Customization Manager

Release 2015

Disclaimer

This document is provided “as-is”. Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it.

Some examples are for illustration only and are fictitious. No real association is intended or inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes only.

Sample Code Warranty disclaimer

Microsoft Corporation disclaims any warranty regarding the sample code contained in this documentation, including the warranties of merchantability and fitness for a particular purpose.

License agreement

Use of this software is covered by a license agreement provided with it. If you have any questions, please call the Customer Assistance Department at 800-456-0025 (in the United States or Canada) or +1-701-281-6500.

Copyright

Publication Date

September 2014

Contents

Introduction

Customization Manager Overview

The Customization Manager module in Microsoft Dynamics® SL allows users, MIS staff, and consultants to modify standard Microsoft Dynamics SL screens quickly and easily. Modification capabilities range from simple changes that the system administrator or user can make to complex customizations that are best performed by a programmer or consultant with experience in Visual Basic® programming and SQL syntax.

The Customization Manager module functions are organized into separate selections on the **Customize** menu. Capabilities of the Customization Manager module include:

- Implementing customizations that apply to a single user or to all users.
- Hiding fields.
- Arranging data entry screens to resemble source document formats.
- Moving fields to make room for new fields or to provide a layout that facilitates data entry and viewing.
- Adding new data items to screens from anywhere in the SQL database, including new records and fields added to the database. In addition to new fields, you can add standard object types such as text boxes, combo boxes, labels, push buttons, frames, and forms.
- Setting or changing default values.
- Creating or modifying edit masks for such items as telephone numbers and social security numbers.

Because no two companies have identical business rules and operations, the need for customization of any software package is inevitable. Once you identify the need for a customization, determine whether it must apply to all or nearly all users or to a small number of users. If only a few users need the customization, create the customization for one user, then use the Customization Export and Import functions to copy it for other users (see “Customization Management” on page 39). If many users need the customization, create two customizations: one for the large number of users who need to use the customized version, and another for the few users who need to use the standard version. Once you have selected the level, use Customization Manager to create and maintain customizations that modify application screens and the objects they contain, such as fields, field labels, and buttons.

You may meet your company’s business needs by simply adding or rearranging objects on an application screen, or your solution may involve adding event logic to display an informational message associated with an object. If your business solution involves adding tables or fields to the Microsoft Dynamics SL database, use Customization Manager to add those objects to application screens.

Use customization import and export functions to copy your customizations from one system to another and to allow other users to access the customizations. To protect entire customized screens or customized objects from unauthorized access, use the software’s security functions. To more thoroughly secure data, you may also wish to secure access to certain Customization Manager functions, such as the Customization Import and Export functions, from unauthorized access.

User Guide Overview

This user guide provides information regarding the setup and use of the General Ledger module. Reviewing the user guide can help you make informed decisions regarding the implementation of the Customization Manager module in your business.

What is Covered in the User Guide?

The user guide consists primarily of procedures and checklists that describe how to perform the various tasks featured in the Customization Manager module. The user guide also contains topics that help you become better acquainted with the capabilities of the module. Topics are arranged in a logical order that builds on information previously presented in other Microsoft Dynamics SL user guides.

Who Should Use the User Guide?

The user guide is designed for readers who are new to Microsoft Dynamics SL. The guide provides the information necessary for making decisions regarding how to use the Customization Manager module to get the most from your system.

How to Use the User Guide

Read the appropriate section of the user guide before proceeding with any system customizations. The user guide presents the procedures and steps required for completing the various customization processes. To assist you in locating information, the user guide contains:

- A “Table of Contents” of logically organized activities and tasks.
- An alphabetized “Quick Reference Task List” of commonly performed tasks.
- An alphabetized “Index” of the information provided in the user guide.

Quick Reference Task List

This list contains tasks that are commonly performed with the Customization Manager module. Each task is cross-referenced to a specific page in the user guide. In the Customization Manager's Help feature, the tasks hypertext link to their corresponding procedure.

How Do I Add...?

- An Object — see "Using the Insert Object Wizard to Add Objects" on page 14 or "Insert Object Wizard (91.252.00)" on page 68
- A Tab Stop to an Existing Object — see "Modifying the Tab Order of a Screen" on page 18
- A Hot Key for an Object — see "Implementing Hot Keys for Objects" on page 18
- A Possible Values List — see "Adding and Modifying Possible Values Lists" on page 28

How Do I Change...?

- The Properties of a Customized Object — see "Making Simple Property Modifications" on page 17
- The Tab Order of a Customized Screen — see "Modifying the Tab Order of a Screen" on page 18
- An Edit Mask or Data Mask — see "Modifying Edit Masks" on page 23
- Default Values for an Object — see "Modifying Default Values" on page 27
- A Possible Values List — see "Adding and Modifying Possible Values Lists" on page 28
- Database Values Using Visual Basic — see "Modifying Database Values Using Visual Basic Code" on page 35
- An Additional Table Once It Has Been Added — see "Implementing and Maintaining Additional Tables" on page 36
- The Customization Export File for a New User — see "Modifying the Export File" on page 44
- The Customization Export File for another System — see "Modifying the Export File" on page 44

How Do I Copy...?

- A Customization for Another User — see "Delivering Customizations to Other Sites and Users" on page 45
- A Customization to Another System — see "Delivering Customizations to Other Sites and Users" on page 45

How Do I Create...?

- A Customization — see "Creating or Editing a Customization" on page 6
- An Additional Table — see "Implementing and Maintaining Additional Tables" on page 36
- A Group Level Customization – see "Definition of Customization Levels" page 40

How Do I Delete...?

- A Customization for an Object — see "Deleting Object Level Customizations" on page 20
- The Tab Stop for an Object — see "Modifying the Tab Order of a Screen" on page 18
- A Tab Object — see "Deleting Object Level Customizations" on page 20
- All Customizations for an Entire Screen — see "Deleting Screen Level Customizations" on page 20

How Do I Export...?

- A Customization — see “Exporting Customizations to a File” on page 42

How Do I Import...?

- A Customization — see “Importing the Customization File” on page 46

How Do I Identify...?

- Whether a Screen Has Been Customized — see “Identifying Customized Screens” on page 50
- Whether a Standard Object Has Been Customized — see “Identifying Customized or New Objects” on page 50
- Whether an Object Has Been Added — see “Identifying Customized or New Objects” on page 50

How Do I Secure...?

- A Customization — see “Securing an Entire Customization” on page 55
- A Customized Object — see “ Securing Customizations to Individual Objects” on page 55
- Access to the Visual Basic Editor — see “Securing Access Visual Basic Editor (VBA IDE” on page 57
- Access to Insert Objects Wizard — see “Securing Access to Insert Object Wizard” on page 57
- Access to Import Customizations (91.510.00) — see “Securing Access to Import Customizations” on page 59
- Access to Export Customizations (91.500.00) — see “Securing Access to Export Customizations” on page 58

How Do I Set...?

- Required Fields — see “Setting Required Fields” on page 25
- Valid Minimum and Maximum Values for a Field — see “Setting Valid Min/Max Values” page 26
- A Tab Stop for an Object — see ”Modifying the Tab Order of a Screen” page 18

Customization Basics

Overview

Use Customization Manager to create and maintain customizations that modify application screens and the objects they contain (such as fields, field labels, and buttons). A customization is the collection of changes and additions made to a screen or to an object on a screen that are applied to an application screen at run time.

"Customization Basics" provides information and procedures for performing basic tasks associated with creating and maintaining customizations. These basic tasks include:

- Creating or Editing a Customization
- Selecting Objects
- Moving Objects
- Using the Customize Menu
- Using the Insert Object Wizard to Add Objects
- Modifying Objects
- Implementing Hot Keys for Objects
- Modifying the Tab Order of a Screen
- Resizing Existing Objects
- Deleting Customizations
- Saving Customizations
- Exiting Customization Manager

Creating or Editing a Customization

Selecting an Application Screen to Customize

To create a customization, display the application screen to be customized. The default customization level is **Self** (see “Setting the Level for a Customization” on page 41).

To edit an existing customization, display the customized application at the level of the customization you wish to edit, then enter customize mode.

In order to enter customize mode, you need to have only one Microsoft Dynamics SL application screen active on the desktop. Note that there may be additional screens linked to the application screen that you are planning to customize. Pressing a button on another screen typically accesses subscreens. To customize the subscreen, be sure to open both screens before entering customize mode by pressing `CTRL+ALT+C`.

Note: All invisible objects become visible while you are in customize mode so that you can change them. This includes standard objects for any Microsoft Dynamics SL modules that may not be available at your business location.

To create or edit a customization:

1. In the Microsoft Dynamics SL window, click the **Administration** button, and then click **Select Customization Level**. *Select Customization Level* (91.260.00) appears.

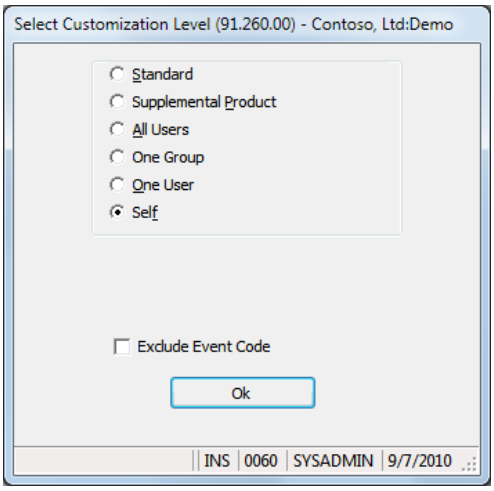

Figure 1: Select Customization Level (91.260.00)

2. In *Select Customization Level* (91.260.00), select the level.
   - If creating a new customization, select the level you want the customization to apply to.
   - If editing an existing customization, select the level to which the existing customization applies.
3. Open the application screen you wish to customize.
4. Click **Customize Mode** on the **Actions** menu to enter customize mode or use the keyboard short cut of CTRL+ALT+C to enter customize mode. This short cut is required when customizing a subscreen.

Opening an Application Screen

To open an application screen:

After you open Microsoft Dynamics SL, follow the steps below to open an application screen.

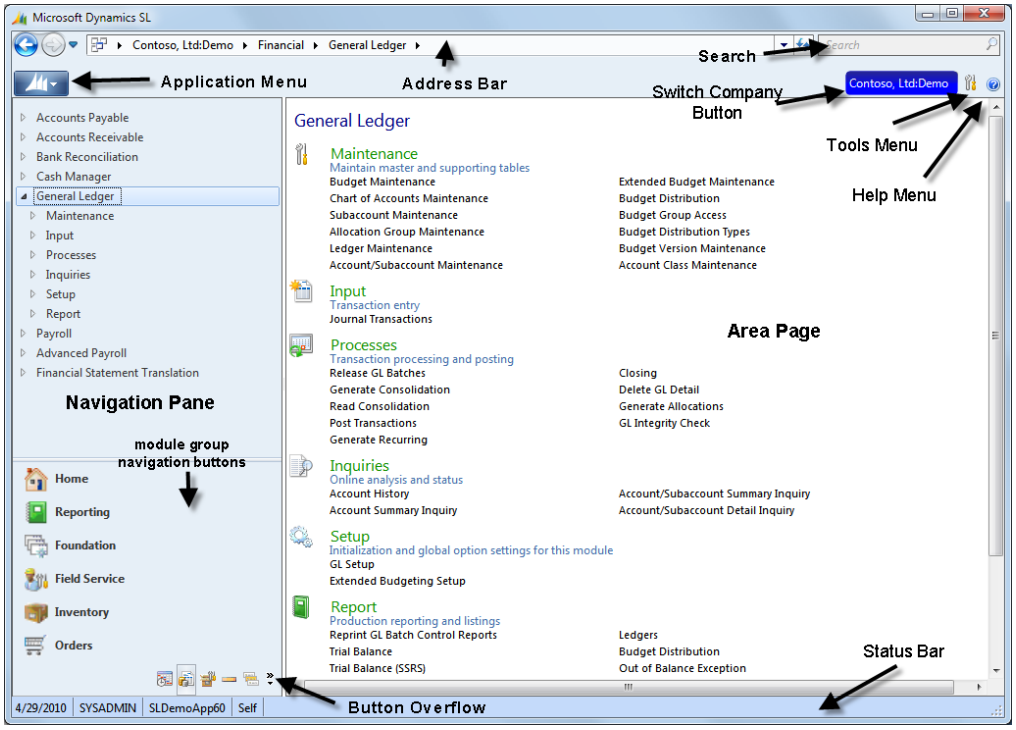

Figure 2: Microsoft Dynamics SL window

1. At the bottom of the navigation pane, click the button for the module group you want to open (for example, Financial to open the General Ledger module), or click **All Modules**.
2. In the navigation pane, locate the module the application screen resides in and click to expand it.
3. Locate the screen either in the navigation pane (double-click to select the screen) or in the application pane (single-click to select it).

- OR -

1. Click **Application** on the menu bar to display the Application menu.
2. On the Application menu, click **Open**, and then select the module to which the application screen belongs. The module's screens and reports appear in the navigation and application panes.
3. Locate the screen either in the navigation pane (double-click to select the screen) or in the application pane (single-click to select it).

Selecting Objects

You must first select an object before you can move, resize, copy, delete or modify its properties. The object you select can be a text box, combo box, check box, label, button, form or frame.

Selecting an Object

You must select an object in order to modify it.

To select an object:

Click on the object. A thin dark border with sizing handles at each corner and on the middle of each side appears around the object. The object's name appears in *Properties*. The following example shows an application screen with **Fiscal Year** selected.

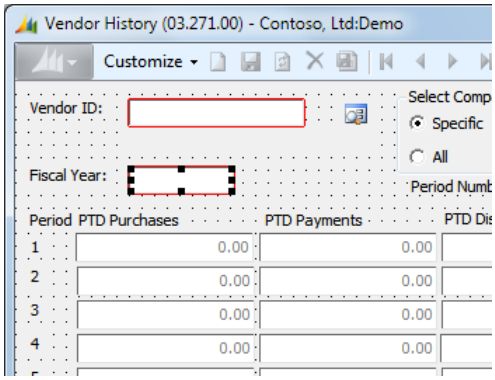

Figure 3: Vendor History (03.271.00)

Note: Occasionally, objects overlay each other, causing you to accidentally select the wrong object or to select the frame the object is in. To select the correct object, first select another object outside the frame the desired object is in, and then select the desired object. Be careful not to accidentally select an area outside the object you are trying to select.

Moving Objects

Once you have selected an object, you may wish to move it to a different location on the screen. To optimize the screen for data entry, you may wish to move several related objects closer to each other to facilitate data analysis or data entry by users. Or, you may wish to move infrequently used objects to the lower right corner of the screen.

Important Points about Moving Objects and Screens

You can reposition either the entire application screen or a specific object that appears within the screen. You can move the entire screen or an object by dragging and dropping it to a new location. Important points about moving objects:

- The name of the object with focus appears in *Properties* so that you know which object is selected.
- Moving a frame also moves any objects it contains.
- You cannot move objects from one application to another.
- You cannot move option buttons outside of the frame or panel that contains them.
- Multiple objects can occupy the same screen location. If all objects are visible, the object placed last appears on top with the others showing around the edge.

Dragging and Dropping an Object

Select the object with the mouse, and then use the mouse to drag the object to the desired location on the screen.

Be sure to move any label associated with the object you are moving.

To move an object using drag and drop:

1. Click and hold on the object to be moved.
2. Drag the object to the desired location.
3. Release the mouse button to drop the new object at its new location.
4. If the object you moved is associated with a label, repeat this procedure to move the label.

To move an object from one tab screen to another:

1. Click and hold on the object to be moved.
2. Drag the object to the desired location by first dragging the item up to the tab label of the current tab and then over to the tab label of the destination tab. As you drag the object over to the destination tab label, that tab and its contents will become visible.
3. Release the mouse button to drop the object at its new location.
4. If the object you moved is associated with a label, repeat this procedure to move the label.

Setting Snap-to-Grid

While moving or resizing objects, set snap-to-grid so that the objects automatically align with each other when you use the mouse to move or resize them. This ensures a more professional looking screen.

To set the snap-to-grid:

1. Choose **Customize | Customize Mode** from the menu bar.
2. Click **Customize | Customization Options** from the menu bar to display Customization Options.
3. Select **Show Grid** so that a grid of dots appears on the display to aid you in aligning objects on the application screen while you are customizing it.
4. If desired, change the **Width:** and **Height:** settings. The larger the number, the farther apart the dots on the display will be.
5. Select **Align Objects to Grid** so the upper left corner of each object you drag will line up with a dot on the grid. This setting only affects objects that you drag or create during this session; it does not rearrange existing objects on the application screen.

Using the Customize Menu

The **Customize** menu appears next to a selected control when you click the right mouse button. This gives you a quick way to access the most common actions used in customization. You can also access the **Customize** menu by clicking **Actions** on the application toolbar and then clicking **Customize Mode**.

The **Customize** menu is available if Customization Manager is part of your Microsoft Dynamics SL installation, and if you have access rights to the module.

For all customization activities except selecting the customization level, you must open the screen to be customized before selecting the customize function.

See “Actions menu” for additional information.

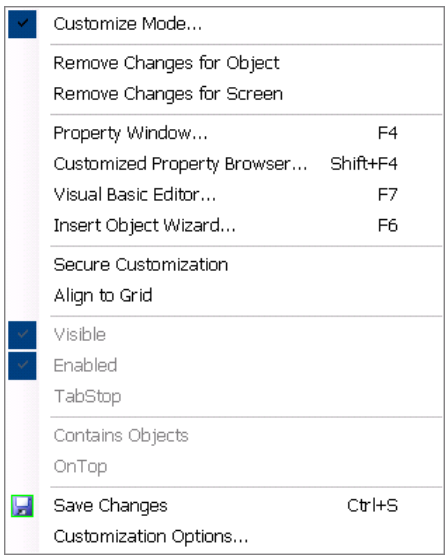

Figure 4: Customize menu – Visual Basic for Applications

Customize Mode

Activate customize mode by first opening the application screen you want to customize, and then selecting **Customize Mode** from the **Actions** menu. You are now in customize mode for the currently open application screen.

To deactivate customize mode, click **Customize Mode** on the **Customize** menu. The software prompts you to save any changes you have made to the application screen (if you have not done so already). The application screen returns to runtime mode. After you close and reopen the screen, all customizations entered during the session become active in the software.

Note:

- You cannot enter customize mode with more than one Microsoft Dynamics SL executable file active on the desktop. However, it is possible to have a screen and one of its subscreens open at the same time.
- All invisible objects become visible while you are in customize mode so that changes can be made to these objects. This includes standard objects for any Microsoft Dynamics SL modules not yet released.

Remove Changes For Object

Enables you to remove all changes for the object that currently has focus. If the object is a new object, it is deleted.

Note: Objects are not removed until you exit customize mode and close, then re-open, the customized screen.

Remove Changes For Screen

Enables you to remove all changes made to the current screen. This causes the screen to display in its standard form.

Note: Objects are not removed until you exit customize mode and close, then re-open, the customized screen.

Property Window

Enables you customize properties of data objects on an application screen. When you select an object and then select **Property Window**, *Properties* displays values for that object, and you can enter customized values for the properties of the object. See "Properties" on page 64 for more information. To check whether objects are already customized, see "Identifying Customized or New Objects" on page 50.

Customized Property Browser

Displays customized properties of an application screen's data objects. Available if **VBA features** was selected when Microsoft Dynamics SL was installed.

Visual Basic Editor

Enables you to enter and associate Visual Basic for Applications code with an object. Each object has specific actions that can be performed on it during runtime.

Insert Object Wizard

Enables you to insert new screen objects. See "Insert Object Wizard (91.252.00)" on page 68 for more information.

Secure Customization

Enables you to secure customization features and access to the functional pieces of the Customization Manager. The software provides security features for preventing users with insufficient access rights from modifying application modules. This security is applicable to individual objects within a customization or to the entire customization. See "Customization Security" on page 55 for more information.

Align to Grid

Forces the selected object to be aligned to the nearest grid coordinate.

Visible, Enabled, TabStop

The current state of the selected objects.

Contain Objects

Changes the order of the selected container on a form. The selected container object will contain all underlying objects.

On Top

Changes the order of the selected container on a form. The objects contained by the selected object will be placed behind it.

Save Changes

Enables you to save (record in the database) all changes or additions made to a customization.

Note: When creating a customization, it is important for you to save your changes frequently. This can prevent the loss of valuable time and effort should some type of system-related error occur.

Customization Options

Enables you to make modifications to Customization environment settings for the current logged-in user.

Using the Insert Object Wizard to Add Objects

You may wish to add objects to a screen to support additional business functions. Or, you may wish to add an object to a screen to spare the user from bringing up a different screen just to look at or enter data in one field. Customization Manager allows you to add two types of objects to an application screen: unbound and data-bound. Unbound objects are not associated with data in the screen or database. Data-bound objects require a relationship bound to data elements (fields) in the database.

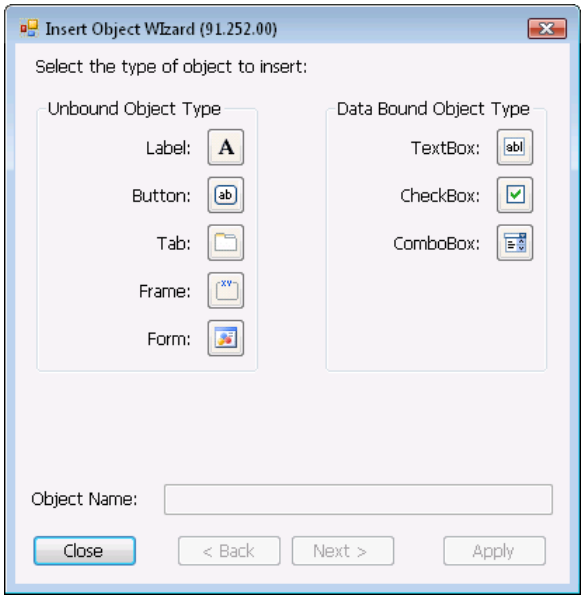

Figure 5: Insert Object Wizard (91.252.00)

Preparing to Add an Object

To prepare to add an object to a customized application screen:

1. Display the screen or subscreen to which you are adding the object.
2. Choose **Customize Mode** from the **Actions** menu.
3. Choose **Customize | Insert Object Wizard** from the **Customize** menu.

Adding an Unbound Object

After performing the steps to prepare for adding an object:

1. From *Insert Object Wizard* (91.252.00), click on a choice in the **Unbound Object Type** grouping (a default object name will appear).
2. You can change the object name to any unused object name.
3. Click **Apply** to add this object to the screen.

Adding a Data-Bound Object

After performing the steps to prepare for adding an object:

1. From *Insert Object Wizard* (91.252.00), click on a choice in the **Data Bound Object Type** grouping (a default object name will appear).
2. You can change the object name to any unused object name.
3. Click **Apply** at any time to add this object to the screen.

4. Click **Next >** to display the next screen of *Insert Object Wizard* (91.252.00).

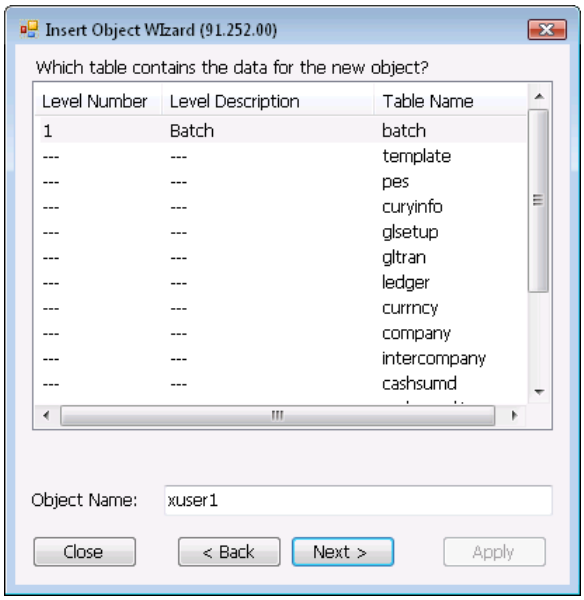

Figure 6: Insert Object Wizard (91.252.00), Table

5. Select the table containing the data for the new object.
6. Click **Next >** to associate a level or field with the new object. If you are finished defining the new object, click **Apply** and then **Close**.
7. If the table you selected is not associated with a level, the *Insert Object Wizard (91.252.00)* window appears. Select the level to which the new object should be bound.

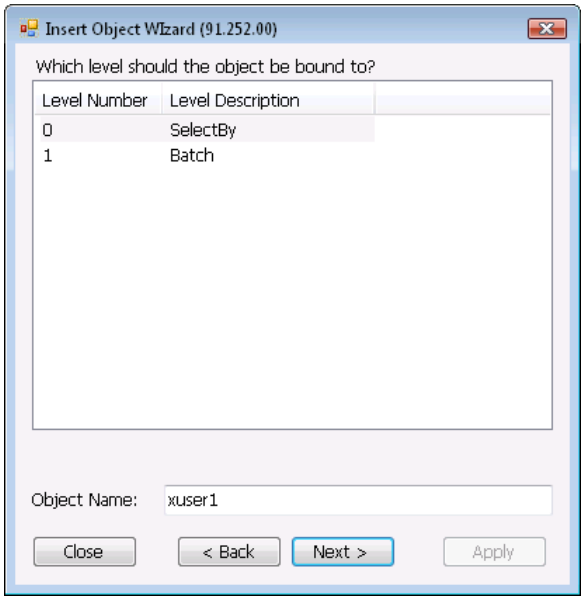

Figure 7: Insert Object Wizard (91.252.00), Level

8. Click **Next >** to associate a field with the new object. If you are finished defining the new object, click **Apply** and then **Close**.

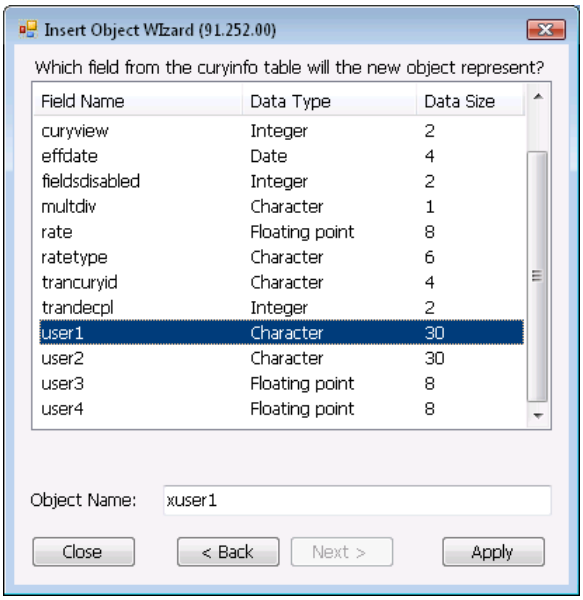

Figure 8: Insert Object Wizard (91.252.00), Field

9. Select the field that the new object represents.
10. Click **Apply** and then **Close**.

Finalizing the Addition of an Object

After performing the steps to add an object:

1. Move the new object to the correct position on the application screen. See “Moving Objects” on page 9.
2. If desired, resize the control. See “Resizing Existing Objects” on page 19.
3. Set the properties on the new control(s). See “Making Simple Property Modifications” on page 17.
4. If the object is a container, move other objects into it. See “Moving Objects” on page 9.

Modifying Objects

You can modify the properties of an object on a screen that you have displayed in customize mode to change how it looks and behaves. For instance, you may wish to color code objects on a screen to help users understand their logical relationship. In addition, you can make the following changes to an object:

- Hide an object from certain users who do not need to see it to simplify their data entry process and unclutter their screen.
- Clear an object's tab stop so the user does not have to waste keystrokes tabbing through fields he or she does not use.
- Set an object's Display Only attribute to allow a user to see it but prevent them from entering data in it.

Making Simple Property Modifications

To make simple property modifications:

1. Select the application or application component to be modified.
2. Choose a specific property to be modified.
3. Confirm that the property needs to be modified.
4. Enter a new property value.

To hide a field:

1. Select the object to be hidden.
2. In the **Customized Value** column, set the value for the **Visible** property to False.

To prevent a user from tabbing to a field:

1. Select the field to be removed from the Tab sequence.
2. In the **Customized Value** column, set the value for the **TabStop** property to False.

Note: To rearrange the sequence in which users tab through the fields, see "Modifying the Tab Order of a Screen" on page 18.

To prevent a user from entering data in a field:

1. Select the object to be restricted.
2. In the **Customized Value** column, set the value for the **Enable** property to False.

To change the label for an object:

1. Select the label.
2. Enter the new value in the **Text** property.

To change the label for a column:

1. Select the data object.
2. Enter the new value in the **Heading** property.

To add or modify a ToolTip:

1. Select the object.
2. Enter the tip in the **ToolTip Text** property.

Implementing Hot Keys for Objects

Implement hot keys to give users quick access to applications without the use of the mouse, tool bars, or popup screens.

To implement a hot key for an object:

1. Select the object to apply a hot key (label, groupbox and button can be modified).
2. Display Properties. Determine if the object you are adding the hot key for contains the **Text** property:
   - If the object has a **Text** property, continue with step 3.
   - If the object does not have a **Text** property select the control immediately preceding the one you wish to apply the hot key for.
3. In the **Customized Value** column, type an ampersand (&) followed by the hot key.

Example: If you want the user to use `ALT+S` as the hot key type `&S` in the Text row.

Modifying the Tab Order of a Screen

Use Customization Manager to modify the order in which the cursor moves through the objects when the user presses the `TAB` key. To optimize data entry, you may wish to customize the tab order for a specific user or for all users who do not typically enter data in a group of objects on the screen. You should also re-sequence the tab order when you move an object. Also, rearrange the tab order after you rearrange fields to ensure that the cursor will move through the fields top to bottom through each "column" of data entry field on the screen when the user presses the `TAB` key. There may be one, two, or three "columns."

To modify the tab order of objects on a screen:

1. Display the screen or subscreen to which you are modifying the tab order.
2. Choose **Customize Mode** from the **Actions** menu.
3. Select the object on the screen for which to change the tab order.
4. Choose **Property Window** from the **Customize** menu.
5. Click on the **TabIndex** row of Properties. The **TabIndex** property is highlighted.
6. Change the **TabIndex** value to the desired result (numerical order of that object in the screen's sequence).
7. Continue to select other objects in the screen, repeating the **TabIndex** changes until they are in the order desired.
8. Save your **TabIndex** property changes before exiting Customization Manager.

Resizing Existing Objects

You may wish to resize an object to make it more visible or to create more open space on a screen. You can resize only one object at a time. Once you select the object, resize it by dragging its borders to the desired size, or by setting a specific size using the object's properties. Some other points about resizing:

- To resize an application screen, click on the window background.
- To resize the selected object horizontally and vertically, position the mouse pointer over a corner of the selected object.
- To resize the selected object in one direction only, position the mouse pointer over the middle of one of the window frame sides.
- The mouse pointer changes to a bi-directional arrow when in the size mode.
- The exact size of the object is represented by the outside surface of the double frame.

Resizing by Dragging

Use the mouse to resize an existing object by dragging the object's border to the desired size.

To resize an object by dragging:

1. Click on the object to select it.
2. Position the mouse pointer over the window frame, then click and hold. Drag in the desired direction until the frame reaches the desired size.
3. Release the mouse button to retain the new size of the object.

Note: An outline representation of the object moves with the pointer. After completing the size operation, the object assumes its new size. Sizing pointers appear for all objects, including those for resizing a horizontal, vertical, or diagonal edge of an object.

Resizing by Simple Property Modification

To resize an object by modifying its properties:

1. Display *Properties*. See "Making Simple Property Modifications" on page 17.
2. Choose the specific property you wish to change, such as the **Height** or **Width**.
3. Enter the specific desired size as the value of that property.

Deleting Customizations

You may wish to delete an object (such as a button or a field label) that was added for the wrong user or when it is no longer needed. Customizations can be deleted for the record or object level.

Deleting Object Level Customizations

To remove customization(s) for an object:

1. Select the object.

   Note: For objects provided by Microsoft Dynamics SL, the customizations will be removed. For added objects, the object itself will be removed. Make sure you select the correct object. The selected object will not be deleted until you exit the screen, so it may not be immediately clear which field will be removed, or which field will have its customization removed.

2. Choose **Remove Changes for Object** from the **Customize** menu.
3. Click **OK** to confirm the deletion.

Deleting Screen Level Customizations

You may wish to view the customized screen to be sure you are deleting the correct customization and level. Use the **Actions** menu to delete the customization while it is displayed. Deleting a currently loaded customization enables you to quickly delete a customization at the desired level.

This operation deletes all customizations for the displayed screen and selected level.

To delete the current customization:

1. Start Microsoft Dynamics SL. The toolbar displays.
2. Choose **Administration | Select Customization Level** to display *Select Customization Level* (91.260.00).
3. In *Select Customization Level* (91.260.00), select the level of the customization you wish to delete.
   - To delete a customization that applies to a specific user, select One User then select the correct UserId.
   - To delete a customization that applies to all users, select All Users.
   - To delete a customization that applies to a specific customization group, select One Group then select the correct GroupId.
4. Open the screen whose customization you wish to delete.
5. Choose **Customize Mode** from the **Actions** menu.
6. Choose **Customize | Remove Changes for Screen**. The customization continues to appear until you close and reload the screen.

Saving Customizations

You can save new customizations by choosing **Save Changes** from the **Customize** menu. Save customizations after:

- Loading an existing customization
- Creating a new customization

Saving a customization saves all customization changes up to the point when you choose **Customize | Save Changes**. Once the customization changes are saved, the customized screen reflects these changes the next time it is loaded.

To save customizations:

1. Ensure the software is in the customize mode.
2. Choose **Customize | Save Changes**.

To save customizations while in the VBA IDE:

Choose **Save** on the File menu.

Exiting Customization Manager

To exit customize mode:

1. Choose **Customize Mode** from the Customize menu. If you have not yet saved your customizations, you will be prompted to save any changes that you have made to the application screen.
2. Choose **Yes** to save customizations and exit customize mode. The application screen returns to runtime mode, and all customizations entered during the session become active.

Data Entry and Validation Rules

Overview

The “Data Entry and Validation Rules” section provides information and procedures for ensuring that data entry on screens is done efficiently and accurately. These tasks include:

- Modifying Edit Masks
- Setting Required Fields
- Setting Valid Min/Max Values
- Modifying Default Values
- Adding and Modifying Possible Values Lists
- Modifying Code Using Visual Basic for Applications

Modifying Edit Masks

Use edit masks to automatically format text entered in fields on a screen and to validate that information entered in a text field is of the correct data type.

Two properties work together to control the values that an operator can enter for a given field. The **Edit Type** property provides a list of pre-defined standard edit masks as well as a **Custom** option. The **Mask** property is used to define a custom data edit mask.

Example: The mask “99/99/99” displays dates entered in the format of “100397” as “10/03/97”.

To add or select a custom edit mask for a text box:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to modify, and then click **OK**.
3. Open the screen for which you want to add a custom edit mask.
4. Choose **Customize Mode** from the **Actions** menu.
5. Click on the text box for which you want to add a custom edit mask.

6. Choose **Property Window** from the **Customize** menu. You can use F4 to open *Properties*.

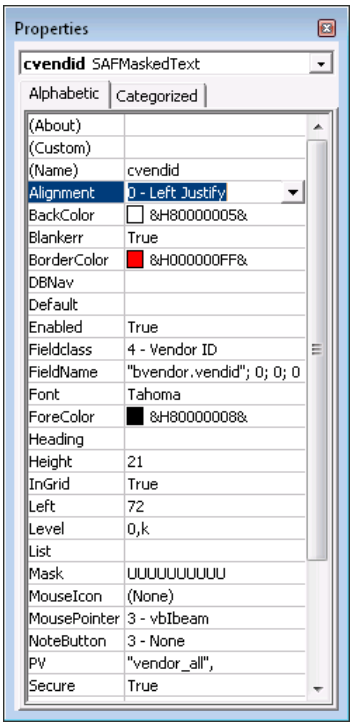

Figure 9: Properties

7. Click in the **Mask** row of *Properties*. The **Mask** property is highlighted.
8. Enter the edit mask for the selected field in the property edit field using the following mask characters:

   X — Accept uppercase and lowercase letters, numbers, and symbols (exceptions: * and ?)

   M — Accept uppercase and lowercase letters, numbers, and symbols, including * and ?

   U — Accept uppercase letters, numbers, and symbols only

   L — Accept lowercase letters, numbers, and symbols only

   A — Accept uppercase and lowercase letters only

   V — Accept uppercase letters only

   N — Accept uppercase and lowercase letters and numbers only; no symbols

   H — Accept hexadecimal characters, that is 0-9 and A-F, only

   W — Accept uppercase letters and numbers only; no symbols

   9 — Accept numbers only

   Caution: Allowing the * or ? character in a key field can create problems if you include the field in a SQL statement that uses the LIKE function.

   Any character entered as part of the edit mask that is not included in the list above is treated as a literal. Literals are included in the display of values entered for the selected object, but are not stored as part of the data.

   Example: If you enter the mask 99-99, the data object accepts entries of up to four numeric digits. The software automatically displays the dash (-) character as part of the number entered. However, Microsoft Dynamics SL does not save the dash with the number in the database.

   - If one of the reserved characters listed above should be treated as a literal, it must be preceded by a forward slash (/).

Example: The edit mask for a domestic telephone number in the software is "(999) 999-9999 E/xt 9999". The slash is required before the "x" in Extension to prevent its use as a placeholder.

- If the standard screen already has an edit mask for the data object for which you enter a mask, you can enter a mask that is more restrictive, but not more permissive, than the existing mask.

Example: If the mask is www (uppercase letters and numbers), you could change it to vvv (uppercase letters) or 999 (numbers) but nothing else (xxx, etc.).

9. Close *Properties*.

Setting Required Fields

Required fields must have information entered by the user before the screen is processed. Required fields are controlled by the **BlankErr** property. If the **BlankErr** property is set to True, an error occurs if the field is blank when the screen is processed. If the **BlankErr** property is set to False, no errors will occur.

Note: You cannot change **BlankErr** to False if its standard value is True.

Note: Required fields for which controls are added through customizations will have colored borders to emphasize that data must be entered in them.

To make a data object a required field:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to modify, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen that contains the field you want to make required.
4. Click **Customize Mode** on the **Actions** menu.
5. Click on the object you want to make required.
6. Click **Customize | Property Window**.
7. Click in the **BlankErr** row of *Properties*.
8. Click the arrow in the column to the right of **BlankErr**.
9. Select **True**.
10. Close *Properties*.

Setting Valid Min/Max Values

Use the **Min** and **Max** properties to specify a range of acceptable data values for an object. Both numbers and dates can be restricted. For a particular object, it is possible to set a minimum value, a maximum value, or both.

Note: The minimum and maximum values for standard Microsoft Dynamics SL objects may be modified, but the new range must be more restrictive than the original. If a larger maximum value or a smaller minimum value is entered, an error occurs.

To set Min/Max values for an object:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to modify, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to set min/max values.
4. Click **Customize Mode** on the **Actions** menu.
5. Click on the object for which you want to set min/max values.
6. Click **Customize | Property Window**.
7. Click in the **Min** row of *Properties*. The **Min** property is highlighted.
8. Enter the minimum acceptable value for the selected object in the value column for **Min**.
9. Click in the **Max** row of *Properties*. The **Max** property is highlighted.
10. Enter the maximum acceptable value for the selected object in the value column for **Max**.
11. Close *Properties*.

Modifying Default Values

Use the **Default** property to automatically populate a field with information. Default values are useful for fields whose value seldom changes.

Example: To automatically display “AZ” in the State field for a billing address, enter AZ in the Default property for that field.

To modify the default value for an object:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to modify, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to set a default value.
4. Click **Customize Mode** on the **Actions** menu.
5. Click on the object for which you want to a default value.
6. Click **Customize | Property Window**.
7. Click in the **Default** row of *Properties*. The **Default** property is highlighted.
8. Enter the value you would like displayed by default for the selected object in the value column for **Default**.

   Example: To set the default for a field to AR, change the property to just the letters AR. Some defaults look like 0;"0";0;127;0. That setting would be changed to just the letters AR.

9. Close *Properties*.

Adding and Modifying Possible Values Lists

The Possible Values lists can be created or changed in two ways. The first is by using the List property for a particular data object. The second is by using the PV property for a particular data object.

Use the List property for simple lists. the PV property incorporates the ability to use an SQL SELECT statement to produce the possible values list.

How to Add or Change the Possible Values by using the List Property

Use the List property to define valid entries for a particular data object. There are two kinds of values lists:

- **Fixed** — The **Fixed** list is only used with standard Microsoft Dynamics SL objects. Operators may not add to or delete from this kind of list, but individual items may be made invisible to users by selecting the **Inactive** property.
- **User Specified** — The **User Specified** type is reserved for use with text fields that do not have a standard values list. If the control is a Text Box, the values are displayed in a Possible Value (PV) window. If the control is a combo box, the values are displayed in the drop-down list for the combo box. For example, you have added the User1 field to a screen's grid and want to specify a list of choices from which the users may select.

To add a user specified possible values list:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to change, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to add a user specified possible values list.
4. Click **Customize Mode** on the **Actions** menu.
5. Click the Text Box or Combo Box for which you want to add a possible values list.
6. Select **Property Window** from the **Customize** menu.
7. Click in the **List** row of *Properties*. The **List** property is highlighted.
8. Click the ellipsis (...) to the right side of the value column for **List**. *Values List* is displayed.

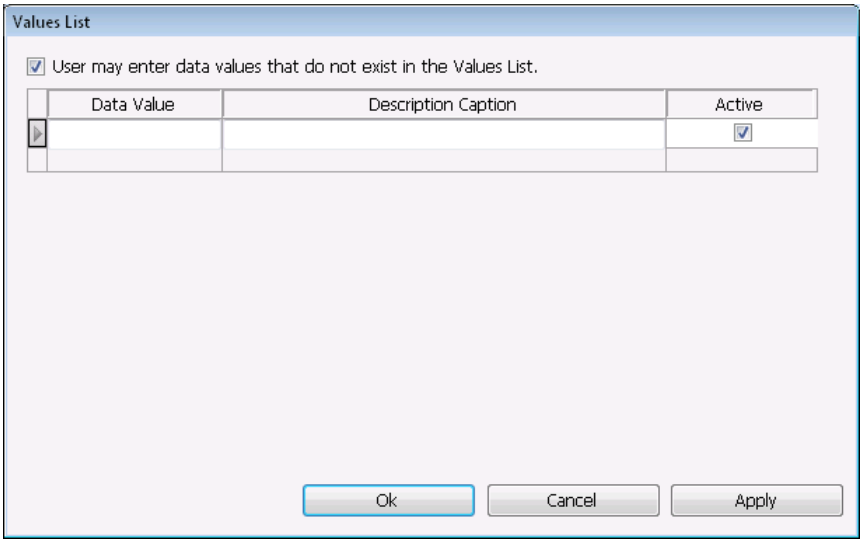

Figure 10: Values List

9. Enter a value for each acceptable value in the **Data Value** column.

10. Enter a complete description of each acceptable value in the **Description Caption** column. Information in the **Description Caption** column is all that is displayed for combo boxes.
11. Do one of the following:
    - To force the entry of one of the possible values entered in *Values List*, click to clear the **User may enter data values that do not exist in the Values List** check box.
    - To make the values be suggested values and let the user make other entries, click to select the **User may enter data values that do not exist in the Values List** check box.
12. Close *Values List*.
13. Close *Properties*.

To change a fixed possible values list:

Use the following instructions to remove individual items from a standard Microsoft Dynamics SL object's possible values list. The items are not actually removed from the list. However, they are, made inactive, and will not be displayed as a possible value.

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization to change, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to set a default value.
4. Click **Customize Mode** on the **Actions** menu.
5. Click the object for which you want to change a possible values list.
6. Select **Property Window** from the **Customize** menu.
7. Click in the **List** row of *Properties*. The **List** property is highlighted.
8. Click the ellipsis (...) to the right side of the value column for **List**. *Values List* is displayed.

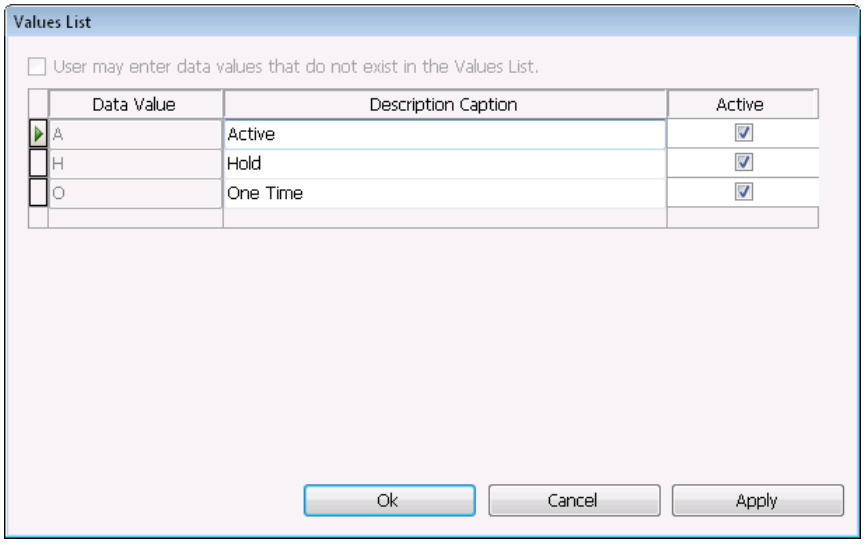

Figure 11: Values List

9. To remove an item from the displayed possible values list for the selected object, clear the **Active** column for that item.
10. Close *Values List.*
11. Close *Properties*.

How to Add or Change the Possible Values by using the PV Property

For the *Possible Value*s window to display, a row must be present in the PVREC table. The PVREC table must be the same value as the stored procedure name. A text file must be created that has the specific format shown here. We recommend that the file name extension is CSV.

The file specifications are shown here with an example:

Element	Example
Stored Procedure Name (also PVID):	• "XCustomer_all",
Quick maintenance Function:	• "0826000",
Language:	• "",
SQL Statement:	• "Select * From Customer Where CustID like @parm1 order by CustID,
PV List Box Caption:	• "Customer",
Field 1 Data: Record.FieldName; Field Header; Field Class; Supplemental Index Number	• "Customer.CustId;Customer ID;003;0",
Field 2 Data: Record.FieldName; Field Header; Field Class; Supplemental Index Number	• "Customer.Name;Name;110;Customer1",
Field 3 Data: Record.FieldName; Field Header; Field Class; Supplemental Index Number	• "Customer.CurrBal;Balance;0;0",
Field 4 Data: Record.FieldName; Field Header; Field Class; Supplemental Index Number	• "Customer.Phone;Phone:0;Customer6",
Field 5 Data: Record.FieldName; Field Header; Field Class; Supplemental Index Number	• "Customer.Zip;Zip;0;Customer7",
Field 6 Data: Record.FieldName; Field Header; Field Class; Supplemental Index	• "Customer.Status;Status;0;Customer2"

NOTE: The line must be one long line in the text file. The semicolons, commas, and quotation marks must be in this exact configuration. No comments are allowed.

This is the text file that is created from this example:

"XCustomer_All","0826000","","Select * From Customer Where CustId like @parm1 order by CustId","Customer","Customer.CustId;CustomerId;003;0","Customer.Name;Name;110;Customer1","Customer.CurrBal;Balance;0;0","Customer.Phone;Phone:0;Customer6","Customer.Zip;Zip;0;Customer7","Customer.Status;Status;0;Customer2"

To add a possible values list by using the PV property:

1. Create an SQL statement.
2. Create a CSV file that is named the same as the SQL statement and using the format that was discussed earlier.
3. In the Microsoft Dynamics SL window, click **Administration | Utilities | Possible Values Import**. The *Possible Values Import* (PV.REC) screen appears.
4. Click **Enter file name** to browse for the CSV import file that you created in step 2.
5. Click **Begin Process** button. If the PV is already on file, a message that indicates that the PVREC already exists will be displayed and request direction.
6. Open the screen for which you want to set the possible value list.

7. Click **Customize Mode** on the **Actions** menu.
8. Click the object for which you want to add or change a possible values list.
9. Select **Property Window** from the **Customize** menu.
10. Type the stored procedure name in the PV box, it must contain quotation marks(" ") around the name and a comma(,) after the name. Using this example, type the following: "XCustomer_all",

Modifying Code Using Visual Basic for Applications

Use Visual Basic for Applications (VBA) code to set properties that are conditional upon other information on a Microsoft Dynamics SL screen.

Note: This option is available to sites that selected VBA Features when they installed Microsoft Dynamics SL.

Example: A company's operational policy mandates that only some users within the organization are allowed to modify the credit limits for certain key customers. Customization Manager, by default, allows fields to be marked Disabled or Invisible for certain users. However, in this case, some users would be allowed access to credit limit changes for certain customers, but not others. To accomplish this, the chk() event for the Customer ID must contain VBA code that evaluates the factors that determine whether a particular user can indeed modify credit limit information. These factors could include any or all of the following: **User ID**, **Customer ID**, **Customer Class**, and **Credit Limit Amount**. Once the VBA code evaluates these factors, it would then enable or disable the credit limit information for the particular user.

To modify a property using VBA Code:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.

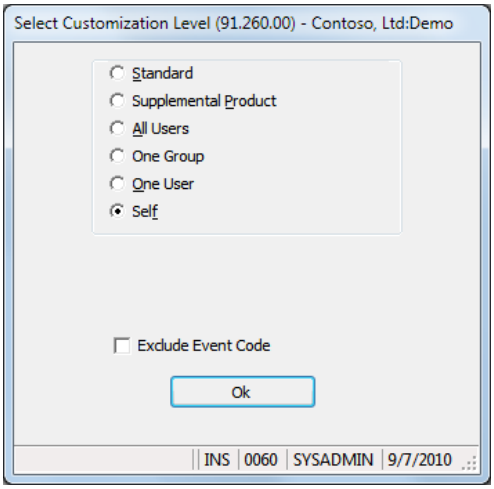

Figure 12: Select Customization Level (91.260.00)

2. Select the level of the customization to modify, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to add conditional logic.
4. Click **Customize Mode** on the **Actions** menu.
5. Click on the object for which you want to add conditional logic.

6. Open the **Customize** menu and click **Visual Basic Editor**. The Microsoft Visual Basic editor window appears.

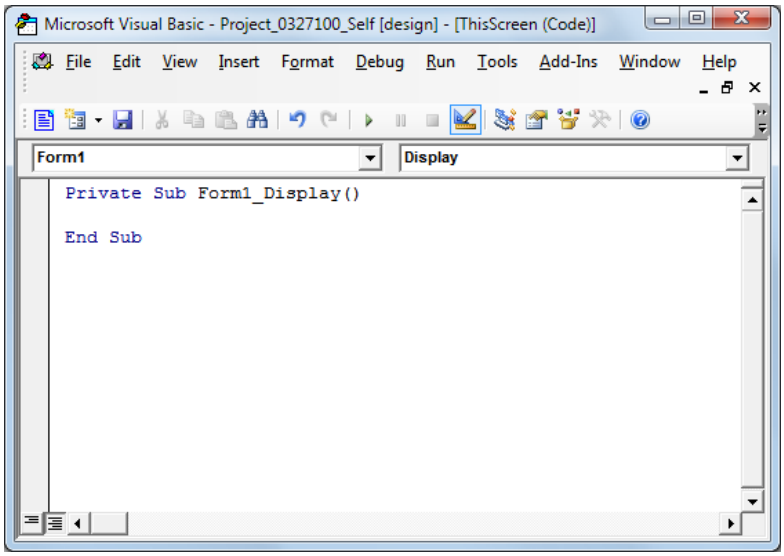

Figure 13: VBA IDE Window

7. In the **Event** box, select the event for which you want to add code.
8. In the Visual Basic editor text box, type the event code for the selected object.

   Note: For detailed information on appropriate syntax and a list of available functions and statements, see the Customization Manager Reference Visual Basic for Applications online help or user guide.

9. Close the *Visual Basic Editor* window.

Advanced Functionality

Overview

The “Advanced Functionality” section provides information and procedures for adding and modifying advanced functionality in modules. The tasks included in this section are:

- Modifying Database Values Using Visual Basic Code
- Implementing and Maintaining Additional Tables
- Using Supplemental Product Level

Modifying Database Values Using Visual Basic Code

Use Visual Basic code to set properties that are conditional upon other information on a Microsoft Dynamics SL screen.

To modify database values using Visual Basic code:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
2. Select the level of the customization you wish to modify, and then click **OK**.
3. Open the screen for which you want to add conditional logic (Visual Basic code).
4. Choose **Customize Mode** from the **Actions** menu.
5. Click on the object for which you want to add conditional logic (Visual Basic code).
6. Choose **Customize | Visual Basic Editor**.
7. In the **Event** box, select the event for which you want to add code.
8. In text box, type the event code for the selected object.

Note: For detailed information on appropriate syntax and a list of available functions and statements, see the Customization Manager, Visual Basic for Applications online help or user guide.

Implementing and Maintaining Additional Tables

You can create and add tables to the SQL database. These tables can then be accessed and updated as if they are part of the original database. Tables are created if the user-defined fields, available on each Microsoft Dynamics SL record, are inadequate.

To create an additional table:

1. Define the table and its index using the SQL Create Table and Create Index statements. For example:

```
Create Table CustInfo (
Custid CHARACTER     char (10),
Trainee1             char (20),
Trained1             smalldatetime,
Trainee2             char (20),
Trained2             smalldatetime,
Tstamp timestamp)
GO
Create UNIQUE Clustered Index Custinfor0 ON Custinfo (Custid)
Go
```

2. Run the Create Table and Create Index statements using the appropriate SQL utilities to actually create these tables in a specific database.

   Note: Experience with SQL syntax, relational database design, and index optimization techniques are prerequisite skills necessary for performing these types of customizations.

To include an SQL table not already accessed by a screen:

Regardless of whether an SQL table is new (one created by you) or an existing Microsoft Dynamics SL SQL table, if it is an additional table not accessed by the screen to be customized, you must access the table manually and update it by entering Visual Basic for Applications statements within the Customization Manager module's *Visual Basic Editor(VBA IDE) Window*.

You must define the SQL table using the Visual Basic for Applications code Type command. This definition is usually in a separate file that is then copied to the Microsoft Dynamics SL Programs directory. For Visual Basic for Applications, the file will need to be added to the project in the *VBA IDE*. Select File | Import File from the *VBA IDE* menu with the modules selected to add a DH file.

1. In the Form1 Load event (under Event), there must be a Setaddr call and a Sqlcursor call for each new record to be accessed. For example:

```
Call Setaddr("bfilename", bfilename, nfilename,
Len(nfilename))
Call Sqlcursor(c1, NOLEVEL)
```

2. Once these calls have been made, the Customization Manager module recognizes these records as being part of the current screen and allows you to add data objects associated with these objects.

Using Supplemental Product Level

You can create and export customizations in a form that can be installed in other systems as a packaged customization. These customizations are installed in their own level and are referred to as supplemental products. Using this function specifies that “supplemental product” customizations should appear on screen.

Customization Management

Overview

Often, a customization is only part of a business solution that is typically developed on a development system, tested, and verified against specifications. The entire solution may consist of one or more customizations, new tables, new indices, a new stored procedure, special reports, and other elements.

Many customizations are created simply to affect the way a specific user works with a particular application screen. Once you create a customization for a specified user, only that user will be affected by that customization. For example, suppose you created a customization that hides sales terms information from an Accounts Payable screen for Joe Brown. In order to hide the sales terms information from Sally Smith, you must create an export file containing the customization you made for Joe, modify the export file for Sally, and import the modified customization.

This section describes how to move a customization to a target system and how to give other users access to the customizations you create. These customization management tasks include:

- Understanding Customization Levels
- Exporting Customizations to a File
- Modifying the Export File
- Delivering Customizations to Other Sites and Users
- Deleting Customizations Using Export and Import
- Determining What Has Been Customized
- Diagnosing Problems

Understanding Customization Levels

Definition of Customization Levels

With user levels, you can define customizations and specify to which user the customization is available. The software places users in groups called “levels.” Using levels, you can define a customization once and have it apply to all users, to a specific user, or to yourself, so that you can support your company’s business rules. Additionally, higher levels are used in diagnosing problems, for supplemental products, and for internalization and language selection.

The common customization levels are:

Level	Applies to
All Users	All users on the system
One User	The user you specify
Self	Your UserId
One Group	All users that are members of a Customization Group

Many other levels exist in the software but are used for special purposes such as selecting the language screens display in. The Standard level is used for diagnosing problems (see “Customization Management” on page 39) and the Supplemental Product level is used for integrating other products into Microsoft Dynamics SL.

When you display a screen, the software loads the customizations at each level, in turn, until all customizations for the selected level and all higher levels are loaded. That is, if you select the All Users level, the software does not load customizations for a specific user. If you select the One User level, the software loads customizations at the All Users level and the customizations for the user you specify.

As a result of this loading process, the customizations for the One User and Self levels depend on values at the All User level. For instance, suppose you add a button at the All Users level, then use the button in another customization at the One User level for a specific user. If you later remove the button from the customization at the All Users level, the button will not load when the screen is displayed and will not display for the specific user.

Strategies for Using Levels

The strategy with which you use levels can simplify your customization management process and effectively support your business rules. By properly using levels, you can create a customization for a specified user without affecting other users.

Example: If you created a customization that hides sales terms information from an Accounts Payable screen for Joe Brown, the customization would not affect any other users. In order to hide the sales terms information from Sally Smith as well, you must export the customization you made for Joe, modify it for Sally, and import the modified customization.

This approach is practical as long as Joe and Sally are the exception; that is, as long as the majority of users need to see the uncustomized version of a screen, and only a few users need to see the customized version.

However, suppose the standard screen allows all users to do something that your business rules say that only a supervisor should be able to do. First, you will want to make a customization apply to all users to prevent anyone from performing the function. Then, you will want to make an additional customization and apply it to the supervisors who need to perform the function.

Suppose, for instance, that the standard vendor screen allows payment terms to be modified and your company’s business rules allow only one supervisor to modify them. You would create two customizations. First, you would create a customization at the All Users level to prevent all users, including the supervisor, from modifying the payment terms. Next, you would create a second

customization at the One User level for the supervisor to allow that supervisor to modify the payment terms.

If you have several sites that all have the same business rule, you would then import both customizations at the other sites. As part of the import process, you would modify the second customization so that the correct supervisor at each site would have access to payment terms. See “Exporting Customizations to a File,” “Modifying the Export File,” and “Delivering Customizations to Other Sites and Users” in “Customization Management“, which begins on page 39.

Setting the Level for a Customization

Before you open the application screen you wish to customize, set the level to which the customization applies.

To set the level for a customization:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.

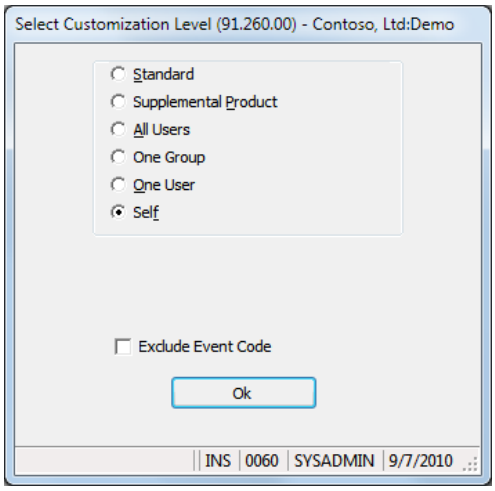

Figure 14: Select Customization Level (91.260.00)

2. Select the level of the customization to modify, and then click **Ok**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

Exporting Customizations to a File

If you create a customization for a specified user, it only affects that user. If you create a customization for all users on one system, it only affects users on that system. In order to apply a customization to other users, or to another system, you first must export the customization into an ASCII file called a .CST file, or "export" file. Then, you must change the customization export file and import the export file to the desired site and users. See "Modifying the Export File" on page 44 and "Delivering Customizations to Other Sites and Users" on page 45.

To export the customization file:

1. In Microsoft Dynamics SL, click **Administration**.
2. Under **Utilities** in the **Administration** details pane, click **Export Customizations**. *Export Customizations* (91.500.00) appears.

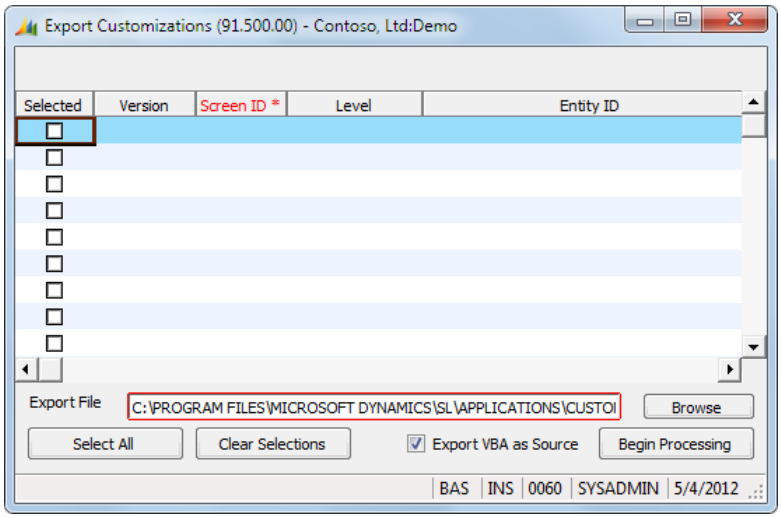

Figure 15: Export Customizations (91.500.00)

3. Name the export file that you want to create, and then select the directory to which it should be exported. Use one of the following two methods:
   - In **Export File**, type the full path and file name for the export file. This includes the drive and folder names.
   - Click **Browse** to move through the directory tree and locate the directory in which to put the export file and use a unique name for the file.

   Note: By default, the customization file is named CUSTOM.CST and is in the Microsoft Dynamics\SL\Applications\ program directory. Any file that already exists by this name in this directory is overwritten unless you rename or move the existing file, or use a different name or directory for the export file that you are creating.

4. Select the customization to be exported. Use one of the following two methods:
   - To select several customizations for export, click **Clear Selections**, and then click to select the **Selected** check boxes to change **Selected** to Yes for each customization that you want to export.
   - To select most of the customizations for export, click **Select All**, and then click to select the **Selected** check boxes to change **Selected** to No for each customization that you do not want to export.

5. Click to select the **Export VBA as Source** check box to export the selected VBA customizations in a source format. Click to clear the **Export VBA as Source** check box to export the selected VBA customizations in a binary format.

   Note:

   - You can export a source format .cst file for VBA customizations that are stored in the database in either binary format or source format.
   - You cannot export a binary format .cst file for VBA customizations that are stored in the database in the source format.
   - For more information about storing VBA customizations in the database in the source format, see the SaveVbaSource setting in the "Appendix A: Solomon.ini Settings" topic in the System Manager Help or user's guide.

6. Click **Begin Processing** to start exporting the selected customizations to the export file. A process status window displays messages about the export process.
7. After the process is completed, click **OK** to close the process status window.

Modifying the Export File

Exporting a customization creates a .CST or export file in ASCII format that contains database keys unique to the system on which the customization was created. In order to make the customization work properly on the target system, you may have to modify the export file before importing it. For instance, if the customization is designed for one user, you will most likely have to change the "UserId" key in the export file in order for the customization to work for the intended user on the target system. Or, you will have to change the "Sequence" key to apply a customization that was made for one user to all users.

If a business solution requires warning or informational messages to display, add these messages through customization event code. The messages can be in the event code or they can be standard messages from the Messages table in the database.

Customizations are stored in the system database and apply to all application databases. The keying structure of the software does not support customizations that apply only to a specific application database.

Some solutions involve database changes. Whether or not these solutions require changes to the export file, the database on the target system has to match the database of the system on which the solution was developed. Database changes that may need to take place in order for a solution to work on the target system include:

- Creating tables.
- Creating indices.
- Creating stored procedures.
- Importing data.

See "Implementing and Maintaining Additional Tables" on page 36 for information on creating tables and indices.

Editing the Export File

Once you have created the export file, modify its keys to the new target location.

To edit the export file:

1. Open the export file in an ASCII text editor, such as Microsoft Notepad or Microsoft WordPad.
2. Change any items that are unique to the system on which the customization was created so they match the corresponding items in the target database.

   Example:

   Before: `Begin Customization Screen: 05260 Sequence: 500 UserId: "SallySmith" CompanyName:""`

   After: `Begin Customization Screen: 05260 Sequence: 500 UserId: "JoeBrown" CompanyName:""`

   Note: In the above example, the sequence is the same on both systems since the customization is for one user. The UserId changes because different people at the different sites use the customization.

3. Save the changes to the file.

Preparing the Database

If the customization requires any new databases, indices, stored procedures or initial data records, apply them to the database on the target system.

Delivering Customizations to Other Sites and Users

Once you have created an export file containing the customization (see “Exporting Customizations to a File” on page 42) and modified it for the desired site or user (see “Modifying the Export File” on page 44), use the Import function to install the customizations in the file on the target system.

The Customization Import and Export functions operate on the same .CST file. The .CST file is in ASCII format, and contains customizations. The .CST file is usually called the export file during the export and file modification processes, and is usually called the import file during the import process.

Before delivering the customization, first verify that it was created for the same version and technical build of Microsoft Dynamics SL that is on the target system, then import the import file.

If the customization uses any custom tables, indices, stored procedures, add them to the target system before using the new customization. See “Implementing and Maintaining Additional Tables” on page 36 for more information on adding tables, indices, and stored procedures. If you have Crystal Reports, see the Crystal Reports user guide for information on implementing any special reports.

Verifying the Microsoft Dynamics SL Version

To protect production data and other files, make sure that the customization you plan to install was developed for and tested against the same release and technical build of the software that is on the target system.

To verify the version:

1. In the Microsoft Dynamics SL window, click **Help | About Microsoft Dynamics SL.** Note that in the *About Microsoft Dynamics SL* window, the version that is installed on the system is listed. The build number is found in the longest segment of the version number.
2. Click **OK** to close *About Microsoft Dynamics SL*.

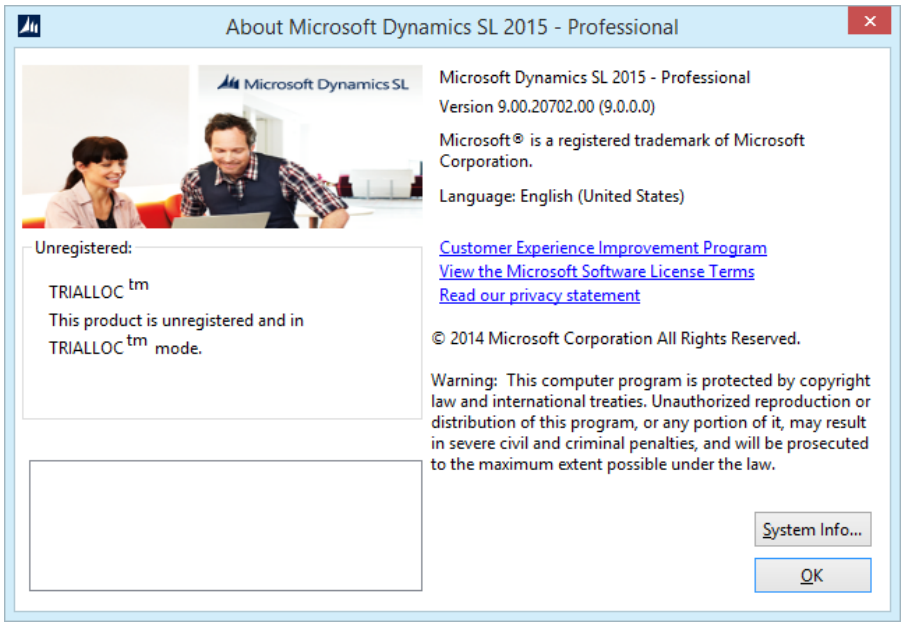

Figure 16: About Microsoft Dynamics SL

Importing the Customization File

Once you have exported a customization and modified it for the desired site or user, use the Import function to install it on the target system. If the customization uses any custom tables, indices, stored procedures, or messages, add them to the target system before using the new customization. See “Advanced Functionality” on page 35 for more information on adding tables, indices, and stored procedures.

To import the customization file:

1. In the Microsoft Dynamics SL window, click the **Administration** button.
2. Under **Utilities** in the application pane, click **Import Customizations**. *Import Customizations* (91.510.00) appears.

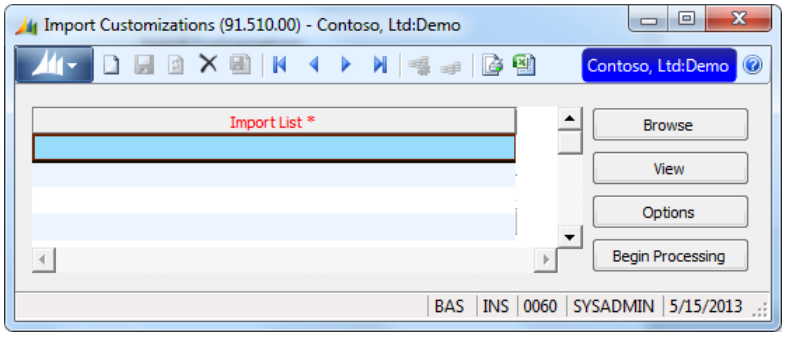

Figure 17: Import Customizations (91.510.00)

3. Click **Browse**. Select the drive where the import file is located. Double-click on directory folders to navigate through the directory tree and locate the directory that contains the import file. By default, the system points to the Microsoft Dynamics SL Application program directory.
   - Select the file that contains the customization to be imported. Highlight the file name and then click **Open.**

   Or,

   - In the **Import list**, type the complete file name including the drive and directory it is in.
4. The file name appears in **Import list**.

   Note: By default, the File Name list includes only customizations; that is, files with the *.CST extension. However, if the list of files includes all types of files or does not include the type of file to be imported, select *.CST from List Files of Type to list only customization files.

5. If desired, click **View** to display the contents of the file in the Import list.

6. Click **Options** to display *Import Options* (91.510.02) and specify how the software should resolve any import file conflicts and errors that may occur.

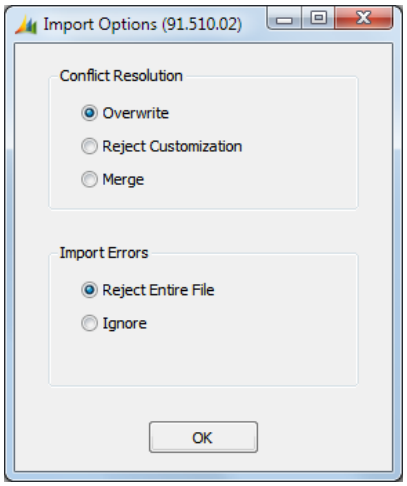

Figure 18: Import Options (91.510.02)

7. In the Conflict Resolution area of *Import Options* (91.510.02), indicate how import file conflicts will be handled. Select one of three options:

   Overwrite — Overwrite the existing customization with the new one. This is the default selection, and ensures that the new version of a customization replaces an existing customization of the same name.

   Reject Customization — Reject the new customization and keep the existing one. This option prevents a new customization from overwriting an existing one if a customization you are importing has the same name.

   Merge — Merge the imported customization with the existing customization. This is an extremely powerful feature that allows two customizations to be merged into one new customization in your system.

8. In the Import Errors area of *Import Options* (91.510.02), indicate how the system should respond to errors that may occur. Select **Reject Entire File**. This is the default selection and rejects the entire import file if any of the customizations had an error.

9. Click **Begin Processing** to start importing the selected file. A process status window displays messages on the import process.

   Note: If an error occurs during the import process, correct the import file and re-import the customization before attempting to use the screen being customized.

10. Once the process is completed, click **OK** to close the process status window.

Deleting Customizations Using Export and Import

Often you create customizations in order to support a business rule. When the business rule changes, you may need to replace or delete a related customization. For instance, if you created a customization to facilitate changing the date on manual checks, you may wish to delete the customization when a policy change requires clerks to stop changing the date on manual checks. You may also need to delete a customization for a specific user when the user leaves the company.

You can use the user interface to delete the currently loaded customization (see “Deleting Customizations” on page 20). Or you can use the software’s Import function to delete any customization without loading it. The latter choice enables a system administrator to remotely create an export file, deliver it to a site, and guide another user through the process of removing customizations.

Preparing the Delete Customization Import File

To prepare the import file to delete a customization:

1. Open the customization import file to be deleted in an ASCII editor such as Microsoft Notepad.
2. Modify the “Begin Customization Screen” statement in the import file as follows:
   - Replace the words “Begin Customization” with “Delete”.
   - Make sure that the rest of the statement lists all database key values necessary to identify a unique customization.

Example:

Before: `Begin Customization Screen: 05260 Sequence: 500 UserId: “JoeSmith” CompanyName:””`

After: `Delete Screen: 05260 Sequence: 500 UserId: “JoeSmith” CompanyName:””`

Note: This import file modification deletes a customization for Joe Smith. If Joe leaves the company, you might send an import file with this statement to the site where Joe used to work to ensure that another employee on site deletes the correct customization.

Note: Place this line anywhere a Begin Customization line can occur. Do not place this line within a Begin Customization and End Customization block of text. The key called Sequence in the above example refers to the level. See “Import Customizations (91.510.00)” on page 78 for more information about the import file. See “Understanding Customization Levels” on page 40 for more information about levels.

To prepare the import file to replace a customization:

1. Use an ASCII editor, such as Microsoft Notepad or Microsoft WordPad, to open the import file containing the customization to be replaced.
2. Copy the "Begin Customization Screen" statement to the bottom of the file.
3. In the "Begin Customization Screen" statement at the top of the file, change the key values (Screen, Sequence, UserId, and Company) as necessary for the target site.
4. In the last statement in the file, replace the words "Begin Customization" with "Delete".

Example:

Top: `Begin Customization Screen: 05260 Sequence: 500 UserId: "JaneDoe" CompanyName:""`

Bottom: `Delete Screen: 05260 Sequence: 500 UserId: "JoeSmith" CompanyName:""`

Note: This import file modification deletes a customization for Joe Smith and replaces it with an identical customization for Jane Doe. If Joe leaves the company and is replaced by Jane, you might send an import file with these statements to the site where Joe used to work. This would ensure that another employee on site deletes the correct customization and that Jane has the customization she needs to do her new job.

To delete the specified customization in the import file:

1. Choose **Administration | Import Customizations** to display *Import Customizations* (91.510.00).
2. Click **Browse** and select the drive where the file containing the customization to be deleted is located.
   a) Double-click on directory folders to navigate through the directory tree and locate the directory that contains the file you wish to delete. Select the file that contains the customization to be deleted. Select the file that contains the customization to be imported. Highlight the file name and then click **Open**.

   Or,

   b) In the **Import list**, type the complete file name including the drive and directory it is in.
3. Highlight the file name in the File Name list, and then click **Open**.
4. In the Import list, type the complete file name including the drive and directory it is in.

   Note: The file name appears in the Import list.

   Note: By default, the File Name list includes only customizations, that is, files with the *.CST extension. However, if the list of files includes all types of files or does not include customization files, select *.CST from List Files of Type to list only customization files.
5. If desired, click **View** to display the contents of the file in the Import List.
6. Click **Options** to display *Import Options* (91.510.02) and specify how to resolve any import file conflicts and errors that may occur.
7. In the Conflict Resolution area of *Import Options* (91.510.02), indicate how import file conflicts will be handled. Select **Overwrite**.
8. In the Import Errors area of *Import Options* (91.510.02), indicate how the system should respond to errors that may occur. Select **Reject Entire File**.
9. Click **Begin Processing** to delete the selected file. A process status window displays messages on the deletion process.
10. Once the process is completed, click **OK** to close the process status window.

Determining What Has Been Customized

When you display a Microsoft Dynamics SL screen, the software loads the customizations at each level, in turn, until it reaches the selected level. If you select the **All Users** level, customizations are not load for a specific user. If you select the **One User** level, the software loads the customizations at the **All Users** level and the customizations for the user you specify.

As a result, the customizations for the One User and Self levels depend on values at the All User level. For instance, suppose you add a button at the All Users level, and then make a customization for a specific user (at the One User level) that depends on the added button. If you later remove the button (at the All Users level), the button will not load when the screen is displayed, and will not be available to be displayed for the specific user.

Knowing which objects have already been customized, and at which level, is important for planning and troubleshooting customizations. In addition, users can determine whether a screen has been customized to ensure that they receive the proper assistance if they contact their system administrator or customer support.

Identifying Customized Screens

Identifying whether a screen has been customized ensures that a user receives the proper assistance if they contact their system administrator or customer support.

To identify a customized screen while it is displayed:

Look for an asterisk (*) to the left of the title of a screen. Customized screens have an asterisk to the left of the title.

Note: If **Option | Save Settings on Exit** is selected, any changes you do such as resizing or moving the window are flagged as customizations and cause the asterisk to appear in the screen's title bar.

To view a list of customized screens and levels:

Choose **Administration | Export Customizations** from the menu bar to display *Export Customizations* (91.500.00). The screen lists each customization and the screen and level it is for.

Identifying Customized or New Objects

Standard objects are provided as part of the software. Before you consider modifying a standard object, determine whether it has already been customized. If you customize a control at the All User level, the custom value becomes the new standard value for lower levels. If you delete a customization that added a control, you will delete the added control and it will no longer be available to anyone who was using it. Therefore, modifying or deleting a control at the All User level may create the need for a customization for a specific user.

To identify whether a control has been customized or added:

1. Display the application screen.
2. Choose **Customize Mode** from the **Actions** menu to enter customize mode.
3. Right-click on the customized application screen to display the **Customize** menu.
4. Select **Customized Property Browser**.
5. For each property shown, compare the values in the **Customized Value** column and the **Standard Value** column.
6. If the **Customized Value** column and the **Standard Value** column both have values, the control has been customized.
7. If the **Customized Value** column is blank but the **Standard Value** column has a value, the control has not been customized.
8. If the **Customized Value** column has a value but the **Standard Value** column is blank, the control has been added.

Note: A value that is set by code will not necessarily appear in the **Customized Value** column in *Properties*, since the logic that sets the customized value may not have run by the time you go into customize mode.

Identifying New Tables

To determine whether an additional table has been added to a customization, inspect the **Form_Load()** event using *VBA IDE*. Additional tables should have a call to the **SetAddr()** function in order to be recognized by Microsoft Dynamics SL.

Identifying Added Routines

To determine whether a particular routine has been added, go to the “General Declarations” section using *VBA IDE* and look for **Sub/Function** declarations.

Diagnosing Problems

Occasionally when customizing an application screen, you may encounter a problem that either causes the customization to work improperly or prevents the customized screen from loading at all. Isolating any customization problem is the first step to diagnosing it.

However, if faulty event code prevents the customization from loading, you may have to bypass the event code in order to load the customization so that you can correct the problem. If a standard application screen appears to have a bug, you may have to load the screen without customizations and note any differences in order to compare the standard version with the customized version.

Isolating Syntax Errors in Event Code

Occasionally, a syntax error in the event code or in a data header may allow the code to compile but prevents the customized application screen from loading. In order to use Customization Manager to correct the event code, you need to bypass the faulty event code so the customized screen loads. Once the customized screen containing the faulty event code loads, you can use Customization Manager to correct the problem.

To isolate a syntax error in event code or a data header that causes an abort:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.

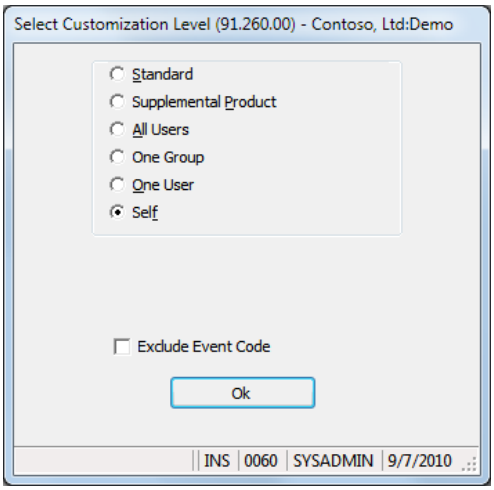

Figure 19: Select Customization Level (91.260.00)

2. Select the **Exclude Event Code** check box.
3. Restart the affected application screen. The customization's event code does not run, but the customized screen displays.
4. Choose **Customize Mode** from the **Actions** menu to enter customize mode.
5. Choose **Customize | Visual Basic Editor**.
6. Click **Events** to view and edit the event code. Correct the syntax error in the event code.
7. Choose **Customize | Customize Mode** from the menu bar to exit customize mode.
8. When prompted to save changes, click **Yes**.
9. Close the affected application screen.
10. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.
11. Clear the **Exclude Event Code** check box, and then click **OK**.
12. Restart the affected application screen.

Note: If the changes were correct, the event code runs completely and the screen displays.

Isolating Changes from Standard Screens

If the standard application appears to have a bug, the event code may include faulty database update or program logic. To diagnose this type of error, run the standard version of the screen without loading any customizations. Customize mode is not available when the Standard level is selected, and no customizations are loaded at screen runtime. The Standard level is for review and comparison only.

To isolate a database or logic error:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.

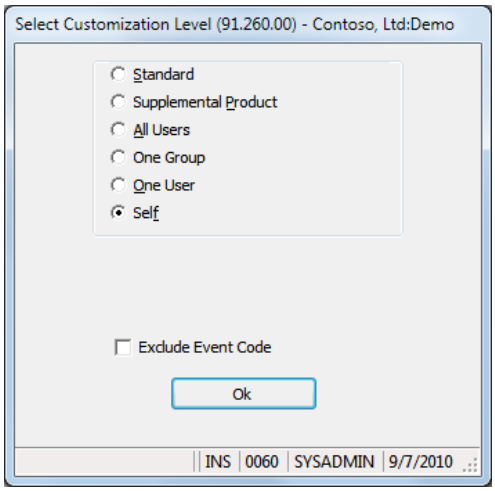

Figure 20: Select Customization Level (91.260.00)

2. Select **Standard**, and then click **OK**.
3. Restart the application screen. The standard or uncustomized version of the screen loads.
4. Evaluate the screen and note any differences between the standard version and the customization.
5. Exit the application screen you are evaluating.
6. Correct the faulty database update or program logic in the event code.

   Note: Customize mode is not available when the Standard level is selected, and no customizations are loaded at screen runtime. The Standard level is for review and comparison only.

7. Choose **Customize Mode** from the **Actions** menu to enter customize mode.
8. Choose **Administration | Select Customization Level** and select the desired level.

Customization Security

Overview

The “Customization Security” section provides information and procedures for securing customizations made to the software. It also includes instructions for securing access to functional pieces of the Customization Manager module. These basic tasks include:

- Securing Customization Functions
- Securing Access to Customization Manager Functional Pieces

Securing Customization Functions

Microsoft Dynamics SL provides security features for preventing users with insufficient access rights from modifying application modules. This security is applicable to individual objects within a customization or to the entire customization.

Once an object is secured, you cannot unsecure the object at lower customization levels. Modifications to secured objects are allowed, but have the following restrictions:

- You cannot change the Enable property that has been disabled.
- You cannot change the BlankErr property of an object.
- You can only narrow the Min/Max value ranges for an object.
- You cannot change the Visible property for Invisible objects.
- You cannot add list items to Combo controls or TextBox objects.

Note: Securing an object does not, in any way, prevent the object from having its cosmetic attributes customized at a lower level.

Securing an Entire Customization

Use the following steps to secure all objects of a particular customization.

To secure an entire customization:

1. Choose **Administration | Select Customization Level** and then select the correct level to specify the level of the customization to secure.
2. Open the application screen to secure.
3. Choose **Customize Mode** from the **Actions** menu to enter customize mode.
4. Choose **Customize | Secure Customization**.

Securing Customizations to Individual Objects

Use the following steps to secure an individual object within a customization.

To secure individual objects:

1. Choose **Administration | Select Customization Level** and then select the correct level to specify the level of the customization to secure.
2. Open the application screen to secure.
3. Choose **Customize Mode** from the **Actions** menu to enter customize mode.
4. Click on the object to secure to select it.
5. Choose **Customize | Property Window**.
6. Click in the **Secure** row of *Properties*. The Secure property is highlighted.

7. Click on the combo arrow and choose **True**.
8. Close *Properties*.

Securing Access to Customization Manager Functional Pieces

You can restrict access to different portions of the Customization Manager module for different users or groups of users. By restricting access to certain parts of the Customization Manager module, you can control the level at which users can make customizations.

Example: You may want to restrict access to Customization Import and Export functions for all users except Administrators. This is accomplished by securing access to *Import Customizations (91.510.00)* and *Export Customizatio*ns (91.500.00) for those users.

Securing Access Visual Basic Editor (VBA IDE)

Use the following steps to secure access to *Visual Basic Editor* (91.251.00) and *VBA IDE* for a user or group of users.

To secure access to Visual Basic Editor Window:

1. Click **Administration | Access Rights Maintenance**.
2. In **Type**, select either **User** or **Group**.
3. Type the group or user ID in **Group / User ID**. The name of the user or group displays in **Name**.
4. Type the company ID in **Company ID** or select **All Companies** to apply the access rights to all companies. If you select **All Companies**, <*ALL*> appears in the **Company ID** box.
5. Type `91.251.00` in **Screen/Report Number**. The **Type**, **Name**, and **Module** boxes display information for *Code Window* (91.251.00).
6. Permit view-only access rights by verifying that each of the following check boxes are cleared:
   - Update
   - Insert
   - Delete
   - Initialization Mode

   Note: If you want to remove View rights, make sure that the 91.251.00 screen does not appear in the **Screen/Report/Query** tab for the user or group.
7. Close *Access Rights Maintenance* (95.270.00) and click **Yes** when prompted to save.

Securing Access to Insert Object Wizard

Use the following steps to secure access to *Insert Object Wizard* (91.252.00) for a user or group of users.

To secure access to the Insert Object Wizard:

1. Choose **Utility | Access Rights**.
2. Click the down arrow in **Type** and highlight either User or Group.
3. Type the group or user ID in **Group/User ID**. The name of the user or group displays in **Name**.
4. Type the company ID in **Company ID** or select All Companies to apply the access rights to all companies. If you select All Companies, <ALL> appears in the **Company ID** field.
5. Type `91.252.00` in **Screen/Report Number**. The **Type**, **Name**, and **Module** fields display information for *Insert Object Wizard* (91.252.00).

6. Disable access rights by clicking the down arrow and highlighting No for each of the following fields:
   - View
   - Update
   - Insert
   - Delete
   - Initialization Mode
7. Close *Access Rights Maintenance* (95.270.00) and click **Yes** when prompted to save.

Securing Access to Export Customizations

Use the following steps to secure access to *Export Customizations* (91.500.00) for a user or group of users.

To secure access to Export:

1. Choose **Utility | Access Rights**.
2. Click the down arrow in **Type** and highlight either User or Group.
3. Type the group or user ID in **Group/User ID**. The name of the user or group displays in **Name**.
4. Type the company ID in **Company ID** or select All Companies to apply the access rights to all companies. If you select All Companies, <ALL> appears in the **Company ID** field.
5. Type `91.500.00` in **Screen/Report Number**. The **Type**, **Name**, and **Module** fields display information for *Export Customizations* (91.500.00).
6. Disable access rights by clicking the down arrow and highlighting No for each of the following fields:
   - View
   - Update
   - Insert
   - Delete
   - Initialization Mode
7. Close *Access Rights Maintenance* (95.270.00) and click **Yes** when prompted to save.

Securing Access to Import Customizations

Use the following steps to secure access to *Import Customizations* (91.510.00) for a user or group of users.

To secure access to Import:

1. Choose **Utility | Access Rights**.
2. Click the down arrow in **Type** and highlight either User or Group.
3. Type the group or user ID in **Group/User ID**. The name of the user or group displays in **Name**.
4. Type the company ID in **Company ID** or select All Companies to apply the access rights to all companies. If you select All Companies, <ALL> appears in the **Company ID** field.
5. Type `91.510.00` in **Screen/Report Number**. The **Type**, **Name**, and **Module** fields display information for *Import Customizations* (91.510.00).
6. Disable access rights by clicking the down arrow and highlighting No for each of the following fields:
   - View
   - Update
   - Insert
   - Delete
   - Initialization Mode
7. Close *Access Rights Maintenance* (95.270.00) and click **Yes** when prompted to save.

Securing Access to Select Customization Level

Use the following steps to secure access to *Select Customization Level* (91.260.00) for a user or group of users.

To secure access to Import:

1. Choose Utility | Access Rights.
2. Click the down arrow in **Type** and highlight either User or Group.
3. Type the group or user ID in **Group/User ID**. The name of the user or group displays in **Name**.
4. Type the company ID in **Company ID** or select All Companies to apply the access rights to all companies. If you select All Companies, <ALL> appears in the **Company ID** field.
5. Type `91.260.00` in **Screen/Report Number**. The **Type**, **Name**, and **Module** fields display information for *Select Customization Level* (91.260.00).
6. Disable access rights by clicking the down arrow and highlighting No for each of the following fields:
   - View
   - Update
   - Insert
   - Delete
   - Initialization Mode
7. Close *Access Rights Maintenance* (95.270.00) and click **Yes** when prompted to save.

Reference

Overview

The “Reference” section contains screens and dialog boxes that you use in the Customization Manager module. For each screen and dialog box, a description of every field and most buttons is included. Use the “Reference” section if you have a specific question about an element on a screen or in a dialog box. If you have a procedural question, consult the user portion of this manual.

Actions menu

Access Customization Manager functions using the **Actions** menu on the application toolbar. The **Actions** menu is available if Customization Manager is part of your Microsoft Dynamics SL installation, and if you have access rights to the module. See “Using the Customize Menu” on page 11 for information about using the **Actions** menu.

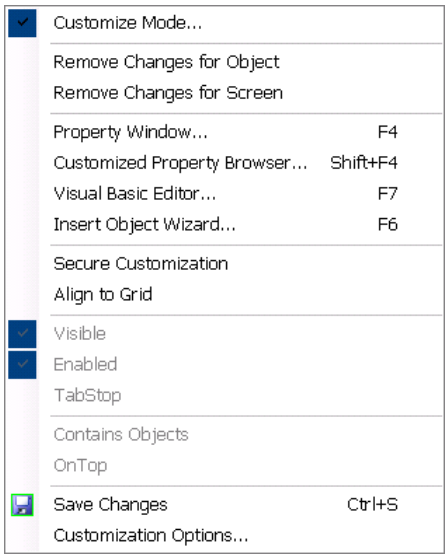

*Figure 21: **Actions** menu – Visual Basic for Applications*

The menu options are:

- **Customize Mode** — Toggle the application screen in and out of Customize mode.
- **Remove Changes for Object** — Remove all changes for the object that currently has focus. If the object is a new, it is deleted.
- **Remove Changes for Screen** — Remove all changes made to the current screen. This causes the screen to display in its standard form.
- **Property Window** — Customize properties of data objects on an application screen.
- **Customized Property Browser** — Display all customized properties of data objects on an application screen. Available if **VBA features** was selected during your Microsoft Dynamics SL installation.
- **Visual Basic Editor** — Open the code to be associated with an object. **Insert Object Wizard** — Insert new screen objects.
- **Secure Customization** — Secure all objects of a particular customization.
- **Align to Grid** — Force the selected object to be aligned to the nearest grid coordinate.
- **Visible, Enabled, TabStop** — Display the current state of the selected object.

- **Contains Objects** — Change the order of the selected container on a form. The selected container object will contain all underlying objects.
- **On Top** — Change the order of the selected container on a form. The objects contained by the selected object will be placed behind it.
- **Save Changes** — Saves customization changes for the application screen.
- **Customization Options** — Customization options window for grid alignment.

Customization Options

Enables you to make modifications to Customization environment settings for the current logged-in user.

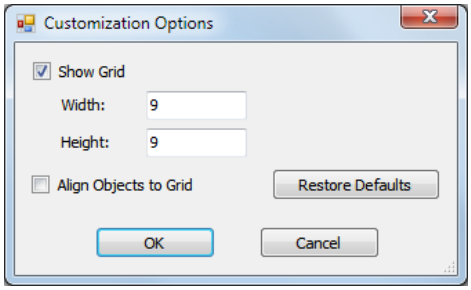

Figure 22: Customization Options

Show Grid

Determines whether to show the object alignment grid during customize mode.

Width

Sets the width of the grid used during customize mode.

Height

Set the height of the grid used during customize mode.

Align Objects to Grid

Forces alignment to the grid during customize mode.

Restore Defaults

Resets the customization options back to original settings.

Properties

Use *Properties* to customize the property values of data objects appearing in Microsoft Dynamics SL application screens.

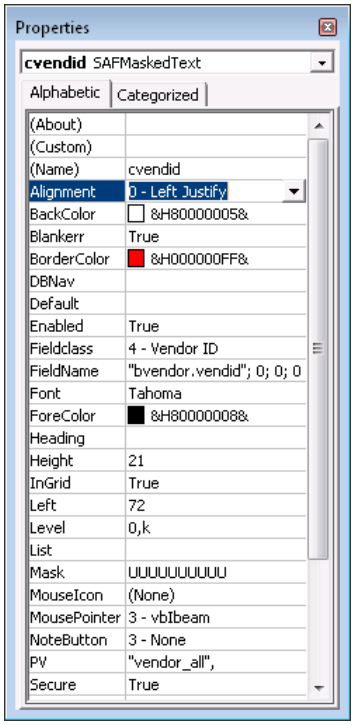

Figure 23: Properties

Selected Object

After selecting an object and then clicking **Property Window** on the **Customize** menu, *Properties* displays. The name of the object selected for customization appears in the drop-down list near the top of the screen.

Property

The first column identifies the properties of the data object.

Value

The second column allows you to enter customized property values. See the Customization Manager, Visual Basic for Applications online help or user guide for instruction regarding property values.

Customized Property Browser

Use *Customized Property Browser* to view the *Properties* window, which contains customized property values of data objects appearing in application screens.

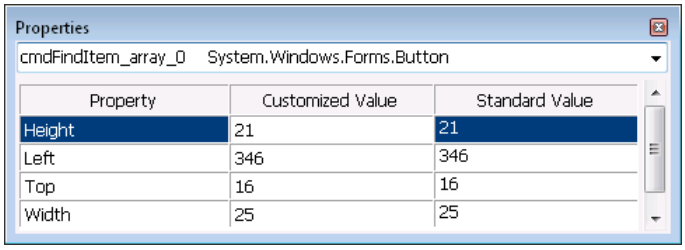

Figure 24: Properties window

Selected Object

After right-clicking on the customized application screen, select **Customized Property Browser** to display *Properties*. The name of the object selected for customization appears in drop-down list near the top of the screen.

Property

The first column identifies the properties of the data object.

Customized Value

The second column allows you to enter customized property values. See the Customization Manager, Visual Basic for Applications online help or user guide for instruction regarding property values.

Standard Value

The third column shows the default values for the data object. The software automatically inserts this value into the data object every time a default event occurs. See the Customization Manager, Visual Basic for Applications online help or user guide for instruction regarding property values.

- You can establish more complex default rules by entering Basic Script Language code. These are presented in more detail in the Customization Manager, Visual Basic for Applications online help or user guide.
- If the standard object already has a default, the one entered here takes precedence over the default.

Visual Basic Editor (VBA IDE)

Use *Visual Basic Editor* to enter Visual Basic for Applications code to be associated with an application screen object. Specific actions are performed on each object during runtime (finish, etc.). These actions are called events.

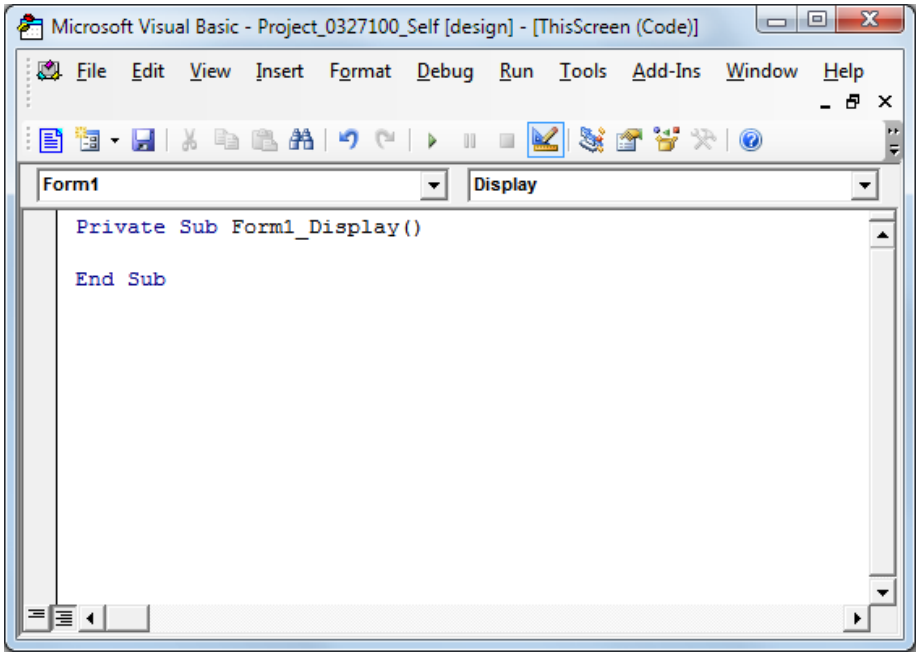

Figure 25: Microsoft Visual Basic Editor

To enter Visual Basic for Applications code for an object, click **Visual Basic Editor** on the **Customize** menu. The *Microsoft Visual Basic Editor* window appears.

Once you are in the *Microsoft Visual Basic Editor*, click the down-arrow in the **Event**, box to view a list of valid events for the object you selected. Then enter the event's VBA code in the text box. See "Modifying Code Using Visual Basic for Applications" on page 32 for more information on using the Visual Basic editor to customize an object.

Note: You can write general-use routines that can be called by other routines in the same screen by selecting the (general) event.

Event procedures must be entered using a specific format. A description of the appropriate format and the available statements and functions are found in the Customization Manager Reference Visual Basic for Applications online help or user guide.

When working with Visual Basic for Applications, it is important to understand the concept of events. Events are simply actions that occur at a point in time, such as when you click the mouse, when the form is loaded, when a record is saved, or when you press `TAB`.

For additional information about this window, see the Microsoft Visual Basic help.

You can add Visual Basic for Applications code to any of the following events:

Event Description	Event Occurrence (Before or After the Underlying Application's Event)
Button clicks	Before
Field default	After
Field chk	After
Field PV	Before application's PV dialog
Grid linegotfocus	After
Grid linechk	After
Form Load	After application's form load
Form Display	After application's form display
Form Hide	Before application closes
Finish	After
Insert	After application's New event
Cancel	After
Update	Before the application calls tranend
Delete	Before the application calls tranend

Insert Object Wizard (91.252.00)

Use *Insert Object Wizard* (91.252.00) to insert new screen objects. The first page displays the list of objects available through customization to be added to the application form. There are two types of objects available, Unbound and Data Bound. Unbound objects are not associated with data in the screen or database. Data-bound objects require a relationship bound to data elements (fields) in the database.

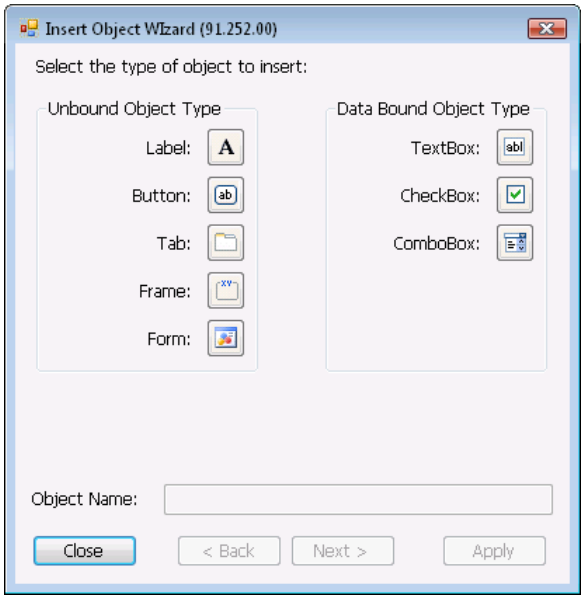

Figure 26: Insert Object Wizard (91.252.00), Type of Object

To select an unbound object, click on a choice in the **Unbound Object Type** grouping. To select a data-bound object, click on a choice in the **Data Bound Object Type** grouping. You can change the object name to any unused object name. Click **Apply** to add the object to the screen. For a data-bound object, click **Next >** to continue defining the data associated with the object. The following series of screens appear.

Insert Object Wizard, Table

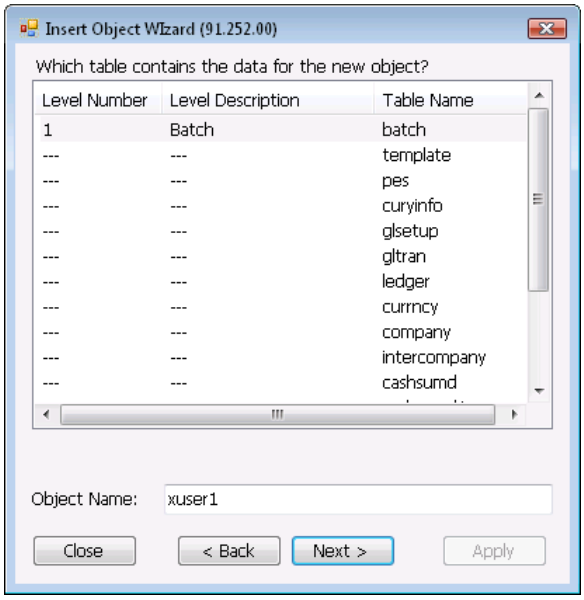

Figure 27: Insert Object Wizard (91.252.00), Table

This page is displayed when you select a data bound object type. It defines which database table contains data for the object to be inserted. When you finish filling in the fields, click **Next >** to continue associating data with the object.

Level Number

Level Number specifies the application screen hierarchical level to which the table belongs (for example, batch, document, or transaction detail).

Level Description

Level Description specifies the description of the level to which the table belongs.

Table Name

Table Name specifies the database table name from which the object to be inserted will be selected.

Insert Object Wizard, Level

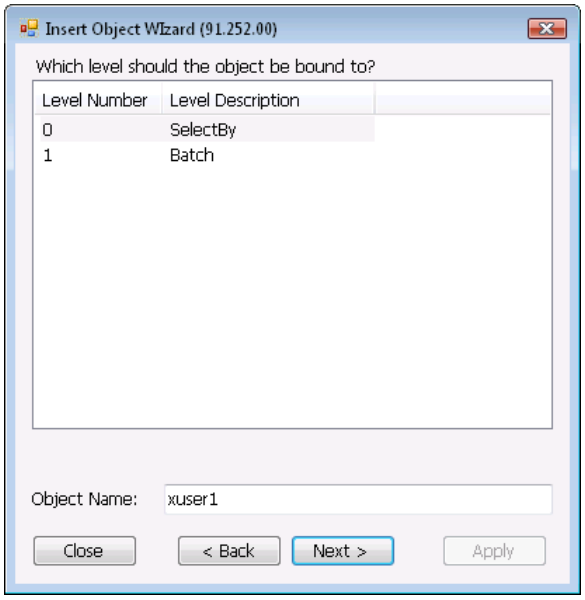

Figure 28: Insert Object Wizard (91.252.00), Level

This page displays if the table name selected does not have an associate level. When you finish filling in the fields, click **Next >** to continue associating data with the object.

Level Number

Level Number specifies the application screen hierarchical level to which the table belongs (for example, batch, document, or transaction detail).

Level Description

Level Description specifies the description of the level to which the table belongs.

Insert Object Wizard, Field

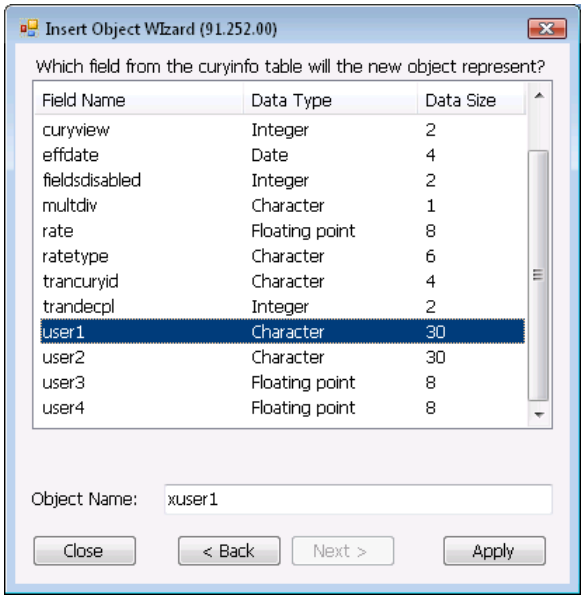

Figure 29: Insert Object Wizard (91.252.00), Field

This page defines the field from the table which the new object will represent. When you finish filling in fields, click **Apply** to add the object to the screen.

Field Name

Field Name specifies the database field that binds the object being inserted.

Data Type

Data Type specifies the type of information (character, logical, etc.) associated with the data object field name (see “Field Name”).

Data Size

Data Size specifies maximum default length, in characters, of the data object field (see “Field Name”).

Values List

Most objects on standard screens are limited as to the acceptable values that may be entered into the control. Use *Values List* to modify lists of standard object value restrictions or, when there is no standard list restriction, to add a list of your own.

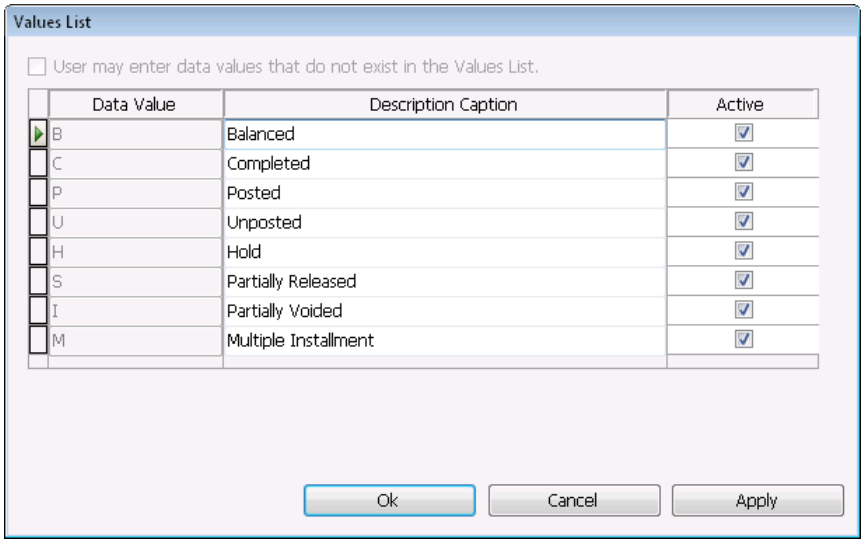

Figure 30: Values List

User may enter data values that do not exist in the Values List

When selected, this check box indicates that the software will accept values whether they are specified or not in the **Values List** for the object. Select this option if the entries in **Values List** are just a list of suggested values for the object (entries on the list can be added).

Data Value

Data Value specifies the object value as it is to appear on the application screen.

Description Caption

Description Caption is an explanation of the object value.

Active

Active specifies whether or not an object value is activated (appears in the **Values List** for its associated data object at runtime).

Select Customization Level (91.260.00)

Use *Select Customization Level* (91.260.00) to set up screen customizations that apply to users other than yourself.

Note: You must select the level at which the changes are applied in this screen before you load the application screen to be customized.

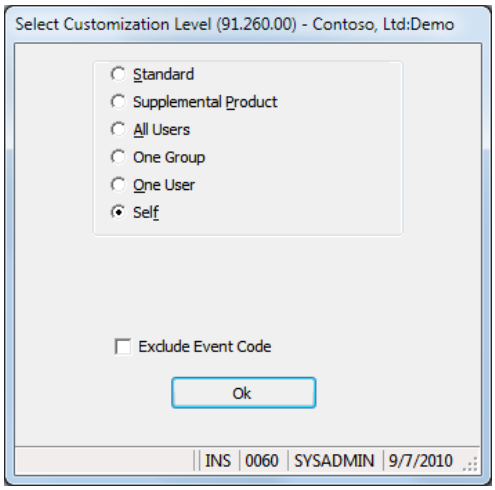

Figure 31: Select Customization Level (91.260.00)

Note: Only system administrators can use or assign access rights to *Select Customization Level* (91.260.00). Users without system administration rights are assumed to be working at the Self level when they are in customize mode.

Standard

Specifies that all screens should appear in their standard format with no customizations active. Since it can be difficult at runtime to tell which aspects of a screen are custom, the standard level can be useful in determining how the standard screen appears. If you select **Standard**, you cannot make changes to screens.

Supplemental Product

Specifies that "supplemental product" customizations should appear on screen. You can create and export customizations in a form that can be installed in other systems as a packaged customization. These customizations are installed in their own level and are referred to as supplemental products. The supplemental products view shows you how screens appear when only their supplemental product changes are active. If you select **Supplemental Product**, you cannot make changes to screens.

All Users

Specifies that each screen for which you enter customize mode should show the standard screen view (see "Standard" in this topic), any supplemental product changes applied to the screen (see "Supplemental Product" in this topic), and any changes that apply to all users in the current installation.

Note: Any changes made using this customization level are visible to all users.

One Group

Specifies that a customization group must be created in *Customization Groups* (91.270.00) to associate a customization group with the screen customization about to be completed. When you select this option and access a screen, the software displays the screen with any previously applied supplemental product and all-user changes (see "Supplemental Product" and "All Users" in this topic).

These changes, plus any additional changes you make, are then saved under the associated customization group ID.

Note: Any changes made using this customization level are visible to users that belong to the specified customization group.

One User

Specifies that a User ID must be entered at **User** to associate a user with the screen customizations about to be completed. When you select this option and access a screen, the software displays the screen with any previously applied supplemental product and all-user changes (see "Supplemental Product" and "All Users" in this topic). These changes plus any additional changes you make are then saved under the user ID entered at **User**.

When the specified user accesses a screen modified in this fashion, that user sees the custom view of the screen you have established for him or her. Any additional changes the user makes to the screen are saved separately and are applied in addition to your changes.

Self

Specifies that the screen changes about to be completed are to apply to the User ID making the changes only. When you select this option and access a screen, the software displays the screen with any previously applied supplemental product and all-user changes (see "Supplemental Product" and "All Users" in this topic).

User

The user to which a one-user customization applies (see "One User" in this topic).

Exclude Event Code

Controls whether or not you can enter customize mode for a customization without also executing the customization's associated event code. This option is useful when developing customizations because mistakes in the event code could prevent the screen from loading.

Customization Groups (91.270.00)

Customization groups can be added and modified in *Customization Groups* (91.270.00).

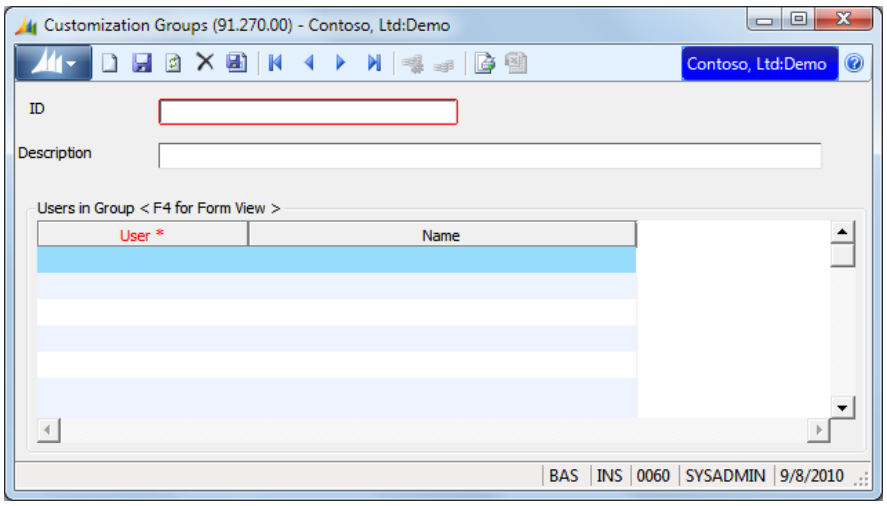

Figure 32: Customization Groups (91.270.00)

ID

An ID for a new or existing customization group.

Description

A short description of this group.

User

A valid user ID for the customization group currently displayed.

Name

The name associated with user ID selected in **User.**

Export Customizations (91.500.00)

Customizations residing in a database can be exported by selecting **Export Customizations** from the **Customization** menu.

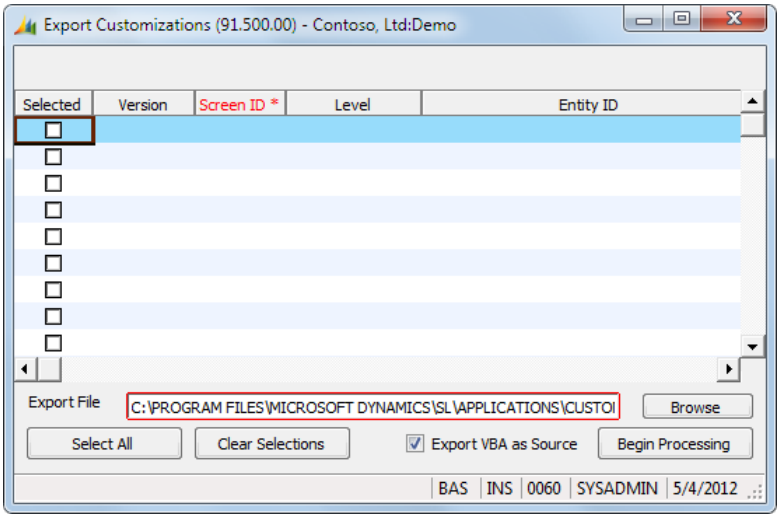

Figure 33: Export Customizations (91.500.00), grid view

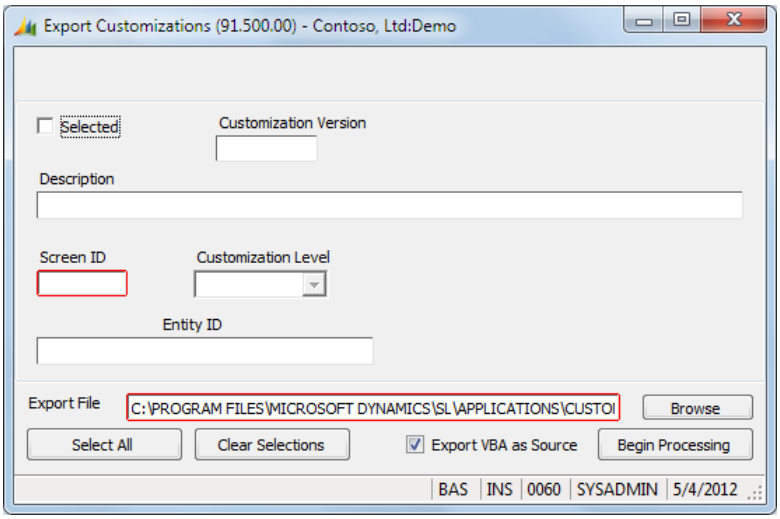

Figure 34: Export Customizations (91.500.00), form view

Selected

Controls whether the customization is exported when you click **Begin Processing**.

Customization Version

Specifies what version the customization was created in. The application screen for which the customization is designated will run versions 2.0 or later versions. Earlier versions are available for export only.

Description

Displays a brief explanation of the customization. At first, a simple description appears. However, the description can be changed in the file that is created by the export process. If a customization is imported, and the description is changed in that file, the file keeps that description, and it appears in the **Description** box when the file is exported.

Screen ID

Indicates the application to which the customization applies.

Example: If the customization applies to the Order Management module's *Sales Order Entry* (40.100.00), then 40100 appears in this field.

Customization Level

Because the software allows for various levels of customization to exist for any particular screen, the **Customization Level** specifies the precise customization to be exported.

Example: Two customizations could exist for the Order Management module's *Sales Order Entry* (40.100.00). One customization could be for All Users and the other could be for One User named THOMAS.

Entity ID

If the **Customization Level** is Self, then **Entity ID** contains the identification of the user who created the customization. If the **Customization Level** is One User, then **Entity ID** contains the identification of the user to which the customization applies. If the **Customization Level** is One Group, then **Entity ID** contains the identification of the customization group to which the customization applies.

Export File

All customizations selected to be processed are exported to the **Export File** in an ASCII file format. The name of the customization export file consists of a fully qualified path (including disk drive location) and file name. By convention, customization **Export Files** have an extension of CST. The default name in **Export File** is CUSTOM.CST. This is generated in the Microsoft Dynamics SL application program directory. To export customizations to a different drive or file name, enter the new file name.

Note: If the Export File already exists, its contents will be overwritten during the export process.

Browse

Allows for browsing of the directory structures to select a new path and file name for export.

Select All

Automatically selects all customizations regardless of Screen ID, Entity ID, or Customization Level.

Example: If you have 50 customizations and you want to export 45, click **Select All** and then click to clear the **Selected** check box for the five customizations that you do not want to export.

Clear Selections

Clears all customizations that were selected for the export process.

Export VBA as Source (check box)

Lets you export binary or source VBA customizations in a source format. Controls the ExportVbaSource setting in the solomon.ini file. Click to select this check box to set ExportVbaSource=Yes. Click to clear this check box to set ExportVbaSource=No. The default for a new database is cleared/No. For more information about this setting, see "Exporting Customizations to a File" on page 42 and "Appendix A: Solomon.ini Settings" in the System Manager Help or user's guide.

Begin Processing

Begins the export process. All customizations selected for export are written to the **Export File** in an ASCII file format. When the process is complete, you can decide to mark additional customizations for export, change the export file name, and press **Begin Processing** again.

Import Customizations (91.510.00)

Import Customizations (91.510.00) allows for customizations which have been exported to be installed on a different Microsoft Dynamics SL database, or to import to the same database with a different User ID. To access *Import Customizations* (91.510.00), select **Import Customizations** from the **Customization** menu.

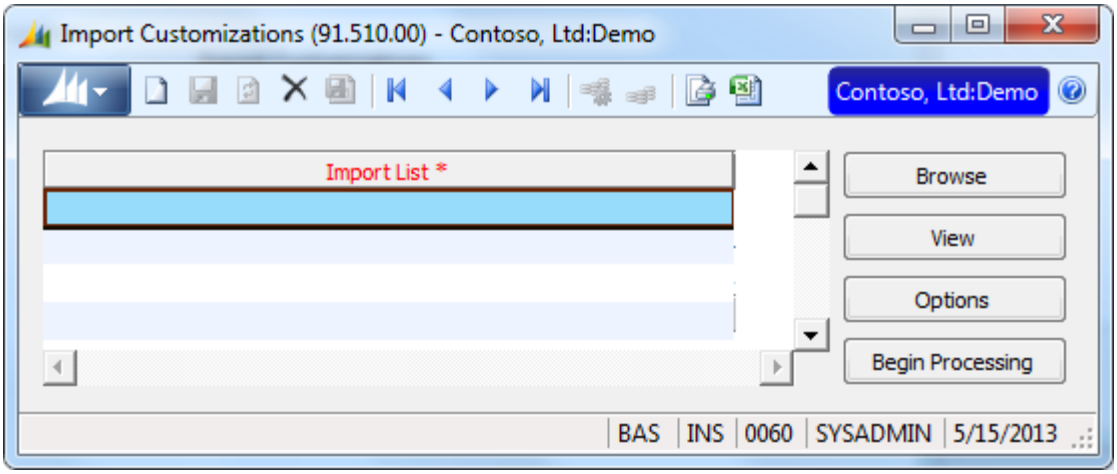

Figure 35: Import Customizations (91.510.00)

Browse

Click **Browse** to select the drive where the import file is located. Double-click on directory folders to navigate through the directory tree and locate the directory that contains the import file. By default, the system points to the Microsoft Dynamics SL Application program directory. Highlight the customization file(s) that you want to import, and then click **Open**. The files appear in the **Import List.**

Import List

A list of the customization files selected to be imported when processing begins. New files are added to this list by either typing in a fully qualified file name (such as A:\CUSTOM.CST), or by double-clicking a particular file while in the *Browse* screen. You can also enter the file name with the drive location into the **Import List**.

View

Displays the contents of the current file in the **Import List**.

Options

Displays *Import Options* (91.510.02), which allows you to specify two types of processing control options, **Conflict Resolution** and **Import Errors**. See “Import Options (91.510.02)” on page 80 for more information.

- Any user with access rights to *Import Customizations* (91.510.00) has the ability to import any customization for any screen at any customization level. You should carefully consider who you allow to access this powerful facility, particularly in a runtime environment as opposed to a development environment.
- In addition to importing new customizations, the **Import Customization** facility can be used to automatically delete existing customizations. See “Deleting Customizations Using Export and Import” on page 48 for more information.
- When you examine the contents of an **Import File**, notice that its composition is similar to a Visual Basic ASCII form file. The following diagram illustrates the fundamental structure of an **Import File**.

```
' Begin Customization …Various key fields identifying a unique
customization…
```

```
        Begin ControlType ControlName
                Customized Properties
        End
        Begin ControlType ControlName
                Customized Properties
        End
        Begin Macro Text
        Sub ProcedureName( )
        End Sub
' End Customization
' Begin Customization …Various key fields identifying a unique
customization…
' End Customization…
' Delete …Various key fields identifying a unique customization…
```

Begin Processing

When you click **Begin Processing**, the system begins importing the customizations contained within each file in the **Import List**.

Import Options (91.510.02)

Use *Import Options* (91.510.02) to specify customization import processing control options. This control option specifies how the software should resolve any import file conflicts and handle any import file errors encountered during the customization import process.

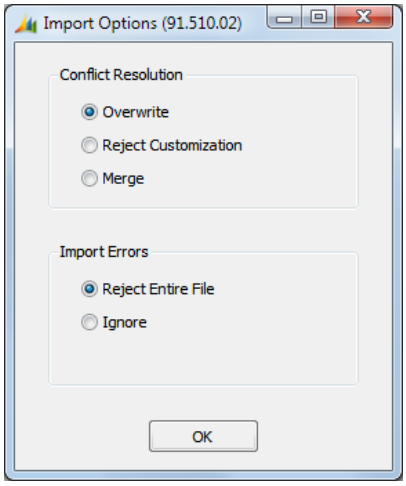

Figure 36: Import Options (91.510.02)

Conflict Resolution

While processing an **Import File**, the system may detect that a customization already exists in the position where a new customization is destined for storage.

Example: If a customization with the key field values Screen ID: 05260, Customization Level: One User, User ID: THOMAS and a blank Company Name exists in the database, and the Import File contains a customization with these same key field values, then an import conflict will occur.

If an import conflict occurs, there are several methods of **Conflict Resolution**:

- If you select **Overwrite**, the new customization overwrites the current customization.
- If you do not want to overwrite any existing customizations, then you can select **Reject Customization**. In the event of a conflict, this option causes the new customization to be rejected.
- The most sophisticated option allows you to **Merge** new customizations together with existing customizations. This is an extremely powerful feature that allows two customizations to be merged into one new customization down to the property level.

  Note: This option does not merge any VBA code customizations. It only merges the cosmetic changes and field properties. Any existing VBA code is removed and only the new VBA code in the file that you import is saved.

Example: If the position of ControlA has been customized so that its on-screen position varies from the standard screen and you want to import a customization that disables ControlA, the control will have both its screen position and enabled properties customized if the Merge option is used.

The only case in which a **Merge** cannot be successfully carried out by the system is in the case where a conflict occurs at the property level. In the example, if the new customization being imported also customizes the screen position of ControlA, then a conflict at the property level results. In such a case, the new customization takes precedence over the existing customization.

Import Errors

The **Import Errors** options allow you to control how the system should respond to errors that may occur during the import operation. An **Import Error** occurs anytime a syntax error is found to exist in the **Import File** currently being processed.

Example: Each customization in the Import File must begin with a line that starts with “Begin Customization” and goes on to specify other key values. If the ASCII text in the Import File is “Beginnn Customization,” then an Import Error results since the keyword Begin is misspelled.

The default option is **Reject Entire File**. This option causes the entire **Import File** to be rejected if any **Import Errors** occur while importing *any* of the customizations contained therein. Since an **Import File** can contain many customizations, this option allows you to easily set up an “all or nothing” type of import operation.

An alternate method of handling **Import Errors** is to **Ignore** them. This option should be used with caution. The usefulness of this option is primarily during development of sophisticated customizations containing advanced Visual Basic for Applications code. The ability to **Ignore** errors essentially provides the ability to compile the entire customization, including all Visual Basic for Applications code, and receive a list of all errors in the entire customization in a single attempt. You can subsequently fix all of the errors and import the customization again this time using the **Reject Entire File** method of handling **Import Errors**.

If an error actually occurs during the import process and you have chosen to **Ignore** errors, the resulting customization may not operate properly. If you receive *any* error during the import process, you should make appropriate corrections to the **Import File** and import the customization again before attempting to actually use the screen being customized.

Appendix A: Requirements for System Table Views Stored Procedures

Security measures require that existing and new stored procedures in the application database that reference system table views (such as VS_Company or VS_Acctsub) have additional special logic. The stored procedure will have permission to the view only if you add the following line to the stored procedure:

```
WITH EXECUTE AS '07718158D19D4f5f9D23B55DBF5DF1'
```

Without this line in a stored procedure that references one of the system table views, permission error 229 will occur.

Example:

```
CREATE Procedure RQ_AcctSub_Acct
   @CpnyID varchar(10),
   @UserID varchar(47),
   @Acct varchar(10)
WITH EXECUTE AS '07718158D19D4f5f9D23B55DBF5DF1'
as
   select *
   from   vs_AcctSub
   where  CpnyID = @CpnyID
   and    Acct in (Select Acct from RQUserAcct where UserID = @UserID)
   and    Acct like @Acct
   and    Active = 1
   order by Acct, Sub
```

Appendix B: Basic Script Language related content

Modifying Code Using BSL

Use Basic Script Language (BSL) code to set properties that are conditional upon other information on a Microsoft Dynamics SL screen.

Note: This option is available to sites that selected BSL Features when they installed Microsoft Dynamics SL.

Example: A company's operational policy mandates that only certain users within the organization are allowed to modify the credit limits for certain "key" customers. Customization Manager, by default, allows fields to be marked Disabled or Invisible for certain users. However, in the case of this particular example, certain users would be allowed access to credit limit changes for certain customers, but not others. To accomplish this, the chk() event for the Customer ID must contain BSL code that evaluates the factors that determine whether a particular user can indeed modify credit limit information. These factors could include any or all of the following; User ID, Customer ID, **Customer Class**, and **Credit Limit Amount**. Once the BSL code evaluates these factors, it would then enable or disable the credit limit information for the particular user.

To modify a property using BSL Code:

1. In the Microsoft Dynamics SL window, click **Administration | Select Customization Level**. *Select Customization Level* (91.260.00) appears.

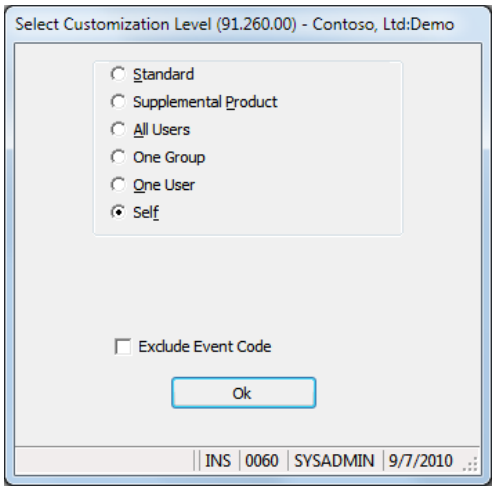

Figure 37: Select Customization Level (91.260.00)

2. Select the level of the customization to modify, and then click **OK**.

Note: You can verify customization level settings by going to **Tools | Options | Customization**.

3. Open the screen for which you want to add conditional logic.
4. Click **Customize Mode** on the **Actions** menu.
5. Click on the object for which you want to add conditional logic.

6. Open the **Customize** menu and click **Code Window**. *Code Window* appears.

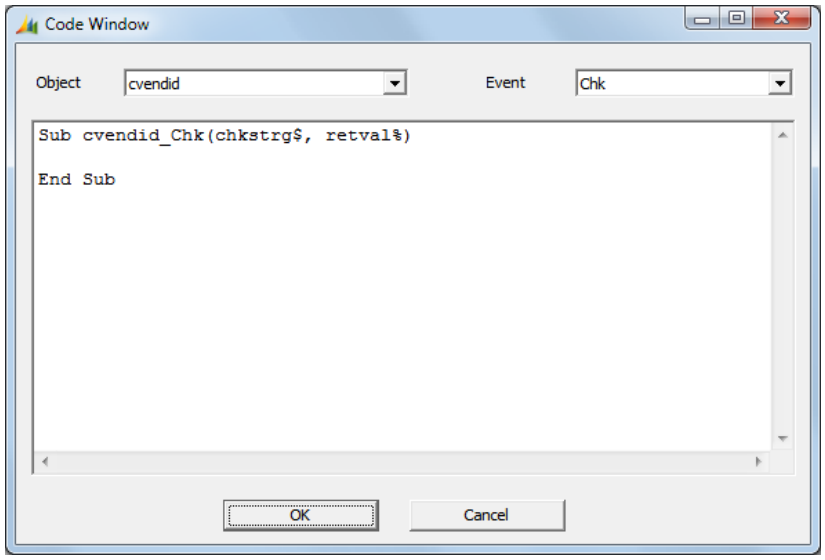

Figure 38: Code Window

7. In the **Event** box, select the event for which you want to add code.
8. In the text box, type the event code for the selected object.

   Note: For detailed information on appropriate syntax and a list of available functions and statements, see the Customization Manager, Visual Basic for Applications online help or user guide.

9. Close *Code Window*.

Modifying Database Values Using BSL Code

Use BSL code to set properties that are conditional upon other information on a Microsoft Dynamics SL screen. In the example that follows, the Journal Entry screen is customized by using BSL code. The code fetches the account description and stores the description in the User1 field in the GLTran table.

Example:

```
'Begin Customization
Screen: 01010
Sequence: 300
Userld: "CompanyName:"
Description: "Customization"
Languageld: "enu"
System Administrator: TRUE

'Begin Macro Text
$include: "bsl.dh"
$include: "account.dh"
$include: "gltran dh"
Dim xCursor as Integer
Declare Subcacct_Chk(chkstrg$, retval%)

Sub cacct_Chkstrg$, retval%

       Dim SqlStr as String
       Dim xDescr as String
       Dim xRetVal as Integer

   'Store the Account description in the Journal Transaction Record.
   SqlStr = 'Select * from account where acct = "& chkstrg
   XRetVal = SqlFetch 1 (xCursor, SqlStr, Account, Len(bAccount))
   Call GetBufferValue(bAccount.Descr, xDescr)
   'Save Account in User 1 field for later reference.
   Call SetBufferValue("bGLTran.User1" , xDescr)
End Sub
'End Customization
```

Code Window

Use *Code Window* to enter Basic Script Language (BSL) code to be associated with an object. Specific actions are performed on each object during runtime (finish, etc.). These actions are called “events.”

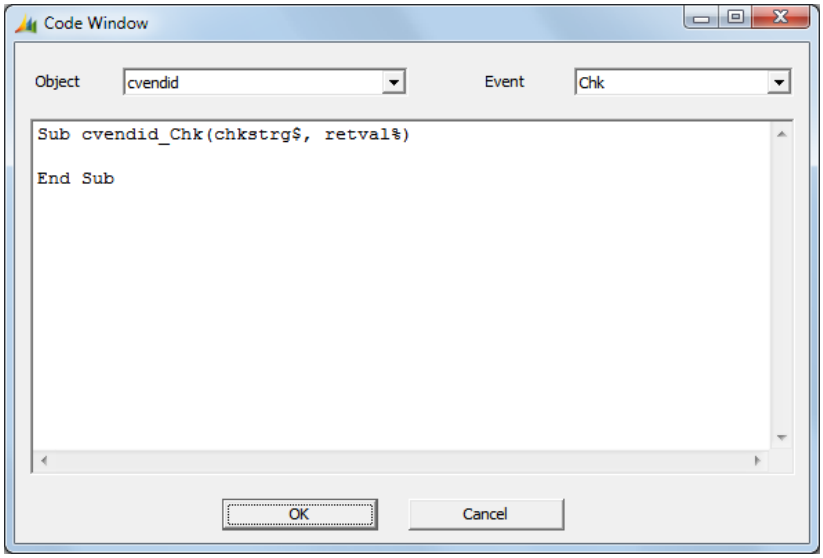

Figure 39: Code Window

To enter BSL code for an object, select **Code Window** from the **Customize** menu.

Once in *Code Window*, select the object event for which you are entering BSL code. Then, enter the event’s BSL code in the text box below **Object** and **Event**.

Note: You can write general-use routines that can be called by other routines in the same screen by selecting the (general) event.

Event procedures must be entered using a specific format. A description of the appropriate format and the available statements and functions are found in Customization Manager, Basic Script Language online help or user guide.

When working with the BSL, it is important to understand the concept of “events.” Events are simply actions that occur at a point in time: when you click the mouse, when the form is loaded, when a record is saved, when you press `TAB`, etc.

You can add BSL code to any of the following events:

Event Description	Event Occurrence (Before or After the Underlying Application's Event)
Button clicks	• Before
• Field default	• After
• Field chk	• After
• Field PV	• Before application's PV dialog
• Grid linegotfocus	• After
• Grid linechk	• After
• Form Load	• After application's form load
• Form Display	• After application's form display
• Form Hide	• Before application closes
• Finish	• After
• Insert	• After application's New event
• Cancel	• After
• Update	• Before the application calls tranend
• Delete	• Before the application calls tranend

Object

The application screen object for which you are entering BSL code.

Event

The object event for which you are entering BSL code. The down arrow displays a list of valid events for the selected object.

Glossary of Terms

Access Rights

A set of permissions that define the parts of the system an authorized user may access as well as the user's rights to alter any information.

Active User

A user who is currently signed on to Microsoft Dynamics SL.

Application

One of the accounting modules, such as Accounts Payable or Accounts Receivable.

Application Database

A database that contains all of the accounting data for an individual financial entity.

Batch

A grouping of documents. The batch fields on a screen apply to all listed documents.

Bound Controls (or Data-Bound Objects)

Controls that have database field associations such that their value is the database field value.

Text

The identifying text used to label objects outside of grids. Normally, an object has a caption or a heading, and some objects have both.

Conflict Resolution

Used when processing any particular **Import File**. The system may detect that a customization already exists in the position where a new customization is destined for storage.

Current Setting

The current property information of the object. This function is useful information when an event has changed any of these properties at runtime.

Custom Setting

Specifies changes actually made to some or all of the object properties.

Customization

The collection of changes and additions made to a screen or to an object on a screen that are applied to an application screen at runtime.

Customization Levels

Used to define customizations and specify to which user the customization is available. The software places users in groups called "levels." Using levels, you can define a customization once and have it apply to all users, to a specific user, or to yourself so that you can support your company's business rules.

Customization Group

A collection of Microsoft Dynamics SL users for whom certain customizations are available. These groups are not like user groups in Microsoft Dynamics SL. A Customization Group has no access rights associated with it. Customization Groups offer a way to tailor customizations to the needs of a specific group of users.

Customize Mode

Used to make custom changes to objects for an application screen.

Default

The default value for the data object. The software automatically inserts this value into the data object every time a default event occurs. A default event occurs whenever you click **New** or whenever you change a field on which the current field depends for its information.

Default Values

Use the **Default** property to automatically populate a field with information. Default values are useful for fields whose value seldom changes.

Document

An individual record in a batch. You can display document fields on a screen in either grid or form view.

Edit Masks

Used to automatically format text entered in fields on a Microsoft Dynamics SL screen and to validate that information entered in a text field is of the correct data type.

Edit Type

Lists pre-defined edits for text fields. You can select one of the pre-defined options or select Custom to enter your own.

Events

Events are simply actions that occur at a point in time, such as when you click the mouse, when the form is loaded, when a record is saved, when you press `TAB`, etc.

Exporting Customizations

Enables you to export customizations residing in a Microsoft Dynamics SL database. When you select **Export Customizations**, the software opens *Export Customizations* (91.500.00).

Field Level

For a data object, **Field Level** displays the level of the current field. If the field is a key field for the specified level, {Key Field} is indicated.

Field Name

For a data object, **Field Name** displays the field name of the object in the SQL database table. The SQL table name is prefixed before the field name to uniquely identify it in the SQL dictionary. A “b” prefix is also provided to identify the field as a buffer when the field is addressed in Visual Basic for Applications code.

Fixed Possible Values List

Only used with standard Microsoft Dynamics SL objects. Operators may not add to or delete from this type of list, but individual items may be made invisible to users by selecting the **Inactive** property.

Grid View

A portion of a data entry screen showing all the data items for all the records in rows and columns. The grid view is useful when comparing values in detail records. Toggle between grid and form view by pressing `F4`. When in Customization mode, pressing F4 will not switch between form view and grid view.

Group

A collection of users who share the same access rights to an application screen.

Heading

The identifying text used to label data objects displayed in grid columns. Normally, an object has a caption or a heading.

Hot Keys

Provide users with quick access to applications without the use of the mouse, toolbars, or popup screens.

Index

The index number of a control array if the current object is an element. A control array is comprised of a group of objects of the same type that share a common object name and a set of event procedures. Each object of a control array has a unique index number.

Insert Object Wizard

Used to add objects to a screen to support additional business functions, or to add an object to a screen to spare the user from bringing up a different screen just to look at or enter data in one field.

Min/Max

Restricts the range of values that the software accepts in a numeric or date field. If the standard screen has a minimum or maximum on the current field already, you can enter a minimum and/or maximum value that is more restrictive.

Object (or Control)

The underlying control on the Application form. All selectable objects are Visual Basic controls (text boxes, push buttons, labels, etc.)

Possible Values Lists

Use the list property to define valid entries for a particular data object.

Possible Values

Values that appear when you press <F3> or double-click in a field. Files containing the lists of possible values are imported into the software using the **Possible Values Import** function on the Utility menu.

Property

You can modify the properties of an object on a screen that you have displayed in customize mode to change how it looks and behaves. Properties include:

- Color codes of an object
- Hiding an object from certain users
- Clearing an object's tab stop
- Setting an object's **Display Only** attribute

Select Customization Level

Enables the Microsoft Dynamics SL system administrator to make screen customizations that apply to other Microsoft Dynamics SL users.

SQL

Structured Query Language.

Standard

Specifies that all screens should appear on screen in their standard format with no customizations active.

Supplemental Product

Specifies that "supplemental product" customizations should appear on screen. You can create and export customizations in a form that can be installed in other systems as a packaged customization. These customizations are installed in their own level and are referred to as supplemental products.

System Database

The database that contains shared site-specific data, such as database field attributes, edit characteristics, product registrations, customizations, and screen-level security.

Tab Area on Tab Control

The area where the captions reside for the tab control as defined by TabHeight and TabOrientation properties. This is the portion of the tab control that is visible in every selected tab page.

Tab Order

The order in which the cursor moves through the objects when the user presses the `TAB` key.

Template

A set of options, such as the printing and sorting options of a report, that is saved for later use.

User Specified Possible Values List

Reserved for use with text fields that do not have a standard values list. If the control is a text box, the values are displayed in a possible value (PV) window. If the control is a combo box, the values are displayed in the drop-down list for the combo box.

Values

Used to establish the list of values that are permitted to be entered for standard Microsoft Dynamics SL objects.

Index

R

S

T

U

V