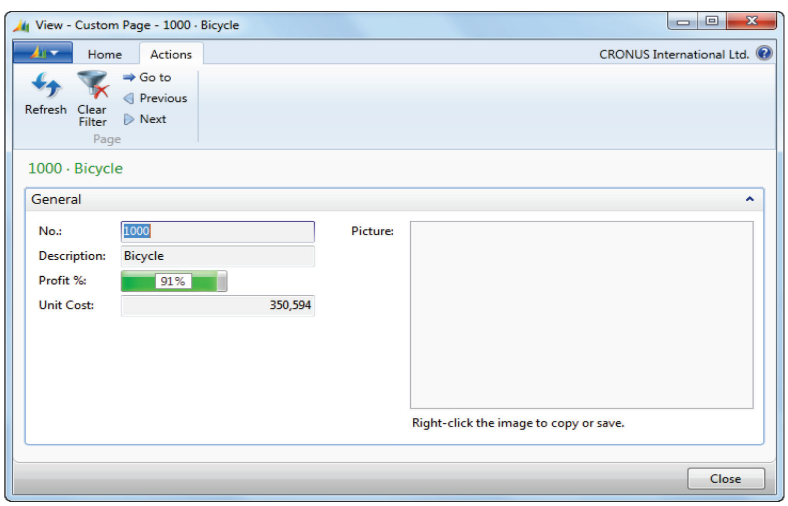


FIGURE 3.18: CUSTOM PAGE THAT HAS PROGRESS INDICATOR CONTROL

   f. Close the **Custom Page** window in the client.

7. Add a FixedLayout control that shows **Inventory, Qty. on Purch. Order, Qty. on Prod. Order, Qty. on Component Lines, Qty. on Sales Order,** and **Qty. on Service Order** fields in a single row.
   a. In the **Object Designer**, select page **90001**, **Custom Page**, and then click **Design**.
   b. Select the last blank line. On the next lines type the following:

Type	SubType	SourceExpr	Caption	Indent
Group	Group		Details	1
Group	FixedLayout			2
Group	Group		Inventory	3
Field		Inventory	Quantity on...	4
Group	Group		Purch. Order	3
Field		"Qty. on Purch. Order"		4
Group	Group		Prod. Order	3
Field		"Qty. on Prod. Order"		4
Group	Group		Component Lines	3
Field		"Qty. on Component Lines"		4
Group	Group		Sales Order	3
Field		"Qty. on Sales Order"		4
Group	Group		Service Order	3
Field		"Qty. on Service Order"		4

Note: *The Indent column represents how many levels each row is indented. To set the indent level, use the* ***Left*** *and* ***Right*** *indent buttons at the bottom of the* ***Page Designer*** *window.*

The Page Preview window updates to show the Details FastTab with the FixedLayout control.

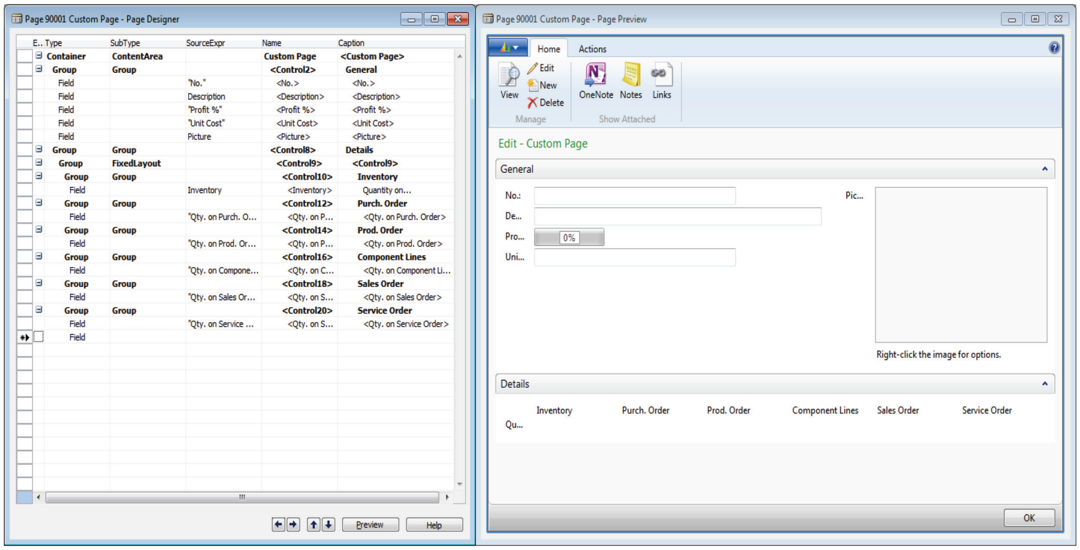


FIGURE 3.19: FIXEDLAYOUT CONTROL IN THE PAGE DESIGNER

c. Compile, save, and close the page.

d. Run the page 90001, **Custom Page**.

FixedLayout control shows noneditable fields without captions and borders, as shown in the Custom Page window.

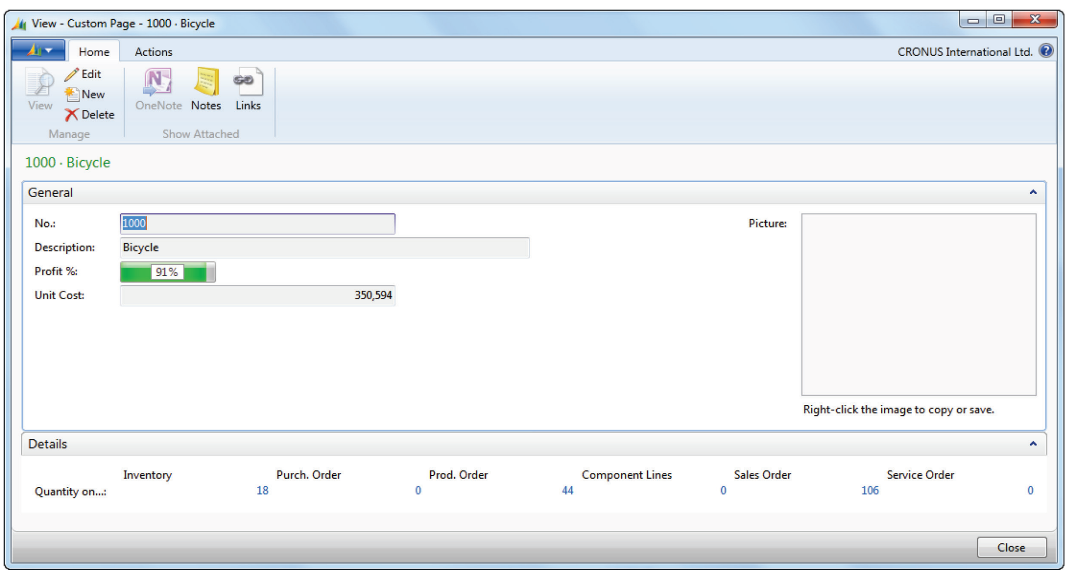


FIGURE 3.20: FIXEDLAYOUT IN THE CLIENT FOR WINDOWS

8. Add an action to run the report 701, **Inventory – List**.
    a. In the **Object Designer**, select page 90001, **Custom Page**, and then click **Design**.
    b. On the **View** menu, click **Preview** to open the **Page Preview** window.
    c. On the **View** menu, click **Page Actions** to open the **Action Designer** window.
    d. In the first blank line, enter the following information:

Type	SubType	Caption
ActionContainer	Reports	

    e. Go to the next blank line. The page element automatically indents.
    f. Enter the following information:

Type	SubType	Caption
Action		Inventory - List

The **Page Preview** window automatically updates to show the **Reports** tab in the ribbon, with **Inventory – List** action.

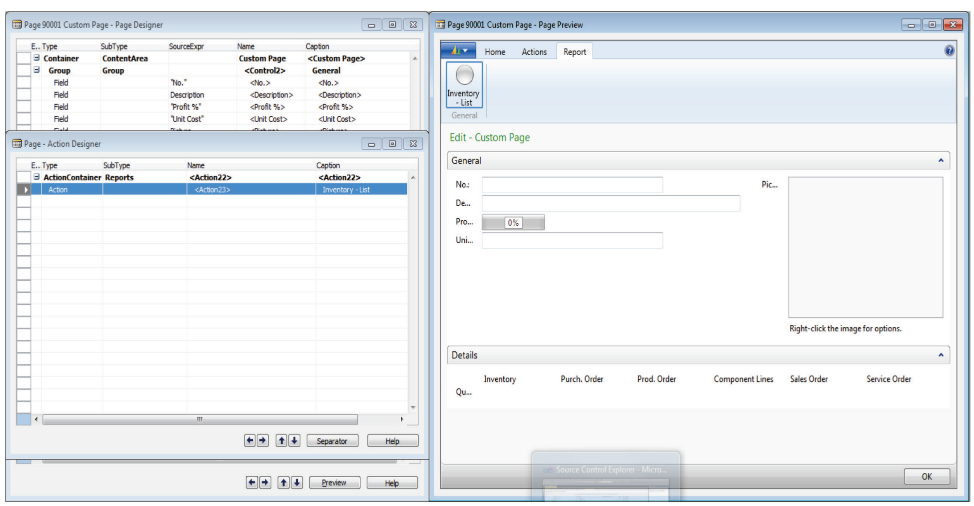


FIGURE 3.21: PAGE PREVIEW WITH ACTIONS

    g. In **Page Preview** window, right-click the **Inventory – List** action, and then in the pop-up menu click **Properties**.

h. Set the following properties:

Property	Value
Image	Report
Promoted	Yes
PromotedCategory	Report
PromotedIsBig	Yes
RunObject	Report Inventory - List

i. Close the **Properties** window, and then close the **Action Designer** window.

The **Page Preview** window updates to show the **Inventory – List** report in the **Report** group on the **Home** tab in the ribbon.

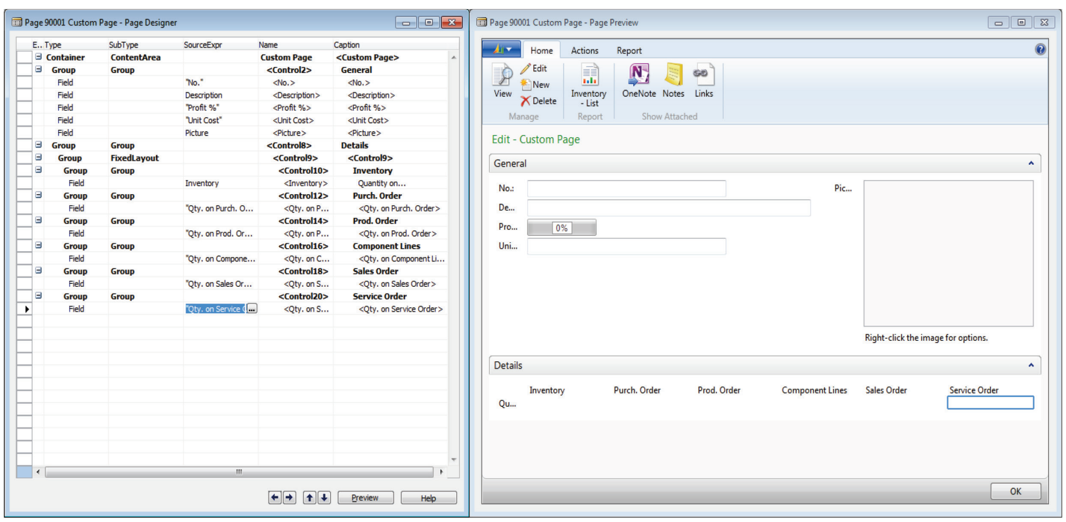


FIGURE 3.22: PROMOTED ACTION ON THE HOME TAB OF THE RIBBON

j. Compile, save and close the page.

k. Run page 90001, Custom Page.

The final look of the page 90001, **Custom Page**, in the client for Windows.

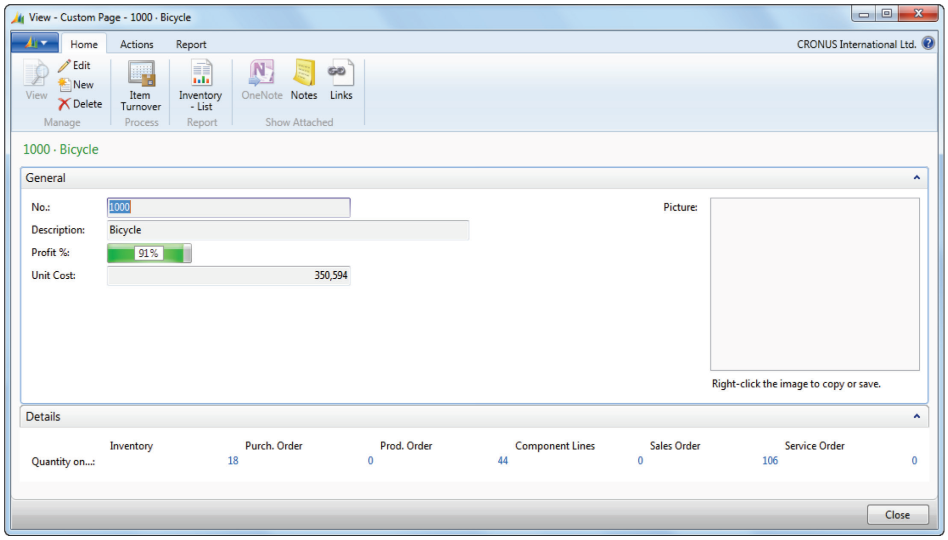


FIGURE 3.23: FINISHED PAGE 90001, CUSTOM PAGE

l. Close the Custom Page window in the client.

9. Add an action to run the page 158, **Item Turnover**.
   a. In the **Object Designer**, select page 90001, **Custom Page**, and then click **Design**.
   b. On the **View** menu, click **Preview** to open the **Page Preview** window.
   c. On the **View** menu, click **Page Actions** to open the **Action Designer** window.
   d. In the first blank line, enter the following information:

Type	SubType	Caption
ActionContainer	ActionItems	

   e. Unindent the **ActionItems** action container completely.

Note: *Page Preview will show a warning about the page specification not being well formed, as long as there are any unsupported settings. Action container that you have added is indented under another action container. This is not a supported configuration. When you correct the indent by unindenting the action container, the Page Preview removes the warning.*

   f. Go to the next blank line. The page element automatically indents.

g. Enter the following information:

Type	SubType	Caption
Action		Item Turnover

__Note:__ The __Page Preview__ window automatically updates, and shows the new action in the __General__ group on the __Actions__ tab in the ribbon.

h. In **Page Preview** window, right-click the **Item Turnover** action, and then in the pop-up menu click **Properties**.

i. Set the following properties:

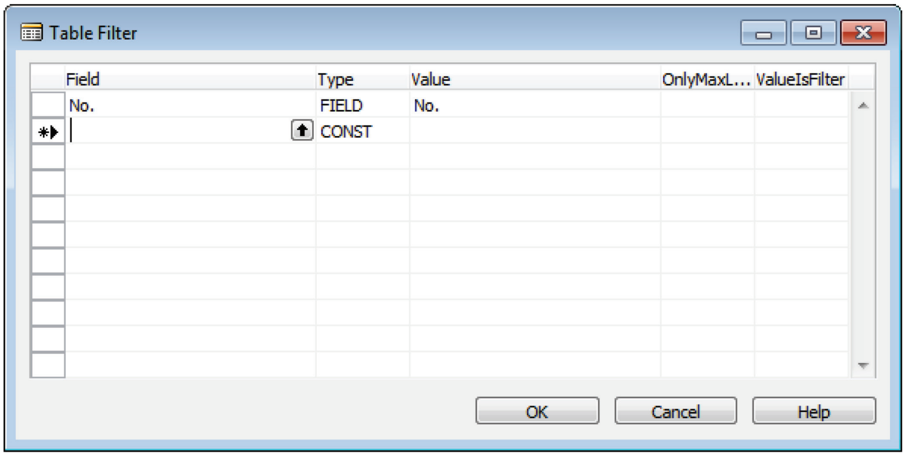
Property	Value
Image	ItemAvailability
Promoted	Yes
PromotedCategory	Process
PromotedIsBig	Yes
RunObject	Page Item Turnover

10. Set the **Item Turnover** action property to show the turnover for the item that is currently selected in **Custom Page** page.

a. In the **RunPageLink** property, click the **AssistEdit** button.

b. In the **Table Filter** window, in the **Field** column, enter “No.”, then in the **Type** column, enter **FIELD**, and then in the **Value** column, enter “No.”. This establishes the link between the **Item Turnover** page and the **Custom Page** page. Therefore, that **Item Turnover** always shows the turnover for the item shown in the **Custom Page** page, when run from the **Item Turnover** action on the **Custom Page** page.

The **Table Filter** window is used to configure the link between the source page, where the action is shown, and the target page that is run by the action.



c. In the **Table Filter** window, click OK.

d. Close the **Properties** window for the **Item Turnover** action.

e. Close the **Action Designer**.

11. Compile, save and run the page 90001, **Custom Page**.

    a. Compile, save and close the page.

    b. Run page 90001, **Custom Page**.

The final look of the page 90001, **Custom Page**, in the client for Windows.

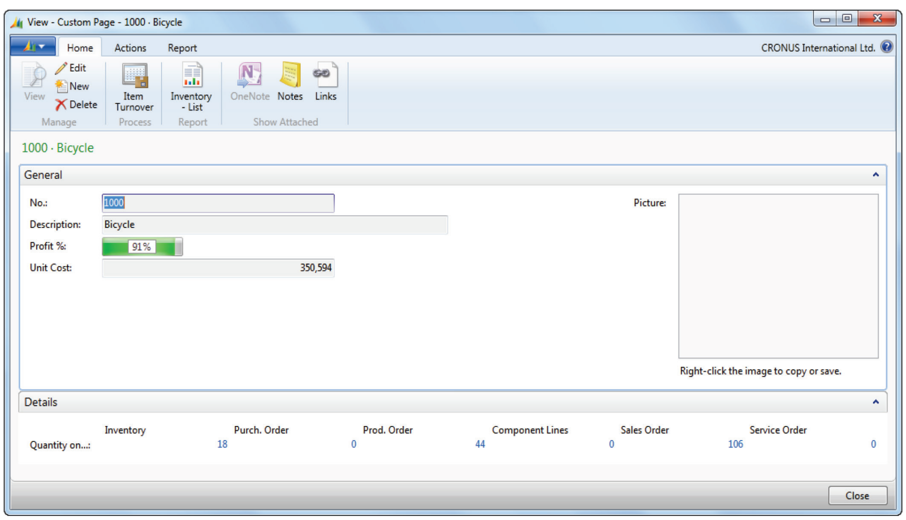


FIGURE 3.25: FINISHED PAGE 90001, CUSTOM PAGE

c. Close the **Custom Page** window in the client.

Page Types and Characteristics

One of the most important page properties is the **PageType** property that is used to specify the page type. Page type determines how the page is displayed in the client. It also controls certain behavioral characteristics of the page. Selecting the correct page type when you design pages is very important, because the pages are interpreted differently in the client, depending on their page type. Each page type also uses different sets of controls. This makes page type the most important concept to understand when you design pages.

Lesson Objectives

Examine different page types and their characteristics.

Card Page

A card page enables users to view and edit one record at a time. Use a card page when there are too many fields to view on one line. A card page always uses FastTabs. This is named **General**. It always is displayed first.

Tables that use card pages only have one field in their primary key. This field always is displayed as the first field in the **General** FastTab.

Naming Card Pages

Use the associated table name to name card pages, followed by the word “Card”. For example, the card page that is associated with the **Customer** table is named **Customer Card**.

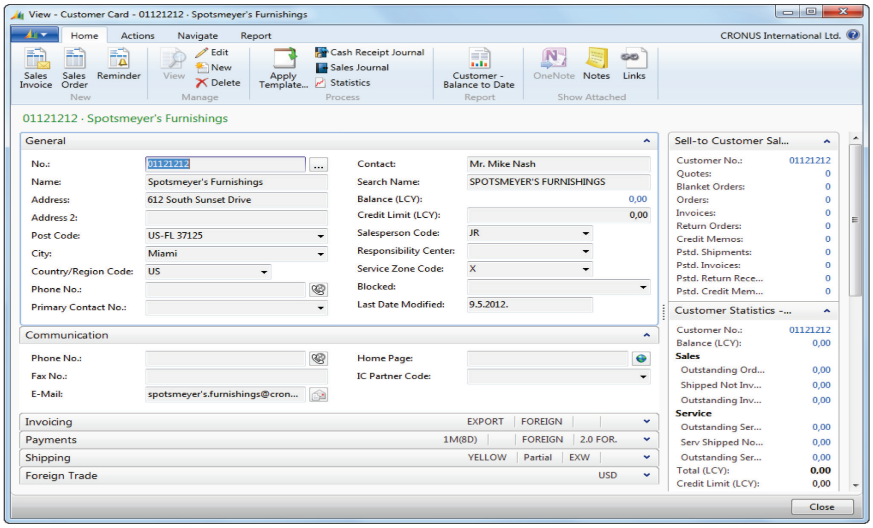


FIGURE 3.26: THE CUSTOMER CARD PAGE

List Page

A list page is a multi-record page that enables users to view multiple records from a table at the same time.

The primary key fields of the associated table are displayed in the left-most columns.

Naming List Pages

If you cannot edit the list page, name it with the name of the associated table, followed by the word "List". For example, the list page that is associated with the **Customer** table is named **Customer List**. If you can edit the list page, name it with the plural of the associated table name. For example, the list page associated with the **Currency** table is named **Currencies**.

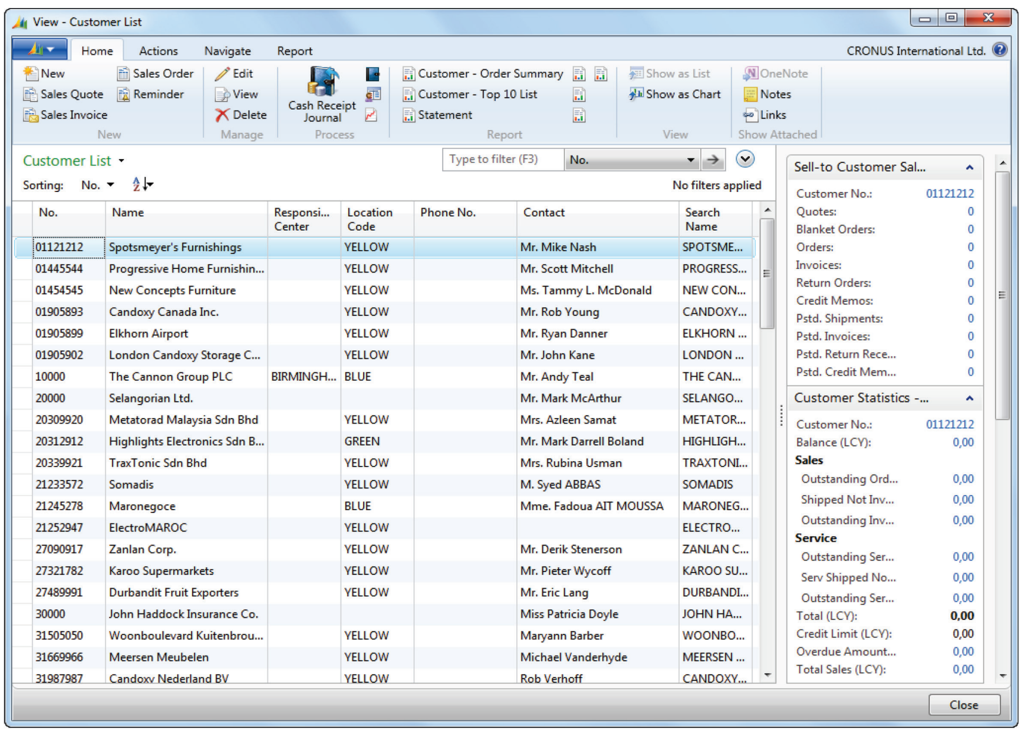


FIGURE 3.27: THE CUSTOMER LIST PAGE

CardPart Page

Use a CardPart page to display fields or a special control like a picture viewer, in a FactBox. Use a CardPart page to create a single-column FactBox.

Naming CardPart Pages

CardPart pages are named with the associated information, followed by the word FactBox. For example, the CardPart page that displays the sales history of the sell-to customer is named **Sales Hist. Sell-to FactBox**.

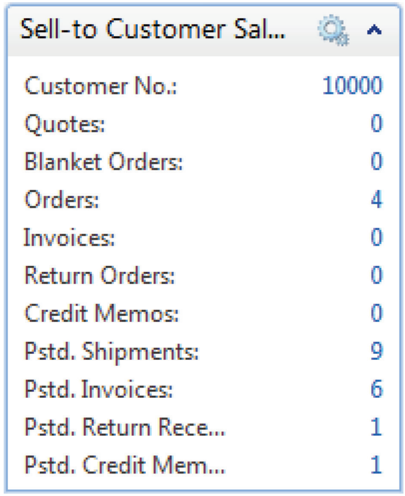


FIGURE 3.28: THE SALES HIST. SELL-TO FACTBOX PAGE

ListPart Page

Use a ListPart page to display subpages for documents. You can also used a ListPart page as a FactBox to display multiple rows and columns from a single table.

Naming ListPart Pages

When ListPart pages are used as FactBoxes, they are named with the associated information, followed by the word “FactBox”. For example, the ListPart page that displays the **Dimensions** is named **Dimensions FactBox**. When you use ListPart pages as subpages for documents, name them with the associated document name, followed by the word “Subform” or “Subpage”. For example, the ListPart page that displays the lines for the **Sales Order** is named **Sales Order Subform**.

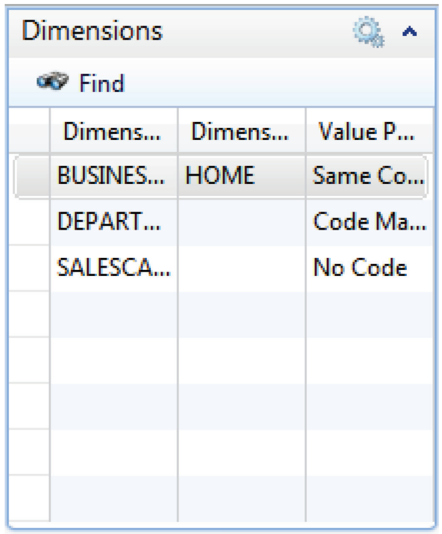


FIGURE 3.29: THE DIMENSIONS FACTBOX PAGE

The following figure displays the ListPart page when it is used as a subpage in a document.

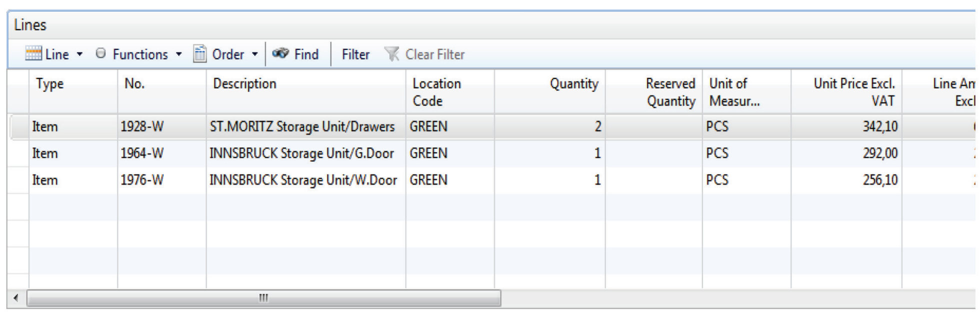


FIGURE 3.30: THE SALES ORDER SUBFORM PAGE

Demonstration: Create a Card Page Using a Wizard

This demonstration shows how to perform the following tasks:

- Create a Card page by using the wizard
- Add FastTabs to the card
- Add fields to FastTabs
- Add FactBoxes to the page

Demonstration Steps

1. Create a new card page for Vendor table by using the Card Page Wizard.
    a. In the **Object Designer**, click **Page**.
    b. Click **New** to show the **New Page** window.

c. In the **Table** field, enter "Vendor", then select the **Create a page using a wizard** option.
d. Click **OK** to start the **Card Page Wizard**.
e. In the **FastTab Name** list, enter "Invoicing" in the first blank line, and then click **Next**.
f. Add the following fields to the **General** FastTab:
   - **No.**
   - **Name**
   - **Address**
   - **City**
   - **Country/Region Code**
   - **Purchaser Code.**
g. Add the following fields to the **Invoicing** FastTab**:**
   - **Budgeted Amount**
   - **Invoice Disc. Code**
   - **Balance (LCY)**
   - **Purchases (LCY)**
   - **Inv. Discounts (LCY)**
   - **Amt. Rcd. Not Invoiced (LCY).**
h. Click **Next**.
i. Add the **Vendor Details FactBox**, and **Vendor Statistics Factbox** page FactBoxes.
j. Add the **Open PO per Location** chart FactBox.
k. Click **Finish**.

2. Set properties on **Vendor Details FactBox** and **Vendor Statistics FactBox** to show the details for the currently displayed vendor.
   a. In the **Page Designer** window, select the **Vendor Details FactBox** page part.
   b. Select **View** > **Properties**.
   c. In the **SubPageLink** property, click the **AssistEdit** button.
   d. In the **Table Filter** window, set the following values:
      - Field to **No.**
      - Type to **FIELD**
      - Value to **No**
   e. Click **OK**.
   f. Select the **Vendor Statistics FactBox** page part.
   g. In the **SubPageLink** property, click the **AssistEdit** button.

h. In the **Table Filter** window, set the following values:

- Field to **No.**
- Type to **FIELD**
- Value to **No**

i. Click **OK**.

3. Compile, save, and close the page.

a. On the **File** menu, click **Save**. The **Save As** dialog box opens.

b. In the **Save As** dialog box, in the **ID** field, enter "90002".

c. In the **Name** field, enter "Vendor Card – New".

d. Make sure that the **Compiled** check box is selected, and then click **OK**.

e. Close the **Page Designer** window.

4. Run page 90002, Vendor Card – New.

a. In the **Object Designer**, locate and select the page 90002, **Vendor Card – New**.

b. Click **Run** to run the page in the client for Windows.

The following figure shows a card page that was created by using **Card Page Wizard**.

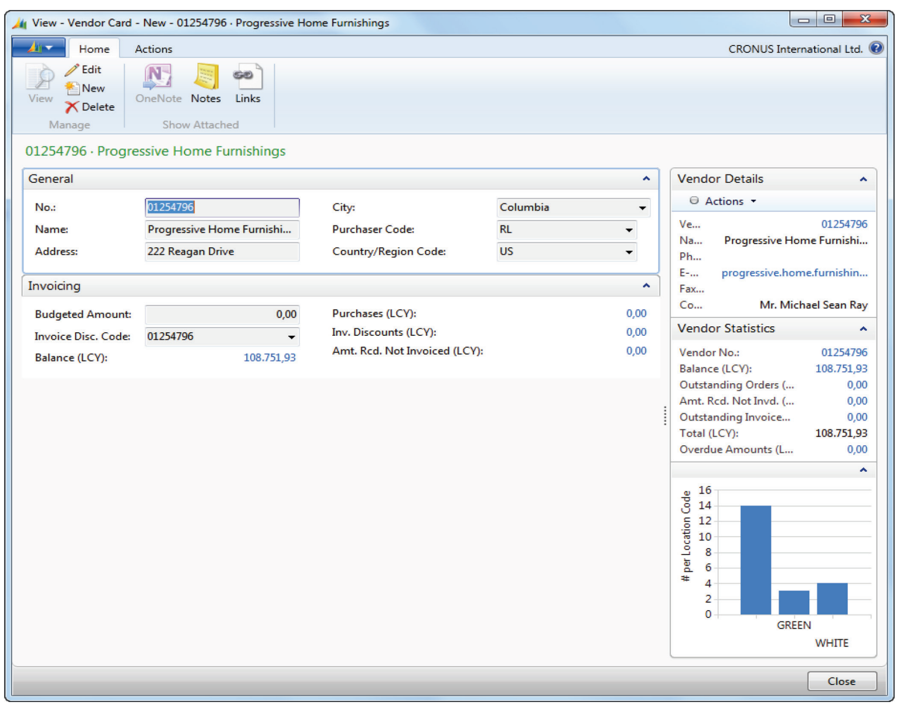


FIGURE 3.31: THE VENDOR CARD – NEW PAGE

Demonstration: Create a List Page Using Wizard

The following demonstration shows how to create a List page by using the wizard, and how to add several fields and FactBoxes to the list page.

Demonstration Steps

1. Create a new card page for Vendor table by using the List Page Wizard.
    a. In the **Object Designer**, click **Page**.
    b. Click **New** to display the **New Page** window.
    c. In the **Table** field, enter "Vendor", and then select the **Create a page using a wizard** option.
    d. Select **List** in the page types list, and then click **OK** to start the **List Page Wizard**.
    e. Add the following fields from the **Available Fields** list:
        - No.
        - Name
        - City
        - Vendor Posting Group
        - Currency Code
        - Purchaser Code
        - Country/Region Code
        - Gen. Bus. Posting Group
    f. Click **Next**.
    g. Add the **Vendor Details FactBox**, **Vendor Statistics Factbox**, **Vendor Hist. Buy-from FactBox**, and **Vendor Hist. Pay-to FactBox** page FactBoxes.
    h. Click **Finish**.

2. Set properties on the factboxes to show the details for the currently shown vendor.
    a. In the **Page Designer** window, select the **Vendor Details FactBox** page part.
    b. Select **View > Properties**.
    c. In the **SubPageLink** property, click the **AssistEdit** button.
    d. In the **Table Filter** window, set the following values:
        - **Field to No.**
        - **Type to FIELD**
        - **Value to No**

e. Repeat these steps for the following FactBoxes:
   - **Vendor Hist. Buy-from FactBox**
   - **Vendor Hist. Pay-to FactBox**
   - **Vendor Hist. Buy-from FactBox**
   - **Vendor Hist. Pay-to FactBox**

3. Set the page properties to prevent changes to the underlying table, and to start the page 90002, **Vendor Card – New**, when a user double-clicks a line in this page.
   a. In the **Page Designer** window, select the first blank line
   b. Select **View > Properties**.
   c. In the **Properties** window, set the **Editable** property to **No**.
   d. In the **Properties** window, set the CardPageID property to “90002”.
   e. Close the **Properties** window.

4. Compile, save and close the page.
   a. On the **File** menu, click **Save**. The **Save As** dialog box opens.
   b. In the **Save As** dialog box, in the **ID** field, enter “90003”.
   c. In the **Name** field, enter “Vendor List – New”.
   d. Verify that the **Compiled** check box is selected, and then click **OK**.
   e. Close the **Page Designer** window.

5. Run page 90003, **Vendor List – New**.
   a. In the **Object Designer**, locate and select the page 90003, **Vendor List – New**.
   b. Click **Run** to run the page in the client for Windows.

The following figure shows a list page that was created by using **List Page Wizard**.

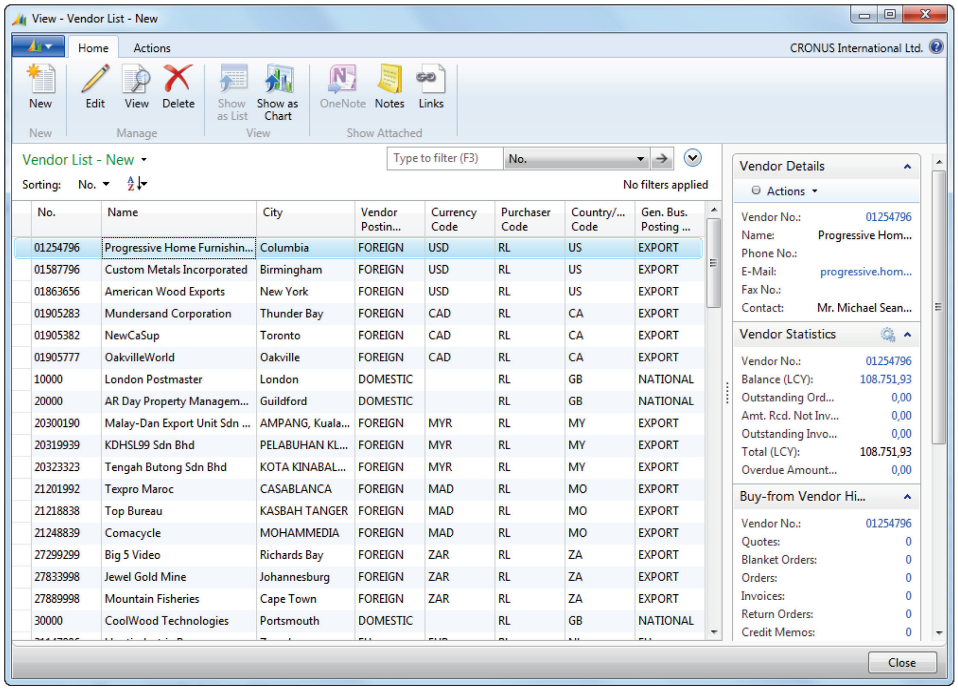


FIGURE 3.32: THE VENDOR LIST – NEW PAGE

6. In the **Vendor List – New** window, verify that **Vendor Card – New** page is opened when you double-click a line.
    a. In the **Vendor List – New** window in the client, double-click any line.
    b. Verify that the **Vendor Card – New** page is shown, and that the **Vendor** record that is selected in the **Vendor List – New** is displayed in the card.

Worksheet Page

A worksheet page is a multi-record page that enables users to view multiple records from a table and edit them. Worksheet pages differ from ordinary list pages. When a user inserts a new record in a Worksheet page, the record does not jump to another position within the page, but remains in the same order in which the user inserted it. The **AutoSplitKey** property of the page, combined with an integer as the last field in the table's primary key enables this order. Worksheet pages let users set filters or options that are shown over the repeater control. Worksheet pages may include a footer which summarizes information from the worksheet or shows detailed information about the currently selected line. You can use worksheet pages as journals.

The primary key fields of the associated table do not show on the worksheet page.

Naming Worksheet Pages

Worksheet pages are named according to the purpose of the associated table. For Journal tables, the name of the Worksheet page ends with the word Journal. For example, the main journal for the **General** area of the **Financial Management** application area is called **General Journal**.

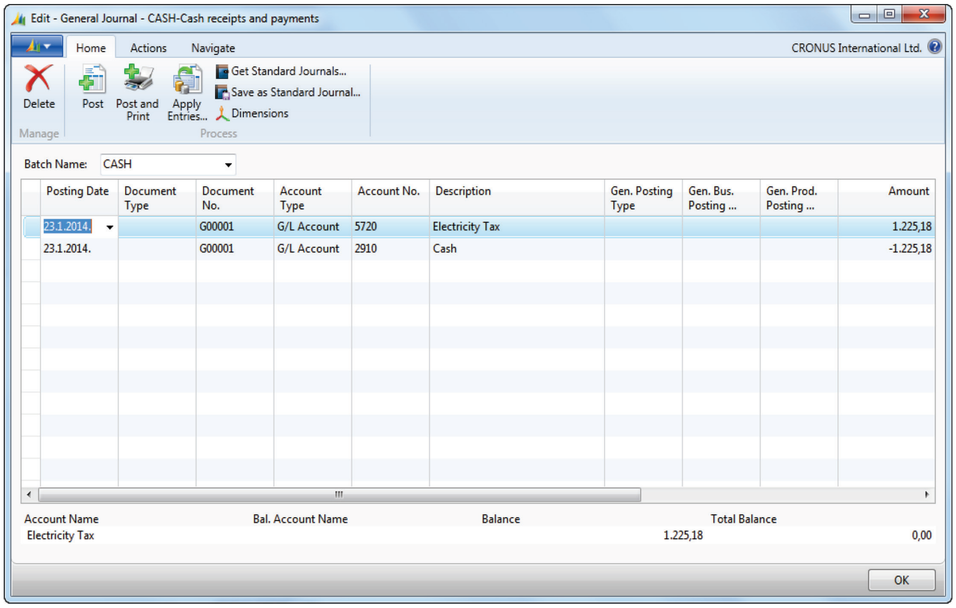


FIGURE 3.33: THE GENERAL JOURNAL PAGE

Document Page

A document page is a page that contains a series of FastTabs and a page part with lines on the same page. It displays information from two tables with the master-detail relationship, such as Sales Header and Sales Line, or Purchase Header and Purchase Line tables.

Naming Document Pages

The name of a document page is the function that it represents, such as, **Sales Order**.

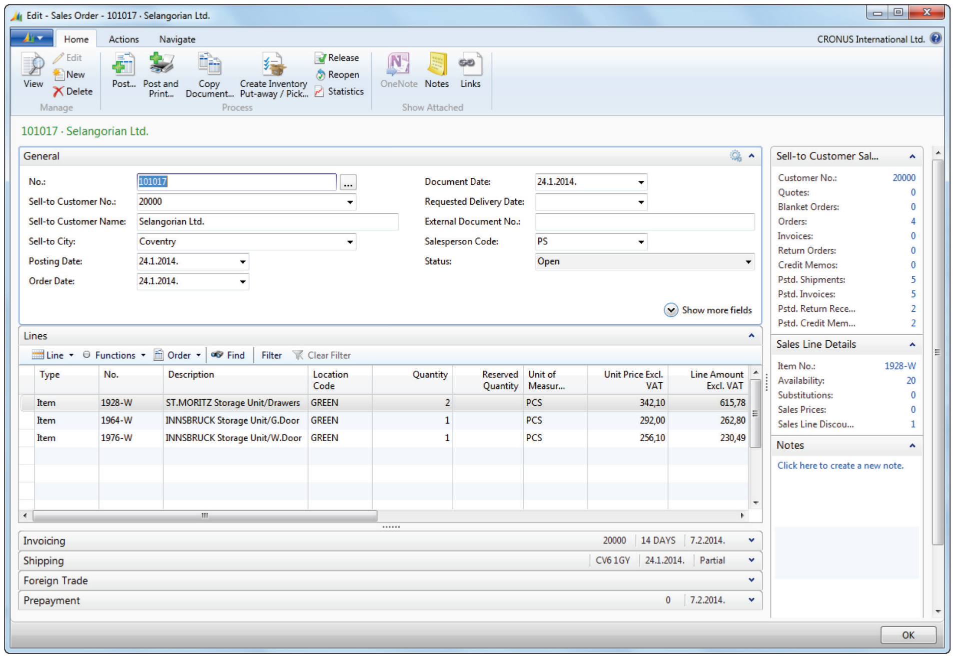


FIGURE 3.34: THE SALES ORDER PAGE

Demonstration: Create a Document Page Using Wizard

The following demonstration shows how to create a Document page by using the wizard, and how to add the Lines subpage to the page.

Demonstration Steps

1. Create a new document page for a **Sales Header** table, by using the Document Page Wizard.
   a. In the **Object Designer**, click **Page**.
   b. Click **New** to show the **New Page** window.
   c. In the **Table** field, enter "Sales Header"
   d. Select the **Create a page using a wizard** option, then select the **Document** in the page types list.
   e. Click **OK** to start the **Document Page Wizard**.
   f. In the **FastTab Name** list, enter "Shipping" in the first blank line, and then click **Next**.
   g. Add the following fields to the General FastTab:
      - **No.**
      - **Sell-to Customer No.**
      - **Sell-to Customer Name**
      - **Sell-to Address**
      - **Sell-to City**

- **Sell-to Post Code**
- **Sell-to Country/Region Code**

h. Add the following fields to the **Shipping** FastTab:
- **Ship-to Code**
- **Ship-to Name**
- **Ship-to Address**
- **Ship-to City**
- **Ship-to Post Code**
- **Ship-to Country/Region Code**

i. Click **Next**, and then click **Finish**.

2. Set the page properties to show only sales orders.
   a. In the **Page Designer**, select the first blank line.
   b. Click **View > Properties** to open the **Page - Properties** window.
   c. In the **SourceTableView** property, click the **AssistEdit** button to open the **Table View** window.
   d. In the **Table View** window, in the **Table Filter** field, click the **AssistEdit** button to open the **Table Filter** window.
   e. In the **Table Filter** window, in the **Field** column, enter "Document Type".
   f. In the **Value** field, enter "Order".
   g. Click **OK** to close the **Table Filter** window.
   h. Click **OK** to close the **Table View** window.
   i. Close the **Page – Properties** window.

3. Create a page part control for **Sales Order Subform** page, and link it to show only the lines for the sales order currently shown in the page.
   a. In the **Page Designer**, select the line for **Shipping** FastTab.
   b. Click **Edit** > **New** to insert a new line.
   c. On the new line, type the following:

Type	SubType	Caption
Part	Page	Lines

   d. With the new line still selected, press the **Left** button one time to unindent the **Lines** page part one level.
   e. Click **View > Properties**.
   f. In the **PagePartID** property, enter "Sales Order Subform".
   g. In the **SubPageLink** property, click the **AssistEdit** button to open the **Table Filter** window.

h. In the **Table Filter** window, enter the following lines:

Field	Type	Value
Document Type	FIELD	Document Type
Document No.	FIELD	No.

i. Click **OK** to close the **Table Filter** window.

j. Close the **Properties** window.

4. Compile, save, and close the page.
   a. Click **File** > **Save**. The **Save As** dialog box opens.
   b. In the **Save As** dialog box, in the **ID** field, enter "90004".
   c. In the **Name** field, enter "Sales Order – New".
   d. Verify that the **Compiled** check box is selected, and then click **OK**.
   e. Close the **Page Designer** window.

5. Run page 90004, Sales Order – New.
   a. In the **Object Designer**, locate and select the page 90004, **Sales Order – New**.
   b. Click **Run** to run the page in the client for Windows.

The following figure shows a document page that was created by using **Document Page Wizard**.

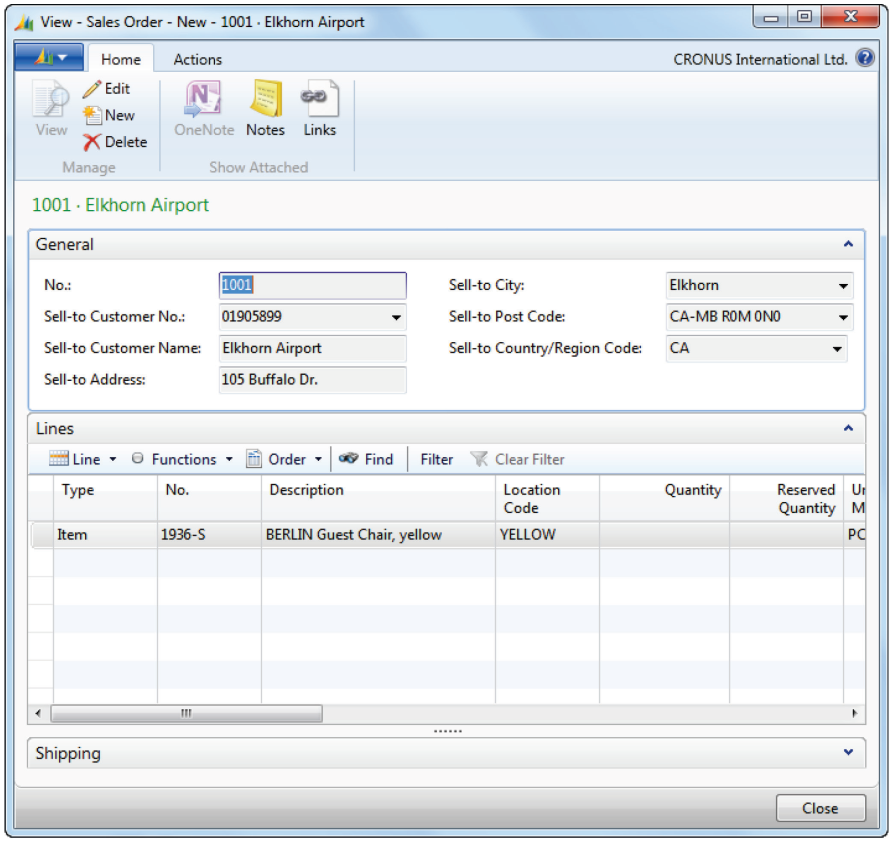


FIGURE 3.35: THE SALES ORDER – NEW PAGE

RoleCenter Page

A role center page represents a customized user home page displays the parts or controls that are relevant to a user's role. A role center page defines the content of the navigation pane, and contains a series of links to list pages. In the central area, a role center points to the user tasks that require their immediate attention, by using cues in the **Activities** part. Role centers may also include the ListParts with the most relevant information for the user, such as **My Customers** or **My Items**. A developer can add system parts, such as **Outlook**, **My Notifications**, and charts, during design time. Users can add these system parts during run time.

Naming Role Center Pages

Role center pages use the name of the associated role, for which it is customized followed by the words "Role Center". For example, the role center page that is associated with the Order Processor role is named **Order Processor Role Center**.

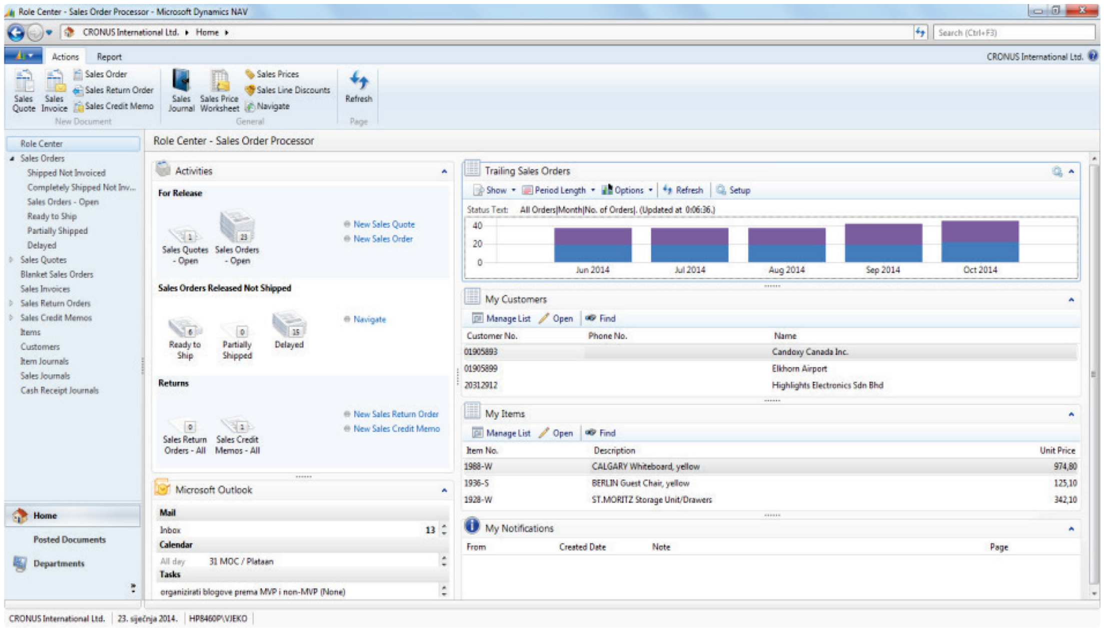


FIGURE 3.36: THE ORDER PROCESSOR ROLE CENTER PAGE

NavigatePage Page

Use the NavigatePage page for searches that are based on individual posting transactions, such as page **344**, **Navigate**. Navigate page type supports showing multiple tabs of information, changing visibility of tabs during run time, and positioning actions in the footer bar. This layout makes NavigatePage pages good candidates for wizards. For example, page **5077**, **Create Interaction**, shows a wizard-like interface that moves the user through a series of questions, before it makes changes to the database.

Naming Navigate Pages

There is no specific requirement for naming NavigatePage pages. Page 344 is named simply Navigate, while typical wizards may be named according to the task they accomplish, such as page **5077**, **Create Interaction**.

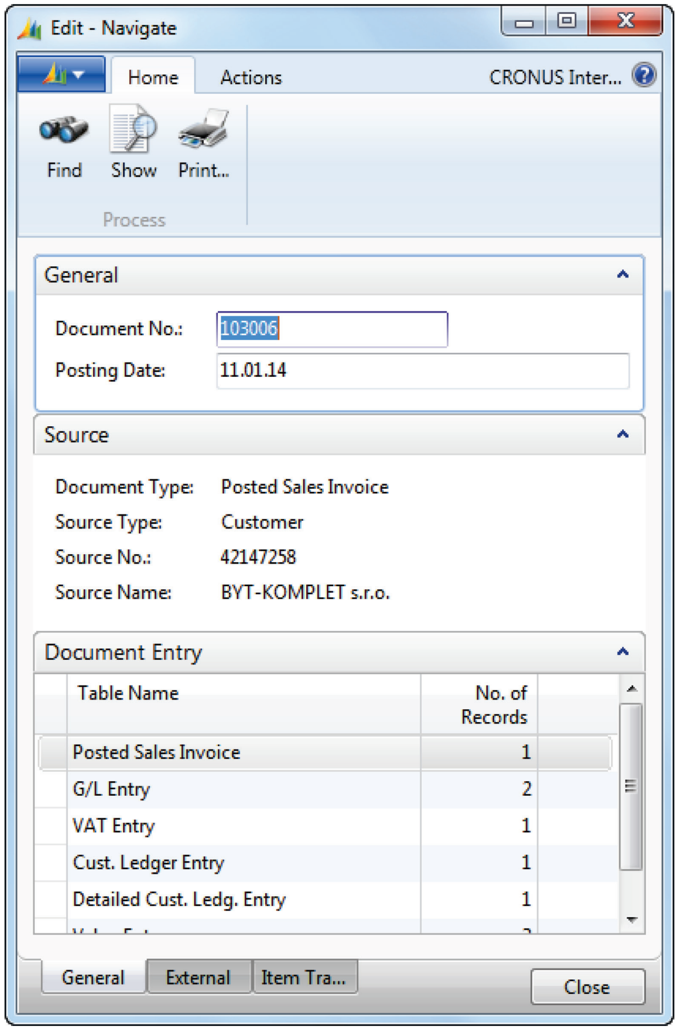


FIGURE 3.37: THE NAVIGATE PAGE

Navigate pages are good candidates for wizards, as they show actions as command buttons, and enable toggling visibility of tabs during run time.

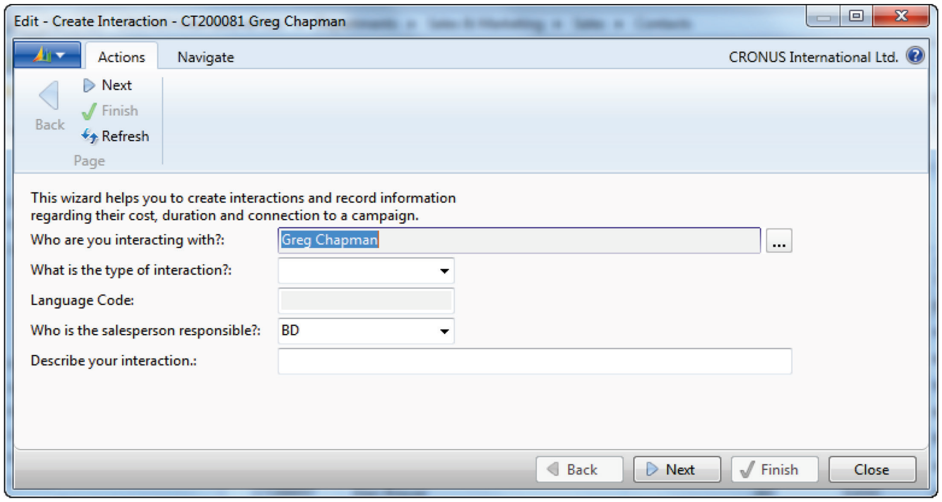


FIGURE 3.38: THE CREATE INTERACTION PAGE

ConfirmationDialog Page

A confirmation dialog asks the user a question that can be answered with minimal data input, and requires the user to click **Yes** or **No**. Pages of this type contain a question, and may contain a single FastTab named Details. This provides the necessary information for the user to make a choice.

Naming ConfirmationDialog Pages

There is no strict convention for naming confirmation dialog boxes. However, you may want to consider using the object of the confirmation preceded by the word Check or Confirm. For example, the confirmation dialog that requires the user to confirm a transaction for a customer who is over the credit limit is named **Check Credit Limit**.

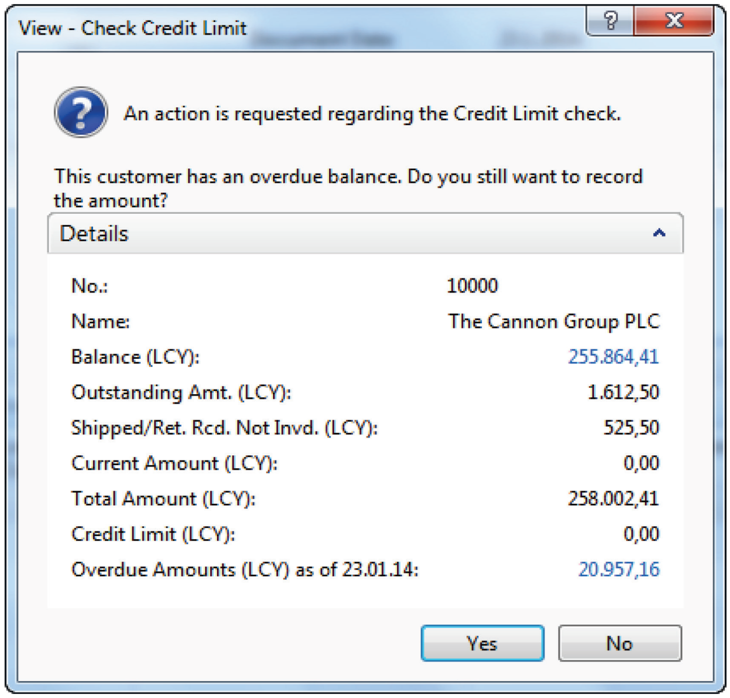


FIGURE 3.39: THE CHECK CREDIT LIMIT PAGE

ListPlus Page

A ListPlus page resembles a document page, because it also contains FastTabs and a ListPart named **Lines**. Whereas document pages typically allow for inserting and deleting lines, and use on the **AutoSplitKey** property, ListPlus pages typically only allow for viewing, or editing existing lines. ListPlus pages do not allow insertion or deletion of lines from the subpage. For example, page **397**, **Sales Invoice Statistics** is a ListPlus page. Because ListPlus pages do not allow the insertion of lines, and show fixed, or pre-determined information in the lines, they may be used instead of card pages for master data management, such as, page **5050**, **Contact Card**.

Naming ListPlus Pages

There is no specific convention regulating the naming of the ListPlus pages. If you use a ListPlus page instead of a card page, such as in the example of page **5050**, **Contact Card**, then it should follow the naming conventions for the card pages.

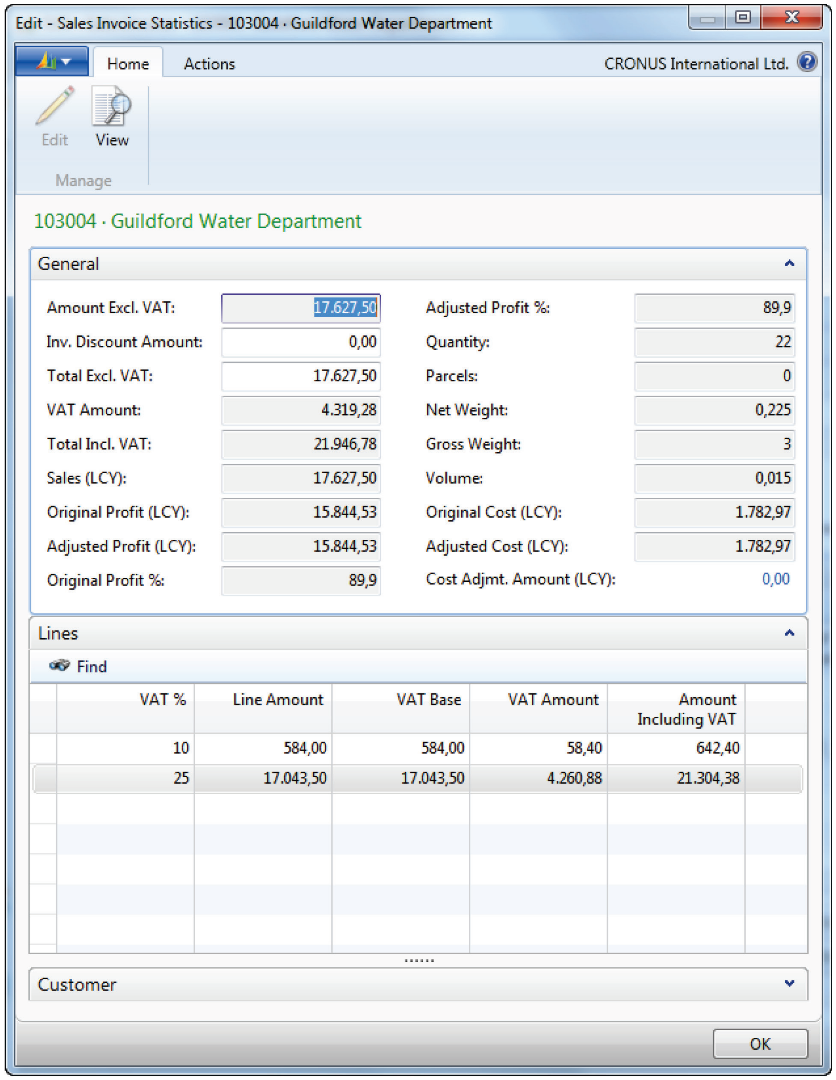


FIGURE 3.40: THE SALES INVOICE STATISTICS PAGE

StandardDialog Page

A standard dialog page is a simple page type that you use when users only input data and do not perform any other actions from the page. Standard dialog boxes do not contain ribbon, FastTabs, or FactBoxes. Users cannot customize the appearance and behavior of these pages. For example, page **511**, **Change Exchange Rate** is of the StandardDialog page type.

Naming StandardDialog Pages

There is no strict convention for naming standard dialog boxes. You may name them according to the task that they perform. For example, the standard dialog for changing the currency exchange rate on documents is named **Change Exchange Rate**.

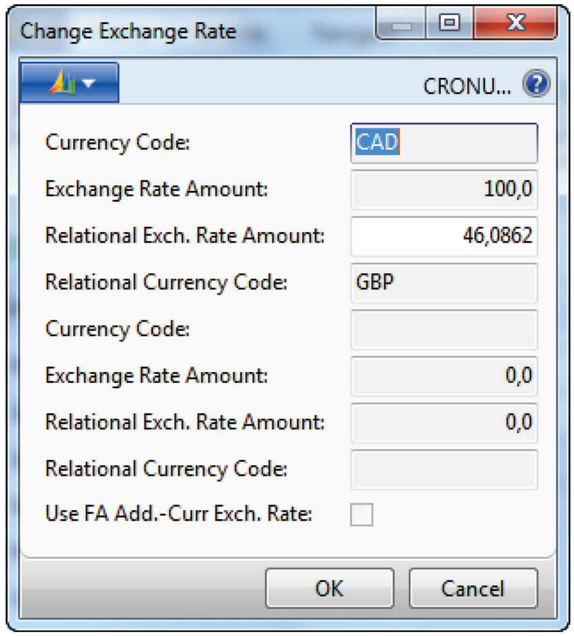


FIGURE 3.41: THE CHANGE EXCHANGE RATE PAGE

Page Design Best Practices

After learning about the components of a page and how they are defined in **Page Designer**, following are best practices when designing pages:

- Consider the types of users who use the new page and list the tasks that they perform.
- Create a list of the fields, actions, and links that are needed for the page.
- Select the page type that best matches the content that is displayed on the page.
- Set the source table of the page to the table that contains the primary set of data that the page shows.
- Design the page by using **Page Designer** to specify the element hierarchy. Determine the details of how data is displayed by adjusting the properties of each element on the page.
- Consider any supplemental sources of information that add value to the page and add these as FactBoxes. Existing FactBoxes can support the design.
- Simplify the user experience by streamlining the user's default view. This can be achieved by doing the following:
    - Promote actions to appropriate groups that have the appropriate sequence and size.

    - Set the **Importance** property to **Additional** for fields that are used less than 30 per cent of the time.
    - Display one to three FactBoxes. This is by default. Add the new page to the menu suite so that it is included in the Departments location.
- Consider specific user profiles if the page is a list place. These user profiles can have a link to the page from their role centers.

Lab 3.1: Create a Card and a List Page

Exercise 1: Create a Card page for the Course table.

Exercise Scenario

Simon wants to follow the standard recommendations for cards and lists. He first creates a card page, names it according to card page naming rules, and saves it.

Task 1: Creating a new page

High Level Steps

1. Create a new **Card** page that has the source table **Course**.
2. Set page properties to show the caption according to the naming conventions for card pages.

Detailed Steps

1. Create a new **Card** page that has the source table **Course**.
   a. In the **Object Designer**, click **Page** to open the page list.
   b. Click **New** to open the **New Page** window.
   c. In the **Table** field, enter "Course".
   d. Verify that **Create blank page** is selected, and then click **OK**.

2. Set page properties to show the caption according to the naming conventions for card pages.
   a. In the **Page Designer window**, click **View** > **Properties**.
   b. In the **Caption** property, enter "Course Card".
   c. Close the **Properties** window.

Task 2: Adding controls

High Level Steps

1. Add a ContentArea container.
2. Add a FastTab.
3. Add the fields.

Detailed Steps

1. Add a ContentArea container.
   a. In the **Page Designer**, select the first blank line.
   b. Verify that **Type** is **Container**, and **SubType** is **ContentArea**.
   c. In the **Caption** column, enter "Course Card".

2. Add a FastTab.
    a. In the **Page Designer**, select the next blank line.
    b. In the **Type** column, enter **Group**.
    c. In the **SubType** column, enter **Group**.
    d. In the **Caption** column, enter “General”.

3. Add the fields.
    a. Click **View** > **Field Menu** to open the **Field Menu** window.
    b. In the **Field Menu**, select the following fields:
        - **Code**
        - **Name**
        - **Description**
        - **Duration**
        - **Price**
        - **Type**
        - **Active**
    c. Click **OK**.
    d. In the confirmation dialog box, click **Yes** to complete adding the fields.

Task 3: Saving the page

High Level Steps

1. Compile, save, and close the page.

Detailed Steps

1. Compile, save, and close the page.
    a. Click **File** > **Save**.
    b. In the **Save As** dialog box, in the **ID** field, enter “90010”.
    c. In the **Name** field, enter “Course Card”.
    d. Verify that the **Compiled** check box is selected.
    e. Click **OK** to complete saving the object.
    f. Close the **Page Designer** window.

Exercise 2: Create a List page for the Course table.

Exercise Scenario

Simon now creates the list page. He follows the list page naming standards, and guarantees that users cannot edit information in this page. He also makes sure that when a user double-clicks a course, the Course Card page displays the selected course. Simon also makes sure that **New**, **Edit**, and **View** actions work as expected in the client.

Task 1: Creating a new page

High Level Steps

1. Create a new List page that has the source table **Course**.
2. Set page properties to display the caption according to the naming conventions for list pages.
3. Set page properties to specify that page type is **List**. Make sure that the list is not editable, and that **Course Card** page opens when users double-click a line in this page.

Detailed Steps

1. Create a new List page that has the source table **Course**.
   a. In **Object Designer**, click **Page** to open the page list.
   b. Click **New** to open the **New Page** window.
   c. In the **Table** field, enter "Course".
   d. Verify that **Create blank page** is selected, and then click **OK**.

2. Set page properties to display the caption according to the naming conventions for list pages.
   a. In the **Page Designer** window, click **View** > **Properties**.
   b. In the **Caption** property, enter "Course List".

3. Set page properties to specify that page type is **List**. Make sure that the list is not editable, and that **Course Card** page opens when users double-click a line in this page.
   a. In the **Editable** property, enter **No**.
   b. In the **PageType** property, enter **List**.
   c. In the **CardPageID** property, enter "Course Card".
   d. Close the **Properties** window.

Task 2: Adding controls

High Level Steps

1. Add a ContentArea container.
2. Add a Repeater.
3. Add the fields.

Detailed Steps

1. Add a ContentArea container.
   a. In the **Page Designer**, select the first blank line.
   b. Make sure that **Type** is **Containe**r, and **SubType** is **ContentArea**.
   c. In the **Caption** column, enter “Course List”.

2. Add a Repeater.
   a. In the **Page Designer**, select the next blank line.
   b. In the **Type** column, enter **Group**.
   c. In the **SubType** column, enter **Repeater**.

3. Add the fields.
   a. On the **View** menu, click **Field Menu** to open the **Field Menu** window.
   b. In the **Field Menu**, select the following fields:
      - **Code**
      - **Name**
      - **Description**
   c. Click **OK**.
   d. In the confirmation dialog box, click **Yes** to complete adding the fields.

Task 3: Saving the page

High Level Steps

1. Compile, save, and close the page.

Detailed Steps

1. Compile, save, and close the page.
   a. On the **File** menu, click **Save**.
   b. In the **Save As** dialog box, in the **ID** field, enter “90011”.
   c. In the **Name** field, enter “Course List”.
   d. Verify that the **Compiled** check box is selected.
   e. Click **OK** to complete saving the object.
   f. Close the **Page Designer** window.

Module Review

Module Review and Takeaways

Pages provide a flexible foundation for building many types of display objects. There are 11 types of pages that control how the page is displayed in the client, and determine the functionality of the page. You design pages in the Page Designer Page Designer is a declarative editor; it does not provide immediate insight into how the page looks. Page Designer includes the Page Preview mode that shows a reliable display of the page in Microsoft Dynamics NAV Development Environment. Page Preview mode streamlines development by integrating with control and action properties and C/AL code editors.

Test Your Knowledge

Test your knowledge with the following questions.

1. How does Microsoft Dynamics NAV 2013 client for Windows show actions?

____

____

____

____

2. Which of the following elements do not have triggers?

   ( ) Page

   ( ) Control

   ( ) Action

   ( ) Ribbon

3. Which of the following types is not a valid page type in Microsoft Dynamics NAV 2013?

   ( ) Card

   ( ) List

   ( ) CardPlus

   ( ) ListPlus

   ( ) CardPart

4. Which page type do you use to enable users to enter information in a wizard-like interface?

   ( ) ListPlus

   ( ) NavigatePage

   ( ) RoleCenter

   ( ) StandardDialog

   ( ) WizardDialog

5. What page type enables users to only input data and not perform any other actions from the page?

____

____

____

____

6. You can define actions at the page level only.

   ( ) True

   ( ) False

7. Which control type allows you to create control actions?

   ( ) CueGroup

   ( ) FastTab

   ( ) Repeater

   ( ) GridLayout

8. Which page types do you use for pages that are shown as subpages in other pages?

____

____

____

____

Test Your Knowledge Solutions

Module Review and Takeaways

1. How does Microsoft Dynamics NAV 2013 client for Windows show actions?

   MODEL ANSWER:

   Microsoft Dynamics NAV 2013 client for Windows shows actions in the ribbon.

2. Which of the following elements do not have triggers?

   ( ) Page

   ( ) Control

   ( ) Action

   (√) Ribbon

3. Which of the following types is not a valid page type in Microsoft Dynamics NAV 2013?

   ( ) Card

   ( ) List

   (√) CardPlus

   ( ) ListPlus

   ( ) CardPart

4. Which page type do you use to enable users to enter information in a wizard-like interface?

   ( ) ListPlus

   (√) NavigatePage

   ( ) RoleCenter

   ( ) StandardDialog

   ( ) WizardDialog

5. What page type enables users to only input data and not perform any other actions from the page?

   MODEL ANSWER:

   StandardDialog page type.

6. You can define actions at the page level only.

   ( ) True

   (√) False

7. Which control type allows you to create control actions?

   (√) CueGroup

   ( ) FastTab

   ( ) Repeater

   ( ) GridLayout

8. Which page types do you use for pages that are shown as subpages in other pages?

   MODEL ANSWER:

   CardPart and ListPart.