

Gentran Integration Suite™

Web Extensions Guide

Version 4.2

Sterling Commerce
An IBM Company

© Copyright 2006 Sterling Commerce, Inc. All rights reserved.
Additional copyright information is located on the Gentran Integration Suite Documentation Library:
<http://www.sterlingcommerce.com/Documentation/GIS42/homepage.htm>

Contents

Chapter 1 Web Extensions Basics 9

Web Extensions Features	9
Obtaining a License for Web Extensions	10
Web Extensions Components	10
Human Interaction Services	10
Web Templates	11
Web Template Designer	11
Web Template Translator	11
Web Template Wizard	11
Web Template Editor	12
File Upload Utility	12
Predefined Business Processes	12
.war File Generation Utility	13
How Web Extensions Works	13
Creating a Business Process For Web Extensions	13
Web Extensions Internationalization and Localization	14
Encoding	14
XML Resource Bundles	15
Role-Based Security	15
Role-Based Security Example	16

Chapter 2 Web Extensions Utilities 17

About the Web Template Designer	17
Requirements for Using the Web Template Designer	18
Installing the Web Template Designer	18
Removing the Web Template Designer	19
Starting the Web Template Designer	19
Starting the Web Template Designer Online	19

Starting the Web Template Designer Offline	19
Exiting the Web Template Designer	19
Navigating the Web Template Designer	20
Title Bar	20
Menu Bar	20
Tool Bar	21
Node Pane	23
Design Pane	23
Code Pane	23
Properties Pane	23
About the Web Template Translator	24
Starting the Web Template Translator	24
Exiting the Web Template Translator	24
Navigating the Web Template Translator	25
Translating Documents Using the Web Template Translator	26
HTML to XForms	27
XML to XForms.	27
About the Web Template Wizard.	30
Starting the Web Template Wizard	30
Exiting the Web Template Wizard.	31
Navigating the Web Template Wizard.	31
Translating DDF Files Using the Web Template Wizard.	31
About the File Upload Utility	32
Uploading Files Using the File Upload Utility	32
Customizing the File Upload Utility	33
Creating a Map	34
Editing the WebSuiteSkin.xrb File	34
Editing the FileUploadTemplate Web Template	34
Editing the FileUploadConfirmSend Web Template	35
Editing the FileUpload Business Process.	35
Editing the Web Extensions HTTP Server Adapter or HTTP Server Adapter	36
Accessing the Customized File Upload Utility.	36
About the Web Archive (.war) File Utility	37

Chapter 3 Creating Web Templates 38

About Web Templates	38
Creating a Web Template	39
Web Template Creation Process	40
Creating a Web Template Using an HTML Editor	41
Creating a Web Template Using the Web Template Designer Method	41
Adding the XForms Structure to the Web Template.	42
Adding the HTML Table to the Web Template	43
Adding the Content, XForms Controls, and HTML Elements to the Web Template.	44
Adding Buttons to the Web Template.	46
Saving, Previewing, Editing and Checking In the Web Template.	48
Creating a Web Template Using the Web Template Translator	49
Adding Buttons to the Web Template.	55
Creating a Web Template Using the Web Template Wizard	57
Adding Buttons to the Web Template.	59
Editing a Web Template	61

Editing a Web Template Using the Web Template Designer	61
Editing a Web Template Using the Web Template Editor.	63
HTML to XForms Code Samples.	63
Managing XForms Structure Elements	65
Managing XML Namespace Elements	66
Adding XML Namespaces	66
Managing XForms Models	67
Adding an XForms Model.	67
Editing an XForms Model Name.	67
Deleting an XForms Model Name	68
Managing XForms Instances	68
Editing an XForms Instance	68
Managing XForms Child Nodes	68
Adding an XForms Child Node	69
Editing an XForms Child Node	69
Deleting an XForms Child Node	70
Managing XForms Child Node Values	70
Adding an XForms Child Node Value.	70
Editing an XForms Child Node Value.	70
Deleting an XForms Child Node Value.	71
Managing XForms Binds.	71
Adding an XForms Bind	71
Editing an XForms Bind	71
Deleting an XForms Bind	73
Managing XForms SubmitInfo Elements.	73
Adding an XForms SubmitInfo Element	73
Editing an XForms SubmitInfo Element	74
Deleting an XForms SubmitInfo Element	74
Managing XForms Controls.	75
Managing XForms Text Field Controls	75
Adding an XForms Text Field	75
Editing an XForms Text Field	76
Deleting an XForms Text Field	76
Managing XForms Button Controls.	76
Adding an XForms Button	77
Editing an XForms Button.	77
Deleting an XForms Button	78
Managing XForms SelectOne Controls.	78
Adding an XForms SelectOne Control	79
Editing an XForms SelectOne Control	80
Deleting an XForms SelectOne Control	80
Managing XForms SelectMany Controls.	80
Adding an XForms SelectMany Control	81
Editing an XForms SelectMany Control	82
Deleting an XForms SelectMany Control	82
Managing XForms File Upload Controls.	82
Adding an XForms File Upload Control	82
Editing an XForms File Upload Control	83
Deleting an XForms File Upload Control	83
Managing HTML Elements	84
Managing HTML Images.	84

Adding an HTML Image	84
Editing an HTML Image	85
Deleting an HTML Image	85
Managing HyperLinks	85
Adding a Hyperlink	85
Editing a Hyperlink	86
Deleting a Hyperlink	86
Managing E-mail Links	86
Adding a E-mail Link.	87
Editing a E-mail Link.	87
Deleting an E-mail Link.	87
Managing HTML Tables	88
Adding an HTML Table.	88
Adding Rows and Columns to an HTML Table	89
Deleting Rows and Columns from an HTML Table	90
Editing and Formatting HTML Items	90
Copying and Pasting Items	90
Cutting and Pasting Items	91
Formatting Text	91
Applying a Font to Text.	91
Applying Bold and Italic Emphasis to Text	91
Applying a Point Size to Text	92
Using Tag Libraries with Web Extensions	92
JSP Tag Libraries	92
Before You Can Use JSP Tag Libraries	92
Run Business Process Tag Libraries	93
runBP	93
runBPnoReturnDoc	93
getDocuments	94
Business Process Resolver	94
Authorization (autho) Tag Library	96
Session Validation (validSession) Tag Library	96
Authentication (authenticate) Tag Library	97
Requirements For Using the JSTL	98
JSTL Concepts and Definitions.	98
Setting Up and Using JSTL.	99
Adding the JSTL to the Web Application.	99
Editing the web.xml File for JSTL	99
Adding the JSTL to a JSP Web Template.	101
Deleting JSTL Code from a JSP Web Template.	101
Update, Compile, and Redeploy the WAR file in Gentran Integration Suite	101
Deploying an External JSP Web Application on Gentran Integration Suite	102
Troubleshooting JSTL Issues	103
JSTL Demonstrations	103

Chapter 4 Managing Web Templates 105

Previewing a Web Template	105
Previewing Web Templates from the Web Templates List	106
Previewing Web Templates in the Web Template Editor	106
Previewing Web Templates Using the XForms Servlet	106

Previewing a Web Template Before Check-In	107
Checking In a Web Template	107
Checking Out a Web Template	108
Searching for a Web Template	109
Deleting a Web Template	110
Maintaining a Web Template Version History	110
Testing Web Templates.	111
Creating the WebTemplateTest Business Process.	112
Checking In the WebTemplateTest Business Process	113
Adding a URI to the Web Extensions HTTP Server Adapter	113
Testing a Web Template.	114
Preinstalled Web Templates	115

Chapter 5 Predefined Business Processes 120

About Predefined Business Processes	120
User Roles and Permissions	124
Expense Report Business Process	125
Expense Report Business Process Example	125
Purchase Requisition Business Process	126
Purchase Requisition Business Process Example	126
Time Sheet Business Process.	128
Time Sheet Business Process Example	128
Purchase Order Business Process	129
Purchase Order Business Process Example	130
Remittance Advice Business Process	131
Remittance Advice Business Process Example	131
Invoice Business Process	132
Invoice Business Process Example	132
Advance Ship Notice Business Process	134
Advance Ship Notice Business Process Example	134
Gentran Integration Suite Work Page	135
View Folders	136
Message List.	137
Search.	138
Administration	138
Create New	139
Accessibility Example	139
Managing the Gentran Integration Suite Work Page.	139
Opening the Gentran Integration Suite Work Page.	139
Changing the Gentran Integration Suite Work Page Display	140
Viewing Documents	140
Searching for Documents in Folders.	140
Changing a User Password	142
Creating and Submitting Documents	142
Approving or Rejecting Submitted Documents	143
Receiving E-mail Notifications of Received Documents	143
Archiving Documents	144
Restoring Documents	144
Selecting Documents for Deletion.	145
Permanently Deleting Documents.	145

Customizing the Gentran Integration Suite Work Page	145
Customizing Predefined Business Processes	146
Accessing Predefined Business Processes	146
Disabling a Predefined Business Process	146
Checking Out a Predefined Business Process	146
Editing a Predefined Business Process	147
Checking In a Predefined Business Process	147
Chapter 6 Deploying Web Applications	149
<hr/>	
About Web Applications	149
Editing the Web.xml File	150
Editing Web Template XML Files	150
Editing Web Template XML Resource Bundle (.xrb) Files	151
Creating and Checking In Web Templates	151
Creating Business Processes That Use the Web Template	152
Setting Permissions for the Web Template and the Business Process	152
Editing the Gentran Integration Suite Work Page	152
Adding Web Templates to the Create New Area	152
Adding Storage Areas to the View Folders Area	153
Managing Web Resources	154
About Resources	154
Web Resources	154
Resource Tags	154
Web Archive (.war) Files	155
Checking In a Web Resource	155
Checking Out a Web Resource	156
Searching for a Web Resource	157
Deleting a Web Resource	158
Maintaining a Web Resource Version History	158
Managing .war Files	159
Generating a Web Archive (.war) File	159
Deploying a Web Application In a DMZ	160
Deploying the XForms Servlet in a DMZ	160
Deploying xforms.war on Apache/Tomcat 3.2	161
Deploying webx.war on Apache/Tomcat 3.2	161
Deploying a .war File in a DMZ	162
Deploying a Web Application in a DMZ Using Perimeter Servers	163
Editing the B2B HTTP Server Adapter	163
Testing the Web Application	164
Index	165
<hr/>	

Web Extensions Basics

Web Extensions is the suite of utilities and services that enable human interaction within business processes while the business processes run in Gentran Integration Suite. Web Extensions also enables you to exchange business documents with your smaller trading partners.

This section covers the following topics:

- ◆ Web Extensions Features
- ◆ Obtaining a License for Web Extensions
- ◆ Web Extensions Components
- ◆ How Web Extensions Works
- ◆ Web Extensions Internationalization and Localization
- ◆ Role-Based Security

Web Extensions Features

Web Extensions features include:

- ◆ Services enabling the addition of human interaction to virtually any business process
- ◆ Translation of Web templates from HTML or XML to XForms format, from XForms to HTML format
- ◆ Translation from Gentran[®] DDF files to XForms format
- ◆ Creation, editing, and customization of Web templates to meet almost any need
- ◆ Support for Java Server Pages (JSP), when using perimeter services
- ◆ Support for localization, international standards, and double-byte character sets (DBCS)
- ◆ Preview of Web template functions before deployment
- ◆ Search and list features to locate specific Web templates
- ◆ Version control for Web templates to manage expanding Web template libraries
- ◆ Customizable predefined business processes using Web-based forms to enable exchange of data with smaller trading partners and between company departments
- ◆ E-mail notification of business processes waiting for human interaction

- ◆ Role-based security features to protect against unauthorized access

Obtaining a License for Web Extensions

To use Gentran Integration Suite Web Extensions, you must obtain a license. You must obtain a license from the Sterling Commerce Self-Service Web site at www.productupdates.stercomm.com.

Web Extensions Components

Web Extensions increases the reach of Gentran Integration Suite to include human interaction through designated Web sites. The Web Extensions suite uses the following tools and services to enable human interaction with the business process:

- ◆ Human Interaction services
- ◆ Web templates
- ◆ Web Template Designer
- ◆ Web Template Editor
- ◆ Web Template Translator
- ◆ Web Template Wizard
- ◆ File Upload utility
- ◆ Predefined business processes
- ◆ .war File Generation utility

Note: In Gentran Integration Suite, fields with colored field names are required. If you skip a required field, a message prompts you to supply the missing information.

Human Interaction Services

To enable human interaction with a business process, Gentran Integration Suite Web Extensions uses Human Interaction services. *Human interaction* is any action taken by a person on a business process while it runs.

An example of human interaction is when a business process contains a purchase order that requires a manager's approval. The business process runs until it comes to the service requiring the manager's approval, where the business process is placed into a waiting status. The manager either receives an e-mail notification of a pending document with a link to the document or simply logs in to a Web site that shows the documents requiring approval. The manager opens the document, views it through the Web browser, approves it making any comments needed, and submits the document using the Submit button. Once submitted, the approved purchase order continues to the next service in the business process.

Types of Human Interaction services are:

- ◆ Human Interaction Event service – Identifies business process data that requires human interaction, flags business processes that require some human action and creates references to the business process data to await human action.
- ◆ Human Interaction Query service – Searches the storage area containing the business processes flagged by the Human Interaction Event service for human interaction and retrieves the business process data for the Human Interaction XForms service.
- ◆ Human Interaction XForms service – Renders the Web templates to display the business process data passed from the Human Interaction Query service. The business process data is passed from the Human Interaction XForms service to the B2B HTTP Server adapter and displayed in HTML format through a Web browser. The Human Interaction XForms service also can accept data from a business process that was started with an HTTP request containing an XForms session parameter. The data is rendered in the appropriate Web template.
- ◆ Human Interaction Document Loader service – Receives a MinedDataId as input and uses this ID to determine which document to load into the business process.

Web Templates

Gentran Integration Suite Web Extensions uses Web templates to display business process data in a Web browser. A *Web template* contains markup tags that enable business process data to be displayed at any point in a business process. Web templates combine client- and server-side markups to enhance reusability by separating content from display. Within Gentran Integration Suite, the Human Interaction services process the Web template server-side markup tags, whereas the Web browser processes the Web template client-side markup tags.

Gentran Integration Suite Web Extensions includes many preformatted Web templates you can customize to save time and resources in implementing Web Extensions with your trading partners. For a list of preinstalled Web templates, see *Preinstalled Web Templates* on page 49.

Web Template Designer

As your business changes, you may need to add different business processes that require human interaction through Web templates. To help create and edit Web templates, Gentran Integration Suite Web Extensions includes the Web Template Designer. Using the Web Template Designer, you can create and edit Web templates and then preview them using the Web Template Editor.

The Web Template Designer includes the Web Template Translator and Web Template Wizard to help you create Web templates.

Web Template Translator

Gentran Integration Suite requires Web templates to be in XForms format (.xfm file extension). To help with this requirement, Gentran Integration Suite Web Extensions includes the *Web Template Translator*, an application that converts a standard HTML form to XForms format, an XForms format document to HTML format, and an XML document to XForms format document. After a document is translated into XForms format and checked in to Gentran Integration Suite, the document is referred to as a Web template.

Web Template Wizard

As you extend your investment, you can use your existing Gentran DDF files in Gentran Integration Suite. To help with this requirement, Gentran Integration Suite includes the Web Template Wizard, which enables you to convert your existing DDF files and convert them to XForms format. For additional information about Gentran DDF files, see your Gentran product documentation.

Web Template Editor

As your business changes, you may need to edit your business processes. This may include changing the Web templates used by Gentran Integration Suite. To help edit your Web templates, Gentran Integration Suite Web Extensions includes the Web Template Editor. Using the Web Template Editor, you can create, edit, and preview Web templates.

File Upload Utility

Gentran Integration Suite provides the File Upload utility, enabling you to translate your documents from one data format to another data format through an associated map. Although the sample File Upload utility is for translating positional data into an XML data format, you can customize this utility to translate from many different formats into other data formats.

Predefined Business Processes

Gentran Integration Suite Web Extensions includes predefined business processes that are ready for use upon installation. Web Extensions predefined business processes are based on typical business scenarios. After you install Web Extensions, you can access the predefined business processes at <http://host:port/websuite>, where *host* is the name of the computer you installed Web Extensions on and *port* is the port number used when installing Gentran Integration Suite. Predefined business processes are either buyer-to-supplier oriented or employee-to-manager oriented.

Buyer-to-Supplier predefined business processes are:

- ◆ Purchase Order
- ◆ Invoice
- ◆ Remittance Advice
- ◆ Advanced Ship Notice

Employee-to-Manager predefined business processes are:

- ◆ Expense Report
- ◆ Purchase Requisition
- ◆ Time Sheet

Data Synchronization predefined business processes are:

- ◆ Subscriber Approve or Reject Process
- ◆ GSS

.war File Generation Utility

As your business changes, you may need to deploy additional Web applications in different environments. To help with this requirement, Gentran Integration Suite provides the .war file generation utility, enabling you to generate Web Archive (.war) files of your Web applications. A *.war file* is a compressed file that contains the directory structures and file contents of a Web application.

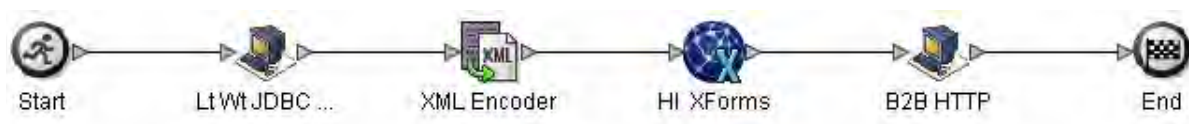
Generating a .war file enables you to:

- ◆ Deploy Web applications easily in a test environment or a production environment.
- ◆ Deploy Web applications easily in a Demilitarized Zone (DMZ) between security fire walls.

How Web Extensions Works

Gentran Integration Suite Web Extensions works by enabling people to make decisions and take actions within business processes while Gentran Integration Suite runs the business processes.

For example, a business process contains a purchase order Web template that is partially populated with data from a database and displayed for the user to complete. This is only one example of how Web Extensions works. The following figure shows a basic business process that queries a database using the Lightweight JDBC™ adapter, creates an XML document containing the query results using the XML Encoder service, locates the appropriate template to populate and displays the information using the Human Interaction XForms service, and displays the Web template with data in a Web browser using the B2B HTTP Server adapter.



Creating a Business Process For Web Extensions

The following table describes how to create Web templates and business processes to use in Web Extensions:

Step	Description
1	The business analyst analyzes the situation and uses the Graphical Process Modeler to create the business process containing the purchase order approval process.
2	The business analyst provides the appropriate business process information to the Web designer and requests that a purchase order Web template and any other Web templates be created for the business process. For more information, see <i>Creating Web Templates</i> on page 41

Step	Description
3	<p>Based on the information supplied by the business analyst, the Web designer creates the Web templates using one of the following methods or a combination of methods:</p> <ul style="list-style-type: none"> ◆ An HTML editor ◆ A text editor ◆ The Web Template Translator ◆ The Web Template Designer <p>For more information, see:</p> <ul style="list-style-type: none"> ◆ <i>Creating Web Templates</i> on page 41 ◆ <i>Web Extensions Utilities</i> on page 17 ◆ www.w3.org
4	<p>After the Web templates are created, the Web designer checks the Web templates in to the Gentran Integration Suite database, enabling them for use in the business process. For more information, see <i>Managing Web Templates</i> on page 39</p>
5	<p>The business analyst opens the Graphical Process Modeler and modifies and tests the configuration of the Human Interaction services, assigning the appropriate templates to each service.</p>
6	<p>After testing, the business analyst deploys the business process in the live environment.</p>

Web Extensions Internationalization and Localization

Web Extensions supports multiple languages (localization) and multiple regional data formats (internationalization). To do so, Web Extensions uses encoding and XML resource bundles.

Encoding is the representation of data in a particular character set. A *character set* is a list of characters (letters, numbers, and symbols such as #, \$, and &) that are recognized by computer hardware and software. For a list of encoding sets, see the *Installation Guide*.

XML resource bundles are groups of XML properties bound together in a package for easy deployment. There is a separate XML resource bundle for each language. For example, there are separate XML resource bundles for English and Japanese. Each XML resource bundle contains the XML properties to localize the date and time, numbers, and currency formats to a specific country or regional format.

Encoding

Web Extensions uses Unicode Transformation Format 8 (UTF-8) to define character sets. *UTF-8* is an encoding set that encompasses every character from the world's major languages and is a subset of *Unicode*, a standard that defines symbols in many of the world's languages. In Unicode, each character is assigned a unique number string, a code point. In UTF-8 each code point is transformed into a unique sequence of values that define the bytes that make up the character.

In addition to UTF-8, Web Extensions supports specific encoding sets that support double-byte character set (DBCS). *Double-byte character set (DBCS)* (a string of 16 numbers) characters instead of single byte (a

string of 8 numbers) characters. DBCS enables correct handling and display of different languages requiring more than the 256 ANSI-supported characters, such as Japanese, Chinese, and other Asian languages.

For example, the Latin capital letter A in ASCII is represented by different numbers in Unicode, Big5, and Shift-JIS. The following table shows examples of the differences in character sets:

Character Set	Code Definition of Latin Capital Letter A
ASCII	0X41
Unicode	0X0041
Big5 (Chinese)	0XA2CF
Shift-JIS (Japanese)	0X41

XML Resource Bundles

Web Extensions uses XML resource bundles to localize Web templates for different countries and regions. Each XML resource bundle contains the XML properties to localize the date and time, numbers, and currency formats to a specific country or regional format. The XML resource bundle is referenced during the display of the Web template in the Web browser to provide the localized data, which is displayed in the language specified by the Web browser. The following table shows examples of how data can be localized:

Date Type	U.S. Format (en-US XML resource bundle)	German Format (de XML resource bundle)
Currency	Dollar = \$	Euro = €
Date	July 24, 2002	24 Juli, 2002
Time	5:00 p.m.	17:00 hours

Role-Based Security

Role-based security provides access to specific files, business processes, Web templates, services, and product features, according to the permissions associated with the user account.

A *user account* contains the groups the user belongs to and the permissions associated with each group or user. *Permissions* provide access to the different modules within Gentran Integration Suite and are the foundation of role-based security.

The user account is linked to your Gentran Integration Suite user name and password. Each time you log in, Gentran Integration Suite verifies you are a valid user and grants access to only the areas you are allowed to use based on the permissions assigned to you by the Gentran Integration Suite system administrator.

Role-based security also helps direct messages and documents to the appropriate user within Gentran Integration Suite Web Extensions. In this circumstance, the Web Extensions Human Interaction Event

service pulls the approving authority's name and e-mail address from the Gentran Integration Suite database and routes the document to that person. This routing feature helps you manage message queues.

Role-Based Security Example

Jack, the Gentran Integration Suite system administrator, sets up user account profiles assigning groups and permissions to two employees, Tom and Sue. Jack sends Tom and Sue unique user names and passwords to access Gentran Integration Suite.

Tom is an employee whose manager is Sue. Tom logs in to Gentran Integration Suite using the user name and password that Jack provided and is verified as a valid user. Tom's Gentran Integration Suite work page opens, showing his Gentran Integration Suite message box options such as Inbox and Outbox and the Web templates that he can use, such as the Expense Report, Time Sheet, and Purchase Requisition. This means that Tom has write permissions to these three Web templates and can create an expense report and submit it to his manager for approval.

Sue, Tom's manager, logs in to Gentran Integration Suite using the user name and password that Jack provided and is verified as a valid user. Sue's Gentran Integration Suite work page does not show any Web templates under the Create New area, because Sue's user profile does not give her permission to edit any Web templates or business processes. She can view the expense reports, time sheets, and purchase requisitions submitted by Tom, approve or reject reports submitted by Tom, but she cannot edit the reports except for providing comments.

Although both Tom and Sue are Gentran Integration Suite Web Extensions users, both cannot perform the same tasks. The tasks they can perform are different based on the permissions provided to them by Jack, the system administrator.

Web Extensions Utilities

Gentran Integration Suite Web Extensions provides a variety of utilities to help you create Web templates and manage Web resources.

The Web Template Translator and the Web Template Wizard are Web Extensions utilities that enable you to convert HTML forms, XML documents, or DDF files to the XForms format. The File Upload utility enables you to map files easily and quickly by uploading them to a business process. The Web Archive File utility enables you to create a Web archive file of resources for deployment in another environment.

This section covers the following topics:

- ◆ About the Web Template Designer
- ◆ About the Web Template Translator
- ◆ About the Web Template Wizard
- ◆ About the File Upload Utility
- ◆ About the Web Archive (.war) File Utility

About the Web Template Designer

The *Web Template Designer* is a utility that enables you to create and edit a Web template using a combination graphical and code display. The following utilities are included with the Web Template Designer:

- ◆ Web Template Translator – Utility that translates XML and HTML documents to XForms format, and XForms to HTML format.
- ◆ Web Template Wizard – Utility that translates one way from Gentran DDF files to XForms format.

You download the Web Template Designer from a Web server and cache it on the client computer. When the resource files are available on the client computer, you can open the Web Template Designer quickly and whenever you need it.

Gentran Integration Suite uses Java™ Web Start to deploy the Web Template Designer. Java Web Start and the Web Template Designer are downloaded from a secure site. When you open the Web Template Designer, Java Web Start checks the Web server to verify whether a new version of the Web Template

Designer is available. When a new version is available, Java Web Start automatically updates the files on the client computer and opens the Web Template Designer.

Requirements for Using the Web Template Designer

Java Web Start requires approximately 5.5 MB of disk space on the client computer. The minimum recommended available disk space for Java Web Start, the Web Template Designer, and Web templates is 100 MB.

Installing the Web Template Designer

You must install Java Web Start only once. If you have already installed it for the Graphical Process Modeler or another tool in Gentran Integration Suite, you only need to click **Go!** next to Run the Web Template Designer in the Web Template Designer area to start the designer.

To download Java Web Start and install the Web Template Designer:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, in the Web Template Designer area, click **Download** next to Java Web Start.
3. In the **File Download** dialog box, click **Save**. The **Save As** dialog box opens.
4. Select the directory on the client computer that you want to save Java Web Start to and click **Save**. The download starts.

The download may take several minutes, depending on the speed of your connection. When the download completes, close the download dialog box if it remains open.

5. Open the directory that contains the downloaded Java Web Start file.
6. Double-click the file **javaws-1_0_1_02-win-int-rt.exe** to start the installation program for Java Web Start.
7. After reviewing the license agreement, click **Accept**.
8. In the **Installation Directory** dialog box, either accept the default directory or click **Browse** to select another directory, and click **Next**.

If you receive a message that setup was unable to detect a usable Java 2 Runtime Environment, either accept the default installation directory or click **Browse** to select another directory.

9. Do you want to restart your computer?
 - ◆ If Yes, click **Yes**.
 - ◆ If No, click **No**. The system updates are completed the next time you restart your computer.

When the installation is complete, the setup program prompts you to read the Readme file.

10. Return to the Web Utilities page in Gentran Integration Suite.
11. In the Web Template Designer area, click **Go!** next to Run the Web Template Designer. This may take a few minutes.

If you receive a security warning, ignore it and click **Start** to start the Web Template Designer.

Removing the Web Template Designer

To remove the Web Template Designer:

1. Open Java Web Start.
2. In the Java Web Start Application Manager window, select the designer to remove.
3. From the **Application** menu, select **Remove Application**. The designer is removed from Java Web Start.

To reinstall the XForms designer, follow the instructions in *Installing the Web Template Designer* on page 18.

Starting the Web Template Designer

You can start the Web Template Designer online or offline.

Starting the Web Template Designer Online

To start the Web Template Designer through Gentran Integration Suite:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Template Designer section, click **Go!**

Starting the Web Template Designer Offline

To start the Web Template Designer through Java Web Start:

1. From the Windows **Start** menu, select **Programs > Java Web Start > Java Web Start**.
2. From the **View** menu, select **Downloaded Applications** to view installed instances of the Web Template Designer.
3. Select the Web Template Designer application and click **Start**.

Exiting the Web Template Designer

Whether you have started the Web Template Designer online through Gentran Integration Suite or offline through Java Web Start, you exit the Web Template Designer in the same way.

To exit the Web Template Designer:

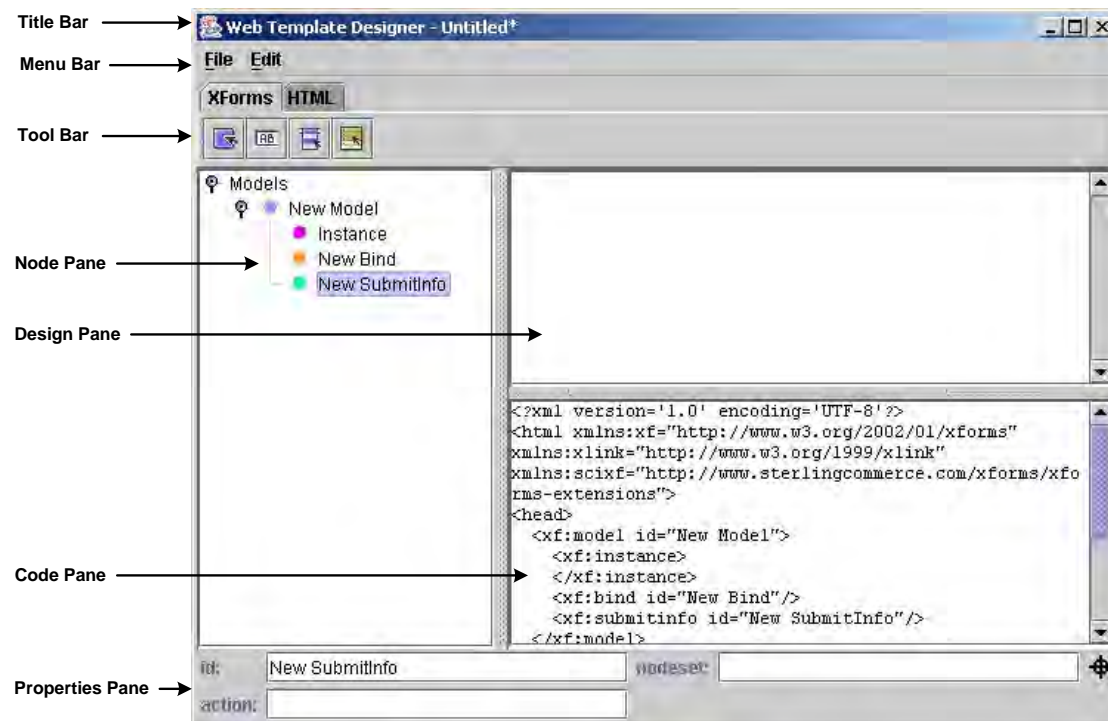
1. From the **File** menu, select **Exit**.

If you have not saved changes to a Web template, the Web Template Designer prompts you to save the changes.

2. Complete one of the following actions:
 - ◆ To save the changes, review the Web template for accuracy, give the Web template a name and click **Yes**.
 - ◆ To exit the Web Template Designer without saving changes to the Web template, click **No**.
 - ◆ To return to the Web Template Designer interface, click **Cancel**.

Navigating the Web Template Designer

The following figure shows the Web Template Designer interface:



Title Bar

The title bar displays the file name of the Web template currently open in the Web Template Designer.

Menu Bar

The menu bar complies with most Windows menu conventions. The following table describes the menu commands available in the Web Template Designer:

Menu	Description
File	<p>The File menu enables you to:</p> <ul style="list-style-type: none"> ◆ Create a new Web template. ◆ Open an existing Web template. ◆ Close and save a Web template. ◆ Quit the Web Template Designer.

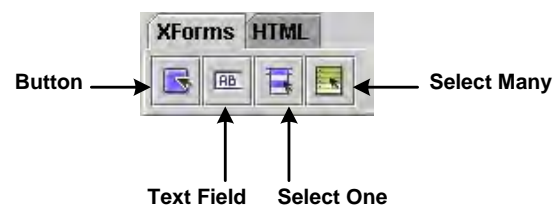
Menu	Description
Edit	<p>The Edit menu enables you to:</p> <ul style="list-style-type: none"> ◆ Copy items. ◆ Cut items. ◆ Paste items. ◆ Delete items. ◆ Undo the last 1000 actions, individually. ◆ Redo actions 1000 actions, individually. ◆ Refresh the design view. <p>Note: You cannot undo actions after you edit code in the code pane or after you use the refresh option on the Edit menu to refresh the design pane.</p>
Tools	<p>The Tools menu enables you to:</p> <ul style="list-style-type: none"> ◆ Run the Web Template Translator. ◆ Run the Web Template Wizard.

Tool Bar

The tool bar displays buttons that provide quick access to the XForms and HTML elements such as drop-down lists and buttons.

XForms Tab

The following figure shows the XForms control icons found on the XForms tab:



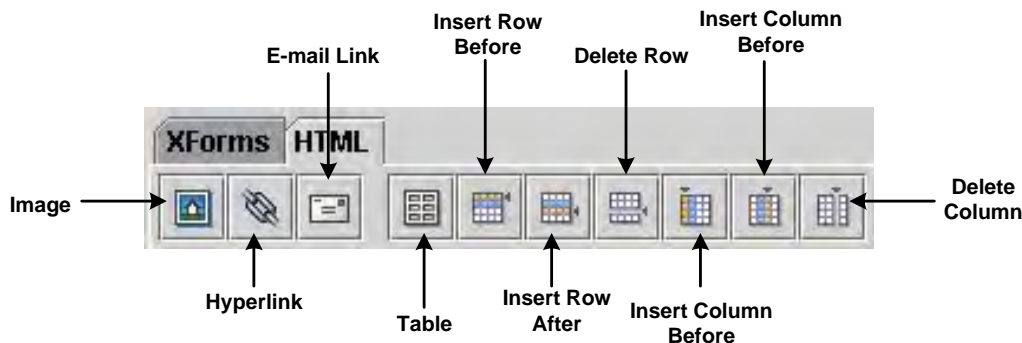
The following table describes each XForms control found on the XForms tab:

Object Name	XForms Tag	Equivalent HTML Tag	Description
TextField	<xf:input> </xf:input>	<input type="text"> </input>	Inserts a text box and the <xf:input> code into the Web template.
Button	<xf:button> </xf:button/>	<input type="button">	Inserts a button and the <xf:button> code into the Web template.

Object Name	XForms Tag	Equivalent HTML Tag	Description
SelectOne	<xf:selectOne> </xf:selectOne>	<select> </select>	Inserts a drop-down list allowing only one entry to be selected and the <xf:selectOne> code into the Web template.
SelectMany	<xf:selectMany> </xf:selectMany>	<select multiple> </select>	Inserts a drop-down list allowing more than one entry to be selected and the <xf:selectMany> code into the Web template.

HTML Tab

The following figure shows the HTML element icons found on the HTML tab:



The following table describes each of the HTML elements on the HTML tab:

Object Name	HTML Tag	Equivalent XForms Tag	Description
Hyperlink	<a>link name	Uses HTML tags	Inserts a link and the link code into the Web template.
Image		Uses HTML tags	Inserts an image into the Web template.
E-mail Link	Test	Uses HTML tags	Inserts a link to an e-mail address enabling a user to send e-mail to the address by clicking the link.
Table	<table> <tr> <td></td> </tr> </table>	Uses HTML tags	Inserts an HTML table that can be used for designing the Web template and organizing information.
Insert Row Before	<tr> </tr>	Uses HTML tags	Inserts a row in the table above the row containing the insertion point.

Object Name	HTML Tag	Equivalent XForms Tag	Description
Insert Row After	<tr> </tr>	Uses HTML tags	Inserts a row in the table below the row containing the insertion point.
Delete Row	N/A	N/A	Deletes the row containing the insertion point.
Delete Column	N/A	N/A	Deletes the column containing the insertion point.
Insert Column After	<td> </td>	Uses HTML tags	Inserts a column in the table to the right of the column containing the insertion point.
Insert Column Before	<td> </td>	Uses HTML tags	Inserts a column in the table to the left of the column containing the insertion point.

Node Pane

The node pane displays the XForms nodes that make up the Web template. A *node* is an element of a document structure. Click a node to view the configurable properties of the specific node. In the node pane, you can add, edit, or delete:

- ◆ *XForms models* – XForms elements that identify what the Web template does and how the Web template is presented.
- ◆ *XForms child nodes* – XForms elements of an XForms instance or another XForms child node.
- ◆ *XForms child node values* – Initial values of XForms child nodes that display in the Web template.
- ◆ *XForms submitInfo elements* – XForms element that sets an action associated with the XForms model.
- ◆ *XForms bind elements* – XForms element that connects an XForms model and a form control in a Web template.

Design Pane

The design pane displays the graphical representation of the Web template objects. You can drag objects into this pane to change the layout of the Web template. When you make a change in the design pane, the code pane changes to reflect the change in the design pane. To refresh the design pane you can either:

- ◆ Click inside the design pane.
- ◆ Click **Edit > Refresh Design View**.

Code Pane

The code pane displays the source code of the Web template. Use this pane to make changes directly in the code. If you make changes in the code pane, the design pane changes to reflect the changes when you click inside the design pane or select Refresh Design View from the Edit menu.

Properties Pane

The properties pane displays the properties associated with a particular XForms element or node. You can type changes to the XForms element properties directly into the properties pane.

About the Web Template Translator

The *Web Template Translator* is a component of the Web Template Designer that converts HTML forms into Web templates in the XForms format and back to HTML forms, and converts XML documents into Web templates in the XForms format. When you download the Web Template Designer to your computer, the Web Template Translator is downloaded as a component of the Web Template Designer to your computer.

Starting the Web Template Translator

To start the Web Template Translator:

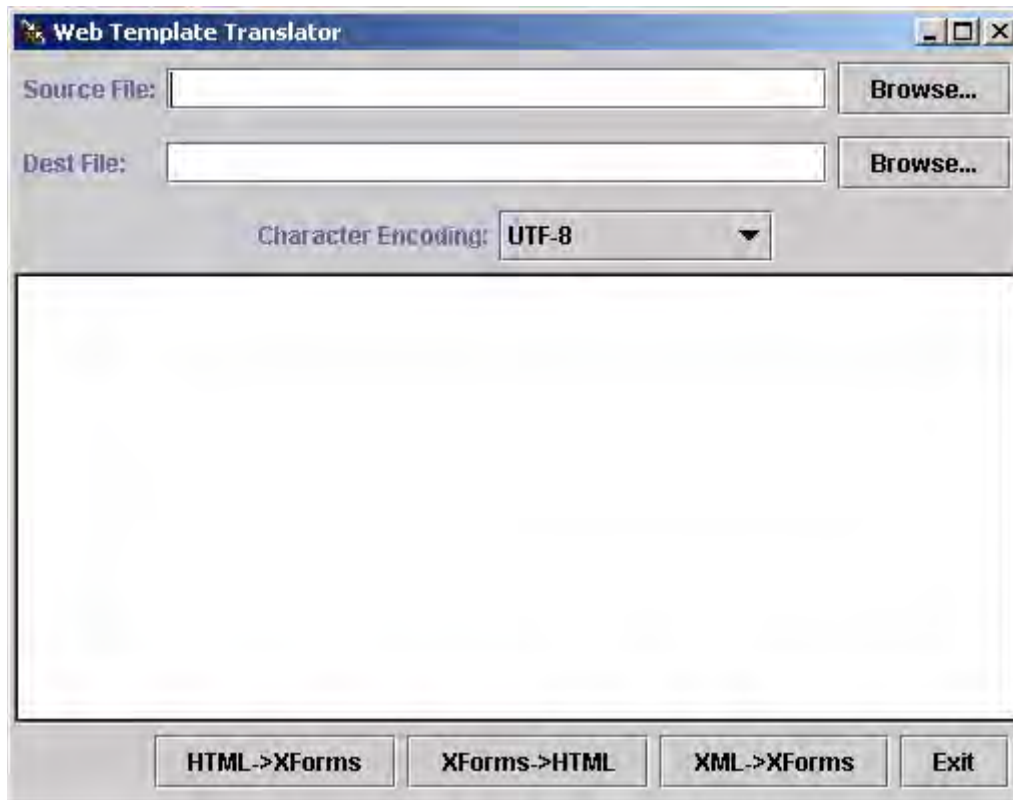
1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, in the Web Template Designer area, click **Go!**
3. In the Web Template Designer, from the **Tools** menu, select **Run Web Template Translator**.

Exiting the Web Template Translator

To exit the Web Template Translator, click the Close button in the upper-right corner of the Web Template Translator dialog box.

Navigating the Web Template Translator

The following figure shows the Web Template Translator interface:



The following table explains the Web Template Translator interface options:

Field or Button	Description
Source File	Location and file name of the file to be translated. Type the location and file name or click Browse to locate the file. Valid file types are .HTML, .xfrm, or .xml.
Dest File	Location and file name that will be the new translated file. Type the location and file name or click Browse to locate a previously translated document. If an existing file is chosen as the destination file, the new file overwrites the existing file.
Character Encoding	Encoding type of the source file. The translated .xfrm file has the encoding type UTF-8, which is the default for Gentran Integration Suite.
Output space	This unlabeled pane shows the results of the translation. This pane does not show the actual code; it shows statements about the translation, errors, or warnings that occurred during translation, and corrective actions to take.
HTML>XForms	Translates an HTML document into an XForms document. Ensure that the source file is an .htm file and the destination file is an .xfrm file.

Field or Button	Description
XForms>HTML	Translates an XForms document into an HTML document. Ensure that the source file is an .xfrm file and the destination file is an .htm file.
XML>XForms	Translates an XML document into an XForms document. Ensure that the source file is an .xml file and the destination file is an .xfrm file.
Exit	Closes the Web Template Translator.

Translating Documents Using the Web Template Translator

The Web Template Translator enables you to translate HTML documents to XForms documents, XForms documents to HTML documents, and XML documents to XForms documents.

To translate a document using the Web Template Translator:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, in the Web Template Designer area, click **Go!**
3. In the Web Template Designer, from the **Tools** menu, select **Run Web Template Translator**.
4. In the **Source File** field, type or browse to the file name of the file to be translated.

If you are translating from:

- ◆ HTML, ensure that the source file has an .htm or .HTML extension.
- ◆ XForms, ensure that the source file has an .xfrm extension.
- ◆ XML, ensure that the source file has an .xml extension.

5. In the **Dest File** field, type the file name that you want for the translated file or browse to the file name of a previously translated file that will be replaced by the newly translated document.

If you are translating to:

- ◆ HTML, ensure that the destination file has an .htm or HTML extension.
- ◆ XForms, ensure that the destination file has an .xfrm extension.

6. Depending on your task, complete one of the following actions:
 - ◆ To translate an HTML document to an XForms document, click **HTML>XForms**.
 - ◆ To translate an XForms document to an HTML document, click **XForms>HTML**.
 - ◆ To translate an XML document to an XForms document, click **XML>XForms**.

The results of the translation display in the output space of the translator.

7. Check the output space for any errors or warnings that occurred during the translation. If an error occurred, make the suggested corrections and retranslate the document, repeating steps 1 - 6.

You are now ready to complete one of the following tasks:

- ◆ Check the Web template in to Gentran Integration Suite.

This option is only available if you translated from XML to XForms. If you translated from HTML to XForms, you must provide additional XForms code before deploying the Web template in Gentran

Integration Suite. For more information, see *Managing XForms Structure Elements* on page 65 and *Managing XForms Controls* on page 75.

- ◆ Open the XForms file in the Web Template Designer and type the additional XForms code to complete the translation to a deployable Web template.

The following sample code shows an HTML to XForms example and an XML to XForms example.

HTML to XForms

HTML document before translation:

```
<HTML>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/HTML; charset=iso-8859-1">
</head>
<p>Before translation using the Web Template Translator</p>
<body bgcolor="#FFFFFF" text="#000000">
</body>
</HTML>
```

HTML document after translation:

```
<?xml version="1.0" encoding="UTF-8"?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink">
<head>
<title>
Untitled Document
</title>
<meta http-equiv="Content-Type" content="text/HTML; charset=iso-8859-1"/>
</head>
<body bgcolor="#FFFFFF" text="#000000">
<p>After translation using the Web Template Translator</p>
</body>
</HTML>
```

XML to XForms

XML document before translation:

```
<?xml version="1.0" ?>
<document type="Purchase Requisition">
- <!-- COMMENT-->
<info>
<EmployeeName>Tom Cool</EmployeeName>
<CostCenter>900800701</CostCenter>
<date>2002-06-23</date>
<comments>Please rush order.</comments>
</info>
<BillingAndShipping>
<Billing>
<BillTo>Test Company</BillTo>
<Address>
<StreetLine1>101 Back North 40</StreetLine1>
```

```

<StreetLine2 />
</Address>
<GeographicLocation>
<CityName>Middle Of Nowhere</CityName>
<StateOrProvinceCode>TX</StateOrProvinceCode>
<PostalCode>99988</PostalCode>
<Country caption="">US</Country>
</GeographicLocation>
</Billing>
<Shipping>
<ShipTo>Winter Again, Inc</ShipTo>
<Address>
<StreetLine1>1500 Cool St.</StreetLine1>
<StreetLine2 />
</Address>
<GeographicLocation>
<CityName>Anytown</CityName>
<StateOrProvinceCode>CA</StateOrProvinceCode>
<PostalCode>55566-4444</PostalCode>
<Country caption="">US</Country>
</GeographicLocation>
</Shipping>
</BillingAndShipping>
</document>

```

XML document after translation to XForms:

```

<?xml version="1.0" encoding="UTF-8"?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:scixf="http://www.sterlingcommerce.com/xforms/xforms-extensions">
  <head>
    <link rel="StyleSheet" type="text/css" href="/websuite/HTML/admin.css"/>
    <link rel="stylesheet" type="text/css" href="/websuite/HTML/templates.css"/>
    <xf:model id="skin">
      <xf:instance xlink:href="/websuite/xrb/WebSuiteSkin.xrb"
scixf:preview-href="/websuite/xrb/WebSuiteSkin.xrb"/>
    </xf:model>
    <xf:model id="myModel">
      <xf:instance xlink:href="inline" scixf:preview-href="inline">
<document type="Purchase Requisition">
<!--COMMENT-->
  <info>
    <EmployeeName>Tom Cool</EmployeeName>
    <CostCenter>900800701</CostCenter>
    <date>2002-06-23</date>
    <comments>Please rush order.</comments>
  </info>
  <BillingAndShipping>
    <Billing>
      <BillTo>Test Company</BillTo>
      <Address>
        <StreetLine1>101 Back North 40</StreetLine1>
        <StreetLine2/>
      </Address>
      <GeographicLocation>
        <CityName>Middle Of Nowhere</CityName>

```

```
        <StateOrProvinceCode>TX</StateOrProvinceCode>
        <PostalCode>99988</PostalCode>
        <Country caption="">US</Country>
    </GeographicLocation>
</Billing>
<Shipping>
    <ShipTo>Winter Again, Inc</ShipTo>
    <Address>
        <StreetLine1>1500 Cool St.</StreetLine1>
        <StreetLine2/>
    </Address>
    <GeographicLocation>
        <CityName>Anytown</CityName>
        <StateOrProvinceCode>CA</StateOrProvinceCode>
        <PostalCode>55566-4444</PostalCode>
        <Country caption="">US</Country>
    </GeographicLocation>
</Shipping>
</BillingAndShipping>
</document>
    </xf:instance>
</xf:model>
    <title>XML to XForms Translation</title>
</head>
<body class="MAINBODY">
    <xf:group id="xform-body" model="myModel">
        <table> The Code Sample continues with a table.
```

The Web template is similar to the following figure:

document	
info	
EmployeeName	Joe Cool
CostCenter	900800701
date	2002-06-23
comments	Please rush order.
BillingAndShipping	
Billing	
BillTo	Test Company
Address	
StreetLine1	101 Back North 40
StreetLine2	
GeographicLocation	
CityName	Middle Of Nowhere
StateOrProvinceCode	TX
PostalCode	99988
Country	US
Shipping	

About the Web Template Wizard

The Web Template Wizard translates one way from Gentran DDF files to XForms format. The XForms document created during translation requires additional XForms code to be added for use with Gentran Integration Suite. You can use the Web Template Designer to add the additional XForms code to the document.

For more information about editing an Web template, see *Editing a Web Template* on page 61.

Starting the Web Template Wizard

To start the Web Template Wizard:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, in the Web Template Designer area, click **Go!**
3. In the Web Template Designer, from the **Tools** menu, select **Run Web Template Wizard**.

Exiting the Web Template Wizard

To exit the Web Template Wizard, click the Close button in the upper-right corner of the Web Template Wizard dialog box.

Navigating the Web Template Wizard

The following table explains the Web Template Wizard interface options:

Field or Button	Description
Source File	Location and file name of the file to be translated. Type the location and file name or click Browse to locate the file. Valid source files are Gentran DDF files.
Dest File	Location and file name of the newly translated file. Type the location and file name or click Browse to locate a previously translated document. If an existing file is chosen as the destination file, the new file overwrites the existing file.
Output space	This unlabeled field shows the results of the translation. This field does not show the actual code; it shows statements about the translation, errors, or warnings that occurred during translation, and corrective actions to take.
DDF>XForms	Translates a DDF file into an XForms document. Ensure that the source file is a DDF file and the destination file is an .xfrm file.
Exit	Closes the Web Template Wizard.

Translating DDF Files Using the Web Template Wizard

The Web Template Wizard enables you to translate DDF files to XForms documents.

Note: If the DDF file has several repeating groups and segments, the Web Template Wizard creates several XForms documents from the initial DDF file. Each separate XForms document displays a layer of data that is nested within the single DDF file.

To convert a DDF file to XForms:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, in the Web Template Designer area, click **Go!**
3. In the Web Template Designer, from the **Tools** menu, select **Run Web Template Wizard**.
4. In the **Source File** field, type or browse to the file name of the DDF file to be translated.

5. In the **Dest File** field, type the file name of the translated document or browse to the file name of a previously translated file that will be replaced by the newly translated document. Ensure that the destination file has an .xfm file extension.
6. Click **DDF > XForms** to complete the translation.
The results of the translation appear in the output space of the translator.
7. Check the output space for any errors or warnings that occurred during the translation. If an error occurred, make the suggested corrections and retranslate the document, repeating steps 1 - 6.

About the File Upload Utility

The *File Upload utility* provided with Gentran Integration Suite Web Extensions enables you to upload to a business process file that requires mapping from one data format to another data format. For example, by default, the File Upload utility translates positional data into an XML data format.

You can use the File Upload utility to enable employees to upload files, or outside of your organization to enable your trading partners to send you any type of files.

The File Upload utility is installed with Gentran Integration Suite Web Extensions. To access the utility, point your Web browser to `http://host:port/websuite/xfm/FileUploadTemplate.xfm`.

To map data formats other than positional data to data formats other than XML, you will need to customize the sample File Upload utility. For more information, see *Customizing the File Upload Utility* on page 33.

Uploading Files Using the File Upload Utility

To upload a file using the File Upload utility:

1. Point your Web browser to the File Upload utility URL provided by your manager or system administrator. The default URL for the File Upload utility is:

`http://host:port/websuite/xfm/FileUploadTemplate.xfm`

For more information, see *Accessing the Customized File Upload Utility* on page 36.

2. In the **File name** field, type or browse to the file name of the file to be uploaded to the business process. The location of the sample data file provided is:

`install_dir/data/websuite/PositionalToXml`

3. Click **Submit** to upload the document to the business process for translation.
4. In the login page, type your user name and password that your manager provides.

Gentran Integration Suite verifies your user name and password against the database.

- ◆ If you do not log in correctly, you receive an error message. Verify your user name and password and try to log in again.
- ◆ If you log in correctly, the file is submitted, the Sending page is displayed in your Web browser, and you are returned to the File Upload Utility page.

5. From the Gentran Integration Suite **Business Process** menu, select **Monitor > Current Processes**.
6. In the ID column of the Monitor page, click the number identifying the instance of the business process that you uploaded the file to. Verify that the document was uploaded and assigned to the TranslatedData document in the process data.

Note: The sample predefined business process for the File Upload utility is FileUpload.bp. This business process translates positional data to XML data using the PositionalToXML map. To translate other types of data, you must customize the File Upload utility. For more information, see *Customizing the File Upload Utility* on page 33.

Customizing the File Upload Utility

You can customize the look of your File Upload utility to provide a branded image to your trading partners.

To customize the File Upload utility:

1. Create, compile, and check in a map.
2. Edit the WebSuiteKkin.xrb file.
3. Edit the FileUploadTemplate.xfm Web template.
4. Edit the FileUploadConfirmSend.xfm Web template.
5. Edit the FileUpload business process.
6. Edit the Web Extensions HTTP Server adapter or the HTTP Server adapter.

Note: If your Web templates are XForms, you can use either the HTTP Server adapter or the Web Extensions HTTP Server adapter. Use the HTTP Server adapter if your system is configured for perimeter servers. If your Web templates are JSPs, you must use the HTTP Server adapter, you cannot use the Web Extensions HTTP Server adapter.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

For each customization, you must edit and save the following files under different names:

- ◆ FileUpload.bp
- ◆ FileUploadTemplate.xfm
- ◆ FileUploadConfirmSend.xfm

For ease of identification, name all related files for each customization using a common syntax. For example, if you customize the File Upload utility to translate EDI data to XML data, you may save the files using the file names in the following table:

Sample File Name	Custom File Name
FileUpload.bp	EDItoXMLUpload.bp
FileUploadTemplate.xfm	EDItoXMLUpload.xfm
FileUploadConfirmSend.xfm	EDItoXMLConfirmSend.xfm

Creating a Map

Using the Gentran Integration Suite Map Editor, you can create, compile, and check in a map for the data formats that you are mapping.

Editing the WebSuiteSkin.xrb File

To edit the WebSuiteSkin.xrb file:

1. In a text editor, open the WebSuiteSkin.xrb file. The standard directory structure is:
applicationserverfolder/webapps/websuite/xrb/WebSuiteSkin.xrb
2. Copy the following code string:
`<phrase d="utility.name1">File Upload</phrase>`
3. Paste the code string you copied in step 2 directly below the line you copied the code string from.
4. In the code string you pasted:
 - ◆ Replace the number *1* after the name element with a different number you choose.
 - ◆ Replace *File Upload* with the name of the customized File Upload utility. For example, *EDI to XML File Upload*.
5. Save the file. Do not change the file name or the location.

Editing the FileUploadTemplate Web Template

To edit the FileUploadTemplate Web template:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, locate the FileUploadTemplate you want to edit by using either the Search or List option.
For more information, see *Searching for a Web Template* on page 109.
3. Click **source manager** next to FileUploadTemplate.xfm.
4. In the **source manager** page, click **edit** next to the FileUploadTemplate to open the Web template in the Web Template Editor.

For more information about editing a Web template, see *Editing a Web Template* on page 61.

5. In the Web Template Editor, make the following changes:
 - a. Locate the <FORM> element under the </xf:group> tag and change the URI to the URI you used in *Editing the Web Extensions HTTP Server Adapter or HTTP Server Adapter* on page 36. The following example shows a before and after change:

```
</xf:group>
```

```
<FORM ENCTYPE="multipart/form-data"
ACTION="http://11.111.111.111:22222/webx/bp/fileupload" METHOD="POST"
NAME="fileform">
```

After you change the URI, your file should resemble:

```
</xf:group>
```

```
<FORM ENCTYPE="multipart/form-data"
ACTION="http://11.111.111.111:22222/webx/bp/URIname" METHOD="POST"
NAME="fileform">
```

- b. In the Name field, type a new name for the FileUploadTemplate.xfm file.
- c. Locate the following code string and replace the number *1* with the same number you replaced the number 1 with in the WebSuiteSkin.xrb file.


```
"XF:output ref=xrb/headings/phrase [@id = 'utility.name1'
```
- d. Click **Save** to save the new Web template in Gentran Integration Suite.

Editing the FileUploadConfirmSend Web Template

To edit the FileUploadConfirmSend Web template:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, locate the FileUploadConfirmSend Web template you want to edit by using either the Search or List option.

For more information, see *Searching for a Web Template* on page 109.

3. Click **source manager** next to FileUploadConfirmSend.
4. In the Web Template Source Manager page, click **edit** next to the FileUploadConfirmSend Web template to open the Web template in the Web Template Editor.
5. Locate the *location.href* element and change the Web template name to the name of the edited FileUploadTemplate.xfm Web template. For example:

```
location.href='/websuite/xfm/NewTemplateName.xfm' file.
```

6. Save the FileUploadConfirmSend Web template with a new unique name that reflects the type of translation the business process completes. For example, EDItoXMLConfirmSend.xfm.

For more information about Web templates, see *Managing Web Templates* on page 105.

Editing the FileUpload Business Process

To edit the File Upload business process:

1. Check out the FileUpload.bp business process from Gentran Integration Suite.

2. From the Gentran Integration Suite **Business Process** menu, select **Manager**.
3. In the **Graphical Modeling** area, click **Go!** next to Run Graphical Process Modeler.
4. In the Graphical Process Modeler, open the FileUpload.bp file and make the following changes:
 - a. In the Translation service, replace the map name with the name of the map you just created and checked in.
 - b. In the Human Interaction XForms service, replace the old template name (FileUploadTemplate.xfm) with the new template name in the TemplateName field.
For more information about naming your template, see *Customizing the File Upload Utility* on page 33.
5. Save the FileUpload.bp file with a new unique name that reflects the type of translation the business process completes. For example, EDItoXML.bp.
6. From the **Tools** menu, select **Source Manager > Process > Add** to add the newly named business process into Gentran Integration Suite.

Editing the Web Extensions HTTP Server Adapter or HTTP Server Adapter

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

To edit the Web Extensions HTTP Server adapter or HTTP Server Adapter:

1. From the Gentran Integration Suite **Deployment** menu, select **Services > Configuration**.
2. Under List, from the **by Service Type** list, select one of the following and click **Go!**
 - ◆ **Web Extensions HTTP Server Adapter**
 - ◆ HTTP Server Adapter
3. Next to the adapter name, click **edit** to edit the adapter configuration. Make the following changes:
 - a. Under Steps on the left side of the page, click **URIs**.
 - b. Next to New URI, click **add** to add a new URI.
 - c. In the **URI** field, type **/webx/bp/URIname**. The URIname entry must match the URIname you changed in *Editing the FileUploadTemplate Web Template* on page 34.
 - d. In the **Business Process** field, select from the list the business process you want to start from the new URI. The business process you select must match the new name of the business process you changed in *Editing the FileUpload Business Process* on page 35.
 - e. Next to Send Raw Messages, select **Yes**.
 - f. Click **Save** to save the adapter configuration.

Accessing the Customized File Upload Utility

To access the customized File Upload utility, point your Web browser to `http://host:port/websuite/xfm/NewTemplateName.xfm`, where *NewTemplateName.xfm* is the Web template name that you used in *Editing the FileUploadTemplate Web Template* on page 34.

About the Web Archive (.war) File Utility

After you have checked in Web resources and associated each resource with a resource tag, you can generate a Web archive (.war) file to deploy the resources in another environment. For example, generating a .war file enables you to:

- ◆ Deploy Web applications easily in a test environment or a production environment.
- ◆ Deploy Web applications easily in a Demilitarized Zone (DMZ).

For more information, see *Managing .war Files* on page 159.

Creating Web Templates

The Web Template Designer enables you to create and edit Web templates that can be used in business processes.

This section covers the following topics:

- ◆ About Web Templates
- ◆ Creating a Web Template
- ◆ Editing a Web Template
- ◆ HTML to XForms Code Samples
- ◆ Managing XForms Structure Elements
- ◆ Managing XForms Controls
- ◆ Managing HTML Elements
- ◆ Editing and Formatting HTML Items
- ◆ Using Tag Libraries with Web Extensions

About Web Templates

Gentran Integration Suite uses Web templates to display business process data in Web browsers. Web templates combine client- and server-side markups to enhance reusability. Within Gentran Integration Suite, the Human Interaction services process the Web template server-side markup tags, whereas the Web browser processes the Web template client-side markup tags.

Prior to creating or editing a Web template, you need:

- ◆ An installed copy of Gentran Integration Suite Web Extensions.
- ◆ A working knowledge of XForms or Java Server Pages, depending on the type of Web templates you implement. For more information about XForms, see www.w3.org.

- ◆ An installed and configured copy of the HTTP Server adapter or Web Extensions HTTP Server adapter. The following table explains when to use each adapter:

Adapter	Use
HTTP Server adapter	<ul style="list-style-type: none"> ◆ When your Web templates are JSPs. For an example, point your Web browser to <code>http://host:port+55/hello-war</code>, where the <code>port+55</code> is 55 added to your port number. Log in using the user name <i>admin</i> and password as the <i>password</i>. ◆ When your Web templates are XForms and you want to use session management features, such as log out requests. For an example of the log out feature, point your Web browser to <code>http://host:port/wssd</code> and log in using the user name <i>sd_buyer</i> and password as the <i>password</i>.
Web Extensions HTTP Server adapter	When your Web templates are XForms and you do not need to use the session management features, such as log out requests.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

- ◆ A defined XML structure and business process information from the business analyst.

Creating a Web Template

You can create Web templates using any one of the following methods:

- ◆ An HTML editor that reads and stores native HTML—that is, HTML that does not store proprietary tags in the HTML document. Gentran Integration Suite Web Extensions provides a Web Template Editor for creating and editing Web templates.
- ◆ Web Template Editor – Uses the Web Template Editor in Gentran Integration Suite as a general text editor to edit Web templates.
- ◆ Web Template Designer – Uses only the Web Template Designer from beginning to end. The Web Template Designer provides the basic XForms structure to get started. You can then add XForms elements, forms controls, and HTML elements to meet your needs.
- ◆ Web Template Translator – Uses the Web Template Translator to translate an XML document into an XForms Web template, then uses the Web Template Designer to edit the Web template. This method provides a robust starting point, as the Web Template Translator provides much of the XForms code that you would otherwise be required to type.
- ◆ Web Template Wizard – Uses the Web Template Wizard to translate a Gentran DDF document into an XForms Web template, then uses the Web Template Designer to edit the Web template. This method provides a robust starting point, as the Web Template Wizard provides much of the XForms code that you would be otherwise required to type.

Web Template Creation Process

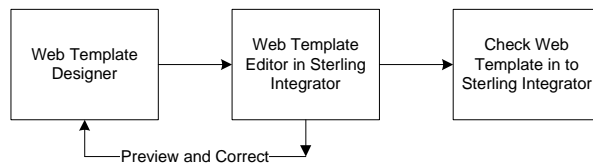
The following steps summarize the Web template creation process:

1. Design the layout and structure of the Web template and split the contents into table cells. This is helpful in determining the number of cells needed when placing content into the table in the Web Template Designer.
2. Create the XForms structure in the Web template.
3. Add an HTML table to the Web template with the number of rows and columns determined in step 1.
4. Add the text, XForms controls, and HTML elements to the cells of the table, based on your design in step 1.

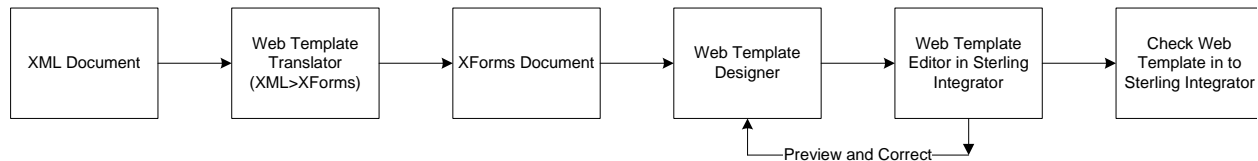
XForms Control and HTML Elements	For More Information, See
XForms button	<i>Managing XForms Button Controls on page 76.</i>
XForms model	<i>Managing XForms Models on page 67.</i>
XForms instance	<i>Managing XForms Instances on page 68.</i>
XForms child node	<i>Managing XForms Child Nodes on page 68.</i>
XForms child node value	<i>Managing XForms Child Node Values on page 70.</i>
XForms bind	<i>Managing XForms Binds on page 71.</i>
XForms text field	<i>Managing XForms Text Field Controls on page 75.</i>
XForms single-choice drop-down list	<i>Managing XForms SelectOne Controls on page 78.</i>
XForms multiple-choice drop-down list	<i>Managing XForms SelectMany Controls on page 80.</i>
HTML image	<i>Managing HTML Images on page 84.</i>
Hyperlinks	<i>Managing HyperLinks on page 85.</i>
E-mail links	<i>Managing E-mail Links on page 86.</i>
HTML table	<i>Managing HTML Tables on page 88.</i>

5. Create the appropriate XForms binds for calculations or verification, if necessary, between the XForms control and the instance data.
6. Save the Web template.
7. Preview the Web template and test the links and features.
8. Make corrections to the Web template.
9. When the Web template meets your needs, check the Web template in to Gentran Integration Suite for use in business processes.

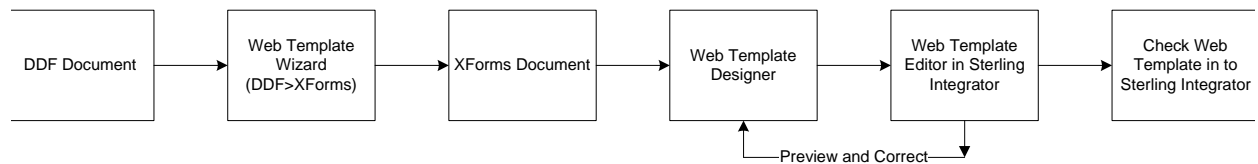
The following figure shows the process to create a Web template using the Web Template Designer method:



The following figure shows the process to complete to create a Web template using the Web Template Translator method:



The following figure shows the process to complete to create a Web template using the Web Template Wizard method:



Creating a Web Template Using an HTML Editor

To create a Web template using an HTML editor:

1. Open a new HTML document in the HTML editor.
2. In the HTML editor, complete one of the following actions:
 - ◆ Paste an existing XForms form written in another HTML editor into the box.
 - ◆ Create an XForms form by typing the XForms code in the box.

You may need to hand-code the XML namespace and XForms model and instance into the Web template. See www.w3.org for additional information about XForms.

3. Save the Web template to your client computer.
4. Preview the Web template.

Note: If you created the Web template, you must include the XML namespace convention in your template to preview the template in the Gentran Integration Suite Web Template Editor. If you have not included the XML namespace tags in your Web template, you will receive an error message when you click the Preview button. For more information, see *Editing a Web Template* on page 61 and *Managing XForms Structure Elements* on page 65.

Creating a Web Template Using the Web Template Designer Method

You can use the Web Template Designer method of creating a Web template when there is no existing Web template, or when you are customizing an existing Web template.

Note: If you implement the examples included in the following instructions, you will have a working Web template that enables you to enter personal information. Using a simple business process, you can check this Web template in to Gentran Integration Suite and test it. For more information, see *Testing Web Templates* on page 111.

To create a Web template using the Web Template Designer method:

1. Start the Web Template Designer.

For more information, see *Starting the Web Template Designer* on page 19.

2. From the **File** menu, select **New** to open a new document.

The design pane displays empty. The code pane displays the basic XForms and HTML structure code that the Web Template Designer provides. The basic structure includes at least one XForms model element with one XForms instance element.

The following sample code shows the basic XForms and HTML structure the Web Template Translator provides:

```
<?xml version='1.0' encoding='UTF-8'?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:scixf="http://www.sterlingcommerce.com/xforms/xforms-extensions">
<head>
</head>
<body>
<xf:group id="xform-body">
</xf:group>
</body>
</HTML>
```

Adding the XForms Structure to the Web Template

1. In the node pane, right-click **Models** > **New Model**.
2. In the **New model name** dialog box, type a unique name for the XForms model and click **OK**. For more information about XForms models, see *Managing XForms Models* on page 67.

For example, create the following XForms models named:

- ◆ myModel
- ◆ docInfo

An XForms instance element is automatically created each time you create an XForms model.

If you implement the examples included in the following instructions, you will have a working Web template that enables you to enter personal information. Using a simple business process, you can check this Web template in to Gentran Integration Suite and test it. For more information, see *Testing Web Templates* on page 111.

- In the node pane, add child nodes to the instance element as required by your design. You can nest a child node within a child node. For more information, see *Managing XForms Child Nodes* on page 68.
For example, possible child nodes are:

Child Node	Child Of
document	Instance element in the myModel element
firstname	myModel/document
lastname	myModel/document
address	myModel/document
zipcode	myModel/document
docInfo	Instance element in the docInfo element
sendURI	docInfo/docInfo
documentType	docInfo/docInfo

- In the node pane, add a value for each child node that you created previously. Each child can contain only one value. For more information about XForms child node values, see *Managing XForms Child Node Values* on page 70.

For example, possible child node and value combinations are:

Child Node	Value
firstname	Tom
lastname	Public
address	555 Anywhere Lane
zipcode	55555-5555
sendURI	/webx/bp/WebTemplateTest
documentType	Test Form

Note: Ensure that all XForms controls are placed between the `<xf:group>` and `</xf:group>` tags. If an XForms control is placed outside of these tags, an error may occur.

Adding the HTML Table to the Web Template

- In the design pane, place an insertion point at the top left corner of the pane.
- From the **HTML** tab, select the **New Table** icon on the tool bar, specify the number of rows and columns of the table to add to the Web template based, on your design layout, and click **OK**.

For example, a table may have 2 columns and 5 rows.

For more information about HTML tables, see *Managing HTML Tables* on page 88.

3. In the code pane, add width and other HTML attributes to the HTML table code. To add width attributes, you must type the width attribute to the `<td></td>` tags for each column in the first row of the table. You can specify width by percent or number of pixels.

For example, `<td width= "50%">` specifies column width in percent, or `<td width= "50">` to specify width in number of pixels.

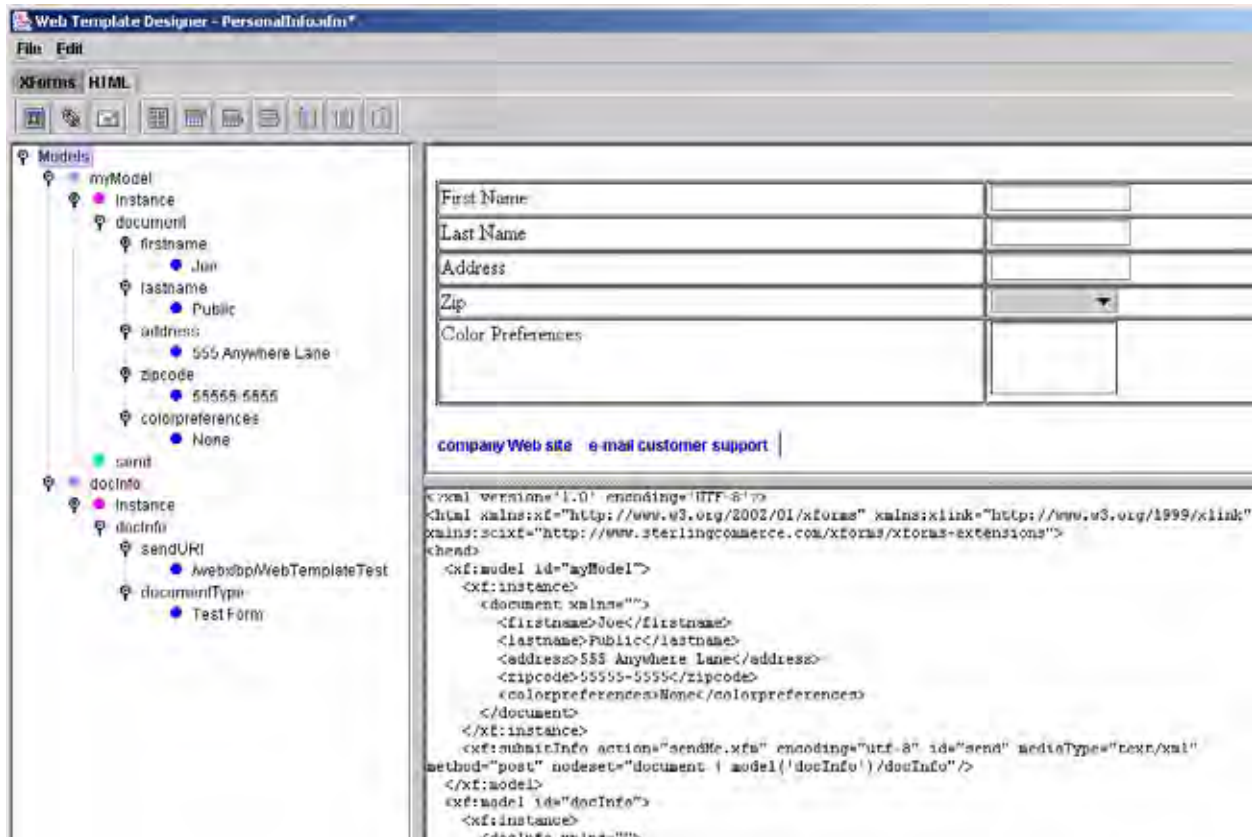
Adding the Content, XForms Controls, and HTML Elements to the Web Template

1. According to your design layout, add text, XForms controls, and HTML elements to the table cells. For more information about XForms controls and HTML elements, see *Managing XForms Controls* on page 75 and *Managing HTML Elements* on page 84.

For example, the following table includes content, XForms controls, and HTML elements to add to the Web template:

Text Content	XForms Control
First Name	Text field. For more information, see <i>Managing XForms Text Field Controls</i> on page 75.
Last Name	Text field. For more information, see <i>Managing XForms Text Field Controls</i> on page 75.
Address	Text field. For more information, see <i>Managing XForms Text Field Controls</i> on page 75.
Zip	SelectOne (static list). For more information, see <i>Managing XForms SelectOne Controls</i> on page 78.
Color Preferences	SelectMany (static list). For more information, see <i>Managing XForms SelectMany Controls</i> on page 80.
company Web site	HTML hyperlink to a Web site. For more information, see <i>Managing HyperLinks</i> on page 85.
e-mail customer support	E-mail link to an e-mail address. For more information, see <i>Managing E-mail Links</i> on page 86.

The following figure shows the XForms structure created, a table with XForms controls, and HTML elements and attributes added to the Web template:



2. In the node pane, expand the node tree, so you can see the child node or value element that you want to link the XForms text field control with.
3. In the design pane, select the XForms text field control for which you want to create a reference to an instance child.
4. In the properties pane, complete the following actions:
 - ◆ In the upper-right corner of the pane, click and drag the cross-hairs to the child node or value element in the node pane to create the link.
 - ◆ The **model** and **ref** fields in the property pane populate with the appropriate information.
 - ◆ In the **size** field, type the display length of the text box in characters. For example, if you type 35, the text box displays the length of 35 characters when you preview or display the Web template. The text box size does not display in the design pane of the Web Template Designer.

- ◆ In the **maxlength** field, type the maximum number of characters that a user can type in the field. For example, if you type 5, the maximum number of characters a user can type into the field is 5.

For example, the following table contains possible XForms controls and the appropriate references to instance child nodes:

XForms Control	Child Node to Link to
First Name text field	document/firstname
Last Name text field	document/lastname
Address	document/address
Zip	document/zipcode
Color Preferences	document/colorpreferences

5. In the node pane, add XForms submitInfo elements to the appropriate XForms models. XForms submitInfo elements are referenced by buttons that initiate the submission of data.

For more information about XForms submitInfo elements, see *Managing XForms SubmitInfo Elements* on page 73.

For example, the following sample code shows an XForms submitInfo control, named *Send*, added to the head section of the Web template:

```
<head>
  <xf:model id="myModel">
    <xf:instance>
    </xf:instance>
    <xf:submitinfo id="send"/>
  </xf:model>
</head>
```

Adding Buttons to the Web Template

1. In the design pane, place an insertion point where you want to place the button that enables you to take action on the data.

For example, place an insertion point directly below the left column of the table. This button enables users to submit data to a business process.

2. On the **XForms** tab, click the **New Button** icon on the tool bar.
3. In the design pane, select the button.

- In the properties pane, in the **Caption** field, type a descriptive name for the button. For example, if the button will submit data, a possible value could be submit.

The following sample code shows a button named Submit added to a Web template:

```
<xf:button>
    <xf:caption>Submit</xf:caption>
</xf:button>
```

- In the properties pane, click **Edit Actions**.
- In the **Actions** dialog box, accept onclick as the Event and click **Add**.
- In the **New Action** dialog box, from the Action Type list, select an action type and click **OK**.

The following table describes the action types available on the action type list:

Action Type	Description
xf:submitInstance	Submits an instance to another URL using the xf:submitInfo attribute.
xf:resetInstance	Refreshes the Web template resetting all entries to default values.
xf:validate	Verifies xf:binds are accurate.
xf:refresh	Refreshes the Web template and recalculates values using the entries that you provided.
xf:insert	Inserts data in to an XForms instance.
xf:delete	Deletes data from an XForms instance.

- In the **Actions** dialog box, under Actions, select an action you want to add attributes to and click **Edit**.
- In the **Edit Action** dialog box, click **Add**.
- In the **New Attribute** dialog box, type the attribute and the corresponding value in the appropriate box and click **OK**.

Valid values are any alphanumeric string; however, all values may not perform a function.

11. In the Actions dialog box, click **OK**.

For more information, see *Managing XForms Button Controls* on page 76.

For example, the following code creates a button to submit data using the XForms submitInfo *send* created previously:

```
<xf:button>
  <xf:caption>Submit</xf:caption>
  <xf:action event="onclick">
    <xf:submitInstance submitInfo="send"/>
  </xf:action>
</xf:button>
```

Ensure that the submitInfo value in the <xf:submitInstance> tag matches the id in the <xf:submitInfo> found in the head section of the Web template; otherwise, an error will occur and the data will not be submitted properly.

12. Repeat *Adding Buttons to the Web Template* on page 46 for each button you are adding.

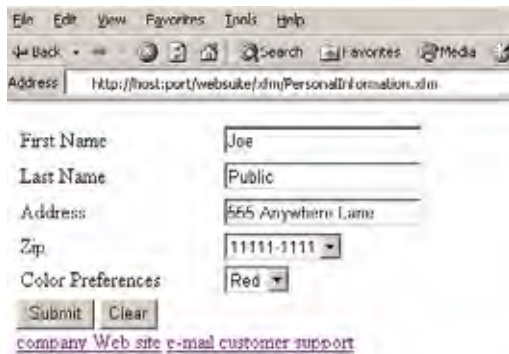
For example, you can add a button and type the following code to create a button named Refresh that refreshes a Web template:

```
<xf:button>
  <xf:caption>Refresh</xf:caption>
  <xf:action event="onclick">
    <xf:refresh/>
  </xf:action>
</xf:button>
```

Saving, Previewing, Editing and Checking In the Web Template

1. From the **File** menu, select **Save As** to save the Web template.
2. Preview the Web template in Gentran Integration Suite.
For more information, see *Previewing a Web Template* on page 105.
3. Make any corrections to the Web template.
For more information, see *Editing a Web Template* on page 61.
4. After the Web template meets your requirements, check the Web template in to Gentran Integration Suite.
For more information, see *Checking In a Web Template* on page 107.

The following figure shows the completed Web template:



Creating a Web Template Using the Web Template Translator

You can use the Web Template Translator method of creating a Web template when there is an existing XML document from which you want to create a Web template. If you implement the examples included in the following procedure, you will have a working Web template that enables you to enter personal information. You can check this Web template in to Gentran Integration Suite and test it using a simple business process. For more information, see *Testing Web Templates* on page 111.

To create a Web template using the Web Template Translator method:

1. Complete one of the following actions:
 - ◆ Locate an XML document created as output from another system.
 - ◆ Use a text editor to write an XML document using information provided by the business analyst.

To follow the scenario, create the following XML document in a text editor:

```
<?xml version='1.0'?>
<document>
<firstname></firstname>
<lastname></lastname>
<address></address>
<zipcode></zipcode>
<colorpreferences></colorpreferences>
</document>
```

2. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
3. In the Web Utilities page, in the Web Template Designer area, click **Go!**
4. From the **Tools** menu, select **Run Web Template Translator**.

5. Using the Web Template Translator, translate the XML document to XForms format. For more information, see *Translating Documents Using the Web Template Translator* on page 26.

The following sample code shows the head section of the XML document translated into XForms format:

```
<?xml version="1.0" encoding="UTF-8"?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:scixf="http://www.sterlingcommerce.com/xforms/xforms-extensions">
  <head>
    <link rel="StyleSheet" type="text/css"
href="/websuite/HTML/admin.css"/>
    <link rel="stylesheet" type="text/css"
href="/websuite/HTML/templates.css"/>
    <xf:model id="skin">
      <xf:instance xlink:href="/websuite/xrb/WebSuiteSkin.xrb"
scixf:preview-href="/websuite/xrb/WebSuiteSkin.xrb"/>
    </xf:model>
    <xf:model id="myModel">
      <xf:instance xlink:href="inline" scixf:preview-href="inline">
<document>
<firstname></firstname>
<lastname></lastname>
<address></address>
<zipcode></zipcode>
<colorpreferences></colorpreferences>
</document>
      </xf:instance>
    </xf:model>
    <title>XML to XForms Translation</title>
  </head>
<body class="MAINBODY">
  <xf:group id="xform-body" model="myModel">
    <table>
      <tr>
        <td>
          
        </td>
      </tr>
    </table>
    <br/>
    <table border="1" width="100%">
```

```
        <tr>
<td align="center" colspan="2">
        <font size="5">
            <B>document</B>
        </font>
</td>
        </tr>
<tr class="sectionHead">
        <td align="center" width="50%">
            firstname
</td>
```

The following sample code shows the body section of the translated document:

```

<td align="center">
    <xf:input ref="/document[1]/firstname[1]/."
model="myModel"></xf:input>
    </td>
</tr>
<tr class="sectionHead">
    <td align="center" width="50%">
        lastname </td>
    <td align="center">
        <xf:input ref="/document[1]/lastname[1]/."
model="myModel"></xf:input>
    </td>
</tr>
<tr class="sectionHead">
    <td align="center" width="50%">
        address
    </td>
    <td align="center">
        <xf:input ref="/document[1]/address[1]/."
model="myModel"></xf:input>
    </td>
</tr>
<tr class="sectionHead">
    <td align="center" width="50%">
        zipcode
    </td>
    <td align="center">
        <xf:input ref="/document[1]/zipcode[1]/."
model="myModel"></xf:input> </td>
</tr>
<tr class="sectionHead">
    <td align="center" width="50%">
        colorpreferences
    </td>
    <td align="center">
        <xf:input ref="/document[1]/colorpreferences[1]/."
model="myModel"></xf:input></td>
</tr>
</table>
</xf:group>
<table border="0" width="100%" cellpadding="0" cellspacing="0"

```

```

height="20">
    <tr>
        <td width="100%" height="20" valign="middle" align="right">
             </td>
    </tr>
</tr>
<tr>
    <td class="hpcopyright" width="100%" height="20"
valign="middle" align="right">
        Powered by Sterling Integrator
        <br/>
        Copyright &#xA9; 2001 Sterling Commerce
    </td>
</tr>
</table>
</body>
</HTML>

```

6. In the Web Template Designer, open the Web template you translated previously.
7. Edit the Web template to meet your requirements. For more information about editing a Web template, see *Editing a Web Template* on page 61. The following table shows possible changes to XForms and references to procedures for each change:

XForms Controls and HTML Elements	For More Information, See
XForms button	<i>Managing XForms Button Controls</i> on page 76.
XForms model	<i>Managing XForms Models</i> on page 67.
XForms instance	<i>Managing XForms Instances</i> on page 68.
XForms child node	<i>Managing XForms Child Nodes</i> on page 68.
XForms child node value	<i>Managing XForms Child Node Values</i> on page 70.
XForms bind	<i>Managing XForms Binds</i> on page 71.
XForms text field	<i>Managing XForms Text Field Controls</i> on page 75.
XForms single-choice drop-down list	<i>Managing XForms SelectOne Controls</i> on page 78.
XForms multiple-choice drop-down list	<i>Managing XForms SelectMany Controls</i> on page 80.
HTML image	<i>Managing HTML Images</i> on page 84.
Hyperlinks	<i>Managing HyperLinks</i> on page 85.
E-mail links	<i>Managing E-mail Links</i> on page 86.
HTML table	<i>Managing HTML Tables</i> on page 88.

8. In the node pane, right-click **Models > New Model**.
9. In the **New model name** dialog box, type a unique name for the XForms model and click **OK**. For more information about XForms models, see *Managing XForms Models* on page 67.

For example, add an XForms model named *docInfo*.

An XForms instance element is automatically created each time you create an XForms model. Each Web template requires at least one XForms model element to display after you check the Web template in Gentran Integration Suite, otherwise an error will occur.

10. In the node pane, add child nodes to the instance element as required by your design. You can nest a child node within a child node.

For more information, see *Managing XForms Child Nodes* on page 68.

For example, add the following child nodes to the *docInfo* model:

Child Node	Child of
docInfo	Instance element in the docInfo element
sendURI	docInfo/docInfo
documentType	docInfo/docInfo

11. In the node pane, add a value for each child node that you added in XForms model *docInfo*. Each child can contain only one value. For more information about XForms child node values, see *Managing XForms Child Node Values* on page 70.

For example, add the following child node and value combinations:

Child Node	Value
sendURI	/webx/bp/WebTemplateTest
documentType	Test Form

12. In the node pane, add XForms submitInfo elements to the appropriate XForms models. XForms submitInfo elements are referenced by buttons that initiated the submission of data.

For more information about XForms submitInfo elements, see *Managing XForms SubmitInfo Elements* on page 73.

For example, the following code sample shows an XForms submitInfo control, named *Send*, added to the head section of the Web template:

```
<head>
  <xf:model id="myModel">
    <xf:instance>
    </xf:instance>
    <xf:submitinfo id="send"/>
  </xf:model>
</head>
```

Note: The example Web template includes a submitInfo element that merges data from two models. For the Web template to work properly, type "document | model ('docInfo')/docInfo" in the nodeset field in the submitInfo property pane.

Adding Buttons to the Web Template

1. In the design pane, place an insertion point where you want to place the button that enables you to take action on the data.

For example, place an insertion point directly below the left column of the table.

2. On the **XForms** tab, click the **New Button** icon on the tool bar.
3. In the design pane, select the button added previously.
4. In the properties pane, in the **Caption** field, type a descriptive name for the button. For example, if the button will submit data, a possible value could be submit.

The following sample code shows a button named Submit added to a Web template:

```
<xf:button>
    <xf:caption>Submit</xf:caption>
</xf:button>
```

5. In the properties pane, click **Edit Actions**.
6. In the **Actions** dialog box, accept onclick as the Event and click **Add**.
7. In the **New Action** dialog box, from the Action Type list, select an action type and click **OK**.

The following table describes the action types available on the action type list:

Action Type	Description
xf:submitInstance	Submits an instance to another URL using the xf:submitInfo attribute.
xf:resetInstance	Refreshes the Web template resetting all entries to default values.
xf:validate	Verifies xf:binds are accurate.
xf:refresh	Refreshes the Web template and recalculates values using the entries that you provided.
xf:insert	Inserts data in to an XForms instance.
xf:delete	Deletes data from an XForms instance.

8. In the **Actions** dialog box, under Actions, select an action you want to add attributes to and click **Edit**.
9. In the **Edit Action** dialog box, click **Add**.
10. In the **New Attribute** dialog box, type the attribute and the corresponding value in the appropriate box and click **OK**.

Valid values are any alphanumeric string; however, all values may not perform a function.

11. In the Actions dialog box, click **OK**.

For more information about XForms buttons, see *Managing XForms Button Controls* on page 76.

For example, the following code creates a button to submit data using the XForms submitInfo *send* created previously:

```
<xf:button>
  <xf:caption>Submit</xf:caption>
  <xf:action event="onclick">
    <xf:submitInstance submitInfo="send"/>
  </xf:action>
</xf:button>
```

Ensure that the submitInfo value in the <xf:submitInstance> tag matches the id in the <xf:submitInfo> found in the head section of the Web template; otherwise, an error occurs and the data will not be submitted properly.

12. Repeat *Adding Buttons to the Web Template* on page 55 to add the required number of buttons.

For example, you can add a button and type the following code to create a button named Refresh that refreshes a Web template:

```
<xf:button>
  <xf:caption>Refresh</xf:caption>
  <xf:action event="onclick">
    <xf:refresh/>
  </xf:action>
</xf:button>
```

13. In the design pane, place an insertion point directly under the table where you want to place an e-mail link and a hyperlink.
14. From the **HTML** tab, add an e-mail link and a hyperlink to the Web template.

For example, the following table includes content and HTML elements to add to the Web template:

Text Content	HTML Element
company Web site	HTML hyperlink to a Web site. For more information, see <i>Managing HyperLinks</i> on page 85.
e-mail customer support	E-mail link to an e-mail address. For more information, see <i>Managing E-mail Links</i> on page 86.

15. From the **File** menu, select **Save As** to save the Web template.

16. Preview the Web template in Gentran Integration Suite.

For more information, see *Previewing a Web Template* on page 105.

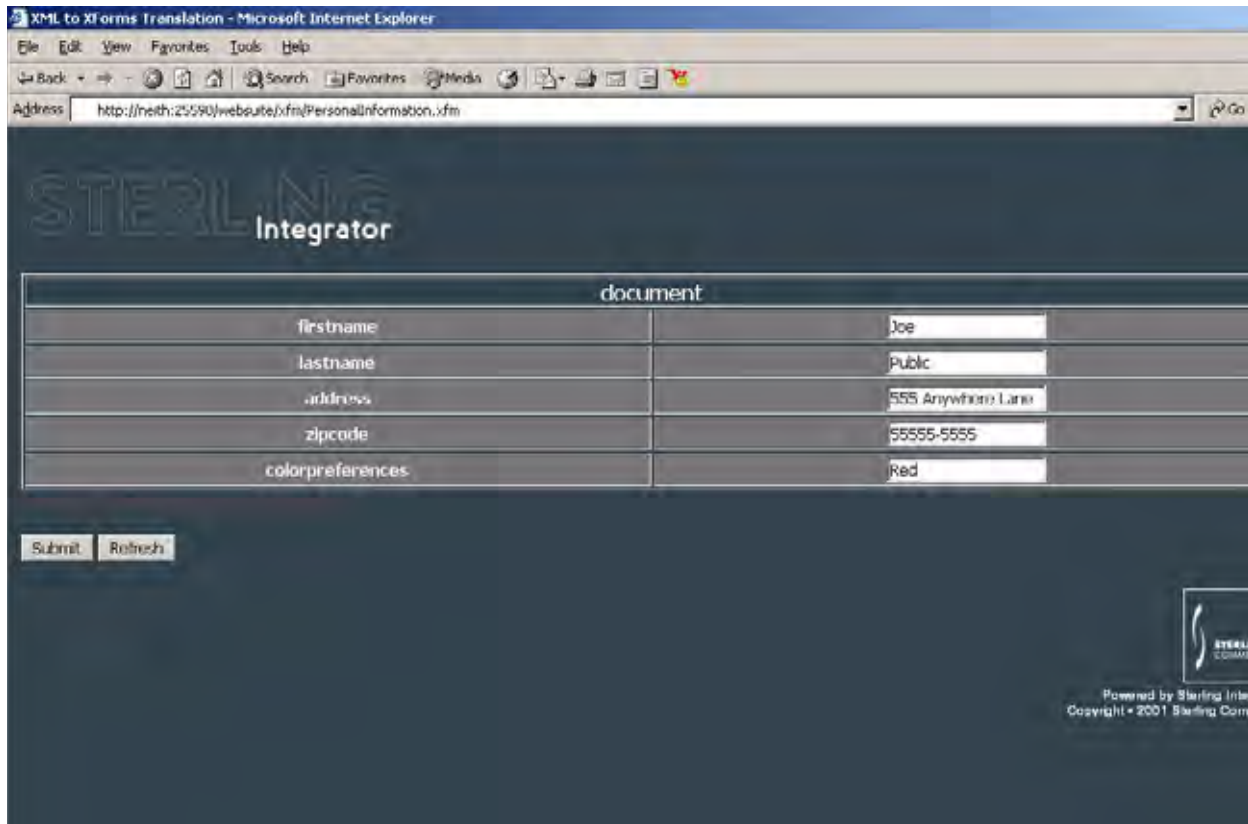
17. Make any corrections to the Web template.

For more information, see *Editing a Web Template* on page 61.

18. When the Web template meets your needs, check the Web template in to Gentran Integration Suite.

For more information, see *Checking In a Web Template* on page 107.

The following figure shows the completed Web template:



Creating a Web Template Using the Web Template Wizard

You can use the Web Template Translator method of creating a Web template when there is an existing XML document from which you want to create a Web template.

Note: If you implement the examples included in the following procedure, you will have a working Web template that enables you to enter personal information. You can check this Web template in to Gentran Integration Suite and test it using a simple business process. For more information, see *Testing Web Templates* on page 111.

To create a Web template using the Web Template Translator method:

1. Locate a Gentran DDF document.
2. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
3. In the Web Utilities page, in the Web Template Designer area, click **Go!**
4. From the **Tools** menu, select **Run Web Template Wizard**.
5. Using the Web Template Wizard, translate the Gentran DDF document to XForms format.

For more information, see *Translating DDF Files Using the Web Template Wizard* on page 31.

6. In the Web Template Designer, open the Web template you translated previously.
7. Edit the Web template to meet your needs. For more information about editing a Web template, see *Editing a Web Template* on page 61.

The following table shows possible changes to XForms and references to procedures for each change:

XForms Controls and HTML Elements	For More Information, See
XForms button	<i>Managing XForms Button Controls</i> on page 76.
XForms model	<i>Managing XForms Models</i> on page 67.
XForms instance	<i>Managing XForms Instances</i> on page 68.
XForms child node	<i>Managing XForms Child Nodes</i> on page 68.
XForms child node value	<i>Managing XForms Child Node Values</i> on page 70.
XForms bind	<i>Managing XForms Binds</i> on page 71.
XForms text field	<i>Managing XForms Text Field Controls</i> on page 75.
XForms single-choice drop-down list	<i>Managing XForms SelectOne Controls</i> on page 78.
XForms multiple-choice drop-down list	<i>Managing XForms SelectMany Controls</i> on page 80.
HTML image	<i>Managing HTML Images</i> on page 84.
Hyperlinks	<i>Managing HyperLinks</i> on page 85.
E-mail links	<i>Managing E-mail Links</i> on page 86.
HTML table	<i>Managing HTML Tables</i> on page 88.

8. In the node pane, right-click **Models** > **New Model**.
9. In the **New model name** dialog box, type a unique name for the XForms model and click **OK**. For more information about XForms models, see *Managing XForms Models* on page 67.

For example, add an XForms model named *docInfo*.

An XForms instance element is automatically created each time you create an XForms model. Each Web template requires at least one XForms model element to display after you check the Web template in Gentran Integration Suite, otherwise an error will occur.

10. In the node pane, add child nodes to the instance element as required by your design. You can nest a child node within a child node.

For more information, see *Managing XForms Child Nodes* on page 68.

For example, add the following child nodes to the docInfo model:

Child Node	Child of
docInfo	Instance element in the docInfo element
sendURI	docInfo/docInfo
documentType	docInfo/docInfo

- In the node pane, add a value for each child node that you added in XForms model *docInfo*. Each child can contain only one value. For more information on XForms child node values, see *Managing XForms Child Node Values* on page 70.

For example, add the following child node and value combinations:

Child Node	Value
sendURI	/webx/bp/WebTemplateTest
documentType	Test Form

- In the node pane, add XForms submitInfo elements to the appropriate XForms models. XForms submitInfo elements are referenced by buttons that initiated the submission of data.

For more information about XForms submitInfo elements, see *Managing XForms SubmitInfo Elements* on page 73.

For example, the following sample code shows an XForms submitInfo control, named *Send*, added to the head section of the Web template:

```
<head>
  <xf:model id="myModel">
    <xf:instance>
    </xf:instance>
    <xf:submitinfo id="send"/>
  </xf:model>
</head>
```

Note: The example Web template includes a submitInfo element that merges data from two models. For the Web template to work properly, type "document | model ('docInfo')/docInfo" in the nodeset field in the submitInfo property pane.

Adding Buttons to the Web Template

- In the design pane, place an insertion point where you want to place the button that enables you to take action on the data.

For example, place an insertion point directly below the left column of the table.

- On the **XForms** tab, click the **New Button** icon on the tool bar.
- In the design pane, select the button added previously.
- In the properties pane, in the **Caption** field, type a descriptive name for the button.

For example, if the button will submit data, a possible value could be submit.

The following sample code shows a button named Submit added to a Web template:

```
<xf:button>
  <xf:caption>Submit</xf:caption>
</xf:button>
```

5. In the properties pane, click **Edit Actions**.
6. In the **Actions** dialog box, accept onclick as the Event and click **Add**.
7. In the **New Action** dialog box, from the Action Type list, select an action type and click **OK**.

The following table describes the action types available on the action type list:

Action Type	Description
xf:submitInstance	Submits an instance to another URL using the xf:submitInfo attribute.
xf:resetInstance	Refreshes the Web template resetting all entries to default values.
xf:validate	Verifies xf:binds are accurate.
xf:refresh	Refreshes the Web template and recalculates values using the entries that you provided.
xf:insert	Inserts data in to an XForms instance.
xf:delete	Deletes data from an XForms instance.

8. In the **Actions** dialog box, under Actions, select an action you want to add attributes to and click **Edit**.
9. In the **Edit Action** dialog box, click **Add**.
10. In the **New Attribute** dialog box, type the attribute and the corresponding value in the appropriate box and click **OK**.

Valid values are any alphanumeric string; however, not all values may perform a function.

11. In the Actions dialog box, click **OK**.

For more information about XForms buttons, see *Managing XForms Button Controls* on page 76.

For example, the following code creates a button to submit data using the XForms submitInfo *send* created previously:

```
<xf:button>
  <xf:caption>Submit</xf:caption>
  <xf:action event="onclick">
    <xf:submitInstance submitInfo="send"/>
  </xf:action>
</xf:button>
```

Ensure that the submitInfo value in the <xf:submitInstance> tag matches the id in the <xf:submitInfo> found in the head section of the Web template; otherwise, an error occurs and the data is not be submitted properly.

12. Repeat *Adding Buttons to the Web Template* on page 55 to add the required number of buttons.

For example, you can add a button and type the following code to create a button named Refresh that refreshes a Web template:

```
<xf:button>
  <xf:caption>Refresh</xf:caption>
  <xf:action event="onclick">
    <xf:refresh/>
  </xf:action>
</xf:button>
```

13. In the design pane, place an insertion point directly under the table where you want to place an e-mail link and a hyperlink.

14. From the **HTML** tab, add an e-mail link and a hyperlink to the Web template.

For example, the following table includes content and HTML elements to add to the Web template:

Text Content	HTML Element
company Web site	HTML hyperlink to a Web site. For more information, see <i>Managing HyperLinks</i> on page 85.
e-mail customer support	E-mail link to an e-mail address. For more information, see <i>Managing E-mail Links</i> on page 86.

15. From the **File** menu, select **Save As** to save the Web template.

16. Preview the Web template in Gentran Integration Suite.

For more information, see *Previewing a Web Template* on page 105.

17. Make any corrections to the Web template.

For more information, see *Editing a Web Template* on page 61.

18. When the Web template meets your needs, check the Web template in to Gentran Integration Suite.

For more information, see *Checking In a Web Template* on page 107.

Editing a Web Template

As your business processes change and your business grows, you may need to change your Web templates. You can also use existing Web templates as a starting point and edit them to meet the needs of new business processes. You can edit Web templates using the Web Template Designer or the Web Template Editor.

Editing a Web Template Using the Web Template Designer

To edit a Web template using the Web Template Designer:

1. Is the Web template checked in to Gentran Integration Suite?
 - ◆ If Yes, check the Web template out of Gentran Integration Suite.
For more information, see *Checking Out a Web Template* on page 108.
 - ◆ If No, go to step 2.
2. Start the Web Template Designer.
For more information, see *Starting the Web Template Designer* on page 19.
3. From the **File** menu, select **Open** to open the Web template.
The Web template opens. The design pane displays the layout and the code pane displays the XForms and HTML code.
4. Make the required corrections.
The following table shows possible edits to Web template elements and references to procedures for each change:

XForms Controls and HTML Elements	For More Information, See
XForms button	<i>Managing XForms Button Controls</i> on page 76.
XForms model	<i>Managing XForms Models</i> on page 67.
XForms instance	<i>Managing XForms Instances</i> on page 68.
XForms child node	<i>Managing XForms Child Nodes</i> on page 68.
XForms child node value	<i>Managing XForms Child Node Values</i> on page 70.
XForms bind	<i>Managing XForms Binds</i> on page 71.
XForms text field	<i>Managing XForms Text Field Controls</i> on page 75.
XForms single-choice drop-down list	<i>Managing XForms SelectOne Controls</i> on page 78.
XForms multiple-choice drop-down list	<i>Managing XForms SelectMany Controls</i> on page 80.
HTML image	<i>Managing HTML Images</i> on page 84.
Hyperlinks	<i>Managing HyperLinks</i> on page 85.
E-mail links	<i>Managing E-mail Links</i> on page 86.
HTML table	<i>Managing HTML Tables</i> on page 88.

5. Preview the Web template.
For more information, see *Previewing a Web Template* on page 105.
6. Review the layout of the Web template and make any changes in the Web Template Designer.
7. When the Web template meets your needs, save the Web template, and check the Web template in to Gentran Integration Suite for use in business processes.

8. In the **Web Template** box, paste the source code from the Web Template Designer and click **Preview**.

Note: If you know the location of the image files or style sheets to use with the template, type the URL in the **Base href** field.

Editing a Web Template Using the Web Template Editor

To edit a Web template using the Web Template Editor:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**. The Web Templates page opens.
2. In the Web Templates page, under **List in the Alphabetically field**, select **ALL** and click **Go!** The Web Templates page opens, listing all Web templates that match your search criteria.
3. Click **source manager** next to the appropriate Web template.
4. In the Web Template Source Manager page, click **edit** next to the Web template.
5. In the Web Template Editor page, in the Web Template box, make any corrections to the Web template.
6. In the **Comments** field type a new description.
7. In the **Base href** box, type the Base href to use when previewing this Web template, using the format `http://hostname:nnnn`. The *Base href* specifies the URL that indicates to Gentran Integration Suite where to locate image files or style sheets to use with the template if the HTML code uses relative paths.
8. Do you want to preview the Web template?
 - ◆ If Yes, click **Preview**.
For more information, see *Previewing a Web Template* on page 105.
 - ◆ If No, go to step 9.
9. Click **Next**.
10. In the Set Default Version page, select the Web template to use as the default Web template in business processes and click **Next**.
11. In the Confirm page, select **Enable for Business Processes** and **Release the lock on the file**, and click **Finish**.

HTML to XForms Code Samples

The following table shows sections of HTML code and corresponding XForms code that can be used to create XForms templates from scratch:

HTML	XForms
<code><form name="form1" method="post" action=""></code>	<code><!--XFORMS<form name="form1" method="post" action="">--></code>

HTML	XForms
<pre><textarea name="textfield" rows="3" cols="30" caption="thecaption" hint="thehint" help="thehelp" ref="myref"></textarea></pre>	<pre><xf:textbox id="textfield" ref="myref" height="3" width="30"> <xf:caption>thecaption</xf:caption> <xf:hint>thehint</xf:hint> <xf:help>thehelp</xf:help> </xf:textbox></pre>
<pre><input type="text" name="textfield2"></pre>	<pre><xf:textbox id="textfield2"> </xf:textbox></pre>
<pre><input type="password" name="textfield3"></pre>	<pre><xf:secret id="textfield3"> </xf:secret></pre>
<pre><input type="checkbox" name="checkboxname" value="checkboxval"></pre>	<pre><xf:selectBoolean id="checkboxname" value="checkboxval"> </xf:selectBoolean></pre>
<pre><input type="submit" name="submitname" value="submitval"> <input type="refresh" name="refreshname" value="refreshval"> <input type="reset" name="resetname" value="resetval"></pre>	<pre><xf:button id="submitname" value="submitval"> <xf:action event="onclick"> <xf:submitInstance/> </xf:action> </xf:button> <xf:button id="refreshname" value="refreshval"> <xf:action event="onclick"> <xf:refreshInstance/> </xf:action> </xf:button> <xf:button id="resetname" value="resetval"> <xf:action event="onclick"> <xf:resetInstance/> </xf:action> </xf:button></pre>
<pre><select name="selectname" size="3" multiple> <option value="item1val">item1</option> <option value="item2val">item2</option> </select></pre>	<pre><xf:selectMany id="selectname" selectUI="listbox" size="3"> <xf:items> <xf:item> <xf:value>item1val</xf:value> <xf:caption>item1</xf:caption> </xf:item> <xf:item> <xf:value>item2val</xf:value> <xf:caption>item2</xf:caption> </xf:item> </xf:items> </xf:selectMany></pre>

HTML	XForms
<pre><select name="select" size="3"> <option value="item1val">item1</option> <option value="item2val">item2</option> </select></pre>	<pre><xf:selectOne id="select" selectUI="listbox" size="3"> <xf:items> <xf:item> <xf:value>item1val</xf:value> <xf:caption>item1</xf:caption> </xf:item> <xf:item> <xf:value>item2val</xf:value> <xf:caption>item2</xf:caption> </xf:item> </xf:items> </xf:selectOne></pre>
<pre><select name="select"> <option value="item1val">item1</option> <option value="item2val">item2</option> </select></pre>	<pre><xf:selectOne id="select" selectUI="pulldown"> <xf:items> <xf:item> <xf:value>item1val</xf:value> <xf:caption>item1</xf:caption> </xf:item> <xf:item> <xf:value>item2val</xf:value> <xf:caption>item2</xf:caption> </xf:item> </xf:items> </xf:selectOne></pre>

Managing XForms Structure Elements

Each Web template created using XForms must include a basic XForms structure. The basic XForms structure includes the following XForms elements:

XForms Element	Location and Requirement
XML namespace	Head section of a Web template. Required.
XForms model	Head section of a Web template. Required.
XForms instance	Head section of a Web template. One XForms instance is required for each XForms model.
XForms group	Body section of a Web template. Required. Note: All XForms controls must be contained between the <xf:group> and the </xf:group> tags; otherwise, errors will occur.
XForms bind	Head section of a Web template. Optional.

XForms Element	Location and Requirement
XForms SubmitInfo	Head section of a Web template. Optional. Note: This element is required if you want to submit the Web template data.

The following code sample shows the basic XForms and HTML structure of a Web template:

```
<?xml version='1.0' encoding='UTF-8'?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:scixf="http://www.sterlingcommerce.com/xforms/xforms-extensions">
<head>
  <xf:model id="new model">
    <xf:instance>
    </xf:instance>
  </xf:model>
</head>
<body>
  <xf:group id="xform-body">
  </xf:group>
</body>
</HTML>
```

The Web Template Designer provides this code for you when you create a new Web template using the **File > New** option. In addition, the Web Template Translator provides this code in the translated Web template, when you translate an XML document into a Web template.

If you create a Web template using methods other than the Web Template Designer or the Web Template Translator, you must type this required information into the Web template.

Managing XML Namespace Elements

An *XML namespace* is an identifier that associates your Web template with a URL, enabling you to view the Web template. After you have added the XML namespace tag to the beginning of your XForms template, you can create shortcuts to the URL using Xlinks instead of retyping it in several places.

The Web Template Translator adds the XML namespace code to the header of the document for you. Predefined Web templates already contain the XML namespace code. If, however, you are editing a Web template that does not contain the XML namespace declaration, you must add it manually.

Adding XML Namespaces

To add XML namespace code to a Web template:

1. Open the Web template in the Web Template Designer.
2. Declare the following new namespace on the HTML element:

```
<HTML xmlns="http://www.w3.org/1999/xhtml"
xmlns:xf="http://www.w3.org/2001/08/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink">
```

```
xmlns:scixf="http://www.Sterlingcommerce.com/xforms/xforms-extensions">
```

3. Add the `scixf:preview-href` attribute to the `xf:instance` element, for example:

```
<xf:instance xlink:href="http://localhost:8080/xforms/xml/catalog.xml"
scixf:preview-href="http://localhost:8080/xforms/xml/catalog2.xml" />
```

In this example, the `catalog.xml` file loads if running in normal mode, and the `catalog2.xml` file loads if running in preview mode.

Managing XForms Models

An *XForms model* element identifies what the Web template does and how the Web template is presented. Each Web template requires at least one XForms model element, but can have multiple XForms model elements within it. Each XForms model must be defined in the HTML header section for the Web template to display correctly.

Adding an XForms Model

You can add XForms model elements at any time during the process of creating or editing a Web template.

To add an XForms model to a Web Template:

1. In the node pane, right-click **Models** > **New Model**.
2. In the **New model name** dialog box, type the name of the model you are adding and click **OK**.
The new XForms model is added to the Web template with one XForms instance. You can view the XForms code in the code pane.
3. Save the Web template.

The following sample code shows an XForms model element added to a Web template:

```
<head>
  <xf:model id="New Model">
    <xf:instance>
    </xf:instance>
  </xf:model>
</head>
```

Editing an XForms Model Name

After you have added an XForms model to the Web template, you can edit the name of the XForms model.

To edit the name of an XForms model in the Web Template Designer:

1. In the node pane, click the XForms model you want to edit.
2. In the properties pane, in the **id** field, type a new name for the XForms model.
3. Save the Web template.

Deleting an XForms Model Name

You can delete an XForms model element when you make changes to your Web template. When you delete an XForms model element, you also delete the instance element and all child nodes and binds contained in the model.

To delete an XForms model element from a Web template in the Web Template Designer:

1. In the node pane, right-click the XForms model you want to delete.
2. From the menu, select **Delete**.
3. Save the Web template.

Managing XForms Instances

The *XForms instance* is an XForms element that contains data that is used in the XForms model while the form is being used. An instance document can be either inline, contained in the Web template itself, or defined in a document outside of the Web template and linked to the Web template.

Each XForms model can have only one XForms instance. When you add an XForms model to a Web template using the Web Template Designer, an XForms instance is automatically created under the XForms model. You can then add your XForms child nodes to create the XML structure.

The XForms instance is often used to populate a Web template with standard information, such as shipping addresses and other repeatedly used information.

Editing an XForms Instance

After you have added an XForms model to the Web template, you can edit the properties of the XForms instance.

To edit the properties of an XForms instance in the Web Template Designer:

1. In the node pane, click the XForms instance you want to edit.
2. In the properties pane, complete the following steps:
 - a. In the **href** field, type the URI of the Web Extensions HTTP Server adapter.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

- b. In the **preview href** field, type the location of the instance data to use to when previewing the Web template.
3. Save the Web template.

Managing XForms Child Nodes

An *XForms child node* is an element of an XForms instance or another XForms child node. If you use the Web Template Translator and Web Template Designer combined method of creating your Web template, the original XML document structure is embedded in the XForms instance of the Web template. Each element of the XML structure is an XForms child node.

If, however, you create your Web template from scratch using only the Web Template Designer, you must build the XML structure by adding XForms child nodes.

Each XForms instance can have only one XForms child node, but each XForms child node can have multiple XForms child nodes. For example, an XForms instance can have only one XForms child node, *Clients*. The XForms child node can have multiple XForms child nodes, *Name*, *Address*, and *Phone*.

The following sample code illustrates the preceding example:

```
<xf:model id="myModel">
  <xf:instance>
    <Clients>
      <Name></Name>
      <Address></Address>
      <Phone></Phone>
    </Clients>
  </xf:instance>
</xf:model>
```

Adding an XForms Child Node

You can add XForms child node elements at any time during the process of creating or editing a Web template.

To add an XForms child node to a Web Template:

1. In the node pane, complete one of the following actions:
 - ◆ Under the XForms model that you want to add a child to, right-click **Instance** and select **New Child** from the menu.
 - ◆ Right-click any child node that you want to add a child to and select **New Child** from the menu.
2. In the **New child name** dialog box, type the name of the child node you are adding and click **OK**.
The new XForms child node element is added to the XForms instance element of the Web template. You can view the XForms code in the code pane.
3. Save the Web template.

The following sample code shows an XForms child node element named *Clients*:

```
<xf:model id="myModel">
  <xf:instance>
    <Clients>
    </Clients>
  </xf:instance>
</xf:model>
```

Editing an XForms Child Node

After you have added an XForms child node to the Web template, you can edit the name of the XForms child node.

To edit the name of an XForms child node in the Web Template Designer:

1. In the node pane, click the XForms child node you want to edit.

2. In the properties pane, in the **Name** field, type a new name for the XForms child node.
3. Save the Web template.

Deleting an XForms Child Node

You can delete an XForms child node when you make changes to your Web template. When you delete an XForms child element, you also delete any child and value elements that are contained in the child node.

To delete an XForms child node from a Web template in the Web Template Designer:

1. In the node pane, right-click the XForms child node you want to delete.
2. From the menu, select **Delete**.
3. Save the Web template.

Managing XForms Child Node Values

XForms child node values are the initial values that display in the Web template. Each XForms child node can have only one value associated with it.

Adding an XForms Child Node Value

You can add an XForms child node value at any time during the process of creating or editing a Web template.

To add an XForms child node value to a Web Template:

1. In the node pane, right-click the child node that you want to add a value to and select **New Value** from the menu.
2. In the **New value** dialog box, type the value of the child node you are adding and click **OK**.
The new XForms child node value is added to the XForms child node of the Web template. You can view the XForms code in the code pane.
3. Save the Web template.

The following sample code shows an XForms child node value of Tom for the XForms child node of Name added to a Web template:

```
<xf:model id="myModel">
  <xf:instance>
    <Clients>
<Client>
<Name>Tom</Name>
</Client>
    </Clients>
  </xf:instance>
</xf:model>
```

Editing an XForms Child Node Value

After you have added an XForms child node value to the Web template, you can edit the XForms child node value.

To edit the XForms child node value in the Web Template Designer:

1. In the node pane, click the XForms child node value you want to edit.
2. In the properties pane, in the **Value** field, type a new value for the XForms child node.
3. Save the Web template.

Deleting an XForms Child Node Value

You can delete an XForms child node value when you make changes to your Web template. When you delete an XForms child node value, you also delete all references to the child node value that are contained in the Web template.

To delete an XForms child node value from a Web template in the Web Template Designer:

1. In the node pane, right-click the XForms child node value you want to delete.
2. From the menu, select **Delete**.
3. Save the Web template.

Managing XForms Binds

An *XForms bind* is a connection between an XForms Model and a form control in a Web template. An XForms bind places constraints on the Web template data. A constraint is an attribute that causes the data to be presented in a specific way. For example, you can include an XForms bind to calculate the sum of two data values, or to have different choices display, depending on the other choices of the user.

Adding an XForms Bind

You can add an XForms bind to an XForms model at any time during the process of creating or editing a Web template.

To add an XForms bind to an XForms model:

1. In the node pane, right-click the XForms model that you want to add a bind to and select **New Bind** from the menu.
2. In the **New bind name** dialog box, type the name of the bind you are adding and click **OK**.

The new XForms bind is added to the XForms model in the Web template. You can view the XForms code in the code pane.

3. Save the Web template.

The following code sample shows an XForms bind called New Bind added to the XForms model called New Model:

```
<xf:model id="New Model">
  <xf:instance>
  </xf:instance>
  <xf:bind id="New Bind"/>
</xf:model>
```

Editing an XForms Bind

After you have added an XForms bind to the Web template, you can edit XForms bind properties in the Web Template Designer properties pane.

To edit the XForms bind properties in the Web Template Designer:

1. In the node pane, click the XForms bind you want to edit.
2. In the properties pane, type values for the following properties:

It is not necessary to use all properties for each bind you add to the Web template.

Bind Property	Description
id	Name of the XForms bind identified in the XForms bind tag. Valid value is any alphanumeric string.
isValid	Validation check ensuring that the data entered is valid for the field. Valid values are: <ul style="list-style-type: none"> ◆ True – Data is valid. ◆ False – Data is not valid.
relevant	Relevancy check ensuring that the fields displayed are relevant to the user's needs. For example, if the user chooses checking as a form of payment, only the fields associated with checking display, while the credit card fields do not display. Valid value is any legal expression.
required	Required field ensuring that the Web template cannot be submitted without the user entering data into the field. Valid values are: <ul style="list-style-type: none"> ◆ true – Field is required. ◆ false – Field is not required.
calculate	Formula to use to calculate a new value. Valid value is any valid expression.
ref	Location where the data is stored. Valid value is any valid XPath expression in to the XForms instance.
reason	Error message that the user sees if the isValid field returns <i>false</i> . Valid value is any alphanumeric message.
readonly	Read-only status. You cannot edit the data in a read-only field. Valid values are: <ul style="list-style-type: none"> ◆ true – Field is read-only. ◆ false – Field is not read-only.

Bind Property	Description
type	Data type of the data in the field. Valid values are: <ul style="list-style-type: none"> ◆ String ◆ Boolean ◆ Integer ◆ Decimal ◆ DateTime ◆ Date ◆ Time ◆ HexBinary ◆ Base64Binary ◆ AnyURI

3. Save the Web template.

Deleting an XForms Bind

You can delete an XForms bind when you make changes to your Web template. When you delete an XForms bind, you delete all references to the bind that in the Web template.

To delete an XForms bind from a Web template in the Web Template Designer:

1. In the node pane, right-click the XForms bind you want to delete.
2. From the menu, select **Delete**.
3. Save the Web template.

Managing XForms SubmitInfo Elements

An *XForms submitInfo* element sets an action to associate with the XForms model. All data found in the node set specified is sent as parameters to the URI defined in the action.

Caution: Ensure that all `<xf:submitInfo></xf:submitInfo>` tags have a capital *I* in *info*; otherwise, errors may occur.

Adding an XForms SubmitInfo Element

You can add an XForms submitInfo element to an XForms model at any time during the process of creating or editing a Web template.

To add an XForms submitInfo element to an XForms model:

1. In the node pane, right-click the XForms model that you want to add an XForms submitInfo to and select **New SubmitInfo** from the menu.

2. In the **New SubmitInfo** dialog box, type the name of the XForms submitInfo you are adding and click **OK**.

The new XForms submitInfo is added to the XForms model in the Web template. You can view the XForms code in the code pane.

3. Save the Web template.

The following sample code shows an XForms submitInfo element called New SubmitInfo added to the XForms model called New Model:

```
<xf:model id="New Model">
  <xf:instance>
</xf:instance>
  <xf:submitInfo id="New SubmitInfo"/>
</xf:model>
```

Editing an XForms SubmitInfo Element

After you have added an XForms submitInfo element to the Web template, you can edit XForms submitInfo properties in the properties pane.

To edit the XForms submitInfo properties in the Web Template Designer:

1. In the node pane, click the XForms submitInfo element you want to edit.
2. In the properties pane, type values for the following properties:

It is not necessary to use all properties for each submitInfo you add to the Web template.

Property	Description
id	Name of the XForms submitInfo element identified in the XForms submitInfo tag. Valid value is any alphanumeric string.
action	Action to associate with the XForms model. Valid value is the Web template.xfm that the data is submitted to.
nodeset	Node set containing the data to submit to the designated URI. Valid value is a valid XPath expression.

3. Save the Web template.

Deleting an XForms SubmitInfo Element

You can delete an XForms submitInfo element when you make changes to your Web template. When you delete an XForms submitInfo element, you also delete all references to the XForms submitInfo element that are contained in the Web template.

To delete an XForms submitInfo element from a Web template in the Web Template Designer:

1. In the node pane, right-click the XForms submitInfo element that you want to delete.
2. From the menu, select **Delete**.
3. Save the Web template.

Managing XForms Controls

XForms controls are areas of the Web template that enable the user to interact with the Web template. For example, a user may type text into a text field or select an item from a list. Each of these is an XForms control that you can add to your Web templates.

These XForms controls provide a great amount of functionality to your Web templates. The Web Template Designer provides tools to add many XForms controls to your Web templates. You can also hand-code XForms controls into the code pane. The XForms controls that you add are visible after the design pane refreshes.

The Web Template Designer provides tools to add the following XForms controls:

- ◆ Text field
- ◆ Button
- ◆ Single choice drop-down list
- ◆ Multiple choice drop-down list

Managing XForms Text Field Controls

An *XForms text field* is an XForms control that enables a user to type text into the Web template. For example, a user may be asked to provide a credit card number for a purchase. The text box that the user types the credit card number into is an XForms text field. There is no restriction on where you can place an XForms text field. You can add an XForms text field to a table, in the middle of a line of text, or stack one on top of another.

Note: You must ensure that the XForms text field control is between the `<xf:group></xf:group>` tags; otherwise, errors may occur.

Adding an XForms Text Field

You can add an XForms text field at any time during the process of creating or editing a Web template.

To add an XForms text field in a Web template:

1. Place an insertion point in the design pane at the location you want to insert the XForms text field.
2. On the **XForms** tab, in the tool bar, click **New TextField**.
3. In the design pane, place an insertion point directly to the left of the text field and type a new label name for the text field.
4. In the node pane, expand the node tree, so you can see the child node or value element that you want to link the XForms text field control with.
5. In the design pane, select the XForms text field control you want to create a link with.
6. In the properties pane, complete the following actions:
 - ◆ In the upper-right corner of the pane, click and drag the cross-hairs to the child node or value element in the node pane to create the link.

The **model** and **ref** fields in the property pane populate with the appropriate information.

- ◆ In the **size** field, type the display length of the text box in characters. For example, if you type 35, the text box displays the length of 35 characters when you preview or display the Web template. The text box size will not display in the design pane of the Web Template Designer.
- ◆ In the **maxlength** field, type the maximum number of characters that a user can type in the field. For example, if you type 5, the maximum number of characters a user can type into the field is 5.

7. Save the Web template.

Editing an XForms Text Field

You can edit an XForms text field at any time during the process of creating or editing a Web template.

To edit an XForms text field in a Web template:

1. In the design pane, select the XForms text field control you want to edit.
2. In the properties pane, highlight and delete the property values that you want to change and complete the following actions:
 - ◆ In the upper-right corner of the pane, click and drag the cross-hairs to the child node or value element in the node pane to create the link.
The **model** and **ref** fields in the property pane populate with the appropriate information.
 - ◆ In the **size** field, type the display length of the text box in characters. For example, if you type 35, the text box displays the length of 35 characters when you preview or display the Web template. The text box size will not display in the design pane of the Web Template Designer.
 - ◆ In the **maxlength** field, type the maximum number of characters that a user can type in the field. For example, if you type 5, the maximum number of characters a user can type into the field is 5.

Deleting an XForms Text Field

You can delete an XForms text field at any time during the process of creating or editing a Web template.

To delete an XForms text field from a Web template:

1. In the design pane, select the XForms text field control you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing XForms Button Controls

An *XForms button* is an XForms control that enables a user to cause the Web template to complete an action. For example, a user may click a Reset button to reset a Web template, click a Submit button to submit a time sheet or a purchase order, or click Cancel to stop processing at some point. There is no restriction on where you can place an XForms button. You can add an XForms button to a table, in the middle of a line of text, or stack one on top of another.

Note: You must ensure that the XForms button control is between the `<xf:group></xf:group>` tags; otherwise, errors may occur.

Adding an XForms Button

You can add an XForms button while creating or editing a Web template.

To add an XForms button to a Web template:

1. Place the insertion point in the design pane where you want to insert the XForms button.
2. On the **XForms** tab, in the tool bar, click the **New Button** icon.
3. In the properties pane, in the **Caption** field, type a descriptive name for the button. For example, if the button will submit data, a possible value could be submit.

The following sample code shows a button named Submit added to a Web template:

```
<xf:button>
  <xf:caption>Submit</xf:caption>
</xf:button>
```

4. In the properties pane, click **Edit Actions**.
5. In the **Actions** dialog box, accept onclick as the Event and click **Add**.
6. In the **New Action** dialog box, from the Action Type list, select an action type and click **OK**.

The following table describes the action types available on the action type list:

Action Type	Description
xf:submitInstance	Submits an instance to another URL using the xf:submitInfo attribute.
xf:resetInstance	Refreshes the Web template resetting all entries to default values.
xf:validate	Verifies xf:binds are accurate.
xf:refresh	Refreshes the Web template and recalculates values using the entries that you provided.
xf:insert	Inserts data in to an XForms instance.
xf:delete	Deletes data from an XForms instance.

7. In the **Actions** dialog box, under Actions, select an action you want to add attributes to and click **Edit**.
8. In the **Edit Action** dialog box, click **Add**.
9. In the **New Attribute** dialog box, type the attribute and the corresponding value in the appropriate box and click **OK**.

Valid values are any alphanumeric string; however, all values may not perform a function.

10. In the Actions dialog box, click **OK**.
11. Save the Web template.

Editing an XForms Button

You can edit an XForms button at any time during the process of creating or editing a Web template.

To edit an XForms button on a Web template:

1. In the design pane, select the XForms button you want to edit.
2. In the properties pane, in the **Caption** field, edit the name of the button, if appropriate.
3. In the properties pane, click **Edit Actions**.
4. In the **Actions** dialog box, accept onclick as the Event.
5. Under **Actions**, select an action from the list and click **Edit**.
6. In the **Edit Action** dialog box, from the Action Type list, select an action type and click **OK**.

The following table describes the action types available on the action type list:

Action Type	Description
xf:submitInstance	Submits an instance to another URL using the xf:submitInfo attribute.
xf:resetInstance	Refreshes the Web template resetting all entries to default values.
xf:validate	Verifies xf:binds are accurate.
xf:refresh	Refreshes the Web template and recalculates values using the entries that you provided.
xf:insert	Inserts data in to an XForms instance.
xf:delete	Deletes data from an XForms instance.

7. In the **Actions** dialog box, under Actions, select an action you want to add attributes to and click **Edit**.
8. In the **Edit Action** dialog box, click **Add**.
9. In the **New Attribute** dialog box, type the attribute and the corresponding value in the appropriate box and click **OK**.

Valid values are any alphanumeric string; however, not all values will perform a function.

10. In the Actions dialog box, click **OK**.
11. Save the Web template.

Deleting an XForms Button

You can delete an XForms button at any time during the process of creating or editing a Web template.

To delete an XForms button from a Web template:

1. In the design pane, select the XForms button you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing XForms SelectOne Controls

An *XForms selectOne control* is an XForms control that enables users to select a single item from a list in a Web template. For example, a user may be asked to provide a country of residency. The user selects the

country of residency from the list. The user can select only one country in a selectOne list. You can add an XForms selectOne control to a table, in the middle of a line of text, or stack one on top of another.

Note: You must ensure that the XForms selectOne control is between the <xf:group></xf:group> tags; otherwise, errors may occur.

Adding an XForms SelectOne Control

You can add an XForms selectOne control at any time during the process of creating or editing a Web template.

To add an XForms selectOne control to a Web template:

1. Place the cursor in the design pane at the location you want to insert the XForms selectOne control.
2. On the **XForms** tab, in the tool bar, click the **New SelectOne** icon.
3. In the design pane, place an insertion point directly to the left of the selectOne box and type a new label name for the selectOne control.

The following sample code shows an XForms selectOne control with text instructing the user to select a payment option:

```
Select a Payment Option:<xf:selectOne selectUI="menu">
  </xf:selectOne>
```

4. Depending on your design, type the appropriate code between the <xf:selectOne> and the </xf:selectOne> tags. The following table shows sample code for different actions:

If You Want to	Use the Following Sample Code
Provide a static list of choices	<pre>Select a Payment Option:<xf:selectOne selectUI="menu"> <xf:choices> <xf:item> <xf:value>credit card</xf:value> <xf:caption>Credit Card</xf:caption> </xf:item> <xf:item> <xf:value>check</xf:value> <xf:caption>Check</xf:caption> </xf:item> </xf:choices> </xf:selectOne></pre>
Provide a dynamic list of choices that is built at run time	<pre>Select a Payment Option:<xf:selectOne ref="//SelectedItem" model="mymodel" selectUI="menu"> <xf:itemset nodeset="//mymodel/option_list" model="mymodel"> <xf:value ref="OptionKey" model="mymodel" /> <xf:caption ref="PaymentOption" model="mymodel" /> </xf:itemset> <xf:action event="onchange"> <xf:refresh/> </xf:action> </xf:selectOne></pre>

5. In the node pane, expand the node tree, so you can see the child node that you want to link the XForms selectOne control with.
6. In the design pane, select the XForms selectOne control you want to create a link with.
7. In the properties pane, in the upper-right corner of the pane, click and drag the cross-hairs to the instance or child node element in the node pane to create the link.

The **model** and **ref** fields in the property pane populate with the appropriate information. These fields indicate the model and the instance or child node elements where the choice will be stored.

8. Save the Web template.

Editing an XForms SelectOne Control

You can edit an XForms selectOne control at any time during the process of creating or editing a Web template.

To edit an XForms selectOne control in a Web template:

1. In the node pane, expand the node tree, so you can see the child node that you want to link the XForms selectOne control with.
2. In the design pane, select the XForms selectOne control you want to create a link with.
3. In the properties pane, in the upper-right corner of the pane, click and drag the cross-hairs to the instance or child node element in the node pane to create the link.

The **model** and **ref** fields in the property pane populate with the appropriate information. These fields indicate the model and the instance or child node elements where the choice will be stored.

4. Save the Web template.

Deleting an XForms SelectOne Control

You can delete an XForms selectOne control at any time during the process of creating or editing a Web template.

To delete an XForms selectOne control from a Web template:

1. In the design pane, select the XForms selectOne control you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing XForms SelectMany Controls

An *XForms selectMany control* is an XForms control that enables users to select more than one item from a list in a Web template. For example, a user may be asked to choose several preferences about color for fabric they are ordering to use in the production of clothing. The user selects the colors from the list. The user can select more than one color from the selectMany list. You can add an XForms selectMany control to a table, in the middle of a line of text, or stack one on top of another.

Note: You must ensure that the XForms selectMany control is between the `<xf:group></xf:group>` tags; otherwise, errors may occur.

Adding an XForms SelectMany Control

You can add an XForms selectMany control at any time during the process of creating or editing a Web template.

To add an XForms selectMany control to a Web template:

1. Place the cursor in the design pane at the location you want to insert the XForms selectMany control.
2. On the **XForms** tab, in the tool bar, click the **New SelectMany** icon.
3. In the design pane, place an insertion point directly to the left of the selectMany box and type a new label name for the selectMany control.

The following sample code shows an XForms selectMany control with text instructing the user to select color preferences:

```
Select Color Preferences:<xf:selectMany selectUI="menu">
    </xf:selectMany>
```

4. Depending on your design, type the appropriate code between the <xf:selectMany> and the </xf:selectMany> tags. The following table shows samples for different actions:

If You Want to	Use the Following Sample Code
Provide a static list of choices	<pre>Select Color Preferences:<xf:selectMany selectUI="menu"> <xf:choices> <xf:item> <xf:value>Blue</xf:value> <xf:caption>Blue</xf:caption> </xf:item> <xf:item> <xf:value>Green</xf:value> <xf:caption>Green</xf:caption> </xf:item> <xf:item> <xf:value>Red</xf:value> <xf:caption>Red</xf:caption> </xf:item> </xf:choices> </xf:selectMany></pre>
Provide a dynamic list of choices that is built at run time	<pre>Select a Payment Option:<xf:selectMany ref="//SelectedItem" model="mymodel" selectUI="menu"> <xf:itemset nodeset="//mymodel/option_list" model="mymodel"> <xf:value ref="OptionKey" model="mymodel" /> <xf:caption ref="ColorOption" model="mymodel" /> </xf:itemset> <xf:action event="onchange"> <xf:refresh/> </xf:action> </xf:selectMany></pre>

5. In the node pane, expand the node tree, so you can see the child node that you want to link the XForms selectMany control with.

6. In the design pane, select the XForms selectMany control you want to create a link to.
7. In the properties pane, in the upper-right corner of the pane, click and drag the cross-hairs to the instance or child node element in the node pane to create the link.

The **model** and **ref** fields in the property pane populate with the appropriate information. These fields indicate the model and the instance or child node elements where the choice the user selects is stored.

8. Save the Web template.

Editing an XForms SelectMany Control

You can edit an XForms selectMany control at any time during the process of creating or editing a Web template.

To edit an XForms selectMany control in a Web template:

1. In the node pane, expand the node tree, so you can see the models that you want to link the XForms selectMany control with.
2. In the design pane, select the XForms selectMany control you want to create a link with.
3. In the properties pane, in the upper-right corner of the pane, click and drag the cross-hairs to the instance or child node element in the node pane to create the link.

The **model** and **ref** fields in the property pane populate with the appropriate information. These fields indicate the model and the instance or child node elements where the choice will be stored.

4. Save the Web template.

Deleting an XForms SelectMany Control

You can delete an XForms selectMany control at any time in the process of creating or editing a Web template.

To delete an XForms selectMany control from a Web template:

1. In the design pane, select the XForms selectMany control you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing XForms File Upload Controls

An *XForms file upload control* is an element that enables you to upload a file to a URI associated with the business process that handles the uploaded files when the form is submitted. The XForms file upload control works in conjunction with the XForms SubmitInfo element to upload the file.

Adding an XForms File Upload Control

You can add XForms file upload controls at any time during the process of creating or editing a Web template.

To add an XForms file upload control to a Web template:

1. In the code pane, locate the XForms submitInfo element that you want to add the XForms file upload control code to.
2. Type the following code sample within the <xf:submitInfo> tag. In the code sample, host:port is your host computer and port number, and URI is the URI in that handles the uploaded files.

```
scixf:uploadURI="http://host:port/URI"
```

For example, the following code sample shows an XForms SubmitInfo named send, which posts the SampleTo1 Web template to the FileUpload business process on host:port, where host and port are your host computer and port number.

```
<xf:submitInfo id="send" action="SampleTo1.xfm" encoding="utf-8"
mediaType="text/xml" method="post"
scixf:uploadURI="http://host:port/webx/bp/FileUpload"/>
```

3. In the code pane, locate the <xf:group> tag.
4. Within the <xf:group> </xf:group> tags, type the following code sample. The location in the document that you type the code, determines the location in the form that the field name and browse button display.

```
<xf:upload name ="the file" accept=image/*"/>
```

Editing an XForms File Upload Control

After you have added an XForms file upload control to a Web template, you can edit the control to change the file to upload and the location that you want the file to upload to.

To edit an XForms file upload control:

1. In the code pane, locate the <xf:submitInfo> tag that contains the URI you want to edit, and make the appropriate changes.
2. In the code pane, locate the <xf:upload> tag and make the appropriate changes.
3. Save the Web template.

Deleting an XForms File Upload Control

You can delete an XForms file upload control when you change your Web template.

Caution: Ensure that you delete both the appropriate entries in the <xf:submitInfo> tag and the <xf:upload> tags to prevent errors in processing.

To delete an XForms file upload control from a Web template in the Web Template Designer:

1. In the code pane, locate the <xf:submitInfo> tag that contains the URI you want to delete, highlight the *scixf:upload URI="host:port..."* code, and press **Delete**.
2. In the code pane, locate the <xf:upload> tag, highlight the entire tag, and press **Delete**.
3. Save the Web template.

Managing HTML Elements

HTML elements are just as important to the Web template as the XForms elements. The Web Template Designer provides tools to add HTML elements to your Web Templates. You can also hand-code HTML elements into the code pane. If you decide to hand-code directly into the code pane, the elements are visible after you refresh the design pane.

Note: If the Web Template Designer is unable to process a specific HTML element, that particular HTML code does not display in the design pane. This should not affect the use of the Web template in Gentran Integration Suite.

The Web Template Designer provides tools to add the following HTML elements:

- ◆ Images
- ◆ Hyperlinks
- ◆ E-mail links
- ◆ Tables

Managing HTML Images

You can add HTML images to your Web templates to customize them to your projects, companies or divisions. For example, you may want to add your company's logo to an invoice Web template. You can add an HTML image to a table, in the middle of a line of text, or stack one on top of another.

All images that you add to a Web template must be:

- ◆ Formatted to display in a Web browser so the image is displayed properly when the Web template is rendered. For example, both the .gif and .jpg image formats are Web browser-friendly.
- ◆ Checked in to Gentran Integration Suite as a Web Resource.

For more information about checking in a Web template, see *Checking In a Web Template* on page 107.

Adding an HTML Image

You can add an HTML image at any time during the process of creating or editing a Web template.

To add an HTML image to a Web template:

1. Place the cursor in the design pane at the location you want to insert the HTML image.
2. On the **HTML** tab, in the tool bar, click the **New Image** icon.

3. In the properties pane, in the **src** field, type or browse to the location where the image is located. This location is used for referencing the image when you display the Web template.

The following sample code shows an HTML image added to a Web template. The image is named `logo.gif` and is found in the `graphics` folder on the applications server:

```
<xf:group id="xform-body">
  
</img>
</xf:group>
```

4. Save the Web template.

Editing an HTML Image

You can change the location of an HTML image, or replace an HTML image with another image at any time during the process of creating or editing a Web template.

To edit an HTML image in a Web template:

1. In the design pane, select the HTML image that you want to edit.
2. In the properties pane, type or browse to the new location where the image is located. The new location is used for referencing the image when you display the Web template.
3. Save the Web template.

Deleting an HTML Image

You can delete an HTML image from a Web template at any time during the process of creating or editing a Web template.

To delete an HTML image from a Web template:

1. In the design pane, select the HTML image you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing HyperLinks

You can add hyperlinks to your Web templates to enable users to jump to other documents or sites for more information about an item on the Web template. For example, you may want to add a hyperlink to your company's Web site on a Web template, so the user can learn more about different products during the purchase order process. You can add a hyperlink to a table, in the middle of a line of text, or stack one on top of another.

Adding a Hyperlink

You can add a hyperlink during the process of creating or editing a Web template.

To add a hyperlink to a Web template:

1. Place the cursor in the design pane at the location you want to insert the hyperlink.

2. On the **HTML** tab, in the tool bar, click the **New Hyperlink** icon.
3. In the properties pane, complete the following steps:
 - a. In the **text** field, type the text that will be linked to the URL. For example, if you are adding a hyperlink to your company's Web site, type My Company Name.
 - b. In the **href** field, type the URL that the hyperlink links to. You can provide either an absolute or relative URL, depending on your needs.

Ensure that all hyperlink references to external Web sites are absolute paths and include the *http://* prefix; otherwise, the XForms process will read the URL as a relative path and errors may occur.

The following sample code shows a hyperlink added to a Web template. The hyperlink is named My Company Name and sends the user to the My Company Name Web site:

```
<xf:group id="xform-body">
<a href="http://www.mycompanyname.com">My Company Name</a>
</xf:group>
```

4. Save the Web template.

Editing a Hyperlink

You can edit a hyperlink at any time during the process of creating or editing a Web template.

To edit a hyperlink in a Web template:

1. In the design pane, select the hyperlink that you want to edit.
2. In the properties pane, make any changes to the hyperlink text or the URL.
3. Save the Web template.

Deleting a Hyperlink

You can delete a hyperlink from a Web template at any time during the process of creating or editing a Web template.

To delete a hyperlink from a Web template:

1. In the design pane, select the hyperlink that you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing E-mail Links

You can add e-mail links to your Web templates to enable users to send e-mail messages directly from the Web template. For example, you may want to add an e-mail link to your company's customer support department on a Web template, so the customer can send an e-mail about a problem they found with the Web template. You can add an e-mail link to a table, in the middle of a line of text, or stack one on top of another.

Adding a E-mail Link

You can add an e-mail link at any time while you are creating or editing a Web template.

To add an e-mail link to a Web template:

1. Place the cursor in the design pane at the location you want to insert the e-mail link.
2. On the **HTML** tab, in the tool bar, click the **New E-mail Link** icon.
3. In the New E-mail Link dialog box, complete the following steps:
 - a. In the **Text** field, type the text that is displayed for the user to click to start the e-mail process. For example, to provide an e-mail link to your customer support department, you might type *E-mail Customer Support*.
 - b. In the **Address** field, type the e-mail address of the person or department to which the user's message is sent. For example, in the previous example, the e-mail address may be *customersupport@mycompanyname.com*.

Ensure that e-mail address is correct and active; otherwise, errors may occur.

The following sample code shows an e-mail link added to a Web template. The e-mail link is named Customer Support and enables the user to e-mail the Customer Support department:

```
<xf:group id="xform-body">  
<a href="mailto:customersupport@mycompanyname.com">Customer Support</a>  
</xf:group>
```

4. Save the Web template.

Editing a E-mail Link

You can edit an e-mail link at any time while you are creating or editing a Web template.

To edit an e-mail link in a Web template:

1. In the design pane, select the e-mail link that you want to edit.
2. In the properties pane, make any changes to the e-mail link text or the e-mail address.
Ensure that you retain the *mailto:* portion of the href; otherwise, errors will occur.
3. Save the Web template.

Deleting an E-mail Link

You can delete an e-mail link from a Web template at any time during the process of creating or editing a Web template.

To delete an e-mail link from a Web template:

1. In the design pane, select the e-mail link that you want to delete.
2. Press **Delete**.
3. Save the Web template.

Managing HTML Tables

HTML tables can help you organize the placement of content, XForms controls, and HTML elements in your Web templates. Tables provide a natural matrix to organize work and to help create the design to meet your needs.

In addition to helping you organize the Web template design, you can use HTML tables to provide information to the user in a more organized manner. You can add an HTML table to a table (nesting), in the middle of a line of text, or stack one on top of another.

Adding an HTML Table

You can add an HTML table at any time during the process of creating or editing a Web template.

To add an HTML table to a Web template:

1. Place the cursor in the design pane at the location you want to insert the HTML table.
2. From the **HTML** tab, in the tool bar, click the **New Table** icon.
3. In the **New Table** dialog box, type the number of rows and columns to add, based on your need, and click **OK**.
4. In the code pane, in the first row of the table, add a width attribute to the each column, and type any other HTML attributes in the HTML table code.

You must type the width attribute to the `<td></td>` tags for each column in the first row of the table. You can specify width by percent or number of pixels. For example, `<td width="40%">` to specify width in percent, or `<td width="40">` to specify width in number of pixels.

The following sample code shows an HTML table added to a Web template:

```
<xf:group id="xform-body">
  <table>
    <tr>
      <td width="40%"></td>
      <td width="60%"></td>
    </tr>
    <tr>
      <td></td>
      <td></td>
    </tr>
    <tr>
      <td></td>
      <td></td>
    </tr>
  </table>
</xf:group>
```


5. Add content, XForms controls, and HTML elements to the table cells to achieve the design to meet your needs.

The following table shows Web template elements and references to the procedures:

XForms Control and HTML Elements	For More Information, See
XForms button	<i>Managing XForms Button Controls on page 76.</i>
XForms model	<i>Managing XForms Models on page 67.</i>
XForms instance	<i>Managing XForms Instances on page 68.</i>
XForms child node	<i>Managing XForms Child Nodes on page 68.</i>
XForms child node value	<i>Managing XForms Child Node Values on page 70.</i>
XForms bind	<i>Managing XForms Binds on page 71.</i>
XForms text field	<i>Managing XForms Text Field Controls on page 75.</i>
XForms single-choice drop-down list	<i>Managing XForms SelectOne Controls on page 78.</i>
XForms multiple-choice drop-down list	<i>Managing XForms SelectMany Controls on page 80.</i>
HTML image	<i>Managing HTML Images on page 84.</i>
Hyperlinks	<i>Managing HyperLinks on page 85.</i>
E-mail links	<i>Managing E-mail Links on page 86.</i>
HTML table	<i>Managing HTML Tables on page 88.</i>

6. Save the Web template.

Adding Rows and Columns to an HTML Table

You can add rows and columns to an HTML table when you create or edit a Web template.

To add a row to an HTML table in a Web template:

1. In the design pane, place an insertion point in a cell of the HTML table to which you want to add a row.
2. From the **HTML** tab, in the tool bar, complete one of the following actions:
 - ◆ Click the **Insert Row Before** icon – Inserts a row in the table above the row containing the insertion point.
 - ◆ Click the **Insert Row After** icon – Inserts a row in the table below the row containing the insertion point.
3. Save the Web template.

To add a column to an HTML table in a Web template:

1. In the design pane, place an insertion point in a cell of the HTML table to which you want to add a column.
2. From the **HTML** tab, in the tool bar, complete one of the following actions:

- ◆ Click the **Insert Column Before** icon – Inserts a column in the table to the left the column containing the insertion point.
 - ◆ Click the **Insert Row After** icon – Inserts a column in the table to the right of the row containing the insertion point.
3. Save the Web template.

Deleting Rows and Columns from an HTML Table

You can delete rows and columns from an HTML table at any time during the process of creating or editing a Web template.

To delete a row from an HTML table in a Web template:

1. In the design pane, place an insertion point in a cell of the row in the HTML table that you want to delete.
2. On the **HTML** tab, in the tool bar, click the **Delete Row** icon. The row is deleted.
3. Save the Web template.

To delete a column from an HTML table in a Web template:

1. In the design pane, place an insertion point in a cell of the column in the HTML table that you want to delete.
2. On the **HTML** tab, in the tool bar, click the **Delete Column** icon. The column is deleted.
3. Save the Web template.

Editing and Formatting HTML Items

Copying and Pasting Items

You can copy items in the Web Template Designer to paste the items to other locations in the Web template. An *item* is either an HTML element in the design pane or code in the code pane. You can only paste the copied items to the same pane. For example, you can copy a button in the design pane and paste it in the design pane, but not in the code pane.

To copy and paste an item in the Web Template Designer:

1. In either the design pane or the code pane, select the item you want to copy.
2. From the **Edit** menu, select **Copy**.
3. In the same pane, place an insertion point where you want to paste the item.
4. From the **Edit** menu, select **Paste**.

Cutting and Pasting Items

You can cut items in the Web Template Designer to paste the items to a different location, or to remove the items completely from the Web template. An item is either an HTML element in the design pane or code in the code pane. You can only paste the cut items to the same pane. For example, you can cut a button in the design pane and paste it in the design pane, but not in the code pane.

To cut and paste an item in the Web Template Designer:

1. In either the design pane or the code pane, select the item you want to cut.
2. From the **Edit** menu, select **Cut**.
3. In the same pane, place an insertion point where you want to paste the item.
4. From the **Edit** menu, select **Paste**.

To cut an item and remove it completely from a Web template:

1. In either the design pane or the code pane, select the item you want to cut.
2. From the **Edit** menu, select **Cut**.

Formatting Text

You can apply virtually any HTML formatting to your Web templates to meet your needs. The Web Template Designer provides tools to help you format text within your Web templates.

Tools provided enable you to apply:

- ◆ Various fonts to text
- ◆ Bold and italic emphasis to text
- ◆ Point sizes to text

Applying a Font to Text

You can apply different fonts to different text elements to enhance the layout and design of your Web templates.

To apply a font to text in a Web template:

1. In the design pane, select the text to which you want to apply the font.
2. In the properties pane, from the font list, select the font you want to apply.

The appropriate attributes are added to the code in the code pane, and the text changes in the design pane.

3. Save the Web template.

Applying Bold and Italic Emphasis to Text

You can apply bold and italic emphasis to different text elements to enhance the layout and design of your Web templates.

To apply bold or italic emphasis to text in a Web template:

1. In the design pane, select the text to which you want to apply emphasis.
2. In the properties pane, complete one of the following actions:
 - ◆ To apply bold to the selected text, click the **Bold** button.
 - ◆ To apply italic to the selected text, click the **Italic** button.
 - ◆ To apply both bold and italic to the selected text, click both the **Bold** and **Italic** buttons.

The appropriate attributes are added to the code in the code pane, and the text changes in the design pane.

3. Save the Web template.

Applying a Point Size to Text

You can apply different point sizes to different text elements to enhance the layout and design of your Web templates.

To apply a point size to text in a Web template:

1. In the design pane, select the text to which you want to apply the point size.
2. In the properties pane, from the **size** list, select the point size you want to apply.

The appropriate point size attributes are added to the code in the code pane, and the text changes in the design pane.

3. Save the Web template.

Using Tag Libraries with Web Extensions

JSP Tag Libraries

JavaServer Pages (JSP) are Web pages that use templates, custom elements, scripting languages, and server-side Java objects to return dynamic content to a Web browser in the form of HTML or XML. You can use the JSP Web templates in Gentran Integration Suite Web Extensions. A *JSP Web template* is a Web form that is used to interface with a Web application, such as a smaller business partner completing a purchase order and invoicing through a Web site.

A *Java Server Pages (JSP) tag library* is a collection of predefined JSP components that you can use in your JSP Web templates. Use JSP tag libraries to:

- ◆ Reduce manual coding efforts of your JSP Web templates
- ◆ Reduce the amount of Java code in your JSP Web templates

Before You Can Use JSP Tag Libraries

Before you can use the JSP tag libraries with Gentran Integration Suite, you must:

- ◆ Have knowledge of Java and Java Server Pages.

- ◆ Have knowledge of application servers.
- ◆ Have read, write, and execute rights to your application server and the server where Gentran Integration Suite is installed.
- ◆ Verify that *install_dir*/jar/woodstock.jar is on the classpath accessible to a Web server.
- ◆ Copy *install_dir*/installed_data/pshttp/exploded_wars/helloworld/WEB-INF/tld/userautho.tld to the WEB-INF/tld directory in your Web application.
- ◆ Specify the location of your JSP tag library in your Web application's WEB-INF/web.xml file. Type the following code between the <web-app> and </web-app> tags:

```
<taglib>
<taglib-uri>http://www.stercomm.com/uix/security</taglib-uri>
<taglib-location>tld/userautho.tld</taglib-location>
</taglib>
```

Run Business Process Tag Libraries

Use the tag libraries to run a single business process per JSP page.

runBP

The runBP tag library runs a business process, which is specified in the bpname attribute. The runBP tag library also sets the pageContext variable called bpResults in the org.w3c.dom.Document type. The business process results are displayed using the bpResults variable.

The following table describes the attributes the runBP tag library uses:

Attribute	Description
user	User ID. Required. This attribute may be specified at the business process run time.
nvp	Name value pairs to be placed into process data in the format name=value&name2=value2. Required. This attribute may be specified at the business process run time.
bpname	Business process to run. Required. This attribute may be specified at the business process run time.
pridoc	Primary document that invokes the business process. Required. This attribute may be specified at the business process run time.

runBPnoReturnDoc

The runBPnoReturnDoc tag library runs a business process with no document being returned from the running business process.

The following table describes the attributes the runBPnoReturnDoc tag library uses:

Attribute	Description
user	User ID. Required. This attribute may be specified at the business process run time.
nvp	Name value pairs to be placed into process data in the format name=value&name2=value2. Required. This attribute may be specified at the business process run time.
bpname	Business process to run. Required. This attribute may be specified at the business process run time.
pridoc	Primary document that invokes the business process. Required. This attribute may be specified at the business process run time.

getDocuments

The getDocuments tag library runs a business process and extracts a document from the business process.

The following table describes the attributes the getDocuments tag library uses:

Attribute	Description
user	User ID. Required. This attribute may be specified at the business process run time.
nvp	Name value pairs to be placed into process data in the format name=value&name2=value2. Required. This attribute may be specified at the business process run time.
bpname	Business process to run. Required. This attribute may be specified at the business process run time.
pridoc	Primary document that invokes the business process. Required. This attribute may be specified at the business process run time.

Business Process Resolver

Use the Business Process Resolver to run multiple business processes on a single JSP page or if you have complex conditional logic. The JSP compiler cannot compile the jsp when using the runBP Tag Library.

Required Imports

```
<%@page import="org.w3c.dom.Document"%>
<%@page import="com.sterlingcommerce.woodstock.util.xml.XMLUtil"%>
<%@pageimport="com.sterlingcommerce.woodstock.webx.xforms.plugin.
BusinessProcessResolver"%>
```

Method Interfaces

```

public org.w3c.dom.Document resolve(String bpName, String processData,
byte[]
    primaryDocument, String username) throws Exception
public org.w3c.dom.Document resolve(String bpName, String processData,
byte[]
    primaryDocument, String username, boolean hasAttachment) throws Exception
public org.w3c.dom.Document resolve(String bpName, String processData,
    Document uploaded_doc, String username, boolean hasAttachment) throws
Exception
public org.w3c.dom.Document resolve(String bpName, String processData,
String
    primaryDocument, String username) throws Exception
public org.w3c.dom.Document resolve(String bpName, String processData,
String
    primaryDocument, String username, boolean hasAttachment) throws Exception
public byte[] resolve(String bpName, String processData, String username)
throws
    Exception

```

Example

```

<%
Document doc;
BusinessProcessResolver bpResolver = new BusinessProcessResolver();
doc = (org.w3c.dom.Document)
bpResolver.resolve("myBPname", "test=123&test2=234", "", "GIS_user_to_run_as");
strBPResultsData = XMLUtil.documentToString(doc, false);
%>

```

com.sterlingcommerce.woodstock.util.XMLUtil

This utility class is used in the previous code example. It provides several interfaces for managing xml nodes, which is what most of the BusinessProcessResolver class interfaces return, in the form of a org.w3c.dom.Document object.

Method Interfaces

```

public static String documentToHTML(Node node, boolean canonical)
public static String documentToString(Node node, boolean canonical)
public static String documentToString(Node node, boolean canonical, boolean
    useReturns)
public static String documentToString(Node node, boolean canonical, boolean
    useReturns)
public static String documentToString(Node node, boolean canonical,
StringBuffer sb,
    String indent)
public static String documentToString(Node node, boolean canonical,
StringBuffer sb,
    String indent, String newline)
public static String documentToString(Node node, boolean canonical,
StringBuffer sb,
    String indent, String newline, boolean useReturns)

```

Authorization (autho) Tag Library

The autho tag library verifies you, as a Gentran Integration Suite user, belong to specified Gentran Integration Suite groups or have specified Gentran Integration Suite permissions. If the authorization fails, the JSP body is not displayed or you are forwarded to an error page. If neither permissions or groups attributes are set, you are considered authorized.

Since authorization is based on at least one match against a permission or group, authorization succeeds. This means that if a user does not have association with any group, but has one permission applied, the authorization succeeds.

The following table describes the attributes the autho tag library uses:

Attribute	Description
user	User ID. Optional. This attribute may be specified at the business process run time. The tag checks the session for a user name attribute if the attribute is not declared.
group	Group ID. Optional. This attribute may be specified at the business process run time.
groups	List or ArrayList of group IDs. Both the group and the groups attributes must be declared at the same time. Authorization succeeds if you have association with at least one group.
permission	Permission ID. Optional. This attribute may be specified at the business process run time.
permissions	List or ArrayList of permissions. Both the permission and the permissions attributes must be declared at the same time. Authorization succeeds if you have association with at least one permission.
login	URL of the login page that you are sent to if your session expires or is invalid. A session is invalid if a user name attribute is not present. Optional. This attribute may be specified at the business process run time. If this attribute is not set and the session expires, the JSP page does not display. The session is valid if the user name attribute is present.
checkExistence	Checks for a group or permission association prior to authorization. Optional. This attribute may be specified at the business process run time. If a permission or group association does not exist and this value is set to true, all non-existent permissions and groups are skipped in the authorization process.

Session Validation (validSession) Tag Library

The validSession tag library validates your session with Gentran Integration Suite has not expired. If the validation fails, an error page or login page displays.

The following table describes the attributes the validSession tag library uses:

Attribute	Description
login	URL of the login page that you are sent to if your session expires or is invalid. A session is invalid if a user name attribute is not present. Required. This attribute may be specified at the business process run time. If this attribute is not set and the session expires, the JSP page does not display. The session is valid if the user name attribute is present.

Authentication (authenticate) Tag Library

The authenticate tag library Authenticates you against the Gentran Integration Suite database or your LDAP server to determine that you are authorized access to the system.

The following table describes the attributes the authenticate tag library uses:

Attribute	Description
user	User ID you enter in the login page. Required. This attribute may be specified at the business process run time.
pass	Password you enter in the login page. Required. This attribute may be specified at the business process run time.
login	URL of the login page that you are sent to if your session expires or is invalid. A session is invalid if a user name attribute is not present. Required. This attribute may be specified at the business process run time. If this attribute is not set and the session expires, the JSP page does not display. The session is valid if the user name attribute is present.

A *Java Server Page Standard Tag Library (JSTL)* is a consolidated grouping of common JavaServer Pages (JSP) code samples that you can use in developing JSP Web templates. JSTL is open source and provides code for common tasks, such as look ups, parsing, date, and time formats. By using the JSTL Standard Tag Library, you can standardize common functions.

The JSTL Standard Tag Library enables you to:

- ◆ Reduce the complexities of developing JSP Web templates by creating standardized code in Web templates to improve consistency of presentation and handling of data
- ◆ Reduce inefficiencies in your business by standardizing code by reducing errors in Web template development
- ◆ Reduce costs and improve revenue generation by speeding the Web template development process
- ◆ Streamline processing of XML messages and XSLT transformations

Note: JSTL is open source. These instructions include how to implement JSTL in a Gentran Integration Suite Web application. Instructions for using JSTL can be found on the open source web site.

Requirements For Using the JSTL

- ◆ JSP Web templates – JSTL does not work with XForms Web templates and therefore cannot be used with the Web Template Designer because it is an XForms Editor.
- ◆ JSP 1.2 container – Gentran Integration Suite uses Jetty, a Java-based HTTP Server and Servlet container that processes a JSP for rendering.
- ◆ Installed copy of Gentran Integration Suite with Web Extensions.
- ◆ Web application created using J2EE.

JSTL Concepts and Definitions

Concept	Definition
Expression Language (EL)	Computer language used in JSTL for expressing simple expressions, which is based on the XPath and JavaScript or JScript languages.
JavaServer Page (JSP)	Web pages that use templates, custom elements, scripting languages, and server-side Java objects to return dynamic content to a Web browser in the form of HTML or XML.
JSP Container	An entity that provides life-cycle management and runtime services to the components of a Web application, just like a servlet, and provides an engine that interprets and processes JSPs.
JSTL Namespace	A unique identifier that defines the URI of the tag library and the prefix used in the tags for that specific functional area library.
Prefix	A unique identifier that defines the functional area that a tag belongs to. The prefix is defined in the JSTL Namespace. For example, prefix="c" means that all tags that start with <c: are part of the core functional area tag library. The c in <c: is the prefix of the tag.
Runtime (RT) Processing	Scripting used in JSPs processed at runtime. This method is being replaced by the EL method in JSTLs.
Tag Handler	Explanation of how to process the tag when it is encountered in the processing of the Web template. Typically, this is defined in a <tag-class/> tag.
Tag Library Descriptor (.tld)	XML file that describes a tag library and contains the tags in the library. Each functional area library has a .tld file.
Universal Resource Indicator (URI)	Trailing part of a Web address that indicates the location of the resource being used. For example, in <code>http://host:port/uri/sample.xml</code> , the URI is located on host and sample.xml is located in the uri folder.
web.xml	A standard file created when the Web application is developed in J2EE and contains properties controlling servlets for the specific Web application. The web.xml file also includes the tag library declaration, the URI, and tld declarations.
Web Application	Web site that delivers dynamic content. For example, a Web site that has a purchase order form that once submitted returns an acknowledgment that the purchase order was created and sent to the appropriate trading partner.
Web Archive (WAR)	A compressed file that contains the directory structure and file contents of a Web application. The directory structure includes the WEB-INF directory containing the web.xml file and classes directory.

Setting Up and Using JSTL

You do not need to configure Gentran Integration Suite to use the JSTL. The JSTL is optional and since this is a library of code, it is never directly tied to any part of Gentran Integration Suite. The library is only referenced by your JSP Web templates.

The following checklist describes what you must do to implement JSTL:

Step	Description	Your Notes
1	Adding the JSTL to the Web Application.	
2	Edit the web.xml file for the Web application to include the tag libraries. See <i>Editing the web.xml File for JSTL</i> .	
3	Add the JSTL Namespace and tags to the JSP Web template.	
4	Update, Compile, and Redeploy the WAR file in Gentran Integration Suite Note: You can use the jar utility that is installed with the JDK to compile the WAR file.	
5	Test the Web template in the Web application.	

Adding the JSTL to the Web Application

To add the JSTL library to your Web application, you must create a .TLD Directory and copy the .tld directory files into it.

1. In your *WebApplicationName*/WEB-INF directory, create a new directory named TLD.
2. Unjar the standard.jar file in the *install_dir/installeddata/patch/jars/jakarta_jstl/1_1_2/* directory.
3. From the expanded standard.jar file, copy the .tld files to the *WebApplicationName*/WEB-INF/TLD directory you created.
4. Edit the web.xml file properties for your specific Web application. See *Editing the web.xml File for JSTL* on page 99.

Editing the web.xml File for JSTL

The web.xml file contains properties controlling servlets for a specific Web application. This is a standard file created when the Web application is created using J2EE. If you are adding JSTL Standard Library functions to multiple Web applications, you must edit the web.xml file associated with each Web application.

To use JSTL, you must declare the <taglib> tags in the web.xml file between the <Web-app> and </Web-app> tags.

To edit the web.xml file:

1. Locate the web.xml file in the Web application directory. The standard directory structure is:
WebApplicationName/WebApp/WEB-INF/web.xml
2. Declare the <taglib> tags for each of the following tag libraries between the <Web-app> and </Web-app> tags:

For This Tag Library	Type The Following Tag
Core (EL)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/core<taglib-uri> <taglib-location>tld/c-1_0.tld</taglib-location> </taglib></pre>
Core (RT)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/core-rt<taglib-uri> <taglib-location>tld/c-1_0-rt.tld</taglib-location> </taglib></pre>
Format (EL)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/fmt<taglib-uri> <taglib-location>tld/fmt-1_0.tld</taglib-location> </taglib></pre>
Format (RT)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/fmt-rt<taglib-uri> <taglib-location>tld/fmt-1_0-rt.tld</taglib-location> </taglib></pre>
SQL (EL)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/sql<taglib-uri> <taglib-location>tld/sql-1_0.tld</taglib-location> </taglib></pre>
SQL (RT)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/sql-rt<taglib-uri> <taglib-location>tld/sql-1_0-rt.tld</taglib-location> </taglib></pre>
XML (EL)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/xml<taglib-uri> <taglib-location>tld/x-1_0.tld</taglib-location> </taglib></pre>
XML (RT)	<pre><taglib> <taglib-uri>http://java.sun.com/jstl/xml-rt<taglib-uri> <taglib-location>tld/x-1_0-rt.tld</taglib-location> </taglib></pre>

3. Save the web.xml file to the original location.

Caution: Do not change the name or the location of the web.xml when saving the file. Doing so may cause errors.

Adding the JSTL to a JSP Web Template

You must add JSTL code to a new JSP Web template or edit an existing JSP Web template to use the JSTL features in your Web application.

1. Open the JSP Web template in an editor.
2. In the line preceding the <HTML> tag, for each functional library that are adding to the JSP, type the Namespace for the functional area library you are using in the JSP Web template:

```
<%a taglib uri="http://java.sun.com/jstl/core" prefix="c" %>
<%a taglib uri="http://java.sun.com/jstl/fmt" prefix="fmt" %>
<%a taglib uri="http://java.sun.com/jstl/xml" prefix="x" %>
<%a taglib uri="http://java.sun.com/jstl/sql" prefix="sql" %>
<%a taglib uri="http://java.sun.com/jstl/core-rt" prefix="c_rt" %>
<%a taglib uri="http://java.sun.com/jstl/fmt-rt" prefix="fmt_rt" %>
<%a taglib uri="http://java.sun.com/jstl/xml-rt" prefix="x_rt" %>
<%a taglib uri="http://java.sun.com/jstl/sql-rt" prefix="sql_rt" %>
```

3. In the appropriate location of the JSP Web template, between the <body> and </body> tags, type the tag code. For example:

```
<fmt:formatDate type="both" timestyle="short" value=${now}">
</fmt:formatDate>
```

4. Save the JSP.

After you create or edit the JSP Web template, you can check your JSP Web template in to the Gentran Integration Suite database. This is optional, unless you intend to use your JSP Web template in Gentran Integration Suite Web Extensions. In this case, you must check in your JSP Web template, so that it can be processed.

Deleting JSTL Code from a JSP Web Template

1. Open the JSP Web template in an editor.
2. In the appropriate location of the JSP Web template, between the <body> and </body> tags, delete the complete JSTL tag code.

For example, if the tag you are deleting is <c:set>, ensure that you delete the </c:set> tag and all attributes associated with the tag.

3. Save the JSP Web template.

Update, Compile, and Redeploy the WAR file in Gentran Integration

Suite

To deploy the Web application in Gentran Integration Suite, you must update and compile the Web application WAR file, which decompresses during deployment.

1. On the Gentran Integration Suite computer, change directories to the *install_dir/container/applications* directory.
2. In the Web application root directory, verify that the following directory structure is followed:
 - ◆ web.xml (Required)
 - ◆ TLD – Directory containing the Tag Library Descriptors (Required)
 - ◆ classes – Directory containing the server-side classes, utility classes, and JavaBeans components (Required)
 - ◆ lib – Directory containing JAR archive of libraries (tag libraries and utility libraries) (Required)
 - ◆ Any miscellaneous subdirectories (Required)
3. Change directories to the *install_dir/bin* directory.
4. Run the Deployer.sh script to deploy all WAR files.

Deploying an External JSP Web Application on Gentran Integration Suite

After you create the JSP Web application WAR file outside of Gentran Integration Suite, you can deploy it to your server that your application server is installed on, so you can use the Web templates you deployed. If you are deploying your Web application in Gentran Integration Suite, complete this procedure:

1. Copy the WAR file to *install_dir/noapp/deploy* directory.
2. Stop Gentran Integration Suite.
3. Start Gentran Integration Suite.

The WAR file is deployed in Gentran Integration Suite, and you can access the Web template by typing <http://localhost:port/webappname>.

Note: If you make changes to the Web application outside of Gentran Integration Suite, you must repackage the Web application as a WAR file and redeploy the Web application in Gentran Integration Suite. You can use the jar utility provided with your JDK to recompile the WAR file.

Troubleshooting JSTL Issues

Follow this methodology when troubleshooting JSTL issues:

Issue	Action
General Errors	<ul style="list-style-type: none"> ◆ Verify that you defined the JSTL Namespace in the JSP Web template. ◆ Verify that you added the complete JSTL tag to the JSP Web template. ◆ Troubleshoot your JSP code.
Class Not Found Errors	<ul style="list-style-type: none"> ◆ Verify that you included the JSTL libraries to the Web application. ◆ Troubleshoot your JSP code.
Page Cannot Display Error	<ul style="list-style-type: none"> ◆ Verify that the prefix defined in JSTL Namespace and tag match and are the same as the <short-name> in the .tld file for that functional area.

JSTL Demonstrations

Gentran Integration Suite contains three demonstrations of simple JSTL functionality located at <http://host:port/helloworld>. Once you go to the helloworld site, you will be prompted for your username and password.

Note: Other demonstrations are available from the helloworld site. Only those listed in the table below are applicable to JSTL.

Demonstration	Description	Steps
JSTL Counting Demonstration	Demonstrates the core functional area features that enable counting in a Web template by letting you type a number to count to and returning the counted sequence.	<ol style="list-style-type: none"> 1. In the Type a number to count text box, type the number you want the computer to count to. 2. Click Go! <p>The list of numbers up to the number you typed in the text box displays in sequence.</p>
JSTL XML Demonstration	Demonstrates the XML functional area features of parsing a document and reformatting the text from the document in a Web browser.	Click Go! next to the list where you selected JSTL XML Demo. The parsed document displays in the JSP Web template.

Demonstration	Description	Steps
JSTL Timezone Demonstration	<p>Demonstrates the formatting functional area Format Timezone feature by letting you select a city to display the time and date of.</p> <p>Note: The demonstration does not adjust for Daylight Savings Time. Therefore, the time may be off by one hour if Daylight Savings Time is in effect for the selected city.</p>	<p>Perform the following steps:</p> <ol style="list-style-type: none"><li data-bbox="1027 302 1370 384">1. From the list, select a city that you want to display the time for.<li data-bbox="1027 396 1166 424">2. Click Go! <p>The time and date for the city you selected displays in the updated JSP Web template.</p>

Managing Web Templates

Gentran Integration Suite uses Web templates to display business process data in Web browsers. The Web template is used to separate the data from the presentation of the data. Managing Web templates is very important to the operations of Gentran Integration Suite Web Extensions.

This chapter covers the following topics:

- ◆ Previewing a Web Template
- ◆ Checking In a Web Template
- ◆ Checking Out a Web Template
- ◆ Searching for a Web Template
- ◆ Deleting a Web Template
- ◆ Maintaining a Web Template Version History
- ◆ Testing Web Templates
- ◆ Preinstalled Web Templates

Previewing a Web Template

You can preview the Web template that you are creating or editing to make sure that it meets your requirements. You can preview your Web templates as many times as needed.

Note: If you created the Web template, you must include the XML namespace convention in your template to preview the template in the Gentran Integration Suite Web Template Editor. If you have not included the XML namespace tags in your Web template, you will receive an error message when you click the Preview button. For more information, see *Preinstalled Web Templates* on page 49.

- ◆ Previewing Web Templates from the Web Templates List
- ◆ Previewing Web Templates in the Web Template Editor
- ◆ Previewing Web Templates Using the XForms Servlet
- ◆ Previewing a Web Template Before Check-In

Previewing Web Templates from the Web Templates List

To preview a Web template from the Web templates list:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, locate the template you want to review by using either the Search or **List** option. For more information, see *Searching for a Web Template* on page 43.
3. Click the **name of the** Web template you want to preview.
4. Preview the Web template in the separate Web browser window and close the Web browser window when you are finished.

Previewing Web Templates in the Web Template Editor

To preview a Web template in the Gentran Integration Suite Web Template Editor:

1. Open a Web template in the Gentran Integration Suite Web Template Editor.
2. In the **Base href** box, type the Base href to use when previewing this Web template, using the format `http://host:port`. The *Base href* specifies the URL that indicates to Gentran Integration Suite where to locate image files or style sheets to use with the template if the HTML code uses relative paths.

For example, `http://hostname:1234`.

If the Web template you want to preview is a preinstalled Web template provided with Gentran Integration Suite, the Base href is already applied.

3. Click **Preview**. The Web template displays graphically in a separate window.
4. Preview the Web template and close the preview window to return to the Web Template Editor.

Previewing Web Templates Using the XForms Servlet

Using the XForms servlet, you can preview any Web template deployed in Gentran Integration Suite that has an `.xfrm` file extension. The XForms servlet reformats the XForms code into HTML, so that a Web browser can display the Web template.

To preview a Web template in a Web browser using the XForms servlet:

1. Point your Web browser to `http://host:port/websuite`.
2. Directly after `websuite` in the Web browser Address box, type a back slash (`/`), the Web template name and the `.xfrm` extension of the Web template you want to preview.

For example, `http://host:port/websuite/webtemplate.xfrm`.

3. To the right of the Address box in your browser, click **Go**. The Web template displays graphically in the Web browser.

Previewing a Web Template Before Check-In

At any time during the process of creating or editing a Web template, you can preview the Web template before you check the Web template in to Gentran Integration Suite for use in business processes. You can preview Web templates to view the layout and verify that the Web template meets your needs.

To preview a Web template before you check the Web template in to Gentran Integration Suite:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**. The Web Templates page opens.
2. In the Web Templates page, under **List in the Alphabetically field**, select **ALL** and click **Go!** The Web Templates page opens, listing all Web templates that match your search criteria.
3. Click **source manager** next to any of the templates.
4. In the Web Template Source Manager page, click **edit** next to the Web template.
5. In the Web Template Editor page, in the Web Template box, select and delete all of the source code.
Do not type a comment in the comment field and click Save after you have deleted all the source code; otherwise, the code for the Web template will be replaced with the blank code in the Web template box.
6. Copy the source code of the Web template.
Ensure that this is the XForms code opened in a text editor or in the Web Template Designer. Do not use code opened in a Web browser, as this may include code that may cause errors in the display of the Web template.
7. In the Web Template box, paste the source code from the Web Template Designer and click **Preview**.
If you know the location of the image files or style sheets to use with the template, type the URL in the Base href field.

Checking In a Web Template

If you create or edit a Web template *outside* of Gentran Integration Suite, you can check it in to the Gentran Integration Suite database to use in business processes.

To check in a Web template:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**. The Web Templates page opens.
2. To the right of Check-in new Web Template, click **Go!**
3. In the **Name** field, type a name for the Web template.
4. Under Select an input mode for defining the new template, select one of the following and click **Next**:
 - ◆ Check-in Web template created by the graphical tool
 - ◆ Web Template Text Editor

5. Complete the steps for your selection in step 4:

If You Selected	Complete These Steps
Check-in Web Template created by the graphical tool	<ol style="list-style-type: none"> 1 In the Web Template filename field, complete one of the following actions: <ul style="list-style-type: none"> ◆ Type the file name and extension of the Web template to check in. ◆ Click Browse to navigate to the location where the Web template was saved, and select the file name of the template. <p>If a version of the Web template you are checking in already exists in the Gentran Integration Suite database, the Version page opens, enabling you to select the version of the template to check in. Select the version and click Next.</p> 2 In the Check-in Comments field, type any comments you want (required). Gentran Integration Suite uses the comments you type in this field to establish a version for each template. Ensure that the comments are descriptive to help you maintain version control. 3 From the Encoding Type list, select the encoding type to use for the Web template you are checking in to Gentran Integration Suite and click Next.
Web Template Text Editor	<ol style="list-style-type: none"> 1 In the Comments field, type any comments you want (required). Gentran Integration Suite uses the comments you type in this field to establish a version for each template. Ensure that the comments are descriptive to help you maintain version control. 2 In the Web Template box, paste the Web template code. 3 In the Base href box, type the Base href to use when previewing this template, using the format <code>http://hostname:nnnn</code>. The Base href specifies the URL that indicates to Gentran Integration Suite where to locate image files or style sheets to use with the template if the HTML code uses relative paths. You can also specify the <code>http://host:port</code> for your Gentran Integration Suite environment to preview Web templates.

4. In the Confirm page, complete the following actions and click **Finish**:

- ◆ Review the information for accuracy.
- ◆ Select **Enable for Business Processes** to enable the Web template to be used in business processes.

Checking Out a Web Template

After a Web template is checked in, you can check it out of the Gentran Integration Suite database. Checking out a template does not remove it from the Gentran Integration Suite database; it saves a copy of the template from the host to your client computer.

To check out a Web template:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.

2. In the Web Templates page, locate the template you want to check out by using either the Search or List option.

For more information, see *Searching for a Web Template* on page 43.

3. Click **source manager** next to the Web template you want to check out.
4. In the Web Template Source Manager page, select the outgoing encoding type from the list next to the Web template you are checking out of Gentran Integration Suite.

The original encoding type, if not UTF-8; otherwise, UTF-8 encoding type is used during checkout.

5. Click **check-out** next to the Web template you want to check out.

You may receive a message prompting you whether to lock the file. Click **OK** to lock the file and continue with the check out process.

6. In the **File Download** dialog box, click **Save** to save the Web template to your computer.
7. In the **Save As** dialog box, select the location where you want to save the template and click **Save**. The copy of the template is saved to the new location.
8. In the **Download complete** dialog box, click **Close**.

The Web template is now saved on your local computer. You can now edit the Web template and then check in a new version.

Searching for a Web Template

After a Web template is checked in, you can search for that template.

To search for a Web template:

1. From the **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, complete one of the following actions:
 - ◆ Under Search in the **Web Template Name** field, type either a portion of the name or the entire name of the Web template you are searching for and click **Go!** The Web Templates page opens, listing all Web templates containing the full or partial name you typed.
 - ◆ Under **List in the Alphabetically** field, select **ALL** or the letter that begins the name of the Web template you are looking for. Selecting **ALL** lists all Web templates that are checked in. Click **Go!** The Web Templates page opens, listing all Web templates that match your search criteria.
3. Depending on your task, complete one of the following actions:
 - ◆ Go to Source Manager to check in or check out the templates. For more information, see *Checking In a Web Template* on page 41 and *Checking Out a Web Template* on page 42.
 - ◆ Go to Version Manager to set a default version of the template and to enable or disable templates for business process use. For more information, see *Maintaining a Web Template Version History* on page 44.
 - ◆ Go to the Web Template Editor to edit the template and save your changes. For more information, see *Editing a Web template* on page 65.

Deleting a Web Template

After a Web template is checked in, you can delete it from the Gentran Integration Suite database. Deleting a Web template removes it completely from the Gentran Integration Suite database. Before you delete a Web template, check it out of Gentran Integration Suite and save a copy to another location for future reference.

To delete a Web template:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, locate the template you want to delete by using either the Search or List option.

For more information, see *Searching for a Web Template* on page 43.

3. Click **source manager** next to the Web template you want to check out.
4. In the Web Template Source Manager page, click the check box in the Delete column next to the Web template you are deleting from Gentran Integration Suite.
5. Next to Delete Selected Versions, click **Go!**
You are prompted to confirm the deletion. Click **OK**.
6. In the Delete Resources page, review the information on the Web template and click **Next**.
7. In the Confirm page, a message displays stating *Deleting this resource may cause processes to fail. This action cannot be reversed*. Click **Delete** to delete the Web template.

The Web template is deleted from the database.

Maintaining a Web Template Version History

Gentran Integration Suite enables you to maintain versions of your Web templates.

A *version* is an edited copy of a Web template. Each time a Web template is edited, a new version is created, but the original Web template remains intact for future use. A Web template version starts when you type a description in the Comments field while you check in a Web template. Because this description provides the basis for maintaining the version for that Web template, you should create the description carefully. In addition, Gentran Integration Suite adds a date and time stamp to each version that can be used for version identification.

To maintain Web template version:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Templates**.
2. In the Web Templates page, locate the template you want to review by using either the Search or **List** option.

For more information, see *Searching for a Web Template* on page 43.

3. Next to the Web template you want to review, click **version manager**.

4. In the Web Template Version Manager page, click **Default** next to the Web template version you want to set as the default template. This is the version of the template to be used for business processes using this template.
5. Do you want to enable the Web template to make it available to business processes?
 - ◆ If Yes, ensure that the check box next to the Web template to enable is selected.
 - ◆ If No, ensure that the check box next to the Web template to disable is cleared.
6. Click **Return**. Your changes are applied to the templates in Gentran Integration Suite.
7. Repeat steps 1 through 6 for other templates you want to review.

Testing Web Templates

To ensure that the Web templates you have designed send data correctly, you should test each Web template before putting the template into production.

Testing a Web template involves the following steps:

1. Create and check your Web templates in to Gentran Integration Suite. This includes all Web templates that will send and receive the data.

For more information, see *Managing XForms Structure Elements* on page 69 and *Checking In a Web Template* on page 41.

So that you can test any Web template you want, create and check in the following sample Web template named, *sendMe.xfm*. If you do not create and check this Web template in to Gentran Integration Suite, you will not be able to run the test as is described in these instructions. You can create your own test scenarios.

```
<?xml version='1.0' encoding="UTF-8"?>
<HTML xmlns:xf="http://www.w3.org/2002/01/xforms"
xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:scixf="http://www.sterlingcommerce.com/xforms/xforms-extensions">
<head>
  <xf:model id="docInfo">
    <xf:instance xlink:href="xforms:request.xml#instanceData/docInfo">
    </xf:instance>
  </xf:model>
  <xf:model id="vars">
    <xf:instance scixf:preview-href="inline" scixf:proxy="1"
xlink:href="{model('docInfo')/docInfo/sendURI}">
      <docInfo xmlns="">
        <sendURI>aaa</sendURI>
        <documentType>bbb</documentType>
      </docInfo>
    </xf:instance>
  </xf:model>
</head>
<body>
  <xf:group id="xform-body">
Document has been submitted to      <xf:output model="docInfo" ref="docInfo/sendURI">
```

```

    </xf:output>
  </xf:group>
</body>
</HTML>

```

2. Create the WebTemplateTest business process using the Graphical Process Modeler.

For more information, see *Creating the WebTemplateTest Business Process* on page 46.

You must complete this step only once, since you can reuse the WebTemplateTest business process to test all of your Web templates.

After you have verified that a Web template sends data correctly, you must change the URIs in the Web template to work with the appropriate business process that Web template was designed to work with.

3. Check the WebTemplateTest business process in to Gentran Integration Suite.

For more information, see *Checking In the WebTemplateTest Business Process* on page 47.

4. Add a URI named WebTemplateTest to the list of URIs of the Web Extensions HTTP Server adapter.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

For more information, see *Adding a URI to the Web Extensions HTTP Server Adapter* on page 47.

You must complete this step only once, because the URI remains in place until it is removed from the Web Extensions HTTP Server Adapter list.

5. Log in to Gentran Integration Suite Web Extensions to authenticate yourself to Gentran Integration Suite.

For more information about the Gentran Integration Suite work page, see *Expense Report Business Process* on page 59.

6. Open the input Web template and enter data to be submitted.

7. View the business process data in Gentran Integration Suite to verify that the data was received properly.

Creating the WebTemplateTest Business Process

To test a Web template before placing the Web template into production you must create a business process that can process the data entered into the Web template. You can create a test business process of the actual business process that you want to use in production, or you can create the WebTemplateTest business process to always use as your test business process.

To create the WebTemplateTest business process:

1. From the Gentran Integration Suite **Business Process** menu, select **Manager**.
2. In the Graphical Modeling section, click **Go!**
3. From the **File** menu, select **New** to open a new document.
4. Open the stencils containing the elements you must create the business process. From the **View** menu, select:

- ◆ **Stencils > BPML**
 - ◆ **Stencils > Communications**
5. Click the title bar for the BPML stencil to expand the stencil.
 6. Add the following activities to the business process workspace by dragging and dropping the icons onto the workspace:
 - ◆ Start activity
 - ◆ End activity
 7. Click the title bar for the Communications stencil to expand the stencil.
 8. Add the **B2B HTTP service** to the business process workspace by dragging and dropping the icon onto the workspace. Place the HTTP service between the **Start** and the **End** activities.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.
 9. Click and drag the arrow to the right side of the **Start** icon to the left side of the **HTTP service** icon.
 10. Click and drag the arrow to the right side of the **HTTP service** icon to the left side of the End activity icon.
 11. Double-click the HTTP service to open the service and display the parameters in the Service editor.
 12. From the **Config** field list, select **WEB_EXTENSIONS_HTTP_SERVER_ADAPTER**.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.
 13. From the **File** menu, select **Save As**.
 14. In the **Validate** dialog box, click **Yes** to validate upon saving.
 15. In the **Save** dialog box, select the location you want to save the business process to, type **WebTemplateTest** in the **File name** box, and click **Save**.

Checking In the WebTemplateTest Business Process

After you create the WebTemplateTest business process, you must check the business process in to Gentran Integration Suite. After you check in the business process, you can use the WebTemplateTest business process to test any Web template that you create.

Note: If you have created the test business process under a different name, ensure that you check in the business process under that name and not WebTemplateTest.

Adding a URI to the Web Extensions HTTP Server Adapter

You must add a URI to the Web Extensions HTTP Server adapter for the business process to work properly.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

Note: An error will occur if you do not assign a URI to the Web Extensions HTTP Server adapter that matches the URI in the Web template you are testing.

To add a URI to the Web Extensions HTTP Server adapter:

1. From the Gentran Integration Suite **Deployment** menu, select **Services > Configuration**.
2. In the Services Configuration page, under List, select the letter **W** and click **Go!**
3. Next to the Web Extensions HTTP Server adapter, click **edit**.
4. In the left-side of the page, under Steps, click **URIs**.
5. In the Web Extensions HTTP Server adapter:URIs page, click **add**.
6. In the URIs:Specification page, in the **URI** field, type **webx/bp/WebTemplateTest**.
If you created the business process under a different name, ensure that the URI matches the URI listed in your Web templates; otherwise, an error may occur.
7. In the **Business Process** field, select from the list **WebTemplateTest**, or the name of the business process you created and click **Next**.
8. Verify that the new URI is on the list and click **Next**.
9. In the Confirm page, verify that the URI is listed and click **Finish**.

The URI is added to the Web Extensions HTTP Server adapter configuration.

Testing a Web Template

You can test a Web template after you check in a Web template, create and check in the WebTemplateTest business process, and add the appropriate URI to the Web Extensions HTTP Server adapter.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

To test a Web template:

1. Point your Web browser to `http://host:port/websuite`.
2. At the login page, type your user name and password that your manager provides.
3. Gentran Integration Suite verifies your user name and password against the database.
 - ◆ If you do not log in correctly, you receive an error message. Verify your user name and password and try to log in again.
 - ◆ If you log in correctly, your personal work page Inbox opens. The default is the Inbox page.

You do not need to log in each time that you want to test a Web template as long as you are already logged in to Gentran Integration Suite Web Extensions or have not closed the Web browser.

4. In the Web browser **Address** field, highlight and delete the contents so that only the following address displays:
`http://host:port/websuite/xfm/`
5. Directly after the xfm/, type the name of the Web template you want to test.
 For example, `http://host:port/websuite/xfm/Testdoc.xfm`
6. In the Web template, type information and make selections as appropriate to test each XForms control element and HTML element, and click the button to perform the required action.
 For example, in the Testdoc.xfm file, complete the following test:
 - a. Type the client information in the appropriate fields.
 - b. Click **Reset** to reset the field entries to the initial values.
 - c. Click **Sterling Commerce** to visit the Sterling Commerce Web site.
 - d. Click your browser's **Back** button.
 - e. Click **Submit** to submit the data to the WebTemplateTest business process.
7. From the Gentran Integration Suite **Business Process** menu, select **Monitor > Current Processes**.
8. Select the ID of the current business process that ran when you submitted the data from the Web template.
9. In the Business Process Detail page, click **info** under Document to verify that the data you submitted was received and displayed.

Preinstalled Web Templates

Web templates contain markup tags that enable business process data to be displayed at any point in a business process. Gentran Integration Suite comes with the following preinstalled Web templates for use in business processes.

Note: Web templates marked with an asterisk (*) are not associated with a predefined business process or application. They can be used as a starting point for developing Web templates to meet your needs.

Web Template	Description
BPSSCorrelationSearch	Displays search parameters for the BPSS Correlation search feature.
BPSSCorrelationSearchResults	Displays a list of the BPSS Correlation search results, that matched the search parameters.
CentralSearch	Displays search parameters for the Central Search feature.
ConfirmSend	Displays while your document is being sent, to confirm that the document is being sent.
CorrelationSearch	Displays search parameters for the Correlation search feature.

Web Template	Description
CorrelationSearchResults	Displays a list of the Correlation search results that matched the search parameters.
DocumentSummary	Displays a summary of documents.
EDICoorelationDetail	Displays details from the EDI correlation search.
EDICorrelationGrpSummary	Displays EDI Group information in a summary format including: status, sender ID, receiver ID, sender ID qualifier, receiver ID qualifier, group ID, group control number, and date.
EDICorrelationIntSummary	Displays EDI Interchange information in a summary format, including: status, sender ID, receiver ID, sender ID, qualifier, receiver ID qualifier, standard, interchange control number, and date.
EDICorrelationSearch	Displays the search parameters for the EDI Correlation search feature.
EDICorrelationSetSummary	Displays the EDI correlation transaction set information in a summary format, including: status, sender ID, receiver ID, sender ID qualifier, receiver ID qualifier, set ID, interchange control number, group control number, transaction set control number, and details.
ErrorPage	Displays when an error is encountered running a service.
FileUploadConfirmSend	Displays while your document is being uploaded, to confirm that the document is being sent.
FileUploadTemplate	Displays the File Upload form that you use to upload documents to a business process.
GDSApprovalRequest	Displays the Approval Request form for GDS, enabling you to approve line item additions and updates in GDS.
GDSApproverList	Displays the list of GDS approvers, including the approver's name, type or approver, and value. You can also set up an approver through this Web template.
GDSApproverView	Displays the form to update approver types and categories.
GDSCatalogAdd	Displays the form to add a catalog item to the catalog.
GDSCatalogList	Displays a list of catalog items including the item number, product name, brand, category, and retail price.
GDSCatalogPending	Displays catalog information for an item that is pending approval.
GDSCatalogPendingList	Displays a list of catalog items pending approval.
GDSCatalogUpdate	Displays the form enabling you to update a catalog entry.
GDSCatalogView	Displays the catalog information for a catalog item.
GDSCategoryManager	Displays the approver, document, and status of an item's approval.
GDSChangePassword	Displays the form enabling you to change your GDS password.
GDSChangePasswordConfirm	Displays a message confirming the password change.

Web Template	Description
GDSConfirmSend	Displays while your document is being sent, to confirm that the document is being sent.
GSSDBConfirm	Displays a message indicating the number of database rows that are affected by the submitted and approved change.
GDSQueryList	Displays the contents of a message box in the Message List area of the work page. For example, if you click Outbox, the template displays the list of messages in the Outbox.
GDSQuerySearch	Displays the form enabling you to search for messages in your message boxes.
GSAvanceSearch	Displays advance search parameters that are available when Gentran:Server [®] UNIX [®] is selected in the Central Search feature in the Gentran Integration Suite interface.
GSLifeCycleDetail	Displays the results of an advance search on the Gentran:Server UNIX Life Cycle database.
GSLifeCycleSummary	Displays a summary of the information in the Gentran:Server UNIX Life Cycle database.
GSSApprovalReq	Displays the Approval Request form for GSS, enabling you to approve line item additions and updates in GSS.
GSSApproverList	Displays the list of GSS approvers, including the approver's name, type or approver, and value. You can also set up an approver through this Web template.
GSSApproverView	Displays the form to update approver types and categories.
GSSCatalogAdd	Displays the form to add a catalog item to the catalog.
GSSCatalogList	Displays a list of catalog items including the item number, product name, brand, category, and retail price.
GSSCatalogPending	Displays catalog information for an item that is pending approval.
GSSCatalogPendingList	Displays a list of catalog items pending approval.
GSSCatalogUpdate	Displays the form enabling you to update a catalog entry.
GSSCatalogView	Displays the catalog information for a catalog item.
GSSCategoryManager	Displays the approver, document, and status of an item's approval.
GSSChangePassword	Displays the form enabling you to change your GSS password.
GSSChangePasswordConfirm	Displays a message confirming the password change.
GSSConfirmSend	Displays while your document is being sent, to confirm that the document is being sent.
GSSDBConfirm	Displays a message indicating the number of database rows that are affected by the submitted and approved change.

Web Template	Description
GSSQueryList	Displays the contents of a message box in the Message List area of the work page. For example, if you click Outbox, the template displays the list of messages in the Outbox.
GSSQuerySearch	Displays the form enabling you to search for messages in your message boxes.
Invoice*	Used to create an invoice.
InvoiceCh*	Used to create an invoice displayed in Chinese.
MailboxAdd	Displays the form to add a message to a mailbox.
MailboxAddDisabled	Displays the form to add a message to a mailbox with disabled browse and send buttons.
MailboxChangePassword	Displays the form enabling you to change your mailbox password.
MailboxChangePasswordConfirm	Displays a message confirming the password change.
MailboxConfirmSend	Displays while your document is being sent, to confirm that the document is being sent.
MailboxDirList	Displays a list of search results from the mailbox query.
MailboxErrorPage	Displays an error message stating that an error was encountered while running the service.
MailboxSearch	Displays the search form enabling you to search the mailbox contents.
POChange*	Used by a reviewer of a purchase order to make changes to the original purchase order.
POChangeAcknowledgment*	Used to create a purchase order change acknowledgment.
PurchaseOrder	Used to create a purchase order in the Purchase Order predefined business process.
PurchaseOrderConfirmSend	Displays while your purchase order is being sent, to confirm that the purchase order is being sent.
PurchaseOrderMenu	Displays the View Folders and Create New areas of the Gentran Integration Suite work page.
PurchaseOrderQuery	Displays the View Folders, Create New, and Message List areas of the Gentran Integration Suite work page.
PurchaseOrderView	Displays the purchase order to the approving authority.
Query*	Search form for searching the storage areas by different parameters.
QueryList*	Displays the results of the Query Web template.
RequestForQuote*	Used to create a request for quote.
RFQResponse*	Used to send a response to a request for quote.
SearchResults	Displays the number of Gentran Integration Suite and Gentran:Server UNIX matches from the search.

Web Template	Description
WebSuiteApprovalNotify	Shows that the document submitted was approved.
WebSuiteASN	Used to create an advance ship notice in the Purchase Order predefined business process.
WebSuiteASNView	Displays the advance ship notice to the buyer.
WebSuiteChangePassword	Displays the form enabling you to change your Web Extensions password.
WebSuiteChangePasswordConfirm	Displays a message confirming the password change.
WebSuiteExpenseReport	Used to create an expense report in the Expense Report predefined business process.
WebSuiteExpenseReportView	Displays the expense report to the approving authority.
WebSuiteInvoice	Used to create an invoice in the Invoice predefined business process.
WebSuiteInvoiceView	Displays the invoice to a buyer.
WebSuitePO	Used to create a purchase order in the Purchase Order predefined business process.
WebSuitePOAck	Used to create a purchase order in the Purchase Order predefined business process.
WebSuitePOAckView	Displays the purchase order acknowledgment to the buyer.
WebSuitePOView	Displays the purchase order to the approving authority.
WebSuitePurchaseReq	Used to create a purchase order in the Purchase Order predefined business process.
WebSuitePurchaseReqView	Displays the purchase order to the approving authority.
WebSuiteQueryList	Displays the contents of a message box in the Message List area of the Gentran Integration Suite work page. For example, if you click Outbox, the template displays the list of messages in the Outbox.
WebSuiteQuerySearch	Displays the form enabling you to search the Web Extensions message boxes.
WebSuiteRemittanceAdvice	Used to create a remittance advice in the Remittance Advice predefined business process.
WebSuiteRemittanceAdviceView	Displays the remittance advice to the supplier.
WebSuiteTimeSheet	Used to create a time sheet in the Time Sheet predefined business process.
WebSuiteTimeSheetView	Displays the time sheet to the approving authority.

When Gentran Integration Suite Web Extensions is installed, these Web templates are checked in to the Gentran Integration Suite database and enabled for use in business processes. Web designers can also edit these templates for specific needs. For more information about locating Web templates, see *Searching for a Web Template* on page 43.

Predefined Business Processes

Gentran Integration Suite Web Extensions includes predefined business processes that are ready for use upon installation. You can save time and money by using these predefined business processes as a starting point and customizing them to meet your needs.

This chapter covers the following topics:

- ◆ About Predefined Business Processes
- ◆ User Roles and Permissions
- ◆ Expense Report Business Process
- ◆ Purchase Requisition Business Process
- ◆ Time Sheet Business Process
- ◆ Purchase Order Business Process
- ◆ Remittance Advice Business Process
- ◆ Invoice Business Process
- ◆ Advance Ship Notice Business Process
- ◆ Gentran Integration Suite Work Page
- ◆ Managing the Gentran Integration Suite Work Page
- ◆ Customizing Predefined Business Processes

About Predefined Business Processes

Web Extensions includes predefined business processes that you can use as a starting point to configure business processes to meet your needs. The Web Extensions predefined business processes support human interaction with the business process.

Each predefined business process:

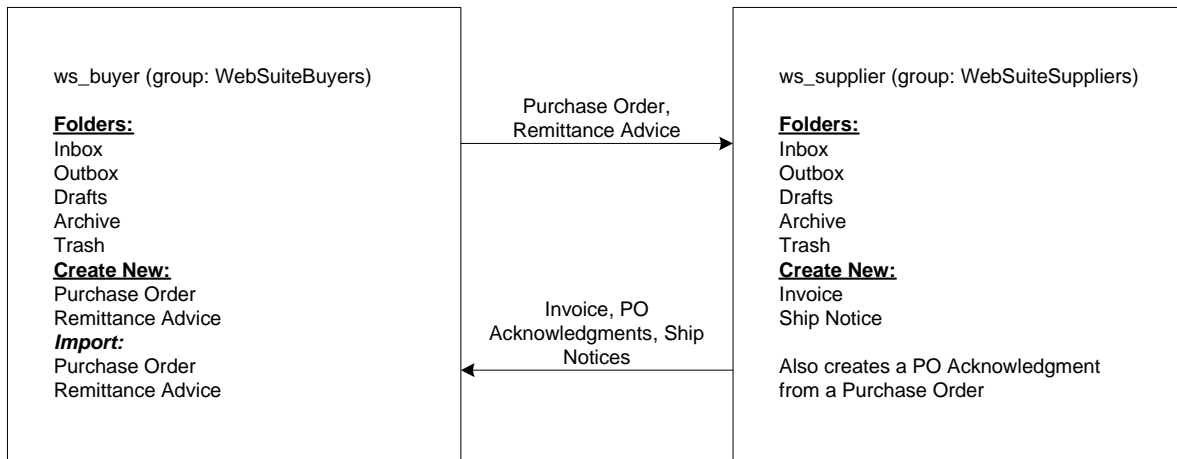
- ◆ Uses the Human Interaction services and Web templates during execution
- ◆ Is preconfigured and ready to use when installed
- ◆ Is configurable to meet your specific needs

- ◆ Uses groups and permissions to determine access rights to business processes and Web templates
- ◆ Is made up of a series of smaller interconnected business processes

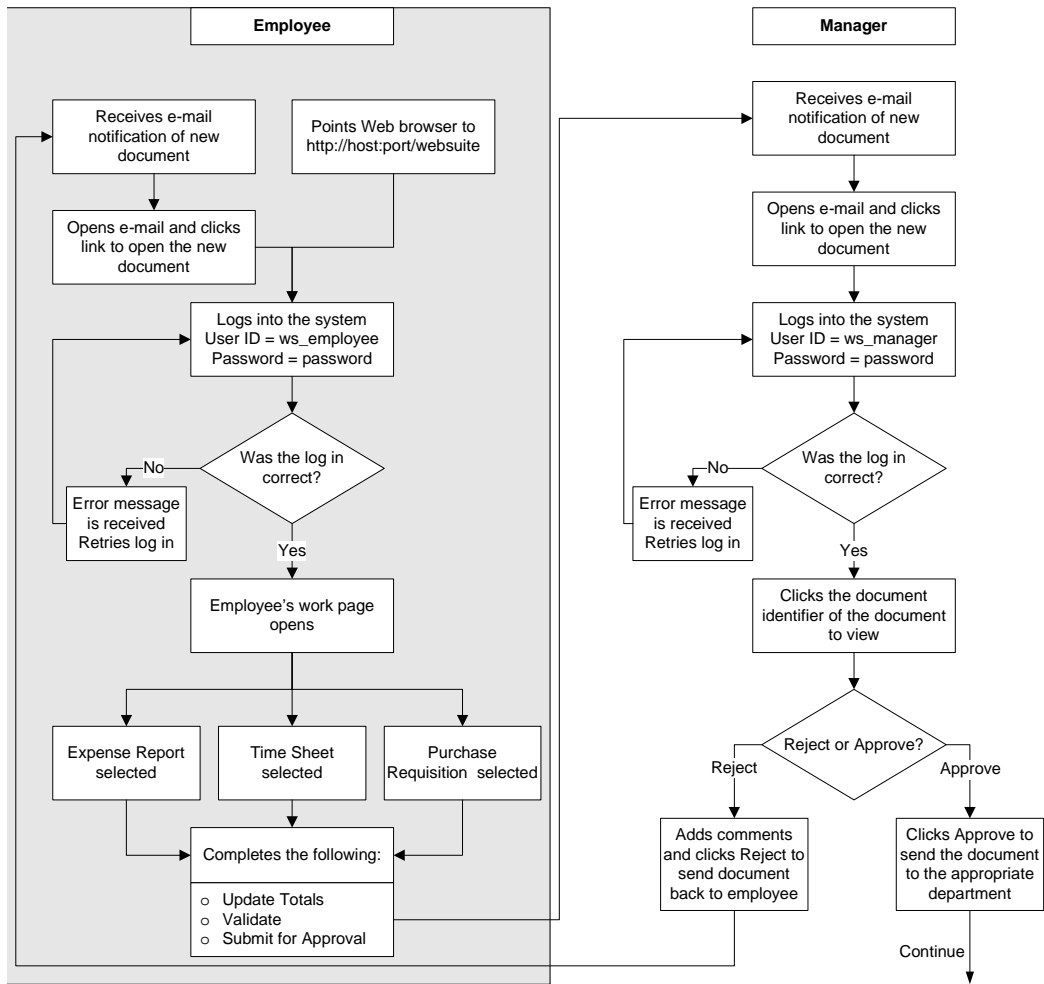
Web Extensions predefined business processes are like any other business processes that you create, in that each consists of a series of activities that accomplishes a business goal. Each step in the business process is an activity that moves the process forward to the business goal.

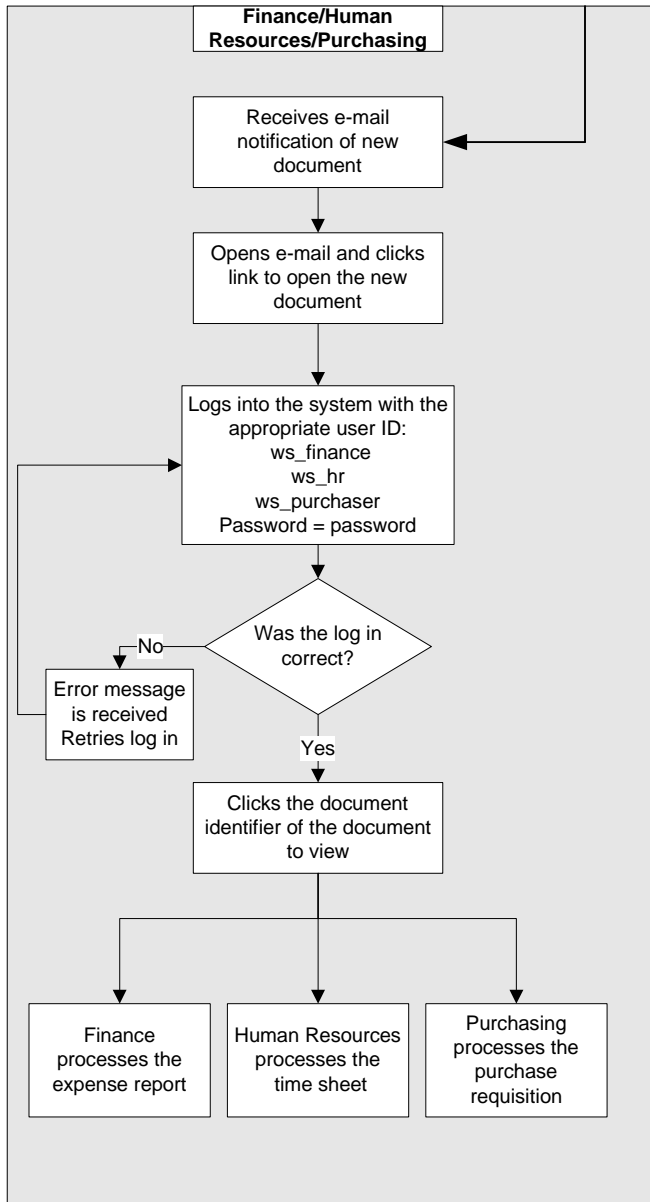
Gentran Integration Suite Web Extensions includes two types of predefined business processes, Buyer-to-Supplier (internal-to-external business partners) business processes and Employee-to-Manager (internal-to-internal employees) business processes. The Employee-to-Manager business processes support interaction between your employees, such as employees completing time sheets, expense reports, and purchase requisitions and sending them to managers for approval. The Buyer-to-Supplier business processes support external company interaction between you and your trading partners, such as the exchange of purchases orders, purchase order acknowledgments, invoices, remittance advices, and advance ship notices.

The following figure shows how the Buyer-to-Supplier predefined business processes are related:



The following figure shows how the Employee-to-Manager predefined business processes are related:





Web templates link the business processes within each additional predefined business process. When initiated, the Web template starts the next business process in the chain and helps move the predefined business process toward its business goal.

The Web Extensions predefined business processes and their associated templates are:

Business Process	Associated Web Template
Advanced Ship Notice	WebSuiteASN and WebSuiteASNView
Expense Report	WebSuiteExpenseReport and WebSuiteExpenseReportView

Business Process	Associated Web Template
Invoice	WebSuiteInvoice and WebSuiteInvoiceView
Purchase Order	WebSuitePO, WebSuitePOAck, WebSuitePOView
Purchase Requisition	WebSuite Purchase Req and WebSuitePurchaseReqView
Remittance Advice	WebSuiteRemittanceAdvice and WebSuiteAdviceView
Time Sheet	WebSuiteTimeSheet and WebSuiteTimeSheetView

User Roles and Permissions

Gentran Integration Suite Web Extensions comes with the following generic user names (password = password) to demonstrate how the role-based security features work within the predefined business processes. After Web Extensions is installed, access <http://host:port/websuite>, and log in using the different user names. The following table shows the different permissions for each role:

User Name	Role	Documents Allowed to Create
Buyer-to-Supplier		
ws_supplier	Supplier of goods to the buyer role.	<ul style="list-style-type: none"> ◆ Invoice ◆ Advance Ship Notice ◆ Purchase Order Acknowledgment
ws_buyer	Buyer of an organization. The ws_buyer buys goods from the supplier role.	<ul style="list-style-type: none"> ◆ Purchase Order ◆ Remittance Advice
Employee-to-Manager		
ws_employee	Employee	<ul style="list-style-type: none"> ◆ Expense Report ◆ Time Sheet ◆ Purchase Requisition
ws_manager	Manager who receives documents from ws_employee.	View-only
ws_hr	Human Resources employee that receives the time sheet from ws_manager.	View-only
ws_finance	Finance employee that receives the expense reports from ws_manager	View-only
ws_purchaser	Purchasing employee that receives the purchase requisitions from ws_manager.	View-only

These generic user names, passwords, and permissions are automatically deployed upon installation of Web Extensions. The Gentran Integration Suite system administrator can change the configuration after installation. For more information about role-based security, see *Role-Based Security* on page 15.

Expense Report Business Process

The Expense Report predefined business process automates and streamlines the process of completing an expense report, submitting that expense report, approving or denying the expense report, and sending the expense report to the company's Finance department.

Expense Report Business Process Example

Tom is an employee of ABC company. He returns from a business trip to Brazil and must complete an expense report for the charges to his credit card during the trip. Tom completes the expense report and sends it to his manager, Sue, for approval. Because the expenses are over \$1000.00 (U.S.), she must review the report for accuracy and either approve or reject the expense report. If Sue approves the expense report without correction, she forwards the expense report to the Finance department for payment. If Sue rejects the expense report, she sends the expense report back to Tom for correction and starts the submit-and-review cycle again. This cycle can repeat an unlimited number of times.

Within the preceding scenario, the Expense Report predefined business process helps Tom and Sue perform their tasks. The business process also helps John, who works in the Finance department, to complete his tasks.

1. Tom points his Web browser to `http://host:port/websuite`. Sue received this URL from the Gentran Integration Suite system administrator.
2. At the login page, Tom logs in to Gentran Integration Suite using "ws_employee" as his user name and "password" as his password.
3. Gentran Integration Suite verifies Tom's user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom's user name and password, Tom has permissions to the Expense Report template. Tom clicks the Expense Report link under the Create New topic. A blank Expense Report Web template displays.
6. Tom completes the Expense Report Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the expense report is technically accurate.

- ◆ **Submit for Approval** to send the completed expense report to Sue for her review.

The Expense report Web template is validated and sent to Sue's Gentran Integration Suite Inbox. An e-mail notification is sent to Sue with a link to Tom's expense report.

7. Sue opens the e-mail message and clicks the link to Tom's expense report. She is prompted for her user name and password. Sue logs in to Gentran Integration Suite using "ws_manager" as her user name and "password" as her password.
8. Sue's Gentran Integration Suite work page shows she has a new expense report from Tom. Sue opens Tom's expense report by clicking the document identifier.
9. As a manager, Sue has only approve or reject privileges to the Expense Report template. She cannot edit the expense report.
10. Does Sue approve Tom's expense report?
 - ◆ If Yes, she clicks **Approve** and sends the expense report to Finance for payment.
 - ◆ If No, Sue types any comments regarding the rejection in the Comments text box, clicks **Reject**, and sends the expense report back to Tom for correction and resubmission. This starts the submit-and-review cycle again.

Sue clicks **Approve** and sends Tom's expense report to Finance for payment.

11. An e-mail notification is sent to John in the Finance department, alerting him that a document is awaiting his attention. John opens the e-mail message and clicks the link to the document.
12. John is prompted to log in. John logs in to Gentran Integration Suite using "ws_finance" as his user name and "password" as his password. John's Gentran Integration Suite work page shows that he has a new expense report from Sue.
13. John opens Tom's expense report by clicking on the document identifier.
14. As a Finance administrator, John has only view privileges to the Expense Report Web template. He cannot edit the expense report. John reviews Tom's expense report and processes it for payment.

Purchase Requisition Business Process

The Purchase Requisition predefined business process automates and streamlines the process of completing a purchase requisition, submitting the purchase requisition, approving or denying the purchase requisition, and sending the purchase requisition to the Purchasing department within a company.

Purchase Requisition Business Process Example

Tom is an employee of ABC company. Tom needs a new computer and must complete a purchase requisition to have the Purchasing department buy the computer for him. Tom completes the purchase requisition and sends it to his manager, Sue, for approval. Sue must review all purchase requests for need and accuracy. Sue then must either approve or reject the purchase requisition. If Sue approves the purchase requisition without correction, she forwards the purchase requisition to the Purchasing department for processing. If Sue rejects the purchase requisition, she sends the purchase requisition back to Tom for correction and starts the submit-and-review cycle again. This cycle can repeat an unlimited number of times.

Within the preceding scenario, the Purchase Requisition predefined business process helps Tom and Sue perform their tasks. The business process also helps Jake, who works in the Purchasing department, to complete his tasks.

1. Tom points his Web browser to `http://host:port/websuite`. Sue received this URL from the Gentran Integration Suite system administrator.
2. At the login page, Tom logs in to Gentran Integration Suite using “ws_employee” as his user name and “password” as his password.
3. Gentran Integration Suite verifies Tom’s user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom’s user name and password, Tom has permission for the Purchase Requisition Web template. Tom clicks the Purchase Requisition link under the Create New topic. A blank Purchase Requisition Web template opens in Tom’s Web browser.
6. Tom completes the Purchase Requisition Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the purchase requisition is technically accurate.
 - ◆ **Submit for Approval** to send the completed purchase requisition to Sue for her review.

The Purchase Requisition Web template is validated and sent to Sue’s Gentran Integration Suite Inbox. An e-mail notification is sent to Sue with a link to Tom’s purchase requisition.
7. Sue opens the e-mail message and clicks the link to Tom’s purchase requisition. She is prompted for her user name and password. Sue logs in to Gentran Integration Suite using “ws_manager” as her user name and “password” as her password.
8. Sue’s Gentran Integration Suite work page shows that she has a new purchase requisition from Tom.
9. Sue opens Tom’s purchase requisition by clicking the document identifier, which is a unique name or numeric string given to each document within Gentran Integration Suite.
10. As a manager, Sue has only approve or reject privileges to the Purchase Requisition Web template. She cannot edit the purchase requisition, but can add comments to the comments text box.
11. Does Sue approve Tom’s purchase requisition?
 - ◆ If Yes, she clicks **Approve** and sends the purchase requisition to Purchasing for processing.
 - ◆ If No, Sue types any comments regarding the rejection in the Comments text box, clicks **Reject**, and sends the purchase requisition back to Tom for correction and resubmission. This starts the submit-and-review cycle again.
12. Sue clicks **Approve** and sends Tom’s purchase requisition to Purchasing for Tom’s computer to be ordered.

13. An e-mail notification is sent to Jake in the Purchasing department, alerting him that a document is awaiting his attention. Jake opens the e-mail message and clicks the link to the document.
14. Jake is prompted to log in. Jake logs in to Gentran Integration Suite using “ws_purchaser” as his user name and “password” as his password.
15. In the Purchasing department, Jake logs in to Gentran Integration Suite, and his Gentran Integration Suite work page shows that he has a new purchase requisition from Sue.
16. Jake opens Tom’s purchase requisition by clicking the document identifier.
17. As a Purchasing administrator, Jake has only view privileges to the Purchase Requisition Web template. He cannot edit the purchase requisition. Jake reviews Tom’s purchase requisition and orders Tom’s new computer.

Time Sheet Business Process

The Time Sheet predefined business process automates and streamlines the process of completing a time sheet, submitting that time sheet, approving or denying the time sheet, and sending the time sheet to the Human Resources department within a company.

Time Sheet Business Process Example

Tom is an employee of ABC company. It is the end of the month and he must complete a time sheet to account for the work hours he spent on different projects and the hours he spent out of the office. Tom completes the time sheet and sends it to his manager, Sue, for approval. Sue must review all time sheets for accuracy and either approve or reject the time sheet. If Sue approves the time sheet without correction, she forwards the time sheet to the Human Resources department for processing. If Sue rejects the time sheet, she sends the time sheet back to Tom for correction and starts the submit-and-review cycle again. This cycle can repeat an unlimited number of times.

Within the preceding scenario, the Time Sheet predefined business process helps Tom and Sue perform their tasks. The business process also helps JoAnn, who works in the Human Resources department, to complete her tasks.

1. Tom points his Web browser to `http://host:port/websuite`. Sue received this URL from the Gentran Integration Suite system administrator.
2. At the login page, Tom logs in to Gentran Integration Suite using “ws_employee” as his user name and “password” as his password.
3. Gentran Integration Suite verifies Tom’s user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.

5. Based on the security permissions associated with Tom's user name and password, Tom has permissions to the Time Sheet Web template. Tom clicks the Time Sheet link under the Create New topic. A blank Time Sheet Web template opens in Tom's Web browser.
6. Tom completes the Time Sheet Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the time sheet is technically accurate.
 - ◆ **Submit for Approval** to send the completed time sheet to Sue for her review.

The Time Sheet Web template is validated and sent to Sue's Gentran Integration Suite Inbox. An e-mail notification is sent to Sue with a link to Tom's time sheet.

7. Sue opens the e-mail message and clicks the link to Tom's time sheet. She is prompted for her user name and password. Sue logs in to Gentran Integration Suite using "ws_manager" as her user name and "password" as her password.
8. Sue's Gentran Integration Suite work page shows that she has a new time sheet from Tom.
9. Sue opens Tom's time sheet by clicking the document identifier.
10. As a manager, Sue has only approve or reject privileges to the time sheet Web template. She cannot edit the time sheet, but can add comments to the Comments text box.
11. Does Sue approve Tom's time sheet?
 - ◆ If Yes, she clicks **Approve** and sends the time sheet to Human Resources for processing.
 - ◆ If No, Sue types any comments regarding the rejection in the Comments text box, clicks **Reject**, and sends the time sheet back to Tom for correction and resubmission. This starts the submit-and-review cycle again.
12. Sue clicks **Approve** and sends Tom's time sheet to Human Resources for processing. An e-mail notification is sent to JoAnn in the Human Resources department, alerting her that a document is awaiting her attention. JoAnn opens the e-mail message and clicks the link to the document.
13. JoAnn is prompted to log in. JoAnn logs in to Gentran Integration Suite using "ws_hr" as her user name and "password" as her password.
14. JoAnn's Gentran Integration Suite work page shows that she has a new time sheet from Sue.
15. JoAnn opens Tom's time sheet by clicking on the document identifier.
16. As a Human Resources administrator, JoAnn has only view privileges to the time sheet Web template. She cannot edit the time sheet. JoAnn reviews Tom's time sheet and processes it.

Purchase Order Business Process

The Purchase Order predefined business process automates and streamlines the process of completing a purchase order, submitting that purchase order to a supplier, having the supplier either approve or deny the purchase order, and having the supplier send either an acknowledgment, invoice, or advance ship notice in response to the purchase order.

Purchase Order Business Process Example

Tom is an employee of ABC company, a manufacturing company. Tom receives notice that ABC's stock of gears just passed the resupply limit. He must complete a purchase order (PO) to order additional gears from his supplier, DEF company. He completes the PO and sends it to his trading partner, DEF company. John at DEF company receives the PO and turns the PO into a PO acknowledgment, an advance ship notice, or an invoice and sends that document to Tom at ABC company for further processing.

Within the preceding scenario, the Purchase Order predefined business process helps Tom and John perform their tasks.

1. Tom points his Web browser to `http://host:port/websuite`. The Gentran Integration Suite system administrator provided Tom with this URL after Gentran Integration Suite was installed at ABC company.
2. At the login page, Tom logs in to Gentran Integration Suite using "ws_buyer" as his user name and "password" as his password.
3. Gentran Integration Suite verifies Tom's user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom's user name and password, Tom has permissions to the Purchase Order Web template. Tom clicks the Purchase Order link under the Create New topic. A blank Purchase Order Web template opens in Tom's Web browser.
6. Tom completes the Purchase Order Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the PO is technically accurate.
 - ◆ **Send** to send the completed PO to John for his review.

The PO is validated and sent to John's Gentran Integration Suite Inbox. An e-mail notification is sent to John with a link to Tom's purchase order.

7. John opens the e-mail message and clicks the link to Tom's purchase order. He is prompted for his user name and password. John logs in to Gentran Integration Suite using "ws_supplier" as his user name and "password" as his password.
8. John's Gentran Integration Suite work page shows that he has a new PO from Tom.
9. John clicks the document identifier and opens Tom's PO. The upper-right corner of the PO identifies Tom as the sender.
10. John reviews the information in the PO and completes one of the following actions:

- ◆ Clicks **Turn to PO Acknowledgment** to turn the PO into an acknowledgment, to send to Tom at ABC company.

Unlike advance ship notice and invoice, you cannot create a PO acknowledgment separately from a PO.

- ◆ Clicks **Turn to Ship Notice** to turn the PO into an advance ship notice to send to Tom at ABC company. For more information, see *Advance Ship Notice Business Process* on page 134.
- ◆ Clicks **Turn to Invoice** to turn the PO into an invoice to send to Tom at ABC company. For more information, see *Invoice Business Process* on page 132.

Remittance Advice Business Process

The Remittance Advice predefined business process automates and streamlines the process of completing a remittance advice and submitting that remittance advice to a supplier.

Remittance Advice Business Process Example

Tom is an employee of ABC company, a manufacturing company. Tom ordered and received a supply of gears from his supplier, DEF company. Tom must pay DEF company for the gears he received. Julie at DEF company receives the remittance advice and processes the document to complete payment.

Within the preceding scenario, the Remittance Advice predefined business process helps Tom and Julie perform their tasks.

1. Tom points his Web browser to `http://host:port/websuite`. The Gentran Integration Suite system administrator provided Tom with this URL after Gentran Integration Suite was installed at ABC company.
2. At the login page, Tom logs in to Gentran Integration Suite using “ws_buyer” as his user name and “password” as his password.
3. Gentran Integration Suite verifies Tom’s user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom’s user name and password, Tom has permissions to the Remittance Advice Web template. Tom clicks the Remittance Advice link under the Create New topic. A blank Remittance Advice Web template opens in Tom’s Web browser.
6. Tom completes the Remittance Advice Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.

- ◆ **Validate** to ensure that all required fields are completed and the remittance advice is technically accurate.
- ◆ **Send** to send the completed remittance advice to DEF company.

The remittance advice is validated and sent to Julie's Gentran Integration Suite Inbox. An e-mail notification is sent to Julie with a link to Tom's remittance advice.

7. Julie opens the e-mail message and clicks the link to Tom's remittance advice. She is prompted for her user name and password. Julie logs in to Gentran Integration Suite using "ws_supplier" as her user name and "password" as her password.
8. Julie's Gentran Integration Suite work page shows that she has a new remittance advice from Tom.
9. Julie clicks the document identifier and opens Tom's remittance advice. The upper-right corner of the remittance advice identifies Tom as the sender.
10. Julie reviews the information in the remittance advice and completes the processing of the payment.

Invoice Business Process

The Invoice predefined business process automates and streamlines the process of completing an invoice from a purchase order and submitting the invoice to a buyer.

Invoice Business Process Example

Tom is an employee of ABC company, a manufacturing company. Tom completes a purchase order (PO) for gears and sends the PO to his trading partner, DEF company. John at DEF company receives the PO that he must create an invoice for Tom's purchase and send the invoice to Tom for payment.

Within the preceding scenario, the Invoice predefined business process helps Tom and John perform their tasks.

1. Tom points his Web browser to `http://host:port/websuite`. The Gentran Integration Suite system administrator provided Tom with this URL after Gentran Integration Suite was installed at ABC company.
2. At the login page, Tom logs in to Gentran Integration Suite using "ws_buyer" as his user name and "password" as his password.
3. Gentran Integration Suite verifies Tom's user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom's user name and password, Tom has permissions to the Purchase Order Web template. Tom clicks the Purchase Order link under the Create New topic. A blank Purchase Order Web template opens in Tom's Web browser.

6. Tom completes the Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the purchase order is technically accurate.
 - ◆ **Send** to send the completed purchase order to John at DEF Company for processing.

The purchase order is validated and sent to John's Gentran Integration Suite Inbox. An e-mail notification is sent to John with a link to Tom's PO.
7. John opens the e-mail message and clicks the link to Tom's PO. He is prompted for his user name and password. John logs in to Gentran Integration Suite using "ws_supplier" as his user name and "password" as his password.
8. John's Gentran Integration Suite work page shows that he has a new PO from Tom.
9. John clicks the document identifier and opens Tom's PO. The upper-right corner of the PO identifies Tom as the sender.
10. John reviews the information in the PO and completes one of the following actions to create an invoice for Tom's PO:
 - ◆ Click **Turn to Invoice** to turn the PO into an invoice to send to Tom at ABC company.

Turn to Invoice populates information in the Invoice Web template with information from the PO received from the trading partner.
 - ◆ Click **Invoice** under Create New.

Choosing to create a new invoice requires you to type all information in the Invoice Web template. No information is populated from the PO received from the trading partner. Use this option when you do not have a PO, but must create an invoice.
11. In the Invoice Web template, John completes the appropriate information for the required fields, makes any changes to the quantity and price, and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the invoice is technically accurate.
 - ◆ **Send** to send the completed invoice to ABC company.

The invoice is validated and sent to ABC company in Tom's Gentran Integration Suite Inbox. An e-mail notification is sent to Tom with a link to John's invoice.
12. Tom opens the e-mail message and clicks the link to John's invoice. He is prompted for his user name and password. Tom logs in to Gentran Integration Suite using "ws_buyer" as his user name and "password" as his password.
13. Tom's Gentran Integration Suite work page shows that he has a new invoice from John.
14. John clicks the document identifier and opens John's invoice. The upper-right corner of the invoice identifies John as the sender.
15. John uses the information in the invoice to remit payment.

Advance Ship Notice Business Process

The Advance Ship Notice predefined business process automates and streamlines the process of completing an advance ship notice and submitting that advance ship notice to the buyer.

Advance Ship Notice Business Process Example

Tom is an employee of ABC company, a manufacturing company. Tom completes a purchase order (PO) for gears and sends the PO to his trading partner, DEF company. John at DEF company receives the PO, creates an advance ship notice for Tom's purchase, and sends the advance ship notice to Tom for payment.

Within the preceding scenario, the Advance Ship Notice predefined business process helps Tom and John perform their tasks.

1. Tom points his Web browser to <http://host:port/websuite>. The Gentran Integration Suite system administrator provided Tom with this URL after Gentran Integration Suite was installed at ABC company.
2. At the login page, Tom logs in to Gentran Integration Suite using "ws_buyer" as his user name and "password" as his password.
3. Gentran Integration Suite verifies Tom's user name and password against the database.
 - ◆ If Tom does not log in correctly, he receives an error message.
 - ◆ If Tom logs in correctly, his personal work page opens. The default is the Inbox page.
4. After Tom is verified, his Gentran Integration Suite work page lists the messages in his Inbox, and the Web templates and business processes he has access to under the permissions given to him by the system administrator.
5. Based on the security permissions associated with Tom's user name and password, Tom has permissions to the Purchase Order Web template. Tom clicks the Purchase Order link under the Create New topic. A blank Purchase Order Web template opens in Tom's Web browser.
6. Tom completes the Purchase Order Web template with the appropriate information and clicks the following buttons to perform the indicated functions:
 - ◆ **Validate** to ensure that all required fields are completed and the PO is technically accurate.
 - ◆ **Send** to send the completed PO to John for his review.The PO is validated and sent to John's Gentran Integration Suite Inbox. An e-mail notification is sent to John with a link to Tom's PO.
7. John opens the e-mail message and clicks the link to Tom's PO. He is prompted for his user name and password. John logs in to Gentran Integration Suite using "ws_supplier" as his user name and "password" as his password.
8. John's Gentran Integration Suite work page shows that he has a new PO from Tom.
9. John clicks the document identifier and opens Tom's PO. The upper-right corner of the PO identifies Tom as the sender.

10. John reviews the information in the PO and completes one of the following actions to create an advance ship notice for Tom's PO:
 - ◆ Click **Turn to Ship Notice** to turn the PO into an advance ship notice to send to Tom at ABC company.
Turn to Ship Notice populates information in the Advance Ship Notice Web template with information from the PO received from the trading partner.
 - ◆ Click **Ship Notice** under Create New.
Choosing to create a new advance ship notice requires you to type all information in the Advance Ship Notice Web template. No information is populated from the PO received from the trading partner. Use this choice when you do not have a PO, but must create an advance ship notice.
11. In the Advance Ship Notice Web template, John completes the appropriate information for the required fields, adds any information for the labels regarding product name and quantity, and clicks the following buttons to perform the indicated functions:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the advance ship notice is technically accurate.
 - ◆ **Send** to send the completed advance ship notice to ABC company.
The advance ship notice is validated and sent to ABC company in Tom's Gentran Integration Suite Inbox. An e-mail notification is sent to Tom with a link to John's advance ship notice.
12. Tom opens the e-mail message and clicks the link to John's advance ship notice. Tom is prompted for his user name and password. Tom logs in to Gentran Integration Suite using "ws_buyer" as his user name and "password" as his password.
13. Tom's Gentran Integration Suite work page shows a new advance ship notice from John.
14. John clicks the document identifier and opens John's advance ship notice. The upper-right corner of the advance ship notice identifies John as the sender.
15. John uses the information in the advance ship notice for planning purposes.

Gentran Integration Suite Work Page

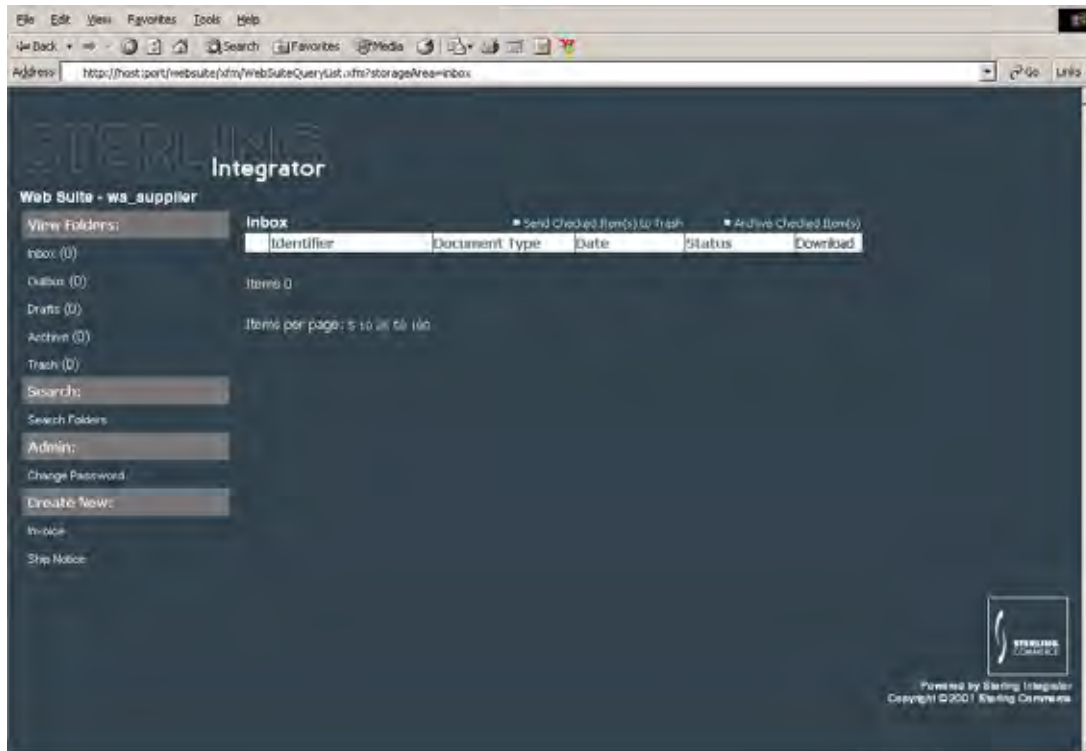
The Gentran Integration Suite work page enables you to interact with the predefined business processes through a common interface. The Gentran Integration Suite work page has five main areas:

- ◆ View Folders area
- ◆ Message List area
- ◆ Search area
- ◆ Administration area
- ◆ Create New area

Each area has a different function to manage your interaction with the predefined business processes. The appearance of the page can vary greatly, because the Create New area changes according to the security permissions associated with your user name and password.

You can customize the work page to include your company information to meet your needs. For more information, see *Editing a Web Template* on page 61 and *Editing the Gentran Integration Suite Work Page* on page 152.

The following figure shows the layout of a typical Gentran Integration Suite work page:



View Folders

The View Folders area of the Gentran Integration Suite work page contains five standard folders for all users. These folders work like the folders in most e-mail systems, as they collect and store documents sent to or from users. To view the contents of a folder, click the name of the folder.

The five standard folders are:

- ◆ Inbox – Receives documents sent to you for you to view and act upon.
- ◆ Outbox – Stores copies of documents you sent to another person.
- ◆ Drafts – Stores documents that are not ready to be sent to another user.
- ◆ Archive – Stores copies of documents that have been acted upon in some manner.
- ◆ Trash – Stores documents that are no longer needed until you permanently delete the documents from Gentran Integration Suite.

Message List

The Message List area of the Gentran Integration Suite work page shows the contents of the folder selected in the View Folders area. If you click Outbox in the View Folders area, the Message List area shows all the documents stored in your Outbox. The default view is the Inbox folder.

You can sort the contents of a folder by clicking the column heading by which you want to sort. Sorting a list makes it easier to locate the correct document. To view a document in the Message List area, click the identifier of the document and the document opens in your Web browser.

The Message List area consists of the following components:

Component	Description and Values
Check box	Checks specific documents for either deletion, archiving, or restoration.
Send Checked Items To Trash	Sends to the Trash folder all documents in the current page that have check boxes selected. The documents are not permanently deleted until you perform this action in the Trash folder.
Archive Checked Items	Sends to the Archive folder all documents in the current page that have check boxes selected.
Restore Checked Items	Restores to active status all documents in the current page that have check boxes selected.
Delete Checked Items	Permanently deletes all documents in the Trash folder that have check boxes selected.
Identifier	<p>ID of the original document. This ID varies by document type:</p> <ul style="list-style-type: none"> ◆ Time sheet and Purchase Requisition documents – Name of the sender. ◆ Expense Report – Cost center of the sender with the amount of the expense report appended. ◆ Purchase Order and PO Acknowledgment – Purchase order number. ◆ Advance Ship Notice – Advance ship notice number. ◆ Invoice – Invoice number. ◆ Remittance Advice – Check number remitted as payment. <p>Note: You can sort by identifier by clicking this column heading.</p>
Document Type	<p>Name of the document type. For example, Time Sheet or Expense Report.</p> <p>Note: You can sort by document type by clicking this column heading.</p>
Date	<p>Date that the document was sent or received:</p> <ul style="list-style-type: none"> ◆ In the Inbox message boxes, this is the date the document was received. ◆ In the Outbox, this is the date the document was sent. ◆ In the Draft message box, this is the date the document was saved as a draft. ◆ In the Trash and Archive message boxes, this is the date the document was either sent, received, or saved as draft. <p>Note: You can sort documents by date by clicking this column heading.</p>

Component	Description and Values
Storage Area	<p>Folder area where the document is located. This component displays only in the Trash, Archive, and Search results lists. Valid values are:</p> <ul style="list-style-type: none"> ◆ Inbox – Receives documents sent to you for you to view and act upon. ◆ Outbox – Stores copies of documents you sent to another person. ◆ Drafts – Stores documents that are not ready to be sent to another user. ◆ Archive – Stores copies of documents that have been acted upon in some manner and that you want to keep. ◆ Trash – Stores documents that are no longer needed before they are permanently deleted from Gentran Integration Suite.
State	<p>State of the document. Valid values are:</p> <ul style="list-style-type: none"> ◆ Active – Document is active. ◆ Archive – Document is archived.
Status	<p>Status of the document. Valid values are:</p> <ul style="list-style-type: none"> ◆ Pending – Document is awaiting action. ◆ Approved – Document was approved by the approving authority. ◆ Rejected – Document was rejected by the approving authority. ◆ Read – Document was read by its receiver. ◆ Unread – Document has not been read by its receiver. ◆ Draft – Document is in draft mode. ◆ Needs Approval – Document is awaiting approval. <p>Note: You can sort by status by clicking this column heading.</p>
Download	Downloads the XML file of the message.
Items	Number of items displayed in the message list.
Items per page	Sets the number of items to display per page in the message list. Valid values are 5, 10, 20, 50, or 100 items listed on each page. The default value is 10 items listed on each page.

Search

The Search area of the Gentran Integration Suite work page enables you to set search parameters to search for documents in the folders in your View Folders area.

Administration

The Administration area of the Gentran Integration Suite work page enables you to change your password in your system account.

Create New

The Create New area of the Gentran Integration Suite work page shows the Web templates and business processes that you have access rights to based on your user name and password. The Gentran Integration Suite system administrator creates these access rights to establish security for the system.

Accessibility Example

The following example explains how two different users can have two different work pages based on the access rights associated with their user names and passwords. Tom is an employee whose manager is Sue. Tom logs in to Gentran Integration Suite and is verified as a valid user. Tom's Gentran Integration Suite work page shows the Expense Report, Time Sheet, and Purchase Requisition Web templates under Create New. This means that Tom has write permissions to these three Web templates and can create an expense report and submit it to his manager for approval. To complete one of the Web templates, Tom clicks the name of the Web template under Create New.

Sue, Tom's manager, logs in to Gentran Integration Suite and her Gentran Integration Suite work page does not show the Create New area. That is because Sue's user name and password do not give her permissions to create or change any Web templates or business processes. She can view the expense reports, time sheets, and purchase requisitions to either approve or reject them, but she cannot edit the reports except for providing comments.

Managing the Gentran Integration Suite Work Page

The Gentran Integration Suite work page enables you to view, send, submit, delete, archive, and restore documents. You can also change your password and customize the look of the Gentran Integration Suite work page.

Opening the Gentran Integration Suite Work Page

To locate your Gentran Integration Suite work page:

1. After Gentran Integration Suite Web Extensions is installed, point your Web browser to `http://host:port/websuite`.
2. At the login page, type your user name and password that your manager provides.
 - ◆ If you do not log in correctly, you receive an error message. Verify your user name and password and try to log in again.
 - ◆ If you log in correctly, your personal work page opens. The default is your Inbox page.

After verification, your Gentran Integration Suite work page lists the messages in your Inbox, and the Web templates and business processes you have access to under the permissions given to you by the system administrator.

Changing the Gentran Integration Suite Work Page Display

You can change the number of items that display in the Message List area on your Gentran Integration Suite work page.

To change the Gentran Integration Suite work page display:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. To the right of **Items per page**, click the number that represents the number of items you want to display in the Message List area on each page of your Gentran Integration Suite work page. For example, click 20 to display 20 items at a time in the message list.
4. If you have more documents in a folder than the items per page selection allows, the Pages links display to the right of the **Items per page**. Click the number that represents the page of items you want to display in the Message List area of your Gentran Integration Suite work page.

For example, if you select 5 items per page and have 7 items in your folder, the first five items will display on page one and the last two documents will display on page two. To see document seven, click the number 2 to the right of **Page**.

Viewing Documents

You can view documents in any of the folders in your Gentran Integration Suite work page. To view a document in any of the folders, click the document identifier.

Searching for Documents in Folders

You can search the folders on the Gentran Integration Suite work page for documents using a combination of search parameters

Note: You can customize the Search form to meet your needs. For more information, see Chapter 3, *Creating Web Templates*.

To search for documents in the folders in the Gentran Integration Suite work page:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. In the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under Search, click **Search Folders**.

4. In the Search page, specify any combination of the following search criteria, and click **Search**:

Field	Description
Document Identifier	ID of the document you are searching for. This ID varies by document type. For more information, see <i>Message List</i> on page 137.
Storage Area	Folder area to search for documents. The default is <i>All Storage Areas</i> . Maintain the default value or select one of the following storage areas: <ul style="list-style-type: none"> ◆ Inbox ◆ Outbox ◆ Drafts
Active Documents	Search only active documents in combination with the other search criteria. For example, if you select Inbox in the Storage Area field and you check Active Documents, then the active documents in the Inbox are searched. Any other documents in the Inbox are not searched.
Archive Documents	Search only archived documents in combination with the other search criteria. For example, if you select Inbox in the Storage Area field and you select Archive Documents, then the archived documents in the Inbox are searched. Any other documents in the Inbox are not searched.
Document Type	Document type you are searching for. For example, Time Sheet or Expense Report. Note: You must type the complete name of the document type; partial names are not permitted and return no results.
Document Status	Status of the document you are searching for. Valid values are: <ul style="list-style-type: none"> ◆ Pending – Document is awaiting action. ◆ Approved – Document was approved by the approving authority. ◆ Rejected – Document was rejected by the approving authority. ◆ Read – Document was read by its receiver. ◆ Unread – Document was not read by its receiver. ◆ Draft – Document is in draft mode. ◆ Needs Approval – Document is awaiting approval. Note: You must type the entire name of the status; partial names are not permitted and return no results.
Creation Date Range	Date range to search for documents that were created on or between dates. The format for these fields is <i>yyyy-mm-dd</i> . The first field is the beginning date in the range and the second field is the ending date in the range.
Creation Time Range	Time range to search for documents that were created on or between specific times. The format for these fields is <i>hh:mm:ss am/pm</i> . The first field is the beginning time in the range and the second field is the ending time in the range.
Modification Date Range	Date range to search for documents that were modified on or between specific dates. The format for these fields is <i>yyyy-mm-dd</i> . The first field is the beginning date in the range and the second field is the ending date in the range.

Field	Description
Modification Time Range	Time range to search for documents that were modified on or between specific times. The format for these fields is <i>hh:mm:ss am/pm</i> . The first field is the beginning time in the range and the second field is the ending time in the range.

5. In the Search Results page, complete one of the following actions:
 - ◆ To view a document, in the **ID** column, click the document identifier of the document.
 - ◆ In the **Message List** area:
 - To sort the search results by the column heading type, click a column heading. The default sort criteria are date and time.
 - To download a document in XML format, click the download icon next to the document.
 - To delete, archive, or restore a specific document or documents, select the check box next to the document identifier.

Changing a User Password

To change your user password:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under Administration, click **Change Password**.
4. In the change password page, complete the following steps and then click **Change**:
 - a. In the Old Password field, type your current password.
 - b. In the New Password field, type your new password.
 - c. In the Confirm New Password field, type the new password again. This password must match the password you typed in the New Password field exactly or an error will occur.

The Password Changed page displays, confirming that you successfully changed your password. After you change your password, you are prompted to provide the new password before completing other tasks in your Gentran Integration Suite work page.

Creating and Submitting Documents

To create and send documents using predefined business processes:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.

3. Under Create New, click the name of the document type you must complete.
4. In the Web template, complete the required fields and any other appropriate fields.
5. Click the following buttons to perform the indicated functions, or click **Reset** to clear the Web template:
 - ◆ **Update Totals** to recalculate the totals to verify accuracy.
 - ◆ **Validate** to ensure that all required fields are completed and the expense report is technically accurate.
 - ◆ **Submit for Approval** to send the completed expense report to the approving authority.

Approving or Rejecting Submitted Documents

To approve or reject submitted documents:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. In the **ID** column, click the document identifier of the document you want to approve or reject.
4. Review the document for accuracy. Do you want to approve or reject the document?
 - ◆ If you want to approve the document, click **Approve** and send the document to the next department for processing.
 - ◆ If you want to reject the document, type any comments regarding the rejection in the **Comments** text box, and click **Reject** to send the document back to the sender for correction and resubmission.

Receiving E-mail Notifications of Received Documents

Gentran Integration Suite Web Extensions predefined business processes are preconfigured to send an e-mail notification to the receiver of the document to alert them that they have received a document awaiting their interaction.

To receive e-mail notifications of pending documents, you must have e-mail addresses identified in the appropriate user accounts in the Gentran Integration Suite database.

The following is a sample e-mail notification:

```
-----Original Message-----
From: user@company.com [mailto:user@company.com]
Sent: Friday, August 16, 2002 12:38 PM
To: Manager, Ws
Subject: Purchase Requisition - Tom Cool
```

```
STERLING Integrator - Web Suite Email Notification
```

```
You have received a new Purchase Requisition in your Web Suite Inbox.
```

To: ws_manager
From: ws_employee
RE: Purchase Requisition - Tom Cool
Location: inbox
Timestamp: 2002-08-16T17:37:33.000Z
Expires (minutes):
http://11.111.111.111:22222/websuite/xfm/WebSuitePurchaseReqView.xfm?documentId=4b63b9:efb34403b4:-70fc

Archiving Documents

After you have submitted, read, and taken action on documents, you can archive them for future reference.

To archive documents using predefined business processes:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under View Folders, click the folder name of the folder containing the document you want to archive.
4. Select the check box next to the document identifier of the document you want to archive. You can select multiple documents to archive at the same time.
5. In the upper-right corner of the Message List area, click **Archive Checked Items**. The documents are removed from the current folder and sent to the Archives folder.

Restoring Documents

After you have archived documents or sent documents to the Trash folder, you can restore them for further action.

To restore documents in predefined business processes:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under View Folders, click the folder from which you want to restore documents:
 - ◆ If you want to restore archived documents, click **Archive**.
 - ◆ If you want to restore documents from the trash folder, click **Trash**.
4. In the folder, select the check box next to the document identifier of the document you want to restore. You can select multiple documents to restore at the same time.
5. In the upper-right corner of the Message List area, click **Restore Checked Items**. The documents are removed from the Archive folder and sent to your Inbox, Draft, or Outbox message box, depending on the folder the message was archived from.

Selecting Documents for Deletion

After you have submitted, read and taken action on, archived, or restored documents, you can delete them.

Deleting documents in predefined business processes is a two-step process. First, you must select the document to be moved to the Trash folder. Second, you must delete the document from the Trash folder. This is a safety measure to ensure you do not inadvertently delete an important document.

To select documents for deletion in predefined business processes:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under View Folders, click the folder name of the folder containing the document you want to select for deletion.
If you click Trash, be sure you want to permanently delete documents.
4. Select the check box next to the document identifier of the document you want to select for deletion. You can select multiple documents for deletion at the same time.
5. In the upper-right corner of the Message List area, click **Send Checked Items to Trash**. The documents are sent to the Trash folder.

Permanently Deleting Documents

Documents permanently deleted from Gentran Integration Suite predefined business processes cannot be restored.

To permanently delete documents from predefined business processes:

1. Point your Web browser to the URL that you were provided by your manager or system administrator. The default URL is `http://host:port/websuite`.
2. At the login page, log in to Gentran Integration Suite.
Your personal Gentran Integration Suite work page opens.
3. Under View Folders, click **Trash**.
Be sure you want to permanently delete documents.
4. Select the check box next to the document identifier of the document you want to delete. You can select multiple documents to delete at the same time.
5. In the upper-right corner of the Message List area, click **Delete Checked Items**. The documents are permanently deleted from Gentran Integration Suite.

Customizing the Gentran Integration Suite Work Page

You can customize the Gentran Integration Suite work page to match your needs.

For more information about customizing the Gentran Integration Suite work page, see *Editing the Gentran Integration Suite Work Page* on page 152.

Customizing Predefined Business Processes

To customize a predefined business process, you must disable the business process, check the business process out of Gentran Integration Suite, edit the business process, and check the business process in to Gentran Integration Suite for use.

Accessing Predefined Business Processes

You can access the predefined business processes to view, customize, or edit them, just as you can any other business process created in Gentran Integration Suite.

To access the predefined business processes:

1. From the Gentran Integration Suite **Business Process** menu, select **Manager**.
2. In the Manager page, use either the Search or List option to locate the business process.
3. When you have located the business process, you can:
 - ◆ View the business process by clicking the name of the business process.
 - ◆ Disable or enable the business process by using Execution Manager.
 - ◆ Check the business process in or out by using Source Manager.
 - ◆ Edit the business process by using Source Manager.

Disabling a Predefined Business Process

To disable a predefined business process:

1. From the Gentran Integration Suite **Business Process** menu, select **Manager**.
2. In the Manager page, use either the Search or List option to locate the business process you want to disable.
3. Next to the business process you want to disable, click **execution manager**.
4. In the Execution Manager page, clear the **Enable** check box for the version of the business process you want to disable.

Checking Out a Predefined Business Process

To check out a predefined business process:

1. From the Gentran Integration Suite **Business Process** menu, select **Manager**.

2. In the Manager page, use either the Search or List option to locate the business process you want to check out.
3. Next to the business process you want to check out, click **source manager**.
4. In the Source Manager page, click **check-out** next to the version you want to check out.
Gentran Integration Suite creates a copy of the business process you are checking out. The original version of the business process remains on the host computer.
5. Do you want to lock the business process?
 - ◆ If Yes, click **OK** to lock the file. No user can edit this business process until you unlock it during the check in process.
 - ◆ If No, click **Cancel**. However, you are not able to edit or check in the business process.
6. In the **File Download** dialog box, click **Save**.
7. In the **Save As** dialog box, select the location to save the business process to on the client computer and click **Save**.

Editing a Predefined Business Process

You can edit predefined business processes by using the Graphical Process Modeler or a text editor. Changes made to business processes by any method other than the Graphical Process Modeler cannot be seen in the Graphical Process Modeler.

To edit a business process using the Graphical Process Modeler:

1. Open the Graphical Process Modeler.
2. From the **File** menu, select **Open**.
3. In the **Open** dialog box, browse to the file for the business process you want to edit. The file has an extension of .bp.
4. Click **Open**. The business process opens in the Graphical Process Modeler.
5. Make the appropriate changes to the business process.
6. Save the new version of the business process.

To use the business process, you must check in the modified version of the business process.

Checking In a Predefined Business Process

To check in a modified predefined business process:

1. From the Gentran Integration Suite **Business Process** menu, select **Manager**.
2. In the Manager page, in the Create section next to Process Definition, click **Go!**
3. In the Editor page, type a name for the business process you are checking in.
4. Under Select an input mode for defining the new process, select **Check-in Business Process created by the graphical modeling tool** and click **Next**.
5. In the Check-in page, browse to the business process file you are checking in.

6. Type a description for the business process and click **Next**.

7. Do you want to enable this business process?

- ◆ If Yes, click **Finish**.
- ◆ If No, clear the Enable Business Process check box and click **Finish**.

Gentran Integration Suite checks in the selected business process. If you enabled the business process during the check-in process, Gentran Integration Suite can begin using it to process documents.

Deploying Web Applications

Using Gentran Integration Suite Web Extensions, you can deploy in Gentran Integration Suite and in demilitarized zones (DMZ) existing Web applications that were created using J2EE™ architecture.

This section covers the following topics:

- ◆ About Web Applications
- ◆ Editing the Web.xml File
- ◆ Editing Web Template XML Files
- ◆ Editing Web Template XML Resource Bundle (.xrb) Files
- ◆ Creating and Checking In Web Templates
- ◆ Creating Business Processes That Use the Web Template
- ◆ Editing the Gentran Integration Suite Work Page
- ◆ Managing Web Resources
- ◆ Managing .war Files
- ◆ Deploying a Web Application In a DMZ

About Web Applications

You can add XForms functions to your Web applications through Web templates and the other features of Web Extensions. A *Web application* is a business application that can be accessed and interacted with through a Web browser. For example, Gentran® Web Suite® is a Sterling Commerce Web application that enables small trading partners to conduct e-business operations with larger trading partners using Web templates displayed through a Web browser.

To add XForms controls and deploy your Web applications in Gentran Integration Suite, you must have:

- ◆ An installed copy of Gentran Integration Suite Web Extensions.
- ◆ A Web application created using J2EE and that has a servlet 2.2 or later container.

The process of adding XForms controls to your Web applications and deploying the Web application includes the following steps:

1. Editing the Web.xml file for your Web application.
2. Creating and checking in your Web template.
3. Creating business processes to use the Web template.
4. Adding a URI to the Web Extensions HTTP Server adapter to point to the business process created.
Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.
5. Setting permissions for the business process and the Web template.
6. Generating a .war file to deploy the Web application in another environment.
7. Deploying the appropriate Gentran Integration Suite services, if you are deploying the Web application in a DMZ.

Editing the Web.xml File

The Web.xml file contains properties controlling servlets for a specific Web application. This is a standard file created when the Web application is created using J2EE. If you are adding XForms functions to multiple Web applications, you must edit the Web.xml file associated with each Web application.

Editing the Web.xml file includes two steps:

1. Declaring the XForms servlet in the Web.xml file between the `<servlet>` and `</servlet>` tags.
2. Associating the .xfm extension and .xrb extension in the Web.xml file between the `<mappings>` and `</mappings>` tags. This association enables the XForms servlet to run each time an .xfm or an .xrb file is located by the XForms servlet.

To edit the Web.xml file:

1. Locate the Web.xml file in the Web application directory. The standard directory structure is:
WebApplicationName/WebApp/WEB-INF/Web.xml
2. Declare the XForms servlet between the `<servlet>` and `</servlet>` tags.
3. Associate the .xfm and the .xrb file extensions with the XForms servlet.
4. Save the **Web.xml** file to the original location.

Caution: Do not change the name or the location of the Web.xml when saving the file. Doing so may cause errors.

Editing Web Template XML Files

Each Web template has two associated XML files that mimic the structure of the Web template in XML:

- ◆ WebTemplateName.xml

◆ WebTemplateNameView.xml

The WebTemplateName.xml file contains no data to populate the Web template. The WebTemplateNameView.xml contains sample data to populate the Web template.

If you edit the Web template, ensure that the structure of the XML files matches the structure of the Web template. If the structures do not match, you may experience errors in displaying your data through the Web template.

To edit the Web template XML files:

1. Locate the XML file for the Web template you are editing in the Web application directory. The standard directory structure is:

install_dir/container/Applications/WebSuite/xml

2. Edit the structure of the XML document to match the Web template structure.
3. Save the *WebTemplateName.xml* file to the location of the new Web application.

Editing Web Template XML Resource Bundle (.xrb) Files

Each Web template has an XML resource bundle (.xrb) file associated with it that contains the IDs for the field names. When the Web template displays, it references the .xrb file to display the correct field labels.

If you edit the Web template, ensure that the structure of the .xrb file matches the structure of the Web template. If the structures do not match, you may experience errors in displaying your data through the Web template.

To edit the Web template .xrb files:

1. Locate the .xrb file for the Web template you are editing in the Web application directory. The standard directory structure is:

install_dir/container/Applications/WebSuite/xrb

2. Edit the structure of the .xrb document to match the Web template structure.
3. Save the **WebTemplateName.xrb** file to the location of the new Web application.

Creating and Checking In Web Templates

After you have edited the Web.xml file, you must create and check your Web template in to the Gentran Integration Suite database.

To create a new Web template, you can either:

- ◆ Type the actual XForms code in an HTML editor.
- ◆ Locate an existing Web template and edit it to meet your needs.

For more information, see *Managing XForms Structure Elements* on page 69 and *Editing a Web template* on page 65.

For more information about checking in your Web template, see *Checking In a Web Template* on page 41.

Creating Business Processes That Use the Web Template

After you create and check in the Web template, you can create a business process to use that Web template. Creating a business process involves configuring services and adapters and linking them in a logical sequence to accomplish a goal. For the business process you create to work properly, you must add a URI to the Web Extensions HTTP Server adapter.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

Creating a business process to use a Web template involves configuring the Human Interaction Event or Human Interaction XForms services to use the Web template.

If you would like to preview the Web template in your Web browser before using it in a business process, you can either:

- ◆ Preview the Web template using the preview feature in Web Extensions.
- ◆ Point your Web browser to `http://host:port/websuite/xfm/templatename.xfm`, where *host:port* is the host name and port number of your Sterling Integration installation and *templatename.xfm* is the name of the Web template to view.

Setting Permissions for the Web Template and the Business Process

After you create the business process, you must associate permissions with the business process and the Web template to allow only authorized users access to the business process and Web template.

Editing the Gentran Integration Suite Work Page

You can add new storage areas or new Web templates to the Gentran Integration Suite work page provided with Gentran Integration Suite Web Extensions predefined business processes. For more information, see *Gentran Integration Suite Work Page* on page 69.

Adding Web Templates to the Create New Area

To add a Web template to the Create New area in the Gentran Integration Suite work page:

1. In the WebSuite/WebApp/Websuite.war directory, locate the websuitemain.xrb file.
2. In a text editor, open the websuitemain.xrb file.

3. Locate the Create New label.

```
<label>Create New:</label>
```

4. Copy a complete `<option></option>` entry.

For example, the following code sample adds a new document named Purchase Order to the existing list:

```
<option visible="true">
<caption>Purchase Order</caption>
<link>WebSuitePO.xfm</link>
</option>
```

5. Paste the complete `<option></option>` entry under the last `</option>` under the Create New label.
6. Replace the `<caption></caption>` value in the pasted `<option> </option>` entry with the name to display in the menu.
7. Replace the `<link></link>` value in the pasted `<option> </option>` entry with the name of the Web template to display when the caption name is clicked on the menu.
8. Save the websuitemain.xrb file.

Caution: Do not change the name or location of the websuitemain.xrb file when saving the file. Doing so may cause errors.

Adding Storage Areas to the View Folders Area

To add a storage area to the View Folders area in the Gentran Integration Suite work page:

1. In the WebSuite/WebApp/Websuite.war directory, locate the websuitemain.xrb file.
2. In a text editor, open the websuitemain.xrb file.
3. Locate the View Folders label:

```
<label>View Folders:</label>
```

4. Copy a complete `<option></option>` entry. For example:

```
<option visible="true">
<caption>Inbox</caption>
<link>Inbox.xfm</link>
</option>
```

5. Paste the complete `<option></option>` entry under the last `</option>` under the View Folders label.
6. Replace the `<caption></caption>` value in the pasted `<option></option>` entry with the name to display in the menu.
7. Replace the `<link></link>` value in the pasted `<option></option>` entry with the name of the Web template to display when the caption name is clicked in the menu.
8. Save the websuitemain.xrb file.

Caution: Do not change the name or location of the websuitemain.xrb file when saving the file. Doing so may cause errors.

Managing Web Resources

Gentran Integration Suite provides tools to manage Web resources that can be used in building Web applications. After you have checked in Web resources, you associate each resource with a resource tag, then you add them to Web archive (.war) files to easily deploy the Web application in another environment.

About Resources

Resources are files, templates, and documents that are deployed in Gentran Integration Suite to perform various actions. Resource types in Gentran Integration Suite include:

- ◆ Trading partner data
- ◆ Business processes
- ◆ XML schemas
- ◆ Web templates
- ◆ XSLT style sheets
- ◆ Service configurations
- ◆ Maps
- ◆ Web resources

Web Resources

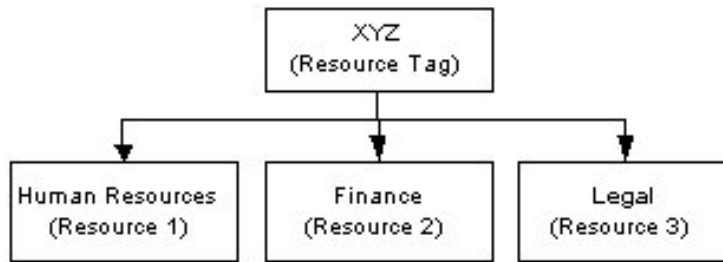
A *Web resource* is a resource that Gentran Integration Suite uses to display and run Web applications and functions in Web browsers. Web resources can be a Java server page file, an HTML file, an image file, an .xml file, a cascading style sheet, or any other file that is used in displaying or running Web applications and functions.

A Web resource is like a department within a company. Each department has a distinct departmental name and function within the company, but if a single department is missing from the company, the company is not complete and will not work as complete company works. For example, a company has Human Resources, Finance, and Legal departments. Together they are a company, but without one of the departments, the company is not complete and will not work as well as when all departments are present.

Resource Tags

A *resource tag* is a name that associates different resources and Web resources, so they can be assigned to a .war file. The resource tag is much like a company name, with each department being a Web resource associated with the resource tag. For example, XYZ company has a Human Resources, Finance, and Legal

department. The company name, XYZ, is the resource tag, and the Human Resources, Finance, and Legal departments are Web resources that are associated with the XYZ resource tag.



Web Archive (.war) Files

A *Web archive (.war) file* is a compressed file that contains the directory structure and file contents of a Web application. In this definition, file contents refers to Web resources. A .war file enables easy transport and deployment of the Web resources to another environment.

Following the analogy discussed in Web Resources and Resource Tags, a .war file is like an airplane containing all of the company departments going on a business trip. All of the departments are located (compressed) within the airplane, travel (are transported) to the location for the business trip, and depart the airplane to be deployed to their appropriate locations to conduct business.

To create a .war file in Gentran Integration Suite, you must complete three steps:

1. Check in the Web resources you want to include in the .war file.
For more information, see *Checking In a Web Resource* on page 90.
2. Create the resource tag associating the Web resources with the resource tag.
3. Generate the .war file for the resource tag.

For more information, see *Generating a Web Archive (.war) File* on page 94.

Checking In a Web Resource

A Web resource can come from many different sources, but before it can be used by Gentran Integration Suite, you must check the Web resource into Gentran Integration Suite. After you have checked in a Web resource, you can search for and check it out of Gentran Integration Suite, and manage versions of the Web resource.

To check in a Web resource:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Resources**.
The Web Resources page opens.
2. Next to the Check-in new Web Resource File, click **Go!**
3. In the **Web Resource File Name** field, complete one of the following actions:
 - ◆ Type the file name and extension of the Web resource to check in.
 - ◆ Click **Browse** to navigate to the location where the Web resource resides, and select the file name of the Web resource.

4. In the **Description** field, type any comments you want (required) and click **Next**. The Naming page opens.

Gentran Integration Suite uses the comments you type in this field to establish a version for each resource. Ensure that the comments are descriptive to help you maintain version control.

For example, if you are checking in a Web resource for which no previous version exists, you could type Version 1 as the description. When you check in subsequent versions of this Web resource, you can type Version 2 and so on.

5. In the **Resource Name** field, type a name that will identify the Web resource in Gentran Integration Suite.

If a name of the Web resource you are checking in already exists in the Gentran Integration Suite database, you are prompted to type a new name for the Web resource.

6. Under **Resource Type**, select the type of Web resource that you are checking in and click **Next**.

Gentran Integration Suite provides the initial type choice based on the file extension of the Web resource you are checking in. You can change the type by selecting a different type from the list. Only one resource type can be associated with a resource name.

7. In the Confirm page, complete the following actions:

- ◆ Select **Enable** to enable the Web resource for use in Gentran Integration Suite.
- ◆ Review the information and click **Finish**.

Checking Out a Web Resource

After a Web resource is checked in, you can check it out of the Gentran Integration Suite database. Checking out a Web resource does not remove it from the Gentran Integration Suite database; it saves a copy of the resource from the host to your client computer. You can choose to check out a read-only copy of the Web resource, or lock the Web resource in the Gentran Integration Suite database for editing.

To check out a Web resource:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Resources**.
2. In the Web Resources page, locate the template you want to check out by using either the Search or List option.

For more information, see *Searching for a Web Resource* on page 92.

3. Click **source manager** next to the Web resource you want to check out.
4. In the Web Resource Source Manager page, click **check-out** next to the Web resource you want to check out.
5. In the message about locking the file, complete one of the following actions:
 - ◆ To lock the Web resource file for editing, click **OK**.
 - ◆ To check out a read-only copy of the Web resource file, click **Cancel**. You cannot edit a read-only copy of the Web resource file.
6. In the **File Download** dialog box, select the **Save** option.

7. In the **Save As** dialog box, select the location where you want to save the Web resources and click **Save**. The copy of the Web resource is saved to the new location.
8. In the Download complete dialog box, complete one of the following actions:
 - ◆ To open the Web resource, click **Open**.
 - ◆ To open the folder where you saved the Web resource, click **Open Folder**.

Searching for a Web Resource

After a Web resource is checked in, you can search for that resource.

To search for a Web resource:

1. From the **Deployment** menu, select **Web Extensions > Web Resources**.
2. In the Web Resources page, complete one of the following actions:
 - ◆ Under Search in the **Resource Name** field, type either a portion of the name or the entire name of the Web resource you are searching for, and click **Go!** The Web Resource Management page opens, listing all Web resources containing the full or partial name you typed.
 - ◆ Under **List in the Alphabetically** field, select **ALL** or the letter that begins the name of the Web resource you are searching for. Selecting **ALL** lists all of the Web resources that are checked in. Click **Go!** The Web Resource Management page opens, listing all of the Web resources that match your search criteria.
 - ◆ Under List in the **by Resource Type** field, select the type of the Web Resource that you are searching for. You can select only one type from the list at a time. The following table describes the Web resource types:

Web Resource Type	Description
JSP	Java Server Page with the file extension .jsp.
JAVASCRIPT	Java scripts with the file extension .js.
HTML	HTML documents with the file extension .htm or .HTML.
XML	XML documents with the file extension .xml.
STYLESHEET	Cascading style sheets with the file extension .css.
PROPERTY	Property files, such as XML Resource bundles, with the file extension .xrb, or property files with the file extension .xml.
IMAGE	Graphic files used in Web pages with the file extensions .gif, .jpg, or .jpeg.
OTHER	All other files that do not fit any of the types listed in this table. File extensions can include .doc for a document file, .txt for text file, .bp for a business process and so forth. Almost any file extension can be included in this Web resource type.

3. Depending on your task, complete one of the following actions:

- ◆ Go to Source Manager to check in or check out the Web resource. For more information, see *Checking In a Web Resource* on page 90 and *Checking Out a Web Resource* on page 91.
- ◆ Go to Version Manager to set a default version of the Web resource and to enable or disable Web resources for business process use. For more information, see *Maintaining a Web Resource Version History* on page 93.

Deleting a Web Resource

After a Web resource is checked in, you can delete it from the Gentran Integration Suite database. Deleting a Web resource removes it completely from the eGentran Integration Suite database. Before you delete a Web resource, check it out of Gentran Integration Suite and save a copy to another location for future reference.

To delete a Web resource:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Resources**.
2. In the Web Resources page, locate the resource you want to delete by using either the Search or List option.

For more information, see *Searching for a Web Resource* on page 92.

3. Click **source manager** next to the Web resource you want to delete.
4. In the Web Resource Source Manager page, click the check box in the Delete column next to the Web resource you are deleting from Gentran Integration Suite.
5. Next to Delete Selected Versions, click **Go!**
You are prompted to confirm the deletion. Click **OK**.
6. In the Delete Resources page, review the information on the Web resource and click **Next**.
7. In the Confirm page, a message displays stating *Deleting this resource may cause processes to fail. This action cannot be reversed*. Click **Delete** to delete the Web resource.

The Web resource is deleted from the database.

Maintaining a Web Resource Version History

Gentran Integration Suite enables you to maintain versions of your Web resources. A *version* is an edited copy of a Web resource. Each time a Web resource is checked out and then checked in, a new version is created, but the original Web resource remains intact for future use.

A Web resource version is designated by the description in the Comments field when you check in a Web resource. Because this description provides the basis for maintaining the version for that Web resource, make the description unique and meaningful to the use of the resource. Doing this makes it easier to quickly choose the resource you need. In addition, Gentran Integration Suite adds a date and time stamp to each version that can be used for version identification.

To maintain Web resource version:

1. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Web Resources**.

2. In the Web Resources page, locate the resource you want to review by using either the Search or **List** option.

For more information, see *Searching for a Web Resource* on page 92.

3. Next to the Web resource you want to review, click **version manager**.
4. In the Web Resource Version Manager page, click **Default** next to the Web resource version you want to set as the default resource. This is the version of the resource to be used for business processes using this resource.
5. Do you want to enable the Web resource to make it available for use by business processes?
 - ◆ If Yes, ensure that the check box next to the Web resource you want to enable is selected.
 - ◆ If No, ensure that the check box next to the Web resource you want to disable is cleared.
6. Click **Save**. Your changes are applied to the Web resources in the Gentran Integration Suite database.

Managing .war Files

After you have checked in Web resources and associated each resource with a resource tag, you can generate a Web archive (.war) file to deploy the resources in another environment. For example, generating a .war file enables you to:

- ◆ Deploy Web applications easily in a test environment or a production environment.
- ◆ Deploy Web applications easily in a Demilitarized Zone (DMZ).

Generating a Web Archive (.war) File

To generate a Web Archive (.war) file:

1. From the **Deployment** menu, select **Web Extensions > Utilities**.
2. In the Web Utilities page, under Export Web Resources, click **Go!** next to Generate a WAR file using system resources.
3. In the **WAR File Name** field, type a file name for the .war file including the .war file extension.
4. In the **Resource Tag** field, select the resource tag from the list that contains the resources you want to include in the .war file and click **Next**.
5. Do you want to save the .war file on the server?
 - ◆ If Yes, ensure that the check box next to Save generated WAR file on server is selected.
 - ◆ If No, ensure that the check box next to Save generated WAR file on server is cleared.
6. In the Confirm page, verify the .war file information and click **Finish**.
7. In the page showing that the system update completed successfully, complete the following actions:
 - ◆ Click the document icon next to View Export Report to verify the .war file generated without errors.

- ◆ Click **Download** next to Download WAR file to download the .war file to your local computer.
8. In the **File Download** dialog box, complete one of the following actions:
 - ◆ To open and view the .war file, click **Open**.
 - ◆ To save the .war file to your local computer, click **Save**.
 - ◆ To cancel the download, click **Cancel**.
 - ◆ For more information about downloading files, click **More Info**.
 9. If you selected Save, make any changes to the file name and browse to the location that you want to save the .war file to and click **Save**.
 10. In the Download Complete dialog box, complete one of the following actions:
 - ◆ To open the file you downloaded, click **Open**.
 - ◆ To open the folder that you downloaded the file to, click **Open Folder**.
 - ◆ To close the Download Complete dialog box, click **Close**.

You are now ready to deploy the downloaded .war file in the new environment.

Deploying a Web Application In a DMZ

You can deploy Web applications in a DMZ to enable someone outside of your company fire wall to access your Web applications.

Deploying the XForms Servlet in a DMZ

If you are deploying a .war file in a DMZ that will reference a Web template created using XForms, you must also deploy the XForms servlet in the same DMZ; otherwise, the Web template will not display.

The XForms servlet enables you to display Web templates created using XForms, without being connected to Gentrans Integration Suite. You deploy the XForms servlet using `xforms.war` found in your application server directory.

A .war file has the following standard directory structure:

- ◆ `rootname` – Identifies the application root directory, which is defined by the Web application developer. For example, if the application is named `xforms`, the Web application developer might name this folder `xforms`.
- ◆ `rootname\WEB-INF` – Contains the necessary `web.xml` file. This folder can also contain resources that are related to the application and specified in the `web.xml` file.
- ◆ `rootname\WEB-INF\classes` – Contains the necessary `servlet.class` files. Any number of subdirectories can correspond to any package names defined in the servlet classes. These classes are automatically loaded by the servlet engine's class loader.
- ◆ `rootname\WEB-INF\lib` – Contains `.jar` and `.zip` files for classes that are automatically available to the class loader even though they are not explicitly defined in the JVM CLASSPATH.

To deploy the XForms servlet with a .war file in a DMZ:

1. Stop the Web server/servlet engine.
2. Copy xforms.war from your application server directory to the Web application root folder.
3. Restart the Web server/servlet engine. The XForms servlet is deployed automatically (some systems may deploy the servlet without a restart).

Deploying xforms.war on Apache/Tomcat 3.2

1. Stop Tomcat.
2. Are you deploying a .war file from Gentran Integration Suite?
 - ◆ If No, go to step 3.
 - ◆ If Yes, complete the following steps:
 - a. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
 - b. In the Download area, select the xforms.war file from the list and click **Go!**
 - c. In the **File Download** dialog box, click **Save**.
 - d. In the **Save As** dialog box, select the location that you want to save the xforms.war file to. Do not change the name of the file.
 - e. Copy **xforms.war** from the location that you saved the file to the Web application folder in the DMZ:

For example, copy the xforms.war to the /usr/local/tomcat3.2.2/webapp directory.

3. Copy the following files located in your Gentran Integration Suite *install_dir*/jar directory to the /tomcat3.2.2/lib folder:
 - ◆ xerces.jar
 - ◆ xercesImpl.jar
 - ◆ xalan.jar
 - ◆ xsdlib.jar
 - ◆ relaxngDatatype.jar
 - ◆ xforms.jar

The xerces.jar is located in the xerces folder and the xalan.jar file is located in the xalan folder. The xsdlib.jar and the relaxngDatatype.jar files are located in a subfolder in the xsdlib folder.

4. Restart Tomcat.

Deploying webx.war on Apache/Tomcat 3.2

If you are deploying a .war file in a DMZ that will invoke a business process in Gentran Integration Suite, you must also deploy the B2B HTTP servlet (webx.war) in the same DMZ environment.

When deploying the webx.war file on a computer other than one that has Gentran Integration Suite installed on it, complete the following steps:

1. Verify that you have an installed copy of Apache/Tomcat on the computer that you want to deploy the webx.war on.
2. Verify that the JDK version installed on the computer is the same version required by Gentran Integration Suite.
3. Deploy the webx.war file on the new applications server.
4. In Gentran Integration Suite, open the B2B HTTP Server adapter configuration and select the Web Extensions HTTP Server Adapter configuration.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

5. Type the IP address of the computer that is in the DMZ.
6. In the applications server where you deployed the webx.war file, locate and open the webx web.xml file.
The web.xml file is located in the /tomcat_install_dir/webapps/webx/web-inf/ directory.
7. In the webx web.xml file, verify that the port number matches the HTTP Server adapter configuration on Gentran Integration Suite.

The web.xml file is located in the /tomcat_install_dir/webapps/webx/web-inf/ directory.

The HTTP Server adapter and the b2bhttp servlet are configured to send data to and receive data from each other.

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

Note: For instructions on deploying .war files on other applications servers, see the documentation for your specific applications server.

To deploy the webx.war file on Apache/Tomcat 3.2:

1. Stop Tomcat.
2. From the Gentran Integration Suite **Deployment** menu, select **Web Extensions > Utilities**.
3. In the Download area, select from the list the **webx.war** file and click **Go!**
4. In the **File Download** dialog box, click **Save**.
5. In the **Save As** dialog box, select the location that you want to save the b2bhttp.war file to. Do not change the name of the file.
6. Copy **webx.war** from the location that you saved the file to the Web application folder in the DMZ:
For example, copy the webx.war to the /usr/local/tomcat3.2.2/webapp directory.
7. Restart Tomcat.

Deploying a .war File in a DMZ

After you have generated a .war file, you can deploy the resources contained in the .war file in a DMZ.

To deploy a .war file in a DMZ:

1. Copy ***anyname.war***, where *anyname* is the name of the file you created, from your local machine to the applications server on which you want to deploy the .war file.
2. Deploy the .war file using the instructions provided with the specific applications server running in the new environment.

If you are deploying a .war file in a DMZ that will invoke a business process in Gentran Integration Suite, you must also deploy the B2B HTTP servlet in the same DMZ environment. For more information, see *Deploying webx.war on Apache/Tomcat 3.2* on page 161. If you are deploying a .war file that will reference a Web template created using XForms, you must also deploy the XForms servlet in the same DMZ; otherwise, the Web template will not display.

3. Stop the application server.
4. In the folder with the same name as the .war file you deployed, create a WEB-INF folder.
5. Copy **web.xml** from the XForms folder from the XForms servlet you deployed to the WEB-INF folder you just created.
6. Restart the application server.

Deploying a Web Application in a DMZ Using Perimeter Servers

You can deploy your Web applications in a DMZ using perimeter servers. Using perimeter servers provides security features and performance enhancements. You must use the perimeter servers configuration if:

- ◆ Your Web templates are JSPs.
- ◆ Your Web templates are XForms and you want use the session management features, such as log out.

Deploying a Web application in a DMZ using perimeter services involves the following four steps:

1. Configuring the perimeter server in Gentran Integration Suite.
2. Deploying the perimeter server to the DMZ.
3. Editing the B2B HTTP Server adapter in Gentran Integration Suite.

Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

For more information about editing the B2B HTTP Server adapter, see *Editing the B2B HTTP Server Adapter* on page 98.

4. Testing the Web application.

For more information about testing the Web application, see *Testing the Web Application* on page 99.

Editing the B2B HTTP Server Adapter

Caution: Because of our continuing efforts to improve services and adapters to align with new technology and capabilities, the B2B HTTP Server adapter has entered the retirement process in Gentran Integration Suite and will be replaced with the HTTP Server adapter.

To edit the B2B HTTP Server adapter:

1. From the Gentran Integration Suite **Deployment** menu, select **Services > Configuration**.
2. Using either the **List** or **Search** feature, locate the HTTP Server adapter.
3. In the Service Configuration List, next to the HTTP Server adapter, click **copy**.
4. In the **Name** field, type a new descriptive name.
5. In the **Description** field, type a new description of the adapter.
6. Choose the appropriate group that you want to associate with this adapter and click **Next**.
7. In the **Host Listen Port** field, type the port number from the DMZ computer on which the adapter listens.
8. In the **Perimeter Server Name** field, select the appropriate name from the list.
9. Choose the appropriate document storage, user authentication, SSL, certificate, and cipher strength options and click **Next**.
10. In the URI page, next to **New URI**, click **add**.
11. In the URI Config page, in the **URI** field, type the URI you are adding to the adapter.
12. Under Launch BP Or War, select **WAR File**, and click **Next**.
13. In the WAR Config page, in the **Enter WAR File Path** field, type the complete path to the .war file and click **Next**.
14. In the URI page, click **Next**.
15. In the Confirm page, verify the settings, select **Enable Service for Business Processes**, and click **Finish**.

Testing the Web Application

After you deploy a Web application in a DMZ using the perimeter servers, you must test the application to ensure that it works properly.

To test a Web application deployed in a DMZ:

1. Point your Web browser to `http://host:port/filename`, where the host is the DMZ IP address, the port is the HTTP Server adapter port, and the filename is the name of the .war file deployed.
2. Log in using your user name and password.

A

- adapter
 - B2B HTTP Server 36
 - Web Extensions HTTP Server 36
- adding
 - button 46, 55, 59
 - content 44
 - e-mail link 87
 - HTML elements 44, 84
 - HTML image 84
 - HTML table 43, 88
 - hyperlink 85
 - storage areas to View Folders 153
 - table rows and columns 89
 - URI 112, 113
 - Web templates to Create New area 152
 - XForms bind 71
 - XForms button 77
 - XForms child node 69
 - XForms child node value 70
 - XForms controls 44, 75
 - XForms file upload 82
 - XForms model 67
 - XForms selectMany 81
 - XForms selectOne 79
 - XForms Structure 42
 - XForms submitInfo 73
 - XForms text field 75
- Advance Ship Notice business process 134
- applying
 - bold and italic 91
 - font 91
 - point size 92
- approving or rejecting documents 143
- archiving documents 144
- ASCII character set 15
- Authentication (authenticate) tag library 97
- Authorization (autho) tag library 96

B

- B2B HTTP Server adapter 36
- Base href 63, 106
- Big 5 (Chinese) character set 15
- bind property 72
- bold, applying 91
- BPSSCorrelationSearch Web template 115
- BPSSCorrelationSearchResults Web template 115
- business process, creation 13
- button, adding 46, 55, 59
- Buyer-to-Supplier predefined business process 12, 121

C

- central search Web template 115
- changing, user password 142
- character encoding 25
- character set
 - ASCII 15
 - Big 5 (Chinese) 15
 - definition 14
 - double-byte character set (DBCS) 14
 - ShiftJIS (Japanese) 15
 - Unicode 15
- checking in
 - predefined business process 147
 - Web resource 155
 - Web template 48, 107, 151
 - WebTemplateTest.bp 113
- checking out
 - predefined business process 146
 - Web resource 156
 - Web template 108
- code pane, Web Template Designer 23
- columns
 - adding 89

- deleting 90
- components, Web Extensions 10
- confirm send Web template 115
- CorrelationSearch Web template 115
- CorrelationSearchResults Web template 116
- Create New area, work page 139
- creating
 - predefined business process documents 142
 - Web template 39
 - Web template (HTML editor) 41
 - Web template (Translator/Designer) 49
 - Web template (Web Template Designer) 41
 - Web Template Wizard 57
 - WebTemplateTest.bp 112
- customizing
 - File Upload utility 33
 - predefined business process 146
- cutting items 91

D

- deleting
 - e-mail link 87
 - HTML image 85
 - hyperlink 86
 - rows and columns 90
 - Web Resource 158
 - Web template 110
 - XForms bind 73
 - XForms button 78
 - XForms child node 70
 - XForms child node value 71
 - XForms file upload 83
 - XForms model 68
 - XForms selectMany 82
 - XForms selectOne 80
 - XForms submitInfo 74
 - XForms text field 76
- deploying
 - .war file (DMZ) 162
 - Web application 149, 160
 - webx.war 161
 - XForms servlet 160
 - xforms.war 161
- design pane, Web Template Designer 23

- destination file
 - Web Template Translator 25
 - Web Template Wizard 31
- disabling predefined business process 146
- documents
 - approving or rejecting 143
 - archiving 144
 - restoring 144
 - routing 16
- DocumentSummary Web template 116
- double-byte character set (DBCS) 14
- downloading, Java Web Start 18
- dynamic list
 - XForms selectMany 81
 - XForms selectOne 79

E

- EDICorrelationDetail Web template 116
- EDICorrelationGrpSummary Web template 116
- EDICorrelationIntSummary Web template 116
- EDICorrelationSearch Web template 116
- EDICorrelationSetSummary Web template 116
- editing
 - e-mail link 87
 - HTML image 85
 - hyperlink 86
 - Web template 48, 61, 63
 - XForms bind 71
 - XForms button 77
 - XForms child node 69
 - XForms child node value 70
 - XForms file upload 83
 - XForms instance 68
 - XForms model 67
 - XForms selectMany 82
 - XForms selectOne 80
 - XForms submitInfo 74
 - XForms text field 76
- e-mail link
 - adding 87
 - deleting 87
 - editing 87
 - managing 86

e-mail notifications, receiving 143

Employee-to-Manager predefined business process 12, 121, 122

encoding
 definition 14
 examples 15
 internationalization 14
 localization 14
 UTF-8 and DBCS 14

error page Web template 116

Expense Report predefined business process 148

F

features, Web Extensions 9

file upload control 82

File Upload utility
 accessing 36
 confirm send Web template, editing 35
 customizing 33
 overview 32
 uploading process 32
 Web template, editing 34

FileUpload.bp 33

FileUploadConfirmSend Web template 116

FileUploadConfirmSend.xfm 33

FileUploadTemplate Web template 116

FileUploadTemplate.xfm 33

font, applying 91

formatting, text 91

G

GDSApprovalRequest Web template 116

GDSApproverList Web template 116

GDSApproverView Web template 116

GDSCatalogAdd Web template 116

GDSCatalogList Web template 116

GDSCatalogPending Web template 116

GDSCatalogPendingList Web template 116

GDSCatalogUpdate Web template 116

GDSCatalogView Web template 116

GDSCategoryManager Web template 116

GDSChangePassword Web template 116

GDSChangePasswordConfirm Web template 116

GDSConfirmSend Web template 117

GDSDBConfirm Web template 117

GDSQueryList Web template 117

GDSQuerySearch Web template 117

Gentran DDF files 30

Gentran Web Suite 149

Graphical Process Modeler (GPM) function 13

GS advance search Web template 117

GSLifeCycleDetail Web template 117

GSLifeCycleSummary Web template 117

GSSApprovalReq Web template 117

GSSApproverList Web template 117

GSSApproverView Web template 117

GSSCatalogAdd Web template 117

GSSCatalogList Web template 117

GSSCatalogPending Web template 117

GSSCatalogPendingList Web template 117

GSSCatalogUpdate Web template 117

GSSCatalogView Web template 117

GSSCategoryManager Web template 117

GSSChangePassword Web template 117

GSSChangePasswordConfirm Web template 117

GSSConfirmSend Web template 117

GSSDBConfirm Web template 117

GSSQueryList Web template 118

GSSQuerySearch Web template 118

H

hello-war 39

HTML
 elements 40, 53, 58, 62
 e-mail links 86

- image 84
 - table 44, 88
- HTML editor, creating Web template 41
- HTML tab
 - e-mail link 22
 - image 22
 - link 22
 - table 22
- HTML>XForms button 26
- HTML/XForms code examples 63
- HTTP Server adapter 33, 36, 39
- Human Interaction Document Loader service
 - function 11
- Human Interaction Event service function 11
- Human Interaction Query service function 11
- Human Interaction service
 - overview 10
 - types 10
- Human Interaction XForms service function 11
- hyperlink
 - adding 85
 - deleting 86
 - editing 86
 - managing 85

I

- image
 - adding 84
 - deleting 85
 - editing 85
 - managing 84
- installing
 - Web Template Designer 18
 - Web Template Wizard 31
- interface, Web Template Designer 20
- internationalization 14
- Invoice
 - (Chinese) Web template 118
 - business process 132
 - Web template 118
- italic, applying 91

J

- Java Server Pages (JSP) 9, 33, 39, 92
- Java Web Start
 - downloading 18
 - starting Web Template Designer 19
- JSP tag libraries
 - prerequisites 92

L

- license file, obtaining 10
- localization
 - overview 14
 - XML resource bundle 15

M

- MailboxAdd Web template 118
- MailboxAddDisabled Web template 118
- MailboxChangePassword Web template 118
- MailboxChangePasswordConfirm Web template 118
- MailboxConfirmSend Web template 118
- MailboxDirList Web template 118
- MailboxErrorPage Web template 118
- MailboxSearch Web template 118
- managing
 - .war file 159
 - e-mail link 86
 - HTML image 84
 - HTML table 88
 - hyperlink 85
 - XForms bind 71
 - XForms button 76
 - XForms child node 68
 - XForms child node value 70
 - XForms instances 68
 - XForms model 67
 - XForms selectMany 80
 - XForms selectOne 78
 - XForms submitInfo 73
 - XForms text field 75
- map, creating 34
- menu bar, Web Template Designer 20

Message List area
work page 137

N

node pane, Web Template Designer 23

notifications, e-mail 143

O

output space box
Web Template Translator 25
Web Template Wizard 31

P

password, Web Extensions 124

pasting items 91

perimeter servers 163

perimeter services 9

permissions
role-based security 15
setting 152
Web Extensions 124

POChange Web template 118

POChangeAcknowledgment Web template 118

point size, applying 92

predefined business process
accessing 146
Advance Ship Notice 134
approving or rejecting documents 143
archiving documents 144
associated Web templates 123
Buyer-to-Supplier 12, 121
checking in 147
checking out 146
creating and submitting documents 142
customizing 146
deleting documents 145
disabling 146
editing 147
Employee-to-Manager 12, 121, 122
Expense Report 148
Invoice 132
overview 12, 120
Purchase Requisition 126

receiving e-mail notifications 143
restoring documents 144
types 121

preinstalled Web templates

BPSSCorrelationSearch 115
BPSSCorrelationSearchResults 115
central search 115
confirm send 115
CorrelationSearch 115
CorrelationSearchResults 116
DocumentSummary 116
EDICorrelationDetail 116
EDICorrelationGrpSummary 116
EDICorrelationIntSummary 116
EDICorrelationSearch 116
EDICorrelationSetSummary 116
error page 116
FileUploadConfirmSend 116
FileUploadTemplate 116
GDSApprovalRequest 116
GDSApproverList 116
GDSApproverView 116
GDSCatalogAdd 116
GDSCatalogList 116
GDSCatalogPending 116
GDSCatalogPendingList 116
GDSCatalogUpdate 116
GDSCatalogView 116
GDSChangePassword 116
GDSChangePasswordConfirm 116
GDSConfirmSend 117
GDSDBConfirm 117
GDSQueryList 117
GDSQuerySearch 117
GS advance search 117
GSLifeCycleDetail 117
GSLifeCycleSummary 117
GSSApprovalReq 117
GSSApproverList 117
GSSApproverView 117
GSSCatalogAdd 117
GSSCatalogList 117
GSSCatalogPending 117
GSSCatalogPendingList 117
GSSCatalogUpdate 117
GSSCatalogView 117
GSSCategoryManager 117
GSSChangePassword 117

GSSChangePasswordConfirm 117
 GSSConfirmSend 117
 GSSDBConfirm 117
 GSSQueryList 118
 GSSQuerySearch 118
 invoice 118
 invoice (Chinese) 118
 MailboxAdd 118
 MailboxAddDisabled 118
 MailboxChangePassword 118
 MailboxChangePasswordConfirm 118
 MailboxConfirmSend 118
 MailboxDirList 118
 MailboxErrorPage 118
 MailboxSearch 118
 POChange 118
 POChangeAcknowledgment 118
 purchase order templates 118
 PurchaseOrderConfirmSend 118
 PurchaseOrderMenu 118
 PurchaseOrderQuery 118
 PurchaseOrderView 118
 query templates 118
 QueryList 118
 request for quote templates 118
 RFQResponse 118
 search results 118
 Web Suite Approval Notify 119
 Web Suite ASN templates 119
 Web Suite expense report templates 119
 Web Suite invoice templates 119
 Web Suite PO templates 119
 Web Suite purchase request templates 119
 Web Suite query list 119
 Web Suite remittance advice templates 119
 Web Suite time sheet templates 119
 WebSuiteChangePassword 119
 previewing Web template 48, 105
 properties pane, Web Template Designer 23
 Purchase Order business process 129
 purchase order Web template 118
 Purchase Requisition business process 126
 PurchaseOrderConfirmSend Web template 118
 PurchaseOrderMenu Web template 118
 PurchaseOrderQuery Web template 118
 PurchaseOrderView Web template 118

Q

query Web templates 118
 QueryList Web template 118
 quitting
 Web Template Designer 19
 Web Template Translator 24
 Web Template Wizard 31

R

relaxngDatatype.jar 161
 Remittance Advice business process 131
 request for quote Web templates 118
 resource tag 154
 resource, Web
 checking in 155
 checking out 156
 locating 157
 version history 158
 restoring documents 144
 RFPRResponse Web template 118
 role-based security
 document routing 16
 permissions 15
 user account 15
 rows
 adding 89
 deleting 90

S

saving, Web template 48
 search results Web template 118
 searching
 Web resource 157
 Web template 109
 security, role-based 16
 sendMe.xfm 111
 service, Human Interaction 10
 Session Validation (validSession) tag library 96
 ShiftJIS (Japanese) character set 15

- source file
 - Web Template Translator 25
 - Web Template Wizard 31
- starting Web Template Designer 19, 42
- static list
 - XForms selectMany 81
 - XForms selectOne 79
- submitting predefined business documents 142

T

- table
 - adding 43, 88
 - adding row 89
 - managing 88
- template, Web
 - checking in 107
 - checking out 108
 - locating 109
 - overview 105
 - previewing 105
 - version history 110
- testing, Web template 111, 114
- text formatting 91
- Time Sheet business process 128
- title bar, Web Template Designer 20
- tool bar
 - HTML tab 22
 - Web Template Designer 21
 - XForms tab 21
- translation process
 - Web Template Translator 26
 - Web Template Wizard 31
- translator, Web template 12

U

- Unicode character set 15
- Unicode Transformation Format 8 (UTF-8) 14, 109
- uninstalling, Web Template Designer 19
- uploading process, File Upload utility 32
- URI 112, 113
- user account, role-based security 15

- user name, Web Extensions 124
- user password, changing 142
- user roles, Web Extensions 124

V

- version history
 - Web resource 158
 - Web template 110
- View Folders area
 - work page 136

W

- Web application
 - definition 149
 - deploying 160
- Web archive (.war) file 155
- Web archive (.war) file utility 37
- Web Extensions
 - basics 9
 - components 10
 - encoding 14
 - example 10, 13
 - features 9
 - internationalization 14
 - localization 14
 - permissions 124
 - user roles 124
 - utilities 17
 - XML resource bundle 14
- Web Extensions HTTP Server adapter 112, 113
- Web resource
 - checking in 155
 - checking out 156
 - general 154
 - locating 157
 - managing 154
 - searching 157
 - version history 158
- Web Suite Approval Notify Web template 119
- Web Suite ASN Web templates 119
- Web Suite expense report Web templates 119
- Web Suite invoice Web templates 119

- Web Suite PO Web templates 119
- Web Suite purchase request Web templates 119
- Web Suite query list Web template 119
- Web Suite remittance advice Web templates 119
- Web Suite time sheet Web templates 119
- Web template
 - adding XML namespace 66
 - checking in 107
 - checking out 108
 - creating 39, 151
 - creation process 40
 - editing 61, 63
 - locating 109
 - overview 11, 105
 - preinstalled 115
 - previewing 105
 - searching 109
 - testing 111, 114
 - version history 110
- Web template .xml files, editing 150
- Web template .xrb files, editing 151
- Web Template Designer
 - code pane 23
 - creating Web template 41
 - description 17
 - design pane 23
 - editing Web template 61
 - general 11
 - installing 18
 - interface 20
 - menu bar 20
 - node pane 23
 - properties pane 23
 - quitting 19
 - requirements 18
 - starting 19
 - title bar 20
 - tool bar 21
 - uninstalling 19
- Web Template Editor 12, 39, 63
- Web Template Translator
 - destination file 25
 - example 27
 - HTML>XForms button 25
 - navigating 25
 - output space 25
 - overview 12
 - quitting 24
 - source file 25
 - translation process 26
 - using 26
 - XForms>HTML button 26
 - XML>XForms button 26
- Web Template Wizard
 - DDF>XForms 31
 - destination file 31
 - general 30
 - installing 31
 - output space 31
 - quitting 31
 - source file 31
 - translation process 31
- Web templates, associated predefined business process 123
- Web.xml 99, 150
- web.xml 163
- WebSuiteChangePassword
 - Web templates 119
- websuitemain.xrb 152, 153
- WebSuiteSkin.xrb, editing 34
- WebTemplateName.xml 150
- WebTemplateName.xrb 151
- WebTemplateNameView.xml 151
- WebTemplateTest.bp, checking in 113
- webx.war 161, 162
- work page 135
 - Create New area 139
 - editing 152
 - Message List area 137
 - overview 135
 - View Folders area 136
- ws_buyer 124
- ws_employer 124
- ws_finance 124
- ws_hr 124
- ws_manager 124
- ws_purchaser 124

ws_supplier 124
www.w3.org 38, 41

X

xalan.jar 161
xerces.jar 161
xercesImpl.jar 161
XForms
 bind 65
 button 55, 59, 76
 child node 43, 54, 58, 68
 child node value 43, 54, 59
 controls 40, 53, 58, 62
 file upload 82
 group 65
 instance 42, 65, 68
 model 42, 54, 58, 65, 67
 selectMany 80
 selectOne 78
 servlet 106
 structure 42, 65
 submitInfo 54, 59, 66
XForms servlet, deploying 160
XForms structure, adding 42
XForms tab
 figure 21
 new button 21
 new selectMany 22
 new selectOne 22
 new textfield 21
xforms.jar 161
xforms.war 161
XForms>HTML button 26
XML
 namespace 41, 65, 66, 105
 resource bundle 14, 15
XML>XForms button 26
xsdlib.jar 161