CA Gen

Web View User Guide

Release 8.5

This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, if you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

CA Technologies Product References

This document references the following CA products:

- CA Gen

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 4: Generating Applications 33

Chapter 5: Deploying the Web View Application 55

Appendix A: Windows GUI Specific Features 61

Appendix B: Application Skins 69

Chapter 1: Introduction

This section contains the following topics:

Overview

This guide explains how to use the Web View UI Generation perspective in CA Gen Studio to deliver a dynamic web user interface that accesses CA Gen business logic. This guide describes the design, generation, and building an application by providing the following information:

- Prerequisites for installing the software
- Technical requirements for Web View UI Generation
- How to set up the required environments
- How to generate the application
- How to deploy the generated application

Audience

The document is intended for Web developers and CA Gen developers who create and deploy content created by Web View to an Application Server. This guide assumes that you are familiar with the application generation methods in CA Gen.

Web View UI Generation

The Web View UI Generation perspective in CA Gen Studio generates HTML pages from an existing model. Use this perspective to open existing models in CA Gen Studio, generate HTML, CSS, JavaScript, and other necessary files for creating a Web View application.

Note: For information about other perspectives in CA Gen Studio—Web Service Access Designer and PStep Interface Designer, see the *CA Gen Studio Overview Guide* or *CA Gen Studio Help*.

Web View Environments

You must install CA Gen Studio as instructed from the CA Gen download folder.

Note: For more information about CA Gen Studio installation, see the *CA Gen Studio Overview Guide* or the *CA Gen Studio Help*.

There are four environments involved in Web View UI Generation that can be located on one machine or four physically separate machines. The following environments are involved in Web View UI Generation:

- Server Environment—The CA Gen runtime is installed for generated server applications. No special setup is required for Web View applications.
- Application Server Environment—The runtime engine for the client is installed on an application server, which enables you to run the generated Web View application on a browser. The application servers must conform to Java EE standard.

  Note: For more information about the supported Java EE version, see the *Technical Requirements* document.
- Development Environment—The Toolset and the CA Gen Studio framework provide the environment to develop and generate CA Gen Web View applications. This environment contains the components that generate the application both locally and remotely.
- Build Environment—This environment enables the generated files from the CA Gen Development Environment to be compiled and packaged into a Web Application Archive (WAR) inside of an Enterprise Application Archive (EAR). The CA Gen Build Environment is often located on the same machine as the CA Gen Development Environment.

The generated product includes the following highlights:

- The application logic resides on the application server in an EAR/WAR file.
- LM.html and TC.html files serve as entry points to generated applications.
- The Web View application is a combination of HTML, JavaScript, and Cascading Style Sheets that are downloaded to the browser.
- Permitted Value and Edit Pattern checking takes place on the browser using JavaScript Edit Pattern checking, and it also takes place on the application server.
- The product enables existing users to reuse their models by generating Web View applications.
- The Web View software can access databases by using CA Gen Server applications or utilizing JDBC on the Application Server environment.

- The Web View application interface can be available on any platform with a supported Web browser.

  Note: For more information, see the *Technical Requirements* document.

- Due to browser and HTML limitations, some CA Gen related features have a different level of support when generated for Web View.

Set Up the Development Environment

To develop a Web View application, you must set up your environment to generate the Web View application. The development environment for Web View requires the CA Gen Toolset and the CA Gen Studio.

Set Up the Build Environment

The Build Tool has tokens that must be set to allow the generated applications to build on the implementation machine. You can set these using the Profile Manager in the Build Tool. Using the Build Tool EAR File Assemble Details dialog, you can assemble an EAR file containing your application in preparation for deployment to the Application Server.

Note: For more information, see the *Build Tool User Guide*.

Set Up the Web View Application Server Environment

You can deploy Web View applications to the following Application Servers:

- WebSphere
- WebLogic
- JBoss

Install the required Application Server to deploy a Web View application.

Set Up the End-User Environment

The Web View application end users can access the application using a supported Web browser. No special setup is necessary.

Chapter 2: Pre-Installation Planning

This chapter provides the infrastructure requirements you must know. The four different environments involved in this feature can be located on one machine or four physically separate machines. The end-user machine must have a supported version of a Web browser.

Note: For more information, see the *Technical Requirements* document.

Microsoft Internet Explorer has a limitation that ignores any CSS selector beyond the 4096th one in a single file and a maximum number of CSS files linked to one HTML page. The Web View assembly process removes empty CSS styles to reduce the chance of reaching the selector limitation. If some controls on the page lose the layout or style, it may be necessary to reduce the number of customizations, such as video property and controls on the page.

This section contains the following topics:

Technical Requirements

Third party software and CA Gen software each have requirements in addition to the CA Gen requirements as shown in the following table:

Environment	Third Party Software	CA Gen Software
CA Gen Server	No change required for Web View applications.	Installed CA Gen cooperative server.
Application Server Machine	The application servers must conform to the Java EE standards. **	Web View Runtime*. Communications Runtime**
Development Client	Windows Operating System **	Workstation Development Toolset for designing and generation of models and CA Gen Studio for generation.
Build Machine	Windows Operating System ** Java Platform, Standard Edition (Java SE)**	CA Gen Build Tool
End User	Web Browser**	

Note:

- * Only needed if not packaged in the application EAR file.
- ** See *Technical Requirements* available on http://ca.com/support for details.

Web Applications Compared to Windows GUI Applications

Due to differences between Windows GUI applications and web applications, some feature differences occur when generated using CA Gen for Web View.

A model developed for Windows GUI may contain elements that are specific to Windows and are not available in Web View.

Some features behave differently in the web environment. In particular, there is no guarantee that events in Web View applications occur in the same order as they occur in Windows GUI applications. If a model relies on the order of events, you need to modify the model for it to behave as expected.

Browser Cache

Browser caching does not retain the state of a Web View application since the application is dynamically updated while it is running. When the user attempts to leave the application, a warning message will be displayed to inform them that the application will be restarted when they return. They may choose whether they wish to stay in the application or not. This will happen regardless of how the user leaves the application including using the browser Back and Forward buttons.

This feature may be turned off by changing the notifyOnUnload setting in the configuration.js for the skin. This file is located in the gen\skins\xp directory in the default Web View skin.

More information:

Override Skins Using the Injection Technique (see page 71)

Chapter 3: Using Existing Models in Web View

This section contains the following topics:

Using Existing Models

You can open existing models in CA Gen Studio and then generate HTML and other required files for a Web View application.

Controls that are customized in the HTML mode retain their customizations and you can generate them accordingly in Web View. Controls that you added in the HTML mode are retained and you can generate them in Web View.

Note: For more information about customizations, see the *Toolset Help*.

Note: Designs made using the CA Gen Studio tools are not backward compatible with Windows, Web Generation, or ASP .NET Web Clients interface.

User Interface Controls

The following list shows the supported user interface controls:

- Windows
- Dialogs
- Menus and menu items
- Hidden fields
- Foreground and Background colors
- Fixed size tables
- Tool bar
- Status bar
- Entry fields
- Push buttons
- Images on push buttons
- List boxes
- Background images in windows and dialogs
- System menu to close primary windows and dialogs
- Message boxes
- Edit patterns
- Tab ordering for fields
- Multi-state bitmaps
- Picture controls on the window
- Check boxes
- Radio buttons
- Combo boxes
- Multiline entry text boxes
- Modeless dialogs
- Modal dialogs

Action Diagram Code Elements

The following list shows the supported Action Diagram code elements:

- Activate and Deactivate events
- ScrollTop and ScrollBottom events
- Math, string, and file functions
- Database statements (Add, Delete, Update, Display, Change, and so on)
- Exit states
- Command statements
- GUI statements
- User-defined action blocks
- External Action Blocks (EABs)

HTML Control and HTML Text

The contents of HTML Control and HTML Text are not compatible between Web View and other CA Gen web products.

The contents must be XHTML compliant. You may verify XHTML compliance using a validator such as http://validator.w3.org/. Place the content that is to be validated in the <body> section of an XHTML document as shown next:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
    <html xmlns="http://www.w3.org/1999/xhtml">
        <head>
          <title>Content To Validate </title>
          <meta http-equiv="X-UA-Compatible" content="IE=8" />
          <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
        </head>
     <body>
        <!–Place the content to be validated inside the body section -->
     </body>
    </html>
```

Do not include CDATA sections in the XHTML document. Even though CDATA sections are valid in an XHTML document, they are not allowed in a Web View HTML Control or HTML Text element.

Special characters must be escaped using the &; notation. The most common character that is likely to be found in user code is the '&' character which is escaped as '&amp;'. Proper validation will help identify special characters in the content.

Minimize the usage of <style> tags as Internet Explorer has limitations on the number of CSS selectors that can be included in a document. Web View uses a technique that stacks opened windows, which in a single HTML document causes a large number of CSS selectors to be loaded.

If the custom content in an HTML control needs to be referenced using the ID, you need to wrap the content inside its own div with the appropriate ID. The Web View controls do not use the same ID naming conventions as Web Generation. If you intend to use the model only for Web View, you can use the className property, which matches the controls name instead of using the ID.

Each HTML Control and HTML Text element is rendered in a separate <iframe> tag in the HTML document. References between the contents of different controls is done by first referencing the <iframe> and then referencing the content relative to the <iframe>.

Example:

An HTML Text element named "HTMLText1" contains the following JavaScript:

```
<script> var mydate=new Date(); </script>
```

From another HTML Control or HTML Text element, the variable "mydate" would be referenced using the following JavaScript syntax:

```
top.document.querySelector(".HTMLText1").contentWindow.mydate
```

As an alternative to specifying HTML content that is inserted into either the <head> or <body> tags of the <iframes> HTML page, an entire document may be included. If the generator reads that the content begins with <!DOCTYPE> statement, the entire content is loaded into the <iframe>, as is, instead of being inserted into a pre-defined HTML page. No validation is done at generation time to ensure that the document is valid. It is the CA Gen developers responsibility to supply valid content.

MAKE Support

The MAKE statement's functionality behaves differently in a few cases for Mozilla Firefox and Internet Explorer. Note that MAKE support may not be effective when dealing with imported HTML pages that were created using third-party web authoring tools external to CA Gen.

MAKE in Internet Explorer

The following table illustrates CA Gen's support for MAKE in Internet Explorer:

Control	Color	Highlight Underline	Highlight Reverse	Intensity Dark	Error	Cursor	Protect
Single-line edit field	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Multi-line edit field	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Check Box	No	No	Yes **	No	Yes **	Yes	Yes
Radio Button	No	No	Yes **	No	Yes **	No	Yes
Drop down List	Yes ***	Yes	Yes	Yes	Yes	Yes	Yes
List box (implemented with HTML tables and links)	Yes	Yes	Yes	Yes	Yes	No	Yes *

Note:

- ** Color changes are on the controls, not on their prompt.
- *** Color changes are on the selected/displayed item.

MAKE in Mozilla Firefox

The following table illustrates CA Gen's support for MAKE in Mozilla Firefox:

Control	Color	Highlight Underline	Highlight Reverse	Intensity Dark	Error	Cursor	Protect
Single-line edit field	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Multi-line edit field	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Check Box	No	No	Yes	No	Yes	Yes	Yes
Radio Button	No	No	Yes	No	Yes	Yes	Yes
Drop down List	Yes ***	Yes	Yes	Yes	Yes	Yes	Yes
List box (implemented with HTML tables and links)	Yes	Yes	Yes	Yes	Yes	No	Yes *

Note:

- * Protected Fields in a List Box are placed within <P> and </P> tags. As a browser feature, when you click on such protected text, the browser will not highlight it. However, if you double-click, Ctl+click or Shift+click, Internet Explorer will highlight the text while Mozilla Firefox will highlight the border of the cell. Events associated with protected rows will not be fired.
- *** Color changes are on the selected/displayed item.

User Exits

A user exit is modifiable source code that you can customize to fit your specific needs. The user exits can be as simple or as complex as you require. The source modules as delivered are referred to as default exits because the logic they contain is executed if no modifications are made. The source for the default exits are in the CA Gen installation directory.

CA Gen generated Web View applications invoke a number of CA Gen supplied routines at execution time to perform various functions. The userOnLoad user exit is supported in Web View applications. This is the only user exit that executes on a browser.

Note: For more information about Web View user exits, see the *User Exit Reference Guide*.

Support for Tabbed Browsing

CA Gen supports tabbed browsing in Web View applications. Tabbed browsing lets you concurrently open, access, and update a Web View application from different tabs of a single browser instance.

A tabbed session is comparable to an individual browser instance of the Web View application. CA Gen independently manages the session states of an application running under different tabs in a Browser Window.

Note: Tabbed browsing support requires a minimum of Internet Explorer 7.x and Mozilla Firefox 2.x.

Support for List Box Selection

The functionality of multiple-selection list boxes and extended-selection list boxes is identical in Web View applications. Use the following keys to select more than one row:

- CTRL+CLICK to select rows non-consecutively
- SHIFT+CLICK to select rows consecutively

Fixed and Varying Size Tables

Resize the Fixed and Varying Size Tables in existing models to display the scroll bar. Fixed Size Tables must be smaller than the actual size and Varying Size Tables must be larger by a couple of rows.

Tab Order

In Web View applications, the Help and Close menu items, and the tool bar buttons are placed at the end of the tab order. The Help button is placed before the Close button in the tab order.

Controls in Dot Notation Syntax

The following controls are available in Dot Notation syntax:

- Application
- Button
- Check Box
- Dialog
- Drop-down List (Selection)
- Drop-down List Prompt
- Group Box
- List Box
- List Column Item
- List Row Item
- Literal

- Menu
- Multiline Edit Field Prompt
- Multiline Edit Field
- Permitted Value List
- Picture
- Radio Button Group
- Radio Button Group (Properties)
- Singleline Edit Field Prompt
- Singleline Edit Field
- Status Bar
- Tool Bar
- Window

The following table lists controls and the corresponding properties that are supported in Microsoft Internet Explorer and Mozilla Firefox:

Legend:

- ✓ denotes that the property is provided for the control
- Empty cell denotes that the property is not provided

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
BackgroundColor	Get				✓	✓		✓	✓			✓			✓					✓		✓	✓	✓		✓
	Set				✓	✓		✓	✓			✓			✓					✓		✓	✓	✓		✓
BitmapBackground	Get		✓																							✓

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
	Set																									
BitmapDisplayMode	Get																✓									✓
	Set																									✓
BitmapName	Get																✓									
	Set																									
Caption	Get		✓	✓	✓			✓										✓	✓	✓						✓
	Set		✓	✓	✓			✓										✓	✓	✓						✓
Check	Get												✓													
	Set												✓													
ColumnCount	Get																									
	Set																									
ColumnType	Get									✓																
	Set																									
CurrentCount	Get					✓			✓																	
	Set																									
DisplayAtTop	Get										✓															
	Set										✓															

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
EditedValue	Get									✓					✓							✓				
	Set														✓							✓				
EditPattern	Get																									
	Set																									
Enabled	Get		✓	✓		✓			✓				✓		✓			✓				✓				
	Set		✓	✓		✓			✓				✓		✓			✓				✓				
Focus	Get		✓	✓		✓			✓						✓			✓				✓				✓
	Set		✓	✓		✓			✓						✓			✓				✓				✓
FontSize	Get		✓	✓		✓	✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
	Set		✓	✓	✓		✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
FontStyle	Get		✓	✓		✓	✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
	Set		✓	✓		✓	✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
FontType	Get		✓	✓		✓	✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
	Set		✓	✓		✓	✓	✓	✓			✓		✓	✓			✓	✓	✓	✓	✓				
ForegroundColor	Get						✓	✓	✓			✓			✓					✓	✓					
	Set						✓	✓	✓			✓			✓					✓	✓					
FullName	Get	✓																								
	Set																									
Handle*	Get		✓	✓		✓		✓				✓			✓		✓	✓		✓		✓	✓	✓		✓
	Set																									

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
Height	Get		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓		✓		
	Set		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓		✓		
Left	Get		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓		✓		
	Set		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓		✓		
Maximized*	Get																									✓
	Set																									
Minimized*	Get																									✓
	Set																									
Multiline	Get														✓							✓				
	Set																									
Name	Get		✓	✓	✓	✓		✓	✓	✓		✓	✓		✓		✓			✓		✓				✓
	Set																									
ObjectType	Get	✓	✓	✓		✓		✓	✓	✓	✓	✓	✓		✓		✓	✓		✓		✓	✓	✓		
	Set																									
Parent	Get	✓	✓	✓				✓	✓	✓	✓	✓	✓		✓		✓	✓		✓		✓	✓	✓		✓
	Set																									
Prompt	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
Prompt Height	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
PromptBackgroundColor	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
PromptFontSize	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
PromptFontStyle	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
PromptFontType	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
PromptForegroundColor	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				
PromptLeft	Get					✓									✓							✓				
	Set					✓									✓							✓				
PromptTop	Get					✓									✓							✓				
	Set					✓									✓							✓				
PromptWidth	Get					✓				✓					✓							✓				
	Set					✓				✓					✓							✓				

		Application	Button	Check Box	Dialog	Dropdown List (Selection)	Dropdown List Prompt	Group Box	List Box	List Column Item	List Row Item	Literal	Menu	Multiline Edit Field Prompt	Multiline Edit Field	Permitted Value List	Picture	Radio Button	Radio Button Group (Methods)	Radio Button Group (Properties)	Singleline Edit Field Prompt	Singleline Edit Field	Status Bar	Tool Bar	Tool Bar Controls	Window
ReadOnly	Get														✓							✓				
	Set																									
Selected	Get			✓							✓															
	Set																									
Top	Get		✓	✓	✓	✓			✓			✓			✓		✓	✓	✓	✓		✓				
	Set		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓				
Value	Get			✓						✓		✓			✓			✓		✓		✓				
	Set											✓								✓						
Visible	Get		✓	✓	✓	✓		✓	✓		✓	✓			✓		✓	✓	✓	✓		✓	✓	✓		✓
	Set		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓	✓	✓		✓
Width	Get		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓				
	Set		✓	✓	✓	✓		✓	✓			✓			✓		✓	✓	✓	✓		✓				

Note: The following notes apply to the corresponding properties marked with * (asterisk) in the previous table:

- The Maximized property always returns a TRUE value and the Minimized property always returns a FALSE value.
- The controls on the tool bar are not buttons and you cannot access the tool bar controls in the same way as you can access buttons.
- Radio Button Group (Methods) are not supported.

- The following list details the specific font types, styles, and sizes that are supported:
  - Font Type
    - Century Gothic
    - Times New Roman
    - Sans Serif
    - Wingdings
    - Courier
  - Font Style
    - Normal
    - Italic
    - Oblique
  - Font Size
    - 8
    - 12
    - 24
- In Bitmap properties, the Scaled BitmapDisplayMode property is not supported completely in Web View applications as it does not scale to the size of the window. The Centered or Tiled BitmapDisplayMode is not supported for Picture control.
- For EditedValue property, validation checking using values in the EditedValue property is not supported in CA Gen Release 8.5.
- The Handle property always returns a value of zero in Web View applications.

Events

The following CA Gen events are not supported by JavaScript and are ignored at runtime:

Note: Read-only entry fields do not intercept any events.

Control	Event Handlers
Window	RightMouseBtnDown RightMouseBtnUp WinMove WinResize
Single or multi-line entry field	RightMouseBtnDown RightMouseBtnUp MouseMove Keypress
Check box	RightMouseBtnDown RightMouseBtnUp MouseMove
Radio Button	RightMouseBtnDown RightMouseBtnUp MouseMove
Non-Enterable Drop down	LeftMouseBtnDown LeftMouseBtnUp RightMouseBtnDown RightMouseBtnUp MouseMove
Non-Enterable List	RightMouseBtnDown RightMouseBtnUp MouseMove MouseEnter MouseExit GainFocus LoseFocus
Picture	RightMouseBtnDown RightMouseBtnUp MouseMove

The following events are not supported in Web View applications:

- Single-click event is not supported in List Boxes
- Gain focus events and Lose focus events

Web Graphics

All bitmaps must be converted to JPG, GIF, or PNG graphic formats. Background bitmaps are implemented by the background image style sheet property applied to the body element (and must be implemented by a JPG, GIF, or PNG graphic file). Before application assembly, create a directory in the model directory with the name *assets* and manually copy the images to the assets directory.

Image file names must be lowercase and the file extension must match what is specified in the model.

Colors

An object's color properties stored in the model include the following:

- background color
- foreground color
- highlighting background color
- highlighting foreground color
- disabled background color
- disabled foreground color
- background text window color
- foreground text window color

The object's foreground color property in the model is used as the value for the color property in the style sheet. The object's background color property in the model is used as the value for the background color property in the style sheet. The other color properties are not supported and are ignored at runtime.

Browser Support

In Mozilla Firefox, you cannot scroll Multi-line Entry Fields without horizontal scroll bars when the number of characters that you typed in the field exceeds the field display area.

If you resize the browser in Web View applications, you may not be able to scroll or access all the elements in the window or dialog.

Asynchronous Support

Web View does not support Asynchronous Processing.

Bi-directional Language Support

Bi-directional language includes text that follows both the right-to-left text orientation (for example, Arabic and Hebrew) as well as the left-to-right text orientation (for example, English).

Web View applications support bi-directional languages partially. You may see some discrepancies in the user interface such as buttons could be placed in the wrong location, radio buttons may not be aligned correctly, text may be partially visible in a text editor, and so on.

Help Support

In Web View applications, all help actions (Help, Help for Help, Extended Help, Keys Help, and Help Index) will display the generated <LoadModuleName>.Help.html in a dialog.

Chapter 4: Generating Applications

You can generate Web View applications based on your model objects using CA Gen Studio. This task is a two-step process. First, you generate source code using the CA Gen Studio generation capabilities, and then you compile and assemble that source code as necessary using the Build Tool or Implementation Toolsets.

This section contains the following topics:

Generation Configuration

A *Generation Configuration* (also known as a configuration instance) defines the model objects and the target environment parameters that you can specify to generate source code. You can create and manage multiple configurations for any model.

By default, a new configuration is file-based. However, if you have the appropriate model access you can save the configuration in the model.

Note: In Client Server Encyclopedia and Host Encyclopedia, we refer to a configuration as Configuration Instance. In CA Gen Studio, we refer to a configuration as Generation Configuration.

Configuration Types

The following configuration types are supported in Generation Configurations:

- Cooperative
- Non-cooperative

You can create new configurations in the Generation Configurations dialog based on these configuration types.

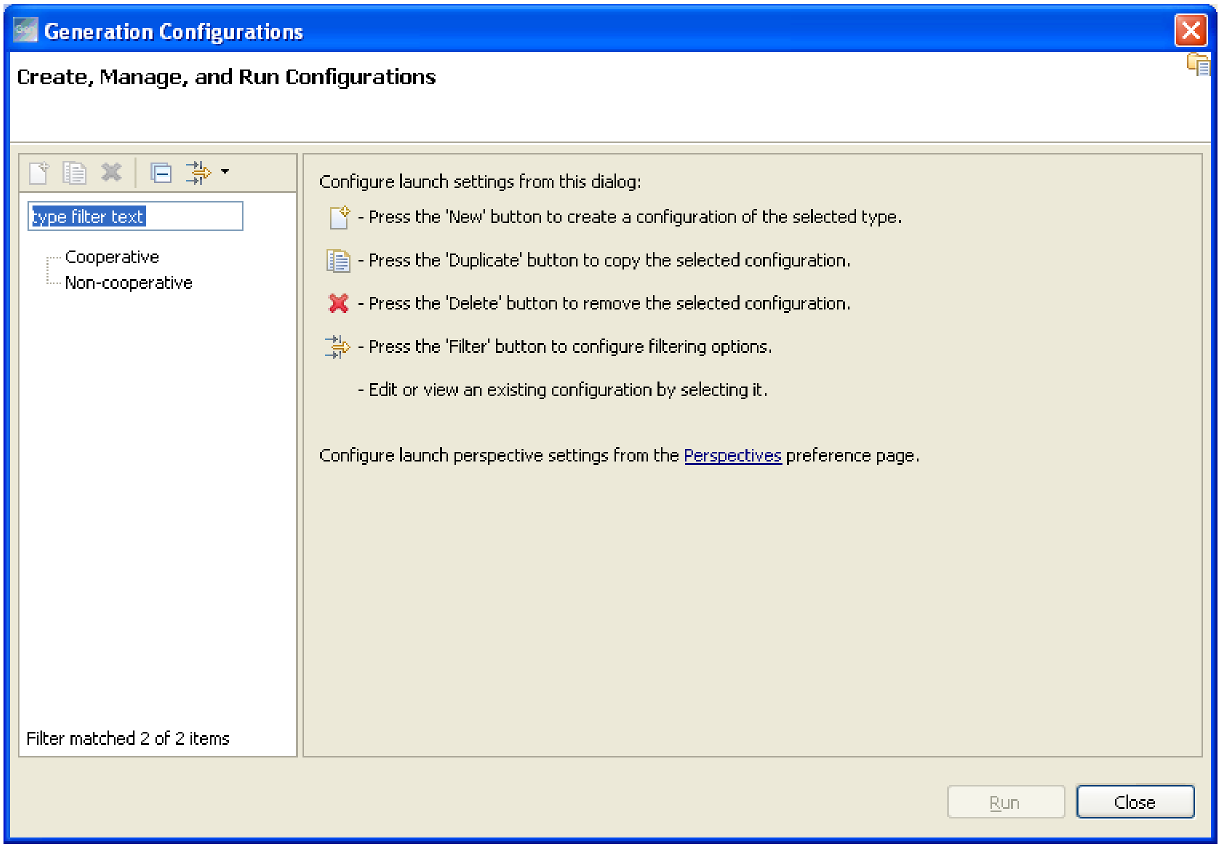
Model-based Configurations and File-based Configurations

Configurations that are saved to a model are *Model-based Configurations*. Configurations that are saved to a file are *File-based Configurations*. By default, new configurations are file-based configurations but with appropriate model access, you can change a file-based configuration to a model-based configuration.

You cannot change a model-based configuration to a file-based configuration if you do not have permissions to delete a configuration from a model. This can occur if the model is read-only or the configuration is not checked out with Delete protection in your subset.

Manage Generation Configurations

You can create, edit, delete, and run configurations using the Generation Configuration option in CA Gen Studio. Open a model in CA Gen Studio and click Construction, Generation Configurations in the main menu to open the Generation Configurations dialog.



Generation Configurations

Create, Manage, and Run Configurations

type filter text

- Cooperative
- Non-cooperative

Filter matched 2 of 2 items

Configure launch settings from this dialog:

- Press the 'New' button to create a configuration of the selected type.
- Press the 'Duplicate' button to copy the selected configuration.
- Press the 'Delete' button to remove the selected configuration.
- Press the 'Filter' button to configure filtering options.
- Edit or view an existing configuration by selecting it.

Configure launch perspective settings from the Perspectives preference page.

Existing configurations do not work with the current release and you need to recreate the configurations. If there are any configurations available in a model, then delete these configurations before installing the latest version.

You can perform the following tasks in the Generation Configurations dialog:

- Create a configuration
- Edit a configuration
- Delete a configuration
- Run a configuration

If a generation configuration has protected objects selected in it, the Name field is disabled and you cannot rename the generation configuration.

Note: The duplicate functionality is not currently available.

More information:

Create a Configuration (see page 35)
Edit a Configuration (see page 47)
Delete a Configuration (see page 47)

Create a Configuration

You can create a configuration using the Generation Configurations dialog. You can create configurations with the same name for different models irrespective of whether the configuration is a file-based configuration or a model-based configuration.

For file-based configurations, the configuration names are not case-sensitive.

For model-based configurations, the encyclopedia stores the model-based configurations with configuration names in a case-sensitive format. You can create two configurations with the same configuration name if you create the configurations in two different subsets. If the two configurations have the same configuration name and case, the upload applies renaming logic and creates a unique configuration name for the second configuration that is created. If the two configuration names are in a different case, then both the configurations are stored in the model without being renamed.

For model-based configurations where the model is checked in, you can rename the configuration from CA Gen Studio only by checking out the configuration from the model with delete protection.

Note: The Encyclopedia functionality provides a function for renaming the configuration outside of CA Gen Studio.

Follow these steps:

1. Open a model in CA Gen Studio.

   The Generation Configuration menu option and toolbar icon are enabled.

2. Click Construction, Generation Configuration from the CA Gen Studio main menu.

   The Generation Configurations dialog opens.

3. Right-click on the configuration type in the left pane and click New from the pop-up menu.

   A new configuration is added under the configuration type in the left pane. By default, a system generated name is assigned to the new configuration; however, you can modify the name by providing a new configuration name in the Name field.

   You can edit the configuration settings in the following tabs on the right pane:

   Main

   Defines the local model, the target environment values, the generation options, and the dialects for the selected generation configuration.

   Database

   Defines the database-specific generation settings for the selected DDL options.

   Model Object Selection

   Lists a tree hierarchy of all the Business Systems that are associated with the model selected in the Main tab. Only the valid objects for the selected configuration type are displayed in the hierarchy.

   Note: If your configuration instance is saved to a model, the model objects that are in Read protection are not displayed in this tab.

   The tree hierarchy lists the following objects in the same order as mentioned:

   Business System

   Packaging Type (Cooperative or Non-cooperative)

   Load Modules

   Procedure Steps

   Action Blocks

   Primary Windows/Dialogs

   Secondary Dialogs

Trace

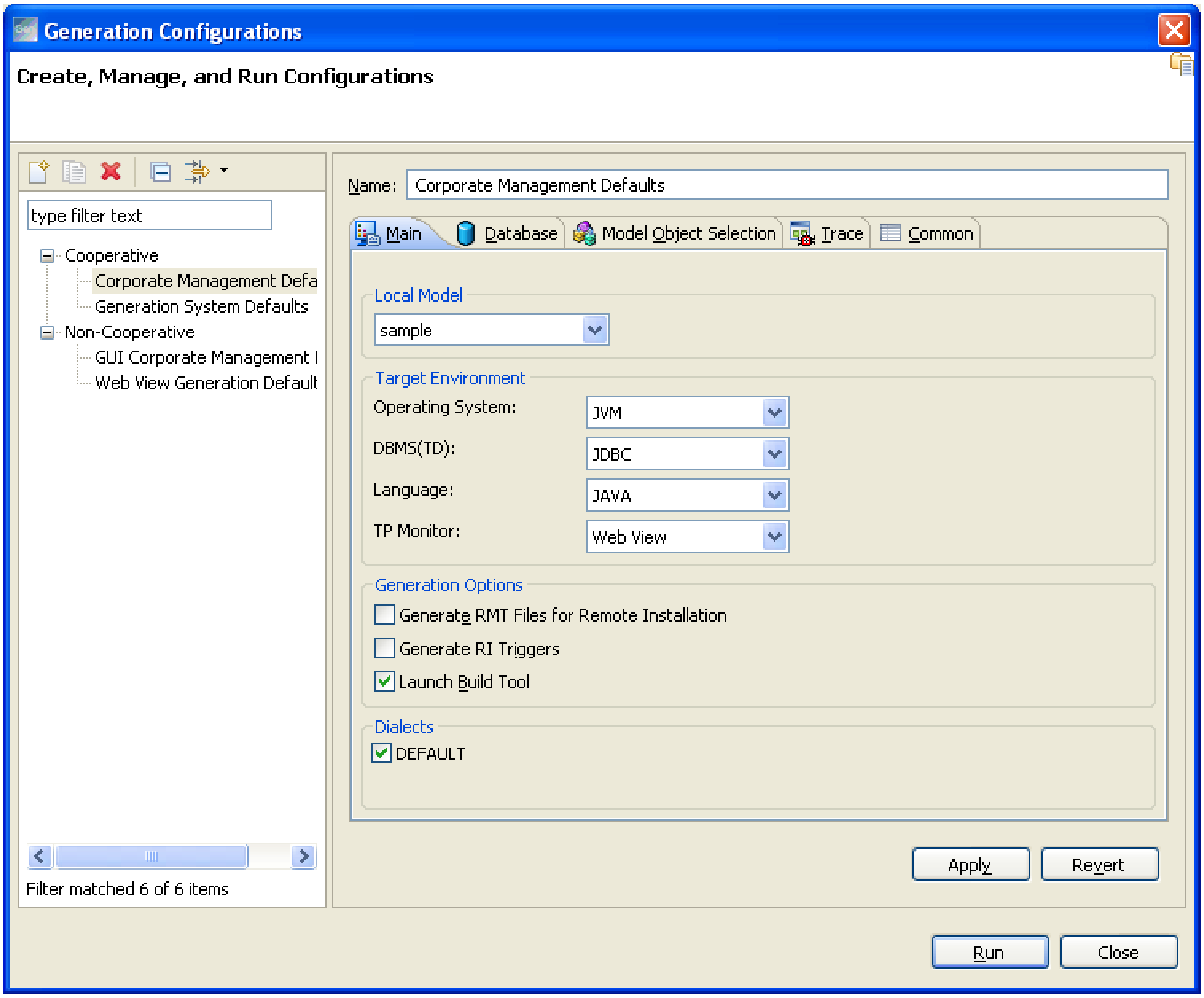
Defines the objects for which you can generate trace information. Only the objects that are selected for generation in the Model Objects Selection tab can be set to trace.

Note: For a file-based configuration, all the objects that are selected for generation in the Model Object Selection tab are also listed in the Trace tab. For a model-based configuration, the objects that are in Read protection are not displayed.

Common

Defines the settings to save the configuration either as a file-based configuration or as a model-based configuration. You can configure the Console view to display the generation output.

4. Select the Local Model, set generation options for the Target Environment, and select the Generation Options in the Main tab.



Generation Configurations

Create, Manage, and Run Configurations

type filter text

- Cooperative
  - Corporate Management Defa
  - Generation System Defaults
- Non-Cooperative
  - GUI Corporate Management I
  - Web View Generation Default

Filter matched 6 of 6 items

Name: Corporate Management Defaults

Local Model

sample

Target Environment

Operating System: JVM

DBMS(TD): JDBC

Language: JAVA

TP Monitor: Web View

Generation Options

□ Generate RMT Files for Remote Installation

□ Generate RI Triggers

☑ Launch Build Tool

Dialects

☑ DEFAULT

Local Model

Displays the model name that is associated with the current configuration. When creating a configuration, the first open model is selected in this drop-down by default. For a model-based configuration, this drop-down list is disabled if the current model or the subset is read-only or portions of the configuration cannot be deleted.

If you change the model selection in this drop-down list, all the selections of the model-specific objects in all the tabs are reset.

For a model-based configuration, you cannot select a read-only model from the Local Model drop-down list.

Target Environment

Specifies the values for the Target Environment. The values selected in these lists are inter-related and each field displays only the valid options that are based on the selection in other fields.

For example: Any change in the Operating System drop-down list updates the DBMS (TD) and the Language drop-down lists in a way such that these lists have only the valid options for the selected Operating System.

Operating System

Specifies the operating system under which the generated application executes.

DBMS(TD)

Specifies the database management system (DBMS) technical design that is used by the generated application.

Note: The DBMS(TD) drop-down list is disabled if the current model or the subset is read-only or the selected database objects are in Read protection. You cannot select the value <NONE> if the configuration is stored in a model and it includes protected database objects that are selected from the table in the Database tab. You cannot select the value JDBC or MS/SQL if the configuration is stored in a model and it includes protected tablespace objects that are selected from the table in the Database tab.

Language

Specifies the high-level programming language in which the Generation Configuration generates an application. The language choices available depend on the configuration of your CA Gen software.

TP Monitor

Specifies the Transaction Processing Monitor under which the application executes. The choices that are listed are valid TP monitors for the specified operating system.

Generate RMT Files for Remote Installation

Creates a remote file and does not invoke the Build Tool.

Generate RI Triggers

Generates Referential Integrity Trigger modules.

Launch Build Tool

Invokes the Build Tool to build and assemble the application after generation is complete. To invoke the Build Tool, select this check box and then select the Load Module in the Model Objects Selection tab, at least for Action Blocks.

If Build Tool is already open, the same instance of Build Tool is used to build the application.

Dialects

Generates dialect windows or dialogs. Select the dialects in the Main tab and the associated windows or dialogs in the Model Object Selection tab.

Apply

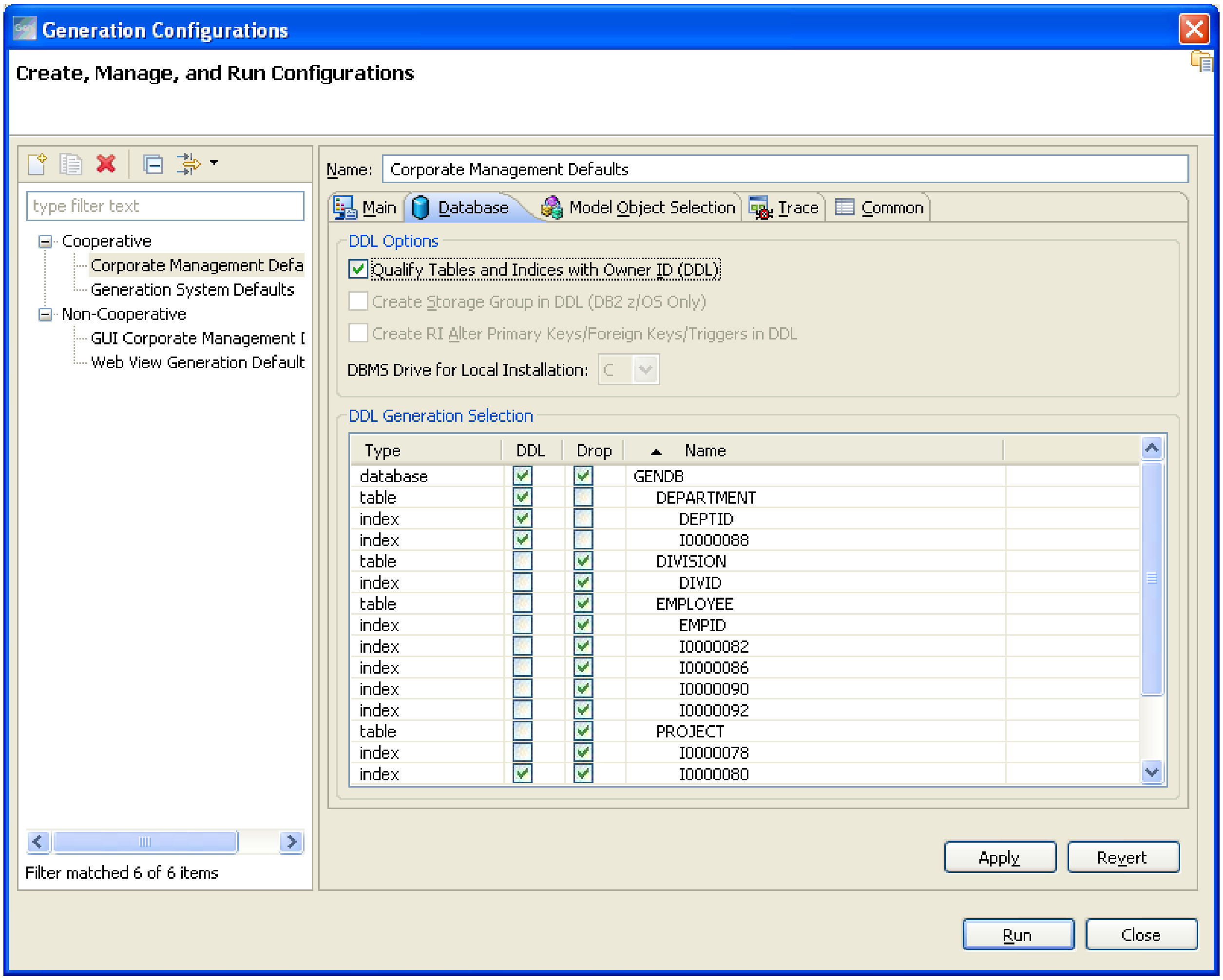
Saves the configuration settings.

Revert

Undoes the last change. You cannot undo your last change after saving the settings.

5. Select the database-specific generation defaults in the Database tab.

   Note: The DDL Options and the DDL Generation Selection are disabled on the Database tab if you select <NONE> in the DBMS(TD) drop-down in the Main tab.


Generation Configurations
Create, Manage, and Run Configurations
type filter text
Cooperative
Corporate Management Defa
Generation System Defaults
Non-Cooperative
GUI Corporate Management I
Web View Generation Default
Filter matched 6 of 6 items
Main
Database
Model Object Selection
Trace
Common
Apply
Revert
Run
Close

Name: Corporate Management Defaults

DDL Options

☑ Qualify Tables and Indices with Owner ID (DDL)

□ Create Storage Group in DDL (DB2 z/OS Only)

□ Create RI Alter Primary Keys/Foreign Keys/Triggers in DDL

DBMS Drive for Local Installation: C

DDL Generation Selection

Type	DDL	Drop	Name
database	☑	☑	GENDB
table	☑	☑	DEPARTMENT
index	☑	□	DEPTID
index	☑	□	I0000088
table	□	☑	DIVISION
index	□	☑	DIVID
table	□	☑	EMPLOYEE
index	□	☑	EMPID
index	□	☑	I0000082
index	□	☑	I0000086
index	□	☑	I0000090
index	□	☑	I0000092
table	□	☑	PROJECT
index	□	☑	I0000078
index	☑	☑	I0000080

Qualify Tables and Indices with Owner ID (DDL)

Adds the Owner ID as a prefix to all the table and index names in SQL DML for all the DBMS except Microsoft SQL.

Create Storage Group in DDL (DB2 z/OS Only)

Generates DDL with storage group definitions.

If you select DB2 z/OS as the database, the Create Storage Group in DDL (DB2 z/OS Only) check box is enabled.

If you select the Create Storage Group in DDL (DB2 z/OS Only) check box and the database contains storage groups, the DDL Generation Selection table contains storage groups for the database.

Create RI Alter Primary Keys/Foreign Keys/Triggers in DDL

Generates DDL that creates alternate Referential Integrity Primary and Foreign Key triggers so that they can be compared against an existing database.

Note: This option creates the RI constraints and primary key statements only. No other DDL statements are generated.

If you select the Create RI Alter Primary Keys/Foreign Keys/Triggers in DDL check box, the Create Storage Group in DDL (DB2 z/OS Only) check box and the Drop column are disabled.

DBMS Drive for Local Installation

Specifies the drive on which the DBMS installs the database for a local installation. This option is valid only for the DB2 UDB database.

DDL Generation Selection

Displays the database objects in a tabular format. You can sort the Name column in ascending or descending order.

The following object types are included in this table:

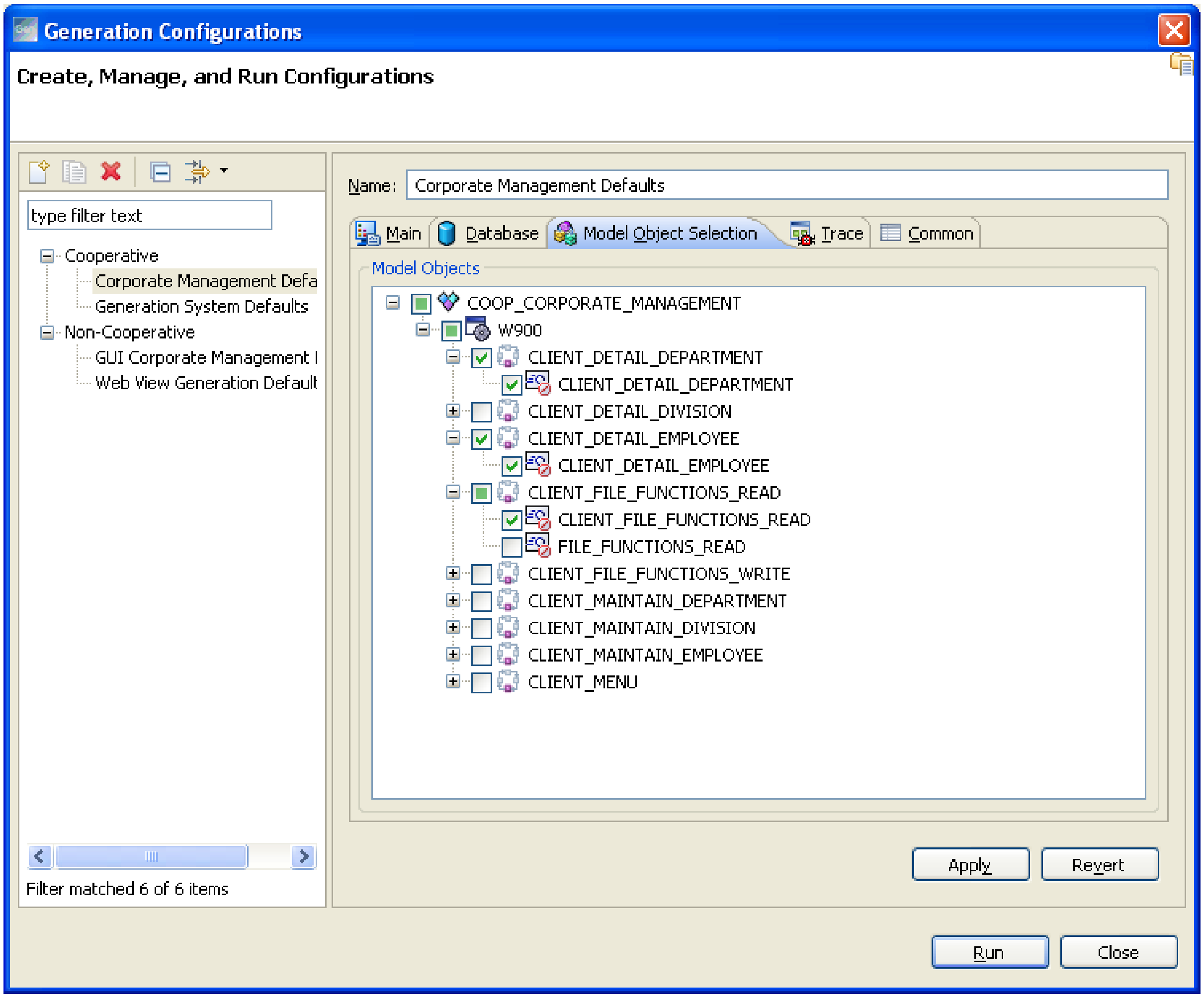
- Database
- Table
- Index
- Tablespace
- Storage Group (DB2 z/OS only)

Note: When a configuration includes the storage group for a database, we recommend you to select the database also in the configuration.

For model-based configurations, you cannot change the selection state of a database object if it has read-only protection.

6. Select the model objects to include in your configuration in the Model Object Selection tab. This tab displays the model objects that are based on the configuration type.

   Note: If you select a Business System, all the Load Modules under the Business System object are selected. Similarly, if you select a Procedure Step, all the Action Blocks under the Procedure Step object are selected.



type filter text

- Cooperative
  - Corporate Management Defa
  - Generation System Defaults
- Non-Cooperative
  - GUI Corporate Management I
  - Web View Generation Default

Filter matched 6 of 6 items

Name: Corporate Management Defaults

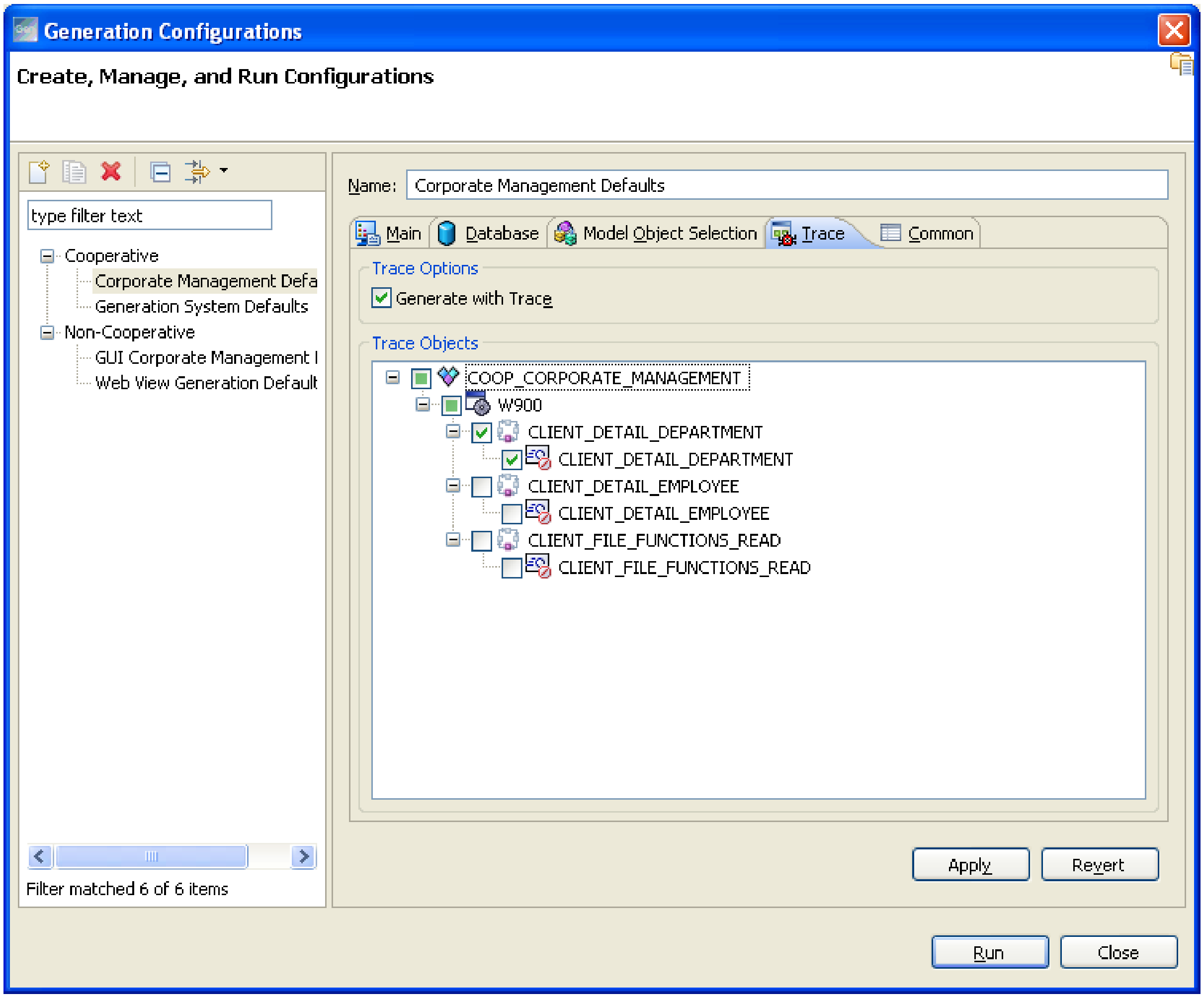
- ▣ COOP_CORPORATE_MANAGEMENT
  - ▣ W900
    - ☑ CLIENT_DETAIL_DEPARTMENT
      - ☑ CLIENT_DETAIL_DEPARTMENT
    - □ CLIENT_DETAIL_DIVISION
    - ☑ CLIENT_DETAIL_EMPLOYEE
      - ☑ CLIENT_DETAIL_EMPLOYEE
    - ▣ CLIENT_FILE_FUNCTIONS_READ
      - ☑ CLIENT_FILE_FUNCTIONS_READ
      - □ FILE_FUNCTIONS_READ
    - □ CLIENT_FILE_FUNCTIONS_WRITE
    - □ CLIENT_MAINTAIN_DEPARTMENT
    - □ CLIENT_MAINTAIN_DIVISION
    - □ CLIENT_MAINTAIN_EMPLOYEE
    - □ CLIENT_MENU

7. Select the objects for which you need to generate trace information in the Trace tab. Only Action Blocks and Procedure Steps that are selected in the Model Object Selection tab are enabled in the Trace tab for selection.

   Note: In Windows 7, the Trace tab shows disabled checkboxes only in the Windows Classic theme.

   Generate with Trace

   Specifies if trace information should be generated for the selected configuration. Selecting this check box enables the Trace Objects panel.



Generation Configurations

Create, Manage, and Run Configurations

type filter text

- Cooperative
  - Corporate Management Defa
  - Generation System Defaults
- Non-Cooperative
  - GUI Corporate Management I
  - Web View Generation Default

Filter matched 6 of 6 items

Name: Corporate Management Defaults

Main | Database | Model Object Selection | Trace | Common

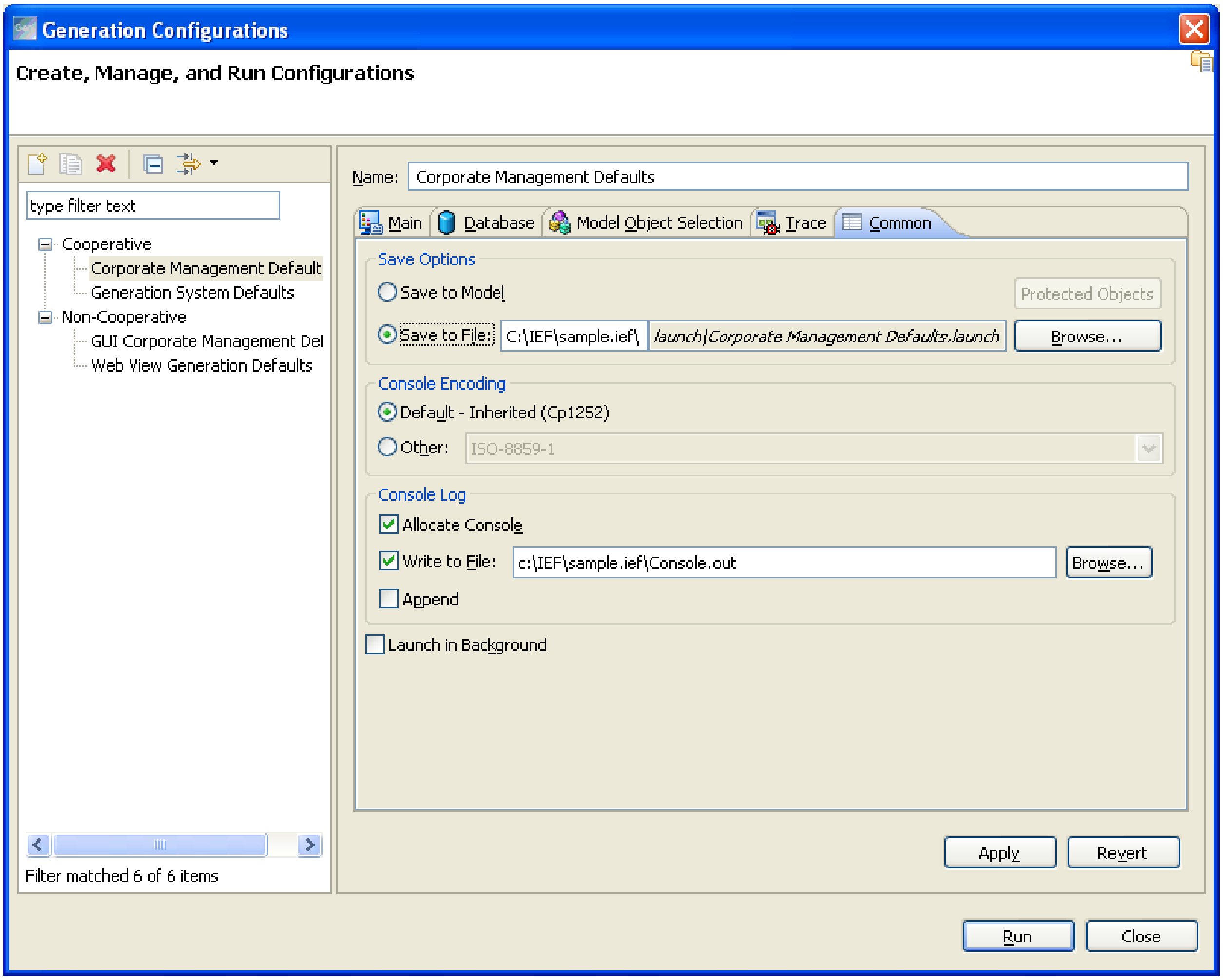
Trace Options

☑ Generate with Trace

Trace Objects

- ■ COOP_CORPORATE_MANAGEMENT
  - ■ W900
    - ☑ CLIENT_DETAIL_DEPARTMENT
      - ☑ CLIENT_DETAIL_DEPARTMENT
    - □ CLIENT_DETAIL_EMPLOYEE
      - □ CLIENT_DETAIL_EMPLOYEE
    - □ CLIENT_FILE_FUNCTIONS_READ
      - □ CLIENT_FILE_FUNCTIONS_READ

8. Set the options to save the configuration settings in the Common tab. You can save the configuration settings either to the model or to a Launch file. You can also set the configuration options to save the generation output in a file in this tab.



Generation Configurations

Create, Manage, and Run Configurations

type filter text

- Cooperative
  - Corporate Management Default
  - Generation System Defaults
- Non-Cooperative
  - GUI Corporate Management Del
  - Web View Generation Defaults

Filter matched 6 of 6 items

Name: Corporate Management Defaults

Main | Database | Model Object Selection | Trace | Common

Save Options

○ Save to Model

● Save to File: C:\IEF\sample.ief\ launch\Corporate Management Defaults.launch

Console Encoding

● Default - Inherited (Cp1252)

○ Other: ISO-8859-1

Console Log

☑ Allocate Console

☑ Write to File: c:\IEF\sample.ief\Console.out

□ Append

□ Launch in Background

Save to Model

Saves the configuration settings to the model selected in the Main tab.

The Save to Model option is disabled in the following cases:

- If the configuration is saved in a file and the model is read-only.
- Protections of selected objects in the Database tab or the Model Object Selection tab prevent moving the configuration to the model.

Protected Objects

Displays a report listing the objects with Read protection that are defined in your file-based configuration. The Protected Objects button is enabled only when the Save to Model option is disabled and you have objects in your configuration with Read protection. The Protected Objects report file defaults to your local model directory.

Save to File

Saves the configuration settings in a launch file, which by default is saved in the launch directory within your model directory.

Type the path in the field that is provided or click Browse to change the location. If the location you specified is valid but the folder is not yet created, a new folder structure is created. If the location you specified is not valid, an error message is displayed.

Note: If the specified directory path in the Save to File field is longer than the size of the field, the Protected Objects and the Browse buttons shift to the right and may not be visible on the dialog. To view these buttons, increase the dialog size accordingly.

The Save to File option is disabled in the following cases for model-based configurations:

- If the configuration is saved in the model and the model is read-only.
- If any of the object protections of the selected objects in the Database tab and the Model Object Selection tab would prevent moving the configuration out of the model.

For example:

- If the configuration instance is checked out with Read, the Save to File option is disabled.
- If the procedure step selected on the Model Object Selection tab is checked out with Read, the Save to File option is disabled.

The launch file is in XML format and encoded in UTF-8 format. The filename is in the following format:

<ConfigurationName>.launch

<ConfigurationName>

Specifies the name of the configuration.

Default: Selected

Console Encoding

Specifies character encoding and displays the generation output.

Default: Encoding used by CA Gen Studio.

Allocate Console

Specifies the use of the Console view to display the generation output.

Write to File

Saves the generation output in a file. By default, the file location is the directory in which the model is saved. *Console.out* is the generation output filename.

Type the path in the field that is provided or click Browse to change the location. If the location you specified is valid but the folder is not yet created, a new folder structure is created. If the location you specified is not valid, an error message is displayed.

If the path specified is not accessible, an error message is displayed in the Console view.

Append

Identifies if the output file is to be configured to the Append mode or the Overwrite mode.

Default: Overwrite mode

Launch in Background

Runs the generation configuration in the background. By default, the generation configuration runs in the foreground and you can see a pop-up dialog showing the generation progress.

9. Click Apply.

The generation configuration is created.

Edit a Configuration

You can edit existing configurations using the Generation Configurations dialog.

Follow these steps:

1. Open a model in CA Gen Studio.

   The Generation Configuration menu option and toolbar icon are enabled.

2. Click Construction, Generation Configuration from the CA Gen Studio main menu.

   The Generation Configurations dialog opens.

3. Select a configuration from the list of configurations in the left pane.
4. Update the information that is required for the type of Configuration.
5. Click Apply to save the configuration.

Delete a Configuration

You can delete an existing configuration from the Generation Configurations dialog. You cannot delete configurations that are stored in a model and that have protected model objects or database objects selected in it.

Follow these steps:

1. Open a model in CA Gen Studio.

   The Generation Configuration menu option and toolbar icon are enabled.

2. Click Construction, Generation Configuration from the CA Gen Studio main menu.

   The Generation Configurations dialog opens.

3. Select a configuration from the list of configurations in the left pane.
4. Click the delete icon to delete the selected configuration. You can also right-click on an existing configuration in the left pane and click Delete from the pop-up menu.

   The configuration is deleted.

Note: In a file-based configuration, the Launch file is deleted when you delete the configuration.

Subsetting Considerations for Configurations

Define a subset for configurations based on the tasks you need to perform as well as the expansion options and the protection levels checked out in your model/subset.

Note: For more information about subsetting, see the *Client Server Encyclopedia Subsetting User Guide* and the *Host Encyclopedia Subsetting User Guide*.

More information:

Model Object Protection Requirements (see page 48)
Read-only Models or Subsets (see page 50)

Model Object Protection Requirements

Protection requirements are applicable only on model-based configurations. If you perform a task for which the model objects included in a subset do not have the appropriate protection, the proposed changes are either not allowed or discarded. An error message is displayed for information purpose.

Protection levels are hierarchical and are from highest to lowest:

1. Delete
2. Modify
3. Access
4. Read

You can delete a configuration only if it is defined with Delete protection and not with Update, Read, or Access protection. You can modify a configuration only if it is defined with Update or Delete protection and not with Read or Access protection.

For example, if a subset is defined with Delete protection on Procedure Step A but the subset is not checked out, another subset containing Procedure A with Delete protection may be checked out and that procedure step may be deleted.

The following requirements govern model object protection with default expansion:

Delete

You can delete existing configurations.

You can add a new configuration and can include target objects unless the target objects are defined with Read protection.

You can update existing configurations including update of target environments. You can select new objects or clear selected objects from the model objects list and update trace selection unless the target object is defined with Read protection.

You can update the generation options and can generate a configuration unless the target objects are defined with Read protection in which case you cannot select the trace option.

Note: The Delete icon is disabled in the Generation Configurations dialog if the configuration is not defined with Delete protection.

Modify

You can add a new configuration and can include target objects unless the target objects are defined with Read protection.

You can update existing configurations including update of target environments. You can select new objects or clear selected objects from the model objects list and update trace selection unless the target objects are defined with Read protection.

You can update the generation options and can generate a configuration unless the target objects are defined with Read protection in which case you cannot select the trace option.

You cannot delete existing configurations.

Access

You can add a new configuration and can include target objects unless the target objects are defined with Read protection.

You cannot update existing configurations including update of target environments. You cannot select new objects or clear selected objects from the model objects list and you cannot update trace selection.

You can generate a configuration but you cannot update the generation options or the trace options.

You cannot delete existing configurations.

Read

You cannot add a new configuration.

You cannot update existing configurations including update of target environments. You cannot select new objects or clear selected objects from the model objects list and you cannot update trace selection.

You cannot delete existing configurations.

You can generate a configuration but you cannot update the generation options or the trace options.

Read-only Models or Subsets

You can generate a configuration from a read-only model or subset.

- You can create, modify, and delete file-based configurations from a read-only model.
- You can create, modify, or delete model-based configurations when the model or the subset is read-only.

Run Generation Configurations

You can run the configurations using any of the following methods:

- From the Generation Configurations dialog.
- Right-click on a valid object in the Web View UI Navigator view and click Generate or Generate including subordinates depending on your requirement.

If you run a generation configuration that includes any of the following generators, a successful generation updates the model if it is not a read-only model:

- Procedure Step
- Action Block
- RI Triggers
- Web Client

Note: DDL generation does not update the model.

The location of the generated files is the same as in Toolset.

You can view the generation output messages in the Console view. Configuration generation errors, if any, are displayed in the Problems view. The Problems view is cleared if the current configuration is regenerated or another configuration is generated. If an object belongs to more than one generation configuration, errors logged for that object in other generation configurations is not cleared.

More information:

Run a Configuration from the Generation Configurations Dialog (see page 51)
Run a Configuration from Web View UI Navigator (see page 52)

Run a Configuration from the Generation Configurations Dialog

You can run the configurations from the Generation Configurations dialog using any of the following methods:

- Select a configuration in the left pane and click Run.
- Select a configuration in the left pane and press the Enter key.
- Double-click the configuration in the left pane.

The following procedure explains how to select a configuration in the Generation Configurations dialog and click Run.

Follow these steps:

1. Open a model in the Web View UI Generation perspective.

   The Generation Configuration menu option and toolbar icon are enabled.

2. Click Construction, Generation Configurations from the CA Gen Studio main menu.

   The Generation Configurations dialog opens. The defined configurations are listed in the left pane.

3. Select a configuration in the left pane or create a new configuration.

   Note: For information about creating a configuration, see Create a Configuration.

4. Click Run.

   If you have selected the configuration to run in the foreground, the Progress Information dialog opens showing the progress.

   Note: If you click Cancel on the Progress Information dialog, the generation stops after completing the current generation step and closes the pop-up dialog.

   The selected objects are generated based on the configurations. You can view the generation output messages in the Console view.

Run a Configuration from Web View UI Navigator

You can run a configuration from the Web View UI Navigator in the Web View UI Generation perspective. You can run configurations of valid objects using this procedure. Running a configuration from the Web View UI Navigator lets you generate a subset of the configuration such as an individual Action Block or a Procedure Step.

Valid Configuration

Specifies a generation configuration for an object in which the object is selected for generation.

Valid Object

Specifies an object in the Web View UI Navigator that has valid configurations. Any object that can be selected for generation in the Model Object Selection tab in the Generations Configuration dialog are valid for generation.

The following objects are valid to run a configuration from the Web View UI Navigator view:

- Model
- Business System
- Procedure Step
- Action Block

Follow these steps:

1. Open the Web View UI Generation perspective in CA Gen Studio.
2. Open a model and expand the model objects in the Web View UI Navigator.
3. Right-click on a valid object that you want to run the configuration.

   Note: Use the Ctrl key to select multiple objects in Web View UI Navigator. When you use the Ctrl key to select multiple objects in the Navigator view, only the selected objects are generated including the child objects that were selected in the configuration.

   A pop-up menu opens showing Generate and Generate including subordinates menu items. The model object and the packaging type object (Cooperative or Non-cooperative) in the Web View UI Navigator view display only the Generate including subordinates menu item. All the other objects in the Web View UI Navigator view display Generate and Generate including subordinates menu items.

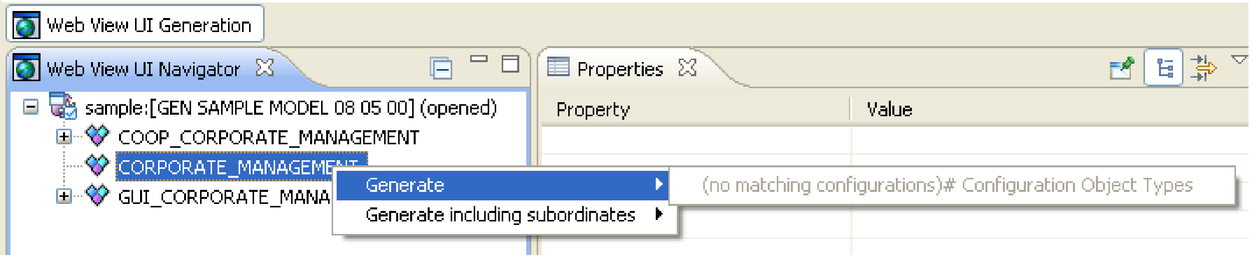
   Generate

   Generates the objects that are selected in the Web View UI Navigator.

   Generate including subordinates

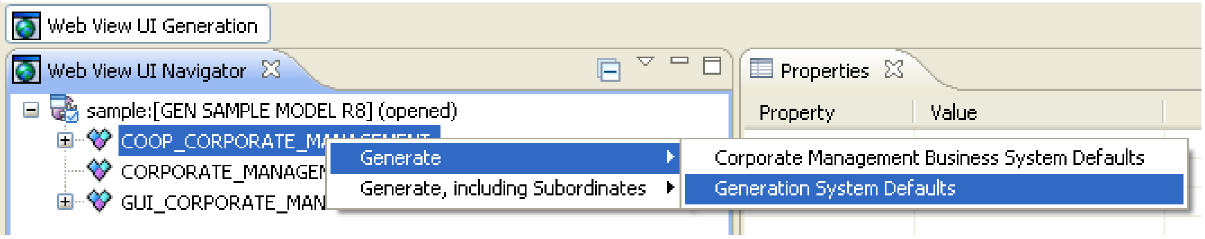
   Generates the selected objects in the Web View UI Navigator along with their child objects that were selected in the configuration.

Note: If there are no valid configurations for an object, the Generate and Generate including subordinates menus display (no matching configurations)# Configuration Object Types as a sub-menu as shown in the following illustration:


Web View UI Generation
Web View UI Navigator
sample:[GEN SAMPLE MODEL 08 05 00] (opened)
COOP_CORPORATE_MANAGEMENT
CORPORATE_MANAGEMENT
GUI_CORPORATE_MANA
Generate
Generate including subordinates
(no matching configurations)# Configuration Object Types
Properties
Property
Value

4. Click Generate or Generate including subordinates depending on your requirement.

   A pop-up menu opens showing only the valid configurations for that object.


Web View UI Generation
Web View UI Navigator
sample:[GEN SAMPLE MODEL R8] (opened)
COOP_CORPORATE_MANAGEMENT
CORPORATE_MANAGEM
GUI_CORPORATE_MAN
Generate
Generate, including Subordinates
Corporate Management Business System Defaults
Generation System Defaults
Properties
Property
Value

5. Click the required configuration name.

   If you have selected the configuration to run in the foreground, the Progress Information dialog opens showing the progress.

   Note: If you click Cancel on the Progress Information dialog, the generation stops after completing the current generation step and closes the pop-up dialog.

   The selected objects are generated based on the configurations. You can view the generation output messages in the Console view.

Chapter 5: Deploying the Web View Application

This section contains the following topics:

Generate and Build the Web View Application

You can build the application using the Build Tool after you generate source code from your model definitions in CA Gen Studio. The Build Tool compiles the source code, assembles the application, and prepares it for deployment.

After generating your application, you have to build it on a Windows platform. The Build Tool works with either ICM files that result from a Local generation, or RMT files that result from a Remote generation. You can set options that affect the build process of your application through the Build Tool Profile.

Note: For more details about the Build Tool and Assemble utility, see the *Build Tool User Guide*.

Deploy the Application

Creating an EAR file with Build Tool's Assemble feature targeting Web View textually compresses static files, such as HTML, CSS, and JavaScript files. If these files are generated, they are compressed during the assemble process in Build Tool. If these files are delivered, these have been already compressed in advance.

However, in some cases, it is necessary to leave the files uncompressed. The OPT.DEBUG token in the Build Tool profile controls the behavior. When OPT.DEBUG is set to 'YES', all textual compression is disabled and the pre-packaged non-debug Web View JavaScript files, WebViewUI_Debug.zip file, and the debug version of YUI library files are packaged into the assembled EAR file.

To be able to run a Web View application, you need to first assemble and then deploy it to an Application Server. You can deploy a Web View application to the following Application Servers:

- WebLogic
- WebSphere
- JBoss

Follow these steps:

1. Open Build Tool.
2. Click Action, Assemble.

   The EAR File Assemble Details dialog opens.
3. Click Application Server on the left panel.

   The Application Server specific settings appear.
4. Select the required application server from the Application Server drop-down.
5. Click OK.

The EAR file is stored in the following location:

```
\\<model directory>\java\deploy.j2ee
```

Format for Load Module and Trancode Filenames

The Web View UI generator will generate startup files for each load module and trancode defined in the window and cooperative packaging. These HTML files are used to start the application.

For load modules, the filenames will be suffixed with '.LM'.

Example:

If the packaging is windows and the load module name is X, the load module HTML file will be generated as 'WIN.X.LM.html'.

For trancodes, the filename will contain the trancode name suffixed with '.TC'.

Example:

If the packaging is cooperative and the trancode name is X, the trancode HTML file will be generated as 'COOP.X.TC.html'.

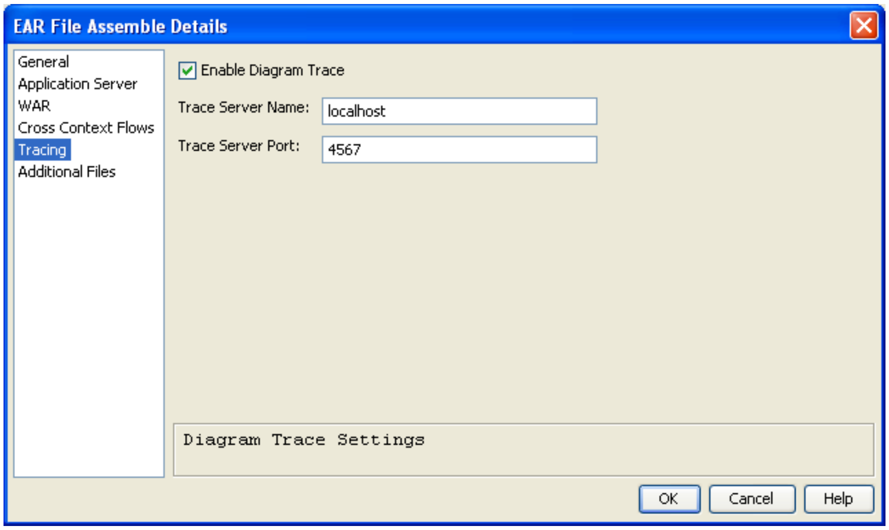
Trace Generated Applications

To debug and test your application you can enable tracing. This section shows how to enable tracing.

Trace a Generated HTML Application Using Diagram Trace Utility

Follow these steps:

1. Select to generate with Trace during Generation.
2. From the Build Tool, in the EAR File Assemble Details dialog, select Enable Diagram Trace under the Tracing tag.



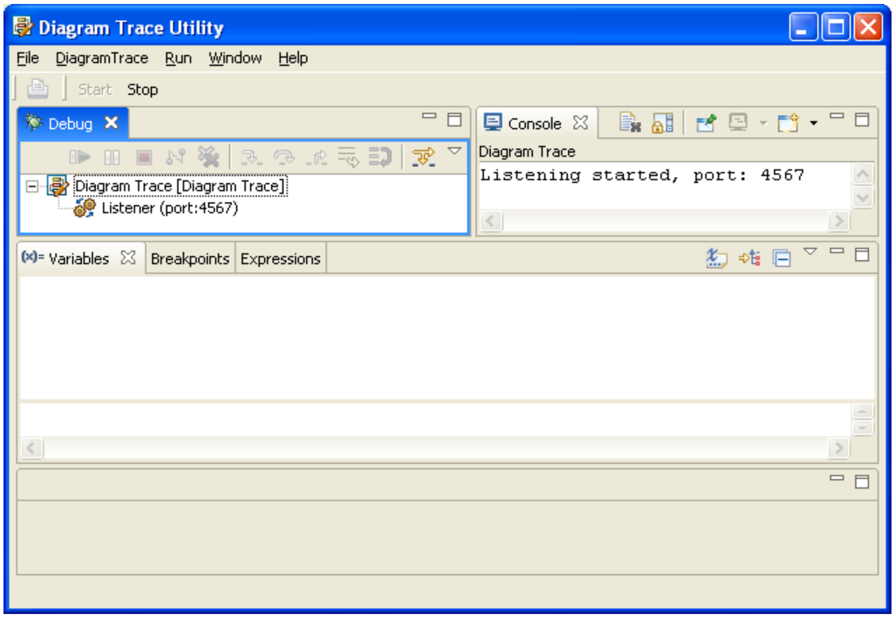
☑ Enable Diagram Trace

Trace Server Name: localhost

Trace Server Port: 4567

3. To start the Diagram Trace Utility, click Start, All Programs, CA, Gen *xx*, Diagram Trace Utility.

   Note: *xx* refers to the current release of CA Gen. For the current release number, see the *Release Notes*. Diagram Trace Utility executes only on Windows.

   ```
   Listening started, port: 4567
   ```

   The Diagram Trace Utility will "autostart" and listen on port 4567. You may change the default port from within the Diagram Trace Utility.

4. Start the application normally from the browser by invoking the TC.html file. You invoke the HTML file by typing the lowercase name into the browser "location" or URL field.

5. Step through the Action Diagram code.

Note: For more information, see the *Diagram Trace Utility User Guide* or the *Diagram Trace Utility Help*.

Run the Web View Application

You need to deploy the generated Web View application to an Application Server. Use the EAR file created by the Build Tool under your model directory: \\<model directory>\java\delpoy.j2ee. Follow your Application Server's deployment instructions to actually deploy the EAR file.

You can invoke your generated application from a supported browser. The URL to access Web View applications will rely on the context specified plus the startup HTML filename. It should follow the following format:

```
<http://<hostname>/<context>/<startupfile>
```

The *web.xml* file is also used to modify the Session Timeout value, which is set to 30 minutes by default.

National Language Support (NLS)

All files generated for Web View are UTF-8 encoded. The model encoding is used to determine the translation used to convert the files to UTF-8.

The dialect name or culture language code is used as part of the context, for example:

```
http://hostname:portnumber/application_DIALECT/WIN(or COOP).trancode(or
loadmodulename)_DIALECT.TC(or LM).html
```

Therefore, NLS requires that the model contain a non-DEFAULT dialect definition.

Appendix A: Windows GUI Specific Features

This section contains the following topics:

Presentation-Related Functions

The following CA Gen presentation-related functions are not supported by browsers:

- Beep
- BackgroundColor
- ClearDisplayProperties
- ForegroundColor
- GetCaption
- GetHandle
- GetHeight
- GetLeft
- GetPromptHeight
- GetPromptLeft
- GetPromptTop
- GetPromptWidth
- GetTop
- GetVisible
- GetWidthMessageBoxBeep
- Mark Command
- Refresh
- RestoreDisplayProperties
- SaveDisplayProperties

- SetCaption
- SetFont
- SetFontDynamically
- SetHeight
- SetLeft
- SetPromptHeight
- SetPromptLeft
- SetPromptTop
- SetPromptWidth
- SetTop
- SetVisible
- SetWidth
- Unmark Command

Tips

Tips for the unsupported presentation-related functions are:

- All the Get and Set (for example, GetHeight, SetTop) functions use Windows API calls that need the HANDLE to a specific control or a window displayed on the desktop to get or set a presentation property of that window or control. These types of Windows API calls are not possible in a browser environment.
  - Dot notation cannot be used with Hypertext links the same as with HTMLText and HTMLControl, which were introduced in Advantage Gen 6.5.
  - The functions Beep and MessageBoxBeep emit a beeping sound on the speaker of the computer where the code is running. Since browsers normally run on a different computer than the Application Server, these features are not useful in browser clients. MessageBoxBeep, however, produces a message box from the browser window. Only the audio beep is not supported.
- The functions ClearDisplayProperties, RestoreDisplayProperties, and SaveDisplayProperties work with the registry entries of the specified window's display properties and use Windows APIs that need the handles of the registry keys. Any changes made to the registry keys only affect the display properties associated with windows on that machine. For browser clients, which normally reside on a different computer, these functions do not provide the desired effect.

- Additionally, some CA Gen Interface Object Methods and Properties are represented internally by Get or Set functions. Thus, they are not supported. The following CA Gen Interface Object Properties are not supported:
  - Caption
  - Enabled
  - Focus
  - FontSize
  - FontStyle
  - FontType
  - Groupbox Caption
  - Handle
  - Height
  - Left
  - Literal Value
  - Maximized
  - Minimized
  - Prompt
  - Prompt FontStyle
  - Prompt FontType
  - Prompt Height
  - Prompt Left
  - PromptFontSize
  - PromptTop
  - PromptWidth
  - Top
  - Visible
  - Width
- The following CA Gen Interface Object Methods are not supported:
  - Clear
  - Click
  - Copy
  - Cut

- Paste
- Redraw
- SetBitmapBackground
- SetSelection
- Undo
- Window.EnterableDropDownList
- Window.EnterableDropDownLists
- Window.EnterableListBox
- Window.EnterableListBoxes
- Window.OLEArea
- Window.OLEAreas
- Window.OLEControl
- Window.OLEControls
- OLE Functions

Message Box

Message boxes are asynchronous in behavior and you may see multiple message boxes displayed at the same time during Action Block or Event processing.

The function MessageBox can only be used in Web View applications after the primary window or primary dialog has been displayed. This function must not be invoked from Procedure Step action logic, from Open or Activate event actions, or from TIREVENTs, which are invoked before the window is displayed.

OLE Functions

The following CA Gen OLE functions are not supported, since Object Linking and Embedding (OLE) technology is not supported by browsers:

- CreateObject
- GetObject
- PrintWindow
- Quit
- RegCloseKeys

- RegCreateKey
- RegDeleteKey
- RegDeleteValue
- RegEnumKey
- RegOpenKey
- RegQueryValue
- RegSetValue

File-Related Functions

The following CA Gen functions are used to create, view, or update files and documents on the local or network drives, and are not supported:

- OpenExcelDocument
- OpenWordDocument
- UpdateExcelCell
- ValueFromExcelCell

The functions OpenExcelDocument, and OpenWordDocument cannot be supported because the client business logic runs on the Application server machine. Access to the file system of the browser's computer is considered a security violation. Also, some of these functions also use OLE objects, which are not supported.

Even though file functions, such as OpenTextFile, are enabled in Web View environments, they are not thread safe. Due to the inherent multi-threaded nature of Web View applications, care must be taken when using file functions. Since no locks are taken on resources accessed by file functions, upon accessing a file from a user session, other users will not be prevented from attempting to access the same resource. This poses potentials for collisions. Furthermore, references to these resources may not be preserved from one request to the next. You will need to Open, Access, and Close within one event action because the information would not be preserved on a second execution if the thread changed. To avoid side effects and collisions between different sessions, you must create a unique file per user session.

Where not supported, action diagram statements are ignored at runtime.

Window/Dialog Properties

The window's properties, such as initial position (Designed, Mouse Alignment, or System Placed) and style (System Menu, Minimize Button, Maximize Button, or Dialog Border) are not currently supported and such properties are ignored at runtime.

All the HTML pages for a given window or dialog have a common appearance based on a predefined style sheet (Cascading Style Sheet). Custom Cascading Style Sheets (CSS) are not supported in Web View.

Vertical menus are supported in Web View applications.

The title property of the window or dialog is used in the <TITLE> element of all HTML files generated for that window or dialog. The icon file associated with the window or dialog is not supported and the property is ignored.

Other Unsupported Features

The following functionalities are unsupported, or partially supported in Web View UI Generation:

1. Due to the limitations imposed by HTML, the following controls are not supported:
   - OLE Area
   - Enterable DropDown List
   - Enterable List
   - ASP.NET controls
2. The following functionalities associated with specific controls are not supported:
   - Disable By for Active X controls
   - Auto tabbing among Entry Fields
   - Extended Selection is only supported for list boxes on Internet Explorer
   - Accelerators for push buttons and Menu or Menu Items
3. The following table discusses the functioning of *Disabled By* logic in Mozilla Firefox and Internet Explorer:

Operation	**Mozilla Firefox**	**Internet Explorer**
Cut, Paste, Delete, and Undo from the Keyboard consistently (without using the Context menu)	*Disabled By* works as expected.	*Disabled By* works as expected.
Cut, Paste, and Undo from the Context menu consistently (without using the keyboard)	*Disabled By* does not work as expected.	*Disabled By* does not work as expected. The state changes once the element loses focus.
Delete by using the Context menu	*Disabled By* does not work as expected.	*Disabled By* does not work as expected. The state changes once the element loses focus.

Operation	Mozilla Firefox	Internet Explorer
Delete key on the keyboard to delete the contents of a control and then use Undo from the Context menu.	*Disabled By* does not work as expected.	*Disabled By* does *not* work as expected. The state changes once the element loses focus.
Select the data to be deleted, use the Delete key on the keyboard to delete the contents of a control, and then use the Undo key.	*Disabled By* does not work as expected.	*Disabled By* works as expected.

4. Tab sequencing among window controls is governed by the controls' order in the generated HTML file. Since a list box is implemented by HTML <TABLE> and Link (Action:) tags, sequencing does not include list boxes.

5. Recursive calls within action blocks (such as, an action block calling itself) are not supported and may result in runtime abends.

6. Multiple selection listbox behavior is not supported.

7. Varying size table is not supported. If such a table exists on a window, it will be rendered as a Fixed size table.

8. The CSSCLASS property is not supported in Web View applications.

9. The HideRadioButtonGroup method is not supported in Web View applications. Use the Visible property to hide the Radio Button Group controls.

10. Customized help and close buttons are not supported.

11. System menu does not provide minimize, maximize, and restore functionality.

12. Cross-Context Flows are not supported in Web View applications.

13. Copying generation configurations or model directories that include generation configurations is not supported.

14. Duplicating generation configurations is not supported.

15. In Web View applications, prompts are part of the list box and it is not possible to detach the prompts from the list box as done in other target environments. If you detach the prompts from the list box in the Navigation Diagram, the list box follows the prompts and it may overlap the controls that are placed between the prompts and the list box in the Navigation Diagram in Toolset.

16. You cannot use single-click and double-click events on the same control in Web View applications.

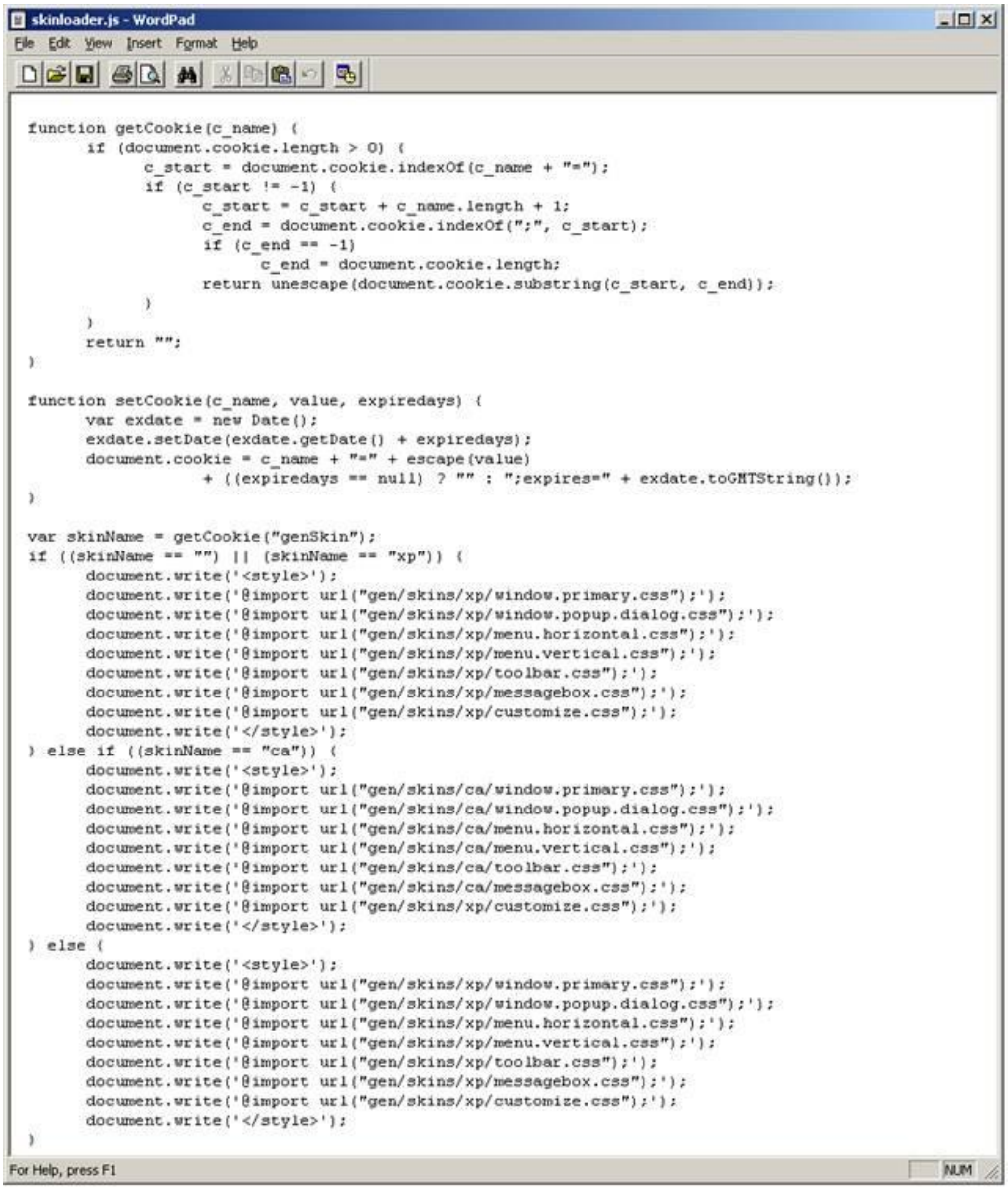
Appendix B: Application Skins

This section contains the following topics:

Select Files to be Loaded in a Skin

Extract the skinloader.js file that is located in the CA Gen installation directory:

<CA Gen Installation Directory>\Gen\webview\ui\WebViewUI.zip



```
function getCookie(c_name) {
        if (document.cookie.length > 0) {
                c_start = document.cookie.indexOf(c_name + "=");
                if (c_start != -1) {
                        c_start = c_start + c_name.length + 1;
                        c_end = document.cookie.indexOf(";", c_start);
                        if (c_end == -1)
                                c_end = document.cookie.length;
                        return unescape(document.cookie.substring(c_start, c_end));
                }
        }
        return "";
}

function setCookie(c_name, value, expiredays) {
        var exdate = new Date();
        exdate.setDate(exdate.getDate() + expiredays);
        document.cookie = c_name + "=" + escape(value)
                        + ((expiredays == null) ? "" : ";expires=" + exdate.toGMTString());
}

var skinName = getCookie("genSkin");
if ((skinName == "") || (skinName == "xp")) {
        document.write('<style>');
        document.write('@import url("gen/skins/xp/window.primary.css");');
        document.write('@import url("gen/skins/xp/window.popup.dialog.css");');
        document.write('@import url("gen/skins/xp/menu.horizontal.css");');
        document.write('@import url("gen/skins/xp/menu.vertical.css");');
        document.write('@import url("gen/skins/xp/toolbar.css");');
        document.write('@import url("gen/skins/xp/messagebox.css");');
        document.write('@import url("gen/skins/xp/customize.css");');
        document.write('</style>');
} else if ((skinName == "ca")) {
        document.write('<style>');
        document.write('@import url("gen/skins/ca/window.primary.css");');
        document.write('@import url("gen/skins/ca/window.popup.dialog.css");');
        document.write('@import url("gen/skins/ca/menu.horizontal.css");');
        document.write('@import url("gen/skins/ca/menu.vertical.css");');
        document.write('@import url("gen/skins/ca/toolbar.css");');
        document.write('@import url("gen/skins/ca/messagebox.css");');
        document.write('@import url("gen/skins/xp/customize.css");');
        document.write('</style>');
} else {
        document.write('<style>');
        document.write('@import url("gen/skins/xp/window.primary.css");');
        document.write('@import url("gen/skins/xp/window.popup.dialog.css");');
        document.write('@import url("gen/skins/xp/menu.horizontal.css");');
        document.write('@import url("gen/skins/xp/menu.vertical.css");');
        document.write('@import url("gen/skins/xp/toolbar.css");');
        document.write('@import url("gen/skins/xp/messagebox.css");');
        document.write('@import url("gen/skins/xp/customize.css");');
        document.write('</style>');
}
```

Add the CSS files that have been created to customize the behavior of the application. Add a line of code that will load your customized CSS file in the portion of the script that begins with “if ((skinName == “”) || (skinName == “xp)) {“. Add this line of code before the document.write(‘<style>’); line. The line of code necessary to load your customized CSS should be in the following format:

```
document.write(‘@import url(“gen/skins/xp/customize.css”);’);.
```

The values of the url parameter provide the location within the WAR file of your customized CSS file. Perform these steps for each customized CSS file needed.

Override Skins Using the Injection Technique

The WebViewUI.zip file that is located in the CA Gen installation directory includes the default skin for Web View applications. You can modify the default skin directly but we recommend that you replace the skin files that you wish to override using the injection technique. This technique keeps your skin overrides from being overwritten with updates to your CA Gen installation.

Use the injection technique either to replace individual skin files or to apply an entirely different skin while assembling the application.

Example: The example below details the steps required to override the skinloader.js file and add an additional CSS file named customize.css.

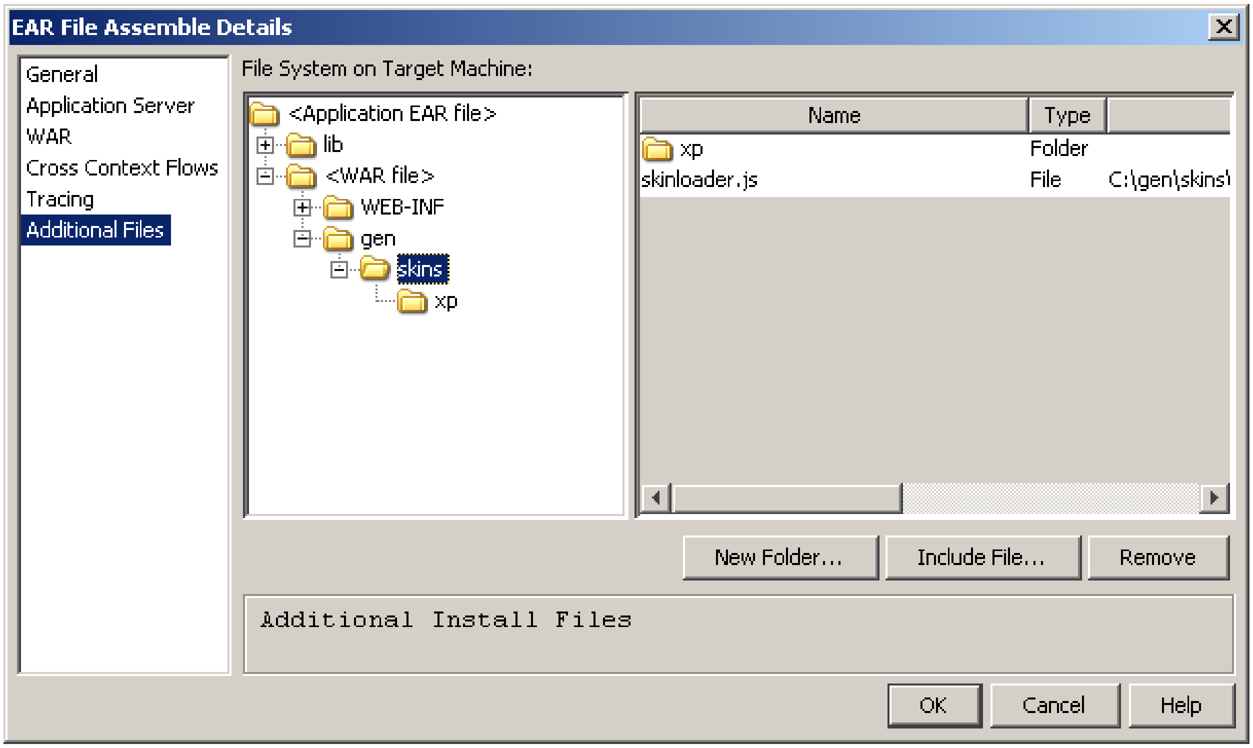
Follow these steps:

1. Select the appropriate load module in Build Tool
2. Right-click the load module and click Assemble.

   The EAR File Assemble Details dialog opens.
3. Click Additional Files in the left pane and provide the files needed for inclusion in the WAR file.
4. Select the <WAR file> node in the tree structure and click New Folder.
5. Type gen in the entry field and click OK.

6. Create a folder *skins* in the gen directory.

   This directory structure is necessary to properly override the skinloader.js file provided in the <CA Gen installation directory>\Gen\webview\ui\WebViewUI.zip file.



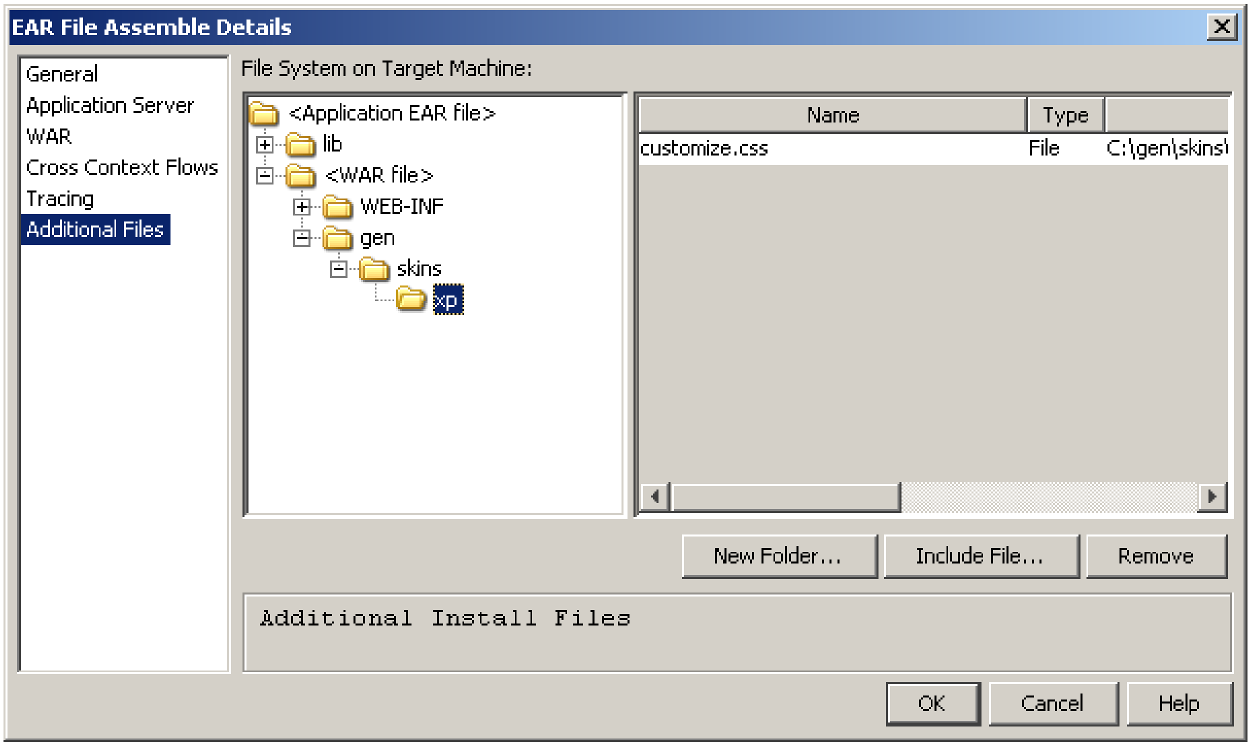
Name	Type	
xp	Folder	
skinloader.js	File	C:\gen\skins\

7. Select the skins folder and click Include File to add the modified skinloader.js file.

   The directory structure is created and the modified skinloader.js file is included in the WAR file.

8. Create the directory structure defined in the skinloader.js for the customized CSS files and include the customized CSS files.

   The directory structure is created and the CSS file is included as shown:



Name	Type	
customize.css	File	C:\gen\skins\

Additional Install Files

Appendix C: Messages

This section contains the following topics:

Message about Active Content

To help protect your security, Internet Explorer has restricted this webpage from running scripts or ActiveX controls that could access your computer.

Reason:

Web View HTML pages reference external templates.

Action:

Modify Internet Explorer settings to stop seeing this message.

Follow these steps:

1. Open Internet Explorer.
2. Click Tools, Internet Options.

   The Internet Options dialog opens.
3. Click the Advanced tab and scroll down to the Security settings.
4. Select the Allow active content to run in files on My Computer check box.

ERROR Writing Console Output - <path> (The system cannot find the path specified)

Reason:

This error message is displayed in the Console view if the path you specified in the Write to File field in the Generation Configurations dialog is not accessible.

Action:

Specify a different path that is accessible to save the generation output in a file.

Index