



Wonderware®  
Information Server  
Win-XML Exporter  
User's Guide

All rights reserved. No part of this documentation shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Invensys Systems, Inc. No copyright or patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this documentation, the publisher and the author assume no responsibility for errors or omissions. Neither is any liability assumed for damages resulting from the use of the information contained herein.

The information in this documentation is subject to change without notice and does not represent a commitment on the part of Invensys Systems, Inc. The software described in this documentation is furnished under a license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of these agreements.

© 2010, 2014 by Invensys Systems, Inc. All rights reserved.

Invensys Systems, Inc.  
26561 Rancho Parkway South  
Lake Forest, CA 92630 U.S.A.  
(949) 727-3200

<http://www.wonderware.com>

For comments or suggestions about the product documentation, send email to [ProductDocumentationComments@invensys.com](mailto:ProductDocumentationComments@invensys.com).

All terms mentioned in this documentation that are known to be trademarks or service marks have been appropriately capitalized. Invensys Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this documentation should not be regarded as affecting the validity of any trademark or service mark.

Alarm Logger, ActiveFactory, ArcestrA, Avantis, DBDump, DBLoad, DT Analyst, Factelligence, FactoryFocus, FactoryOffice, FactorySuite, FactorySuite A<sup>2</sup>, InBatch, InControl, IndustrialRAD, IndustrialSQL Server, InTouch, MaintenanceSuite, MuniSuite, QI Analyst, SCADAAlarm, SCADASuite, SuiteLink, SuiteVoyager, WindowMaker, WindowViewer, Wonderware, Wonderware Factelligence, and Wonderware Logger are trademarks of Invensys plc, its subsidiaries and affiliates. All other brands may be trademarks of their respective owners.

---

# Contents

	Welcome .....	7
	About This Book .....	7
	Wonderware Information Server Documentation .....	7
	Document Conventions .....	8
	Technical Support Information .....	9
<b>Chapter 1</b>	<b>Introduction to the Win-XML Exporter .....</b>	<b>11</b>
	About Converted InTouch Windows .....	12
	About Win-XML Projects .....	12
	About WindowSets and TableWeaverSets .....	12
	Data Sources for Converted Windows .....	13
	Using WindowViewer as a Data Source .....	14
	Using Data Sources Other than InTouch .....	15
	Application Logic for Converted Windows .....	15
	About Publishing Windows .....	15
	Security for Viewing Published Windows .....	16
	Combined Internet/Intranet Access .....	17
<b>Chapter 2</b>	<b>Converting and Publishing InTouch Windows ..</b>	<b>19</b>
	Installing the Win-XML Exporter .....	19
	Upgrading the Win-XML Exporter .....	20

Starting the Win-XML Exporter .....	21
Win-XML Exporter Toolbar Buttons .....	22
General Steps for Using the Win-XML Exporter .....	22
Working with Projects .....	23
Creating a New Project .....	23
Opening an Existing Project .....	25
Viewing or Editing Project Properties .....	25
Deleting a Project .....	26
Working with WindowSets .....	26
Creating a WindowSet .....	26
Renaming a WindowSet .....	27
Editing Properties for a WindowSet .....	27
Adding Files to a WindowSet .....	28
Deleting a WindowSet .....	28
Working with TableWeaverSets .....	29
Creating a TableWeaverSet .....	29
Renaming a TableWeaverSet .....	29
Editing Properties for a TableWeaverSet .....	30
Adding Files to a TableWeaverSet .....	31
Deleting a TableWeaverSet .....	31
Working with InTouch Windows .....	31
Opening an InTouch Application .....	32
Converting InTouch Windows .....	32
Viewing the Window Conversion Status .....	33
Publishing Windows to Wonderware Information Server .....	34
Previewing a Converted Window .....	34
Disabling Automatic Window Resizing .....	35
Deleting a Window .....	35
Using Multiple Web Servers for WindowSets and TableWeaverSets .....	36
<b>Chapter 3 Designing InTouch Application Windows .....</b>	<b>39</b>
Technologies Used by Converted Windows .....	39
XML .....	40
XSL and XSLT .....	40
VML .....	40
Client Connection Speeds .....	41
Text Conversions .....	41
Graphic Conversions .....	42
Rendering Bitmaps .....	43
Rendering Complex Graphic Elements .....	43

---

Script Conversions .....	44
Script Functions Supported .....	45
Operators Supported .....	48
Support for Iterative Statements .....	48
Support for Variables .....	48
Workaround for QuickFunctions .....	49
Animation Conversions .....	50
Window Conversions .....	51
Limiting Window Complexity .....	51
Dynamic Scaling of InTouch Windows .....	52
Using Multiple Windows for a Single InTouch Screen .....	52
Support for Popup Windows .....	52
Support for ActiveX Controls .....	53
Unsupported InTouch Elements .....	53
Index.....	55



# Welcome

## About This Book

This guide provides information on how to best design InTouch application windows so they can be successfully converted to web pages and published to Wonderware Information Server (formerly known as SuiteVoyager). This guide also describes how to use the conversion and publishing application, the Win-XML Exporter.

You may need to adjust the InTouch windows before they can be converted. This guide assumes you are an experienced InTouch user and are familiar with the applications you are converting. You must also have a working knowledge of client/server technologies, Internet technologies, and the Microsoft Windows Server operating system.

## Wonderware Information Server Documentation

The Wonderware Information Server documentation set includes the following guides:

**Wonderware Information Server Administration Guide.** This guide describes how to install Wonderware Information Server, administer and maintain an installed Wonderware Information Server, and how to access factory information over the Internet/intranet via the Wonderware Information Server. The installation section includes hardware and software requirements and migration instructions. The administration section includes information on configuring data sources, managing security, and performing customizations, accessing factory information over the Internet/intranet via the Wonderware Information Server, and using the TableWeaver and reporting features.

**Wonderware Information Server Win-XML Exporter Guide.**

This guide describes how to use the Wonderware Information Server Win-XML Exporter to convert InTouch application windows to web pages and publish them to the Wonderware Information Server software. This guide also describes how to best design InTouch windows so that they can be successfully converted.

**ArchestrA Web Exporter Guide.** This guide describes how to use the ArchestrA Web Exporter to publish ArchestrA Symbols and InTouch Windows to Wonderware Information Server. This guide also describes how to add, delete and modify WindowSets and TableWeaverSets of the project and preview them in the preview window of the ArchestrA Web Exporter.

All of these guides are provided in PDF form and can be printed. Also, information included in the Administration and Win-XML guides is provided in browser-based online Help, which can be accessed from the Wonderware Information Server software.

## Document Conventions

This documentation uses the following conventions:

---

<b>Convention</b>	<b>Used for</b>
Initial Capitals	Paths and filenames.
<b>Bold</b>	Menus, commands, dialog box names, and dialog box options.
Monospace	Code samples and display text.

---



## Technical Support Information

Wonderware Technical Support offers a variety of support options to answer any questions on Wonderware products and their implementation.

Before you contact Technical Support, refer to the relevant section(s) in this documentation for a possible solution to the problem. If you need to contact technical support for help, have the following information ready:

- The type and version of the operating system you are using.
- Details of how to recreate the problem.
- The exact wording of the error messages you saw.
- Any relevant output listing from the Log Viewer or any other diagnostic applications.
- Details of what you did to try to solve the problem(s) and your results.
- If known, the Wonderware Technical Support case number assigned to your problem, if this is an ongoing problem.



# Chapter 1

## Introduction to the Win-XML Exporter

You can use the Wonderware Information Server Win-XML Exporter to create a conversion project, convert InTouch windows to XML, organize these windows into WindowSets and SymbolSets, and publish the windows to Wonderware Information Server. You do not need to know any HTML or XML coding to use the Win-XML Exporter.

Converted windows on Wonderware Information Server can be downloaded and viewed by casual users or decision makers over the Internet/intranet using only Microsoft Internet Explorer, thus providing a thin client solution. The converted windows are shown and animated using standard web technologies and a downloadable client VML graphics program.

You can use the ArcestrA Web Exporter to convert ArcestrA Symbols and managed InTouch application windows into .xap files, which can be accessed in the Web portal. This is similar to the Win-XML Exporter, which is used to convert InTouch windows to XML. The ArcestrA Web Exporter produces a rich feature output using ArcestrA Symbols. For more information about the ArcestrA Web Exporter, see the *ArcestrA Web Exporter Guide*.

## About Converted InTouch Windows

InTouch windows that run over the Internet in a browser are different than applications that run within InTouch WindowViewer. Although both are created using InTouch WindowMaker, not all WindowMaker functionality is compatible with current Internet technology.

Wonderware Information Server does not replace InTouch. Rather, it adds an additional capability: casual users can get real-time data from the factory floor over the Internet, using a standard web browser and no additional tools.

The browser shows the converted InTouch windows to the casual user, and allows the user to interact with the factory floor via dynamic HTML (DHTML), XML, and ActiveX controls. The data travels between the browser and the Wonderware Information Server web server, which communicates with a running InTouch application on the application server node.

Most InTouch windows can be easily converted into a web page viewable in Internet Explorer. However, not all InTouch functionality is compatible with web technology. For more information on how InTouch windows are converted, see Chapter 3, "Designing InTouch Application Windows."

## About Win-XML Projects

The Win-XML Exporter uses project files to save global settings such as where converted XML windows are published to and what data sources are used for the windows. Project files are saved with the .wxe extension. Only one project can be open in the Win-XML Exporter at a time.

Projects can contain one or more WindowSets, which are named groupings of converted windows for use in Wonderware Information Server, or SymbolSets, which are for use specifically with TableWeaver.

## About WindowSets and TableWeaverSets

WindowSets and TableWeaverSets allow you to organize converted windows within a Win-XML Exporter project. WindowSets are used for converting information for use in Wonderware Information Server, and TableWeaverSets are used for converting information for use specifically in TableWeaver.

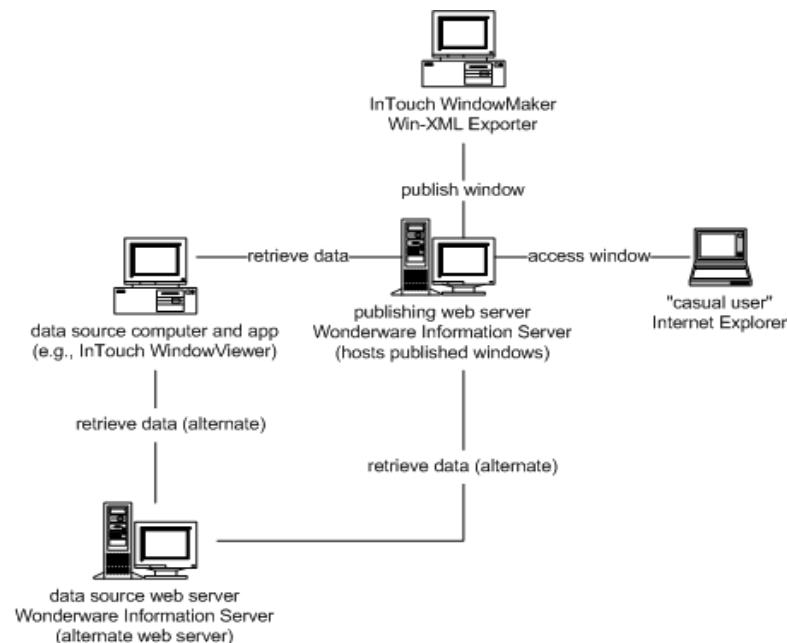
A converted window must be part of a WindowSet or TableWeaverSet, and WindowSets and TableWeaverSets must be part of a project. By default, the Win-XML Exporter creates a WindowSet to contain all of the InTouch windows converted from a single application.

## Data Sources for Converted Windows

You can set the data sources for converted windows at the project level or at the WindowSet or SymbolSet level. The data source definition for any given project, WindowSet, or SymbolSet is a combination of:

- **A data provider web server.** This web server handles all run-time data requests to the data source. Typically, this server is the same Wonderware Information Server you publish your converted windows to. However, you can specify a different Wonderware Information Server. If so, the data provider web server must be able to access the publishing web server via the Internet or intranet. If your users use the Secure Sockets Layer (SSL) protocol to access the site, you must include `HTTPS://` in the web server URL for the data source.
- **A computer (node) on the control network.** This is the computer on which the application is loaded that supplies the run-time data.
- **An application running on the specified node.** Typically, the application is running an InTouch application (the default). However, you can specify other applications such as the Wonderware Historian, or I/O Servers.

The following graphic shows the relationship between the different computers:



By default, a WindowSet or SymbolSet inherits the data source properties of the project to which it belongs. However, each WindowSet or SymbolSet can be configured to use a different data source than the project.

For example, you have several windows show information about a production line. You have several production lines that work the same, but use different data sources. You can use the same windows, but you need to copy them to different WindowSets or SymbolSets and set the data sources for each WindowSet or SymbolSet to the appropriate production line.

Each WindowSet or SymbolSet can potentially use a different InTouch node, and you may want to design windows that, by referencing separate nodes, effectively act as new applications.

## Using WindowViewer as a Data Source

The data source for a converted InTouch window is defined in the WindowSet or SymbolSet properties. The preferred method is to use a running InTouch application as the data source. This provides the InTouch application developer with many advantages over acquiring data directly from an I/O Server, such as support for InTouch dot fields, human readable tagnames, InTouch remote tags, alarming, scaling, and the invocation of server-based (InTouch) scripting. There are some advanced I/O Servers offering some of this capability, but most do not.

If the InTouch application uses remote tags, your window can have as many data sources as are defined by the InTouch application. For example, if an InTouch window is receiving data from six remote InTouch computers, then the converted window has seven data sources - one for the main application and six pointing to the remote tag data sources. When the window is converted and published to Wonderware Information Server, the remote tags receives data from the various data sources and effect changes in the graphics window. Also, the administrator can enable to write-back capabilities to modify the values for the tags at the source.

In some cases, you may want to bypass WindowViewer and retrieve data directly from an I/O Server. You can do this by specifying to use the I/O Server as a data source when setting up a conversion project in the Win-XML Exporter. You can also host converted windows on one site and retrieve data from another. For more information, see [Creating a New Project](#) on page 23.

## Using Data Sources Other than InTouch

You can specify applications such as InControl, the Wonderware Historian, or I/O Servers to provide data for published windows. The following table provides examples for how these data sources are defined.

	<b>I/O Server</b>	<b>Wonderware Historian</b>	<b>InControl</b>
<b>Web Server URL</b>	http://MyWebServer	http://MyWebServer	http://MyWebServer
<b>Data Node Name</b>	IOServer1	Historian1	InControl1
<b>Application</b>	Modbus.Exe	aaHIOSVRSvc.Exe	RTEngine
<b>Topic</b>	Tagname	Tagname	Tagname

## Application Logic for Converted Windows

InTouch windows that have been converted to XML do not contain any application logic; that is, they do not execute scripts locally. What this means for InTouch application developers is that the InTouch windows shown to end users are separated from the data sources and any application logic (for example, scripts) that allows a user to interact with the data sources.

Browser-based windows can execute scripts, but these scripts must not be of the application type and must only be used when the logic pertains to the individual use of the individual window. The browser-based windows interact and change according to the application logic running in the InTouch plant floor application.

The logic and data are provided by a specified InTouch computer (node), which also provides the animation information and actual data for the window.

## About Publishing Windows

Publishing a project moves all WindowSets, SymbolSets, converted windows, image files, and configuration files to the Wonderware Information Server web server specified as the publishing host. After the entire project has been published, you can publish incremental changes by just publishing new individual WindowSets, SymbolSets, or windows. The new files are merged into the WindowSets and SymbolSets on the server.

To publish windows to Wonderware Information Server, you must:

- Have been given permission to publish.
- Have an HTTP connection to the site.

You do not have to be on the web server computer to publish a window. Also, you do not need to have a connection with the application node supplying the data.

By default, any user can download and install the Win-XML Exporter, but only server administrators can publish windows. If the administrator of the web server wants to give others the right to publish, then these users must be added to the web server's local Windows security group, which is usually named *computername\_Admins* or *computername\_Authors*. These security groups are created by the Front Page Server Extensions and are used to control who can publish to the web server. Usually, a typical setup of the extensions adds the local administrator group to the *computername\_Admins* group.

For more information on security, see your *Wonderware Information Server Administration Guide*.

---

**Note:** You cannot publish windows via SSL.

---

## Security for Viewing Published Windows

Wonderware Information Server is fully integrated with Windows security. Wonderware Information Server administrators can give known Windows domain users privileges to read, write, or configure the system. Wonderware Information Server also supports the SSL Internet protocols for additional security, and the system works behind firewalls.

Access, right-privileges, and other security settings are determined by the role to which a user is assigned by the Wonderware Information Server administrator. For example, you have an InTouch window containing a setpoint. In WindowViewer, this setpoint is usually protected by a disable link on the setpoint object. The user who logs into InTouch with the appropriate access level can change the setpoint. Each object a user can change can have different access levels. In Wonderware Information Server, no user can effect changes (write-back) unless this has been enabled for the site by the Wonderware Information Server administrator, there is a specific write-back license installed, and that user has been assigned to a role with write-back privileges. A user with write-back privileges can change any value that can be changed.



There are also advanced security options beyond what Wonderware Information Server provides. For detailed information on configuring security, see the *Wonderware Information Server Administration Guide*.

## Combined Internet/Intranet Access

If your users are accessing Wonderware Information Server from both the Internet and a company intranet, then both the Internet and intranet URL for your web server must be listed in your corporate DNS server or LMHOSTS files. This is due to the fact that a web server is usually known by one name on the Internet and another on the intranet. Another solution is to create different WindowSets or SymbolSets for the different users each using the web server URL that can be resolved by their system.



---

## Chapter 2

# Converting and Publishing InTouch Windows

Converting and publishing InTouch application windows can be accomplished by using the Win-XML Exporter application. The Win-XML Exporter can be installed from Wonderware Information Server.

For conceptual information on using the Win-XML Exporter, see "Introduction to the Win-XML Exporter" on page 11.

For information on designing InTouch windows so that they convert successfully, see "Designing InTouch Application Windows" on page 39.

## Installing the Win-XML Exporter

The Win-XML Exporter can be installed from a link on the Wonderware Information Server home page. If necessary, obtain the URL or address from your Wonderware Information Server administrator. The Win-XML Exporter does not require a license upgrade.

The Win-XML Exporter setup program:

- Installs the application to your local computer
- Adds a shortcut to the Win-XML Exporter on your Windows Start menu (For example, \Start\Programs\Wonderware\InTouch\Win-XML Exporter)

---

**Note:** You must install the Win-XML Exporter on the same computer as InTouch WindowMaker if the windows you are converting contain ActiveX controls.

---

### To install the Win-XML Exporter

- 1 Access the Wonderware Information Server Home Page.
- 2 In the **System** panel, expand **Administration** and then click Win-XML Exporter. The Win-XML Exporter page appears.
- 3 Click **Download Win-XML Exporter Now**. The **File Download** dialog box appears.
- 4 Click **Run this program from its current location** and then click **OK**. The **Security Warning** dialog box appears.
- 5 Click **Yes**. The **Wonderware Information Server Win-XML Exporter Setup - Welcome** dialog box appears.
- 6 Click **Next**. The **Wonderware Information Server Win-XML Exporter Setup - Destination Folder** dialog box appears.
- 7 Specify a destination folder for the program files. This is where the Win-XML Exporter is installed.
- 8 Click **Next**. You are prompted to verify that you want to install.
- 9 Click **Next**. After the application files are installed, the **Setup Complete** dialog box appears.
- 10 Click **Finish**.

## Upgrading the Win-XML Exporter

If you have a previous version of Win-XML Exporter on your system, the old version must be removed first. Your converted project windows can still be used from your old projects, but you may want to re-convert to take advantage of new features, such as remote data source support. However, you do not need to re-convert and re-publish the InTouch windows for them to run on the latest version of the Wonderware Information Server.

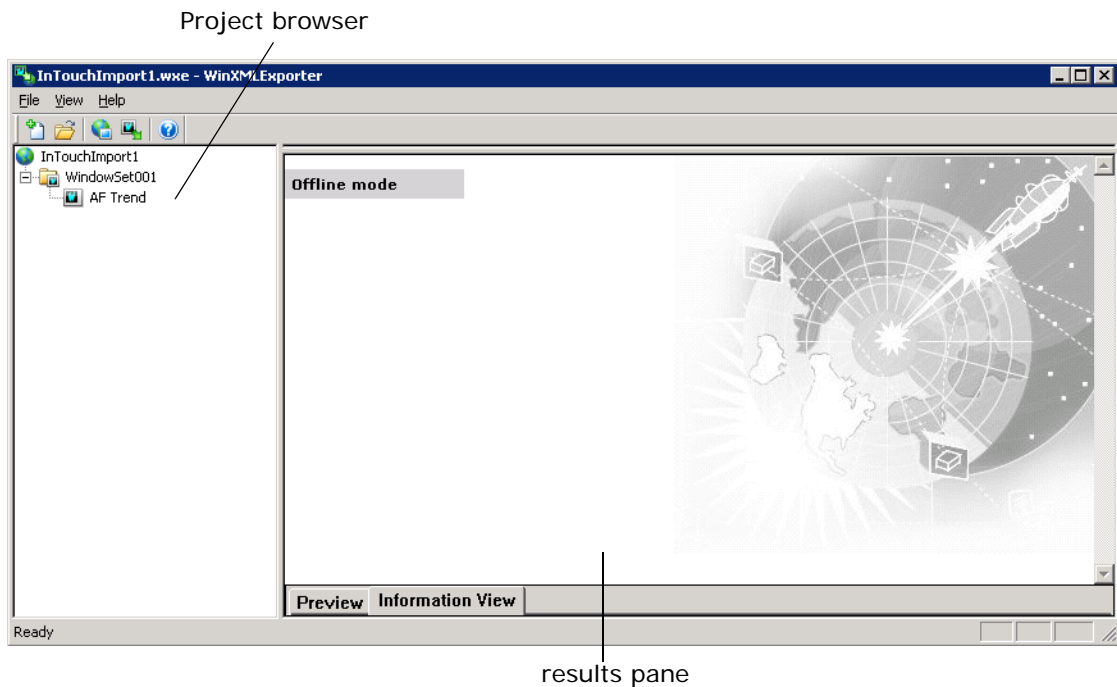
### To remove the Win-XML Exporter

- 1 On the Windows **Start** menu, point to **Settings**, point to **Control Panel**, and then click **Add/Remove Programs**.
- 2 Locate the Win-XML Exporter in the list and then click **Remove**.

# Starting the Win-XML Exporter

## To start the Win-XML Exporter

- 1 To start the exporter, do either of the following:
  - On the Windows Start menu, point to Programs and then click the Win-XML Exporter item in your InTouch program group.
  - In **WindowMaker**, double-click the Wonderware Information Server Win-XML Exporter icon in the Application browser.
- 2 The Win-XML Exporter appears.








The Win-XML Exporter has three main areas:

- The project browser, which shows all of the associated WindowSets, SymbolSets, and windows.
- The results pane, which shows information regarding the success of a window conversion (Information View) or a preview of the converted window that is selected in the browser (Preview).
- The InTouch window pane, which lists InTouch windows available for conversion. The InTouch pane is not initially shown. You must drag the bar at the top of the results pane to see the InTouch pane.

## Win-XML Exporter Toolbar Buttons

The following table describes the tool bar buttons:

Button	Description
	Create a new project file. For more information, see "Creating a New Project" on page 23.
	Opens an existing project file. For more information, see "Opening an Existing Project" on page 25.
	Switches the project to Online or Offline mode. For more information, see "Previewing a Converted Window" on page 34.
	Opens an InTouch application to convert. For more information, see "Converting InTouch Windows" on page 32.
	Opens the Help file.

## General Steps for Using the Win-XML Exporter

The general steps for using the Win-XML Exporter are as follows:

- 1** Create a new project file. The project file is used to store information such as the URL to the Wonderware Information Server where you are publishing and the path to local project files. Local project files also include converted windows, and WindowSets, SymbolSets, and default data source information. For more information, see "Creating a New Project" on page 23.
- 2** Rename the default WindowSet or SymbolSet for the project to something meaningful. For more information, see "Renaming a WindowSet" on page 27 and "Renaming a TableWeaverSet" on page 29.
- 3** Open an InTouch application that contains the windows you want to convert. For more information, see "Opening an InTouch Application" on page 32.
- 4** Convert the InTouch application windows. For more information, see "Converting InTouch Windows" on page 32.

- 5 (Optional) Create additional WindowSets or SymbolSets to contain the converted windows from a single InTouch application. For more information, see "Creating a WindowSet" on page 26 and "Creating a TableWeaverSet" on page 29.
- 6 Publish the entire project to Wonderware Information Server. You can also publish incrementally. For more information, see "Publishing Windows to Wonderware Information Server" on page 34.

There are a few WindowMaker elements that are not compatible with Internet technology, however, but all other elements, including text, graphics, action scripts and animation are automatically converted for you. For more information on conversions, see Chapter 3, "Designing InTouch Application Windows."

## Working with Projects

The first thing to do in the Win-XML Exporter is to create a new project file. This file stores all the configuration information for the InTouch windows you want to publish.

For more information, see "About Win-XML Projects" on page 12.

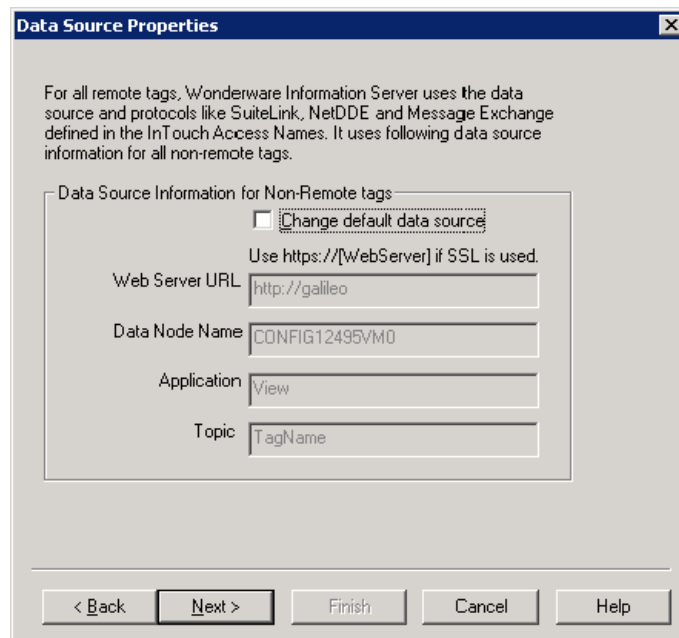
## Creating a New Project

### To create a new project

- 1 On the **File** menu, click **New Project**. The **Project Properties** dialog box appears.
- 2 In the **Project Name** box, type a name for the project. A new folder is created using this name.
- 3 To change the default working folder for the project, click the ellipsis button to the right of the **Local Project Path** box. This project path can point to any folder on your computer.
- 4 Click **Next**. The **Publishing Properties** dialog box appears.
- 5 In the **Wonderware Information Server URL**, type the URL where the Wonderware Information Server is located. Converted windows are published to this server. For example, `http://MyWebServer/InformationServer`. You must include the `/InformationServer` virtual directory as part of the URL.

The wizard does not validate access to the server.

- 6 Click **Next**. The **Data Source Properties** dialog box appears.



- 7 If your application will not be receiving data from a running WindowViewer application on the data node, select **Change default data source** and then configure the data source options:

#### **Web Server URL**

The URL for the web server that serves as the data source. This server can be a different web server than the one you are publishing the converted windows to, allowing converted windows to be hosted on one portal and retrieve data from another. If you are using secure socket layers (SSL), remember to type "https:" instead of "http:" If this host name is not static in your environment (for example, if clients will access the web server across a NAT router), then you can use "localhost" as the URL, as long as the Runtime DB Handler service is on the same Wonderware Information Server hosting the graphics.

For information on using multiple web servers, see "Using Multiple Web Servers for WindowSets and TableWeaverSets" on page 36.

#### **Data Node Name**

The name of the computer to receive data from. This can be either the computer on the plant floor that is running the InTouch application, an I/O Server, Wonderware Historian, and so on. This node will supply the data for the converted window.



**Application**

The executable name of the application on the data node from which to receive data.

For example, for WindowViewer, specify VIEW.

**Topic**

The specific subgroup of data elements for the specified application. Examples of topics are Tagname for InTouch, or the name of an Excel spreadsheet.

If you retrieve data from an I/O Server, you must select the **Use Tagname as Item Name** option in the WindowMaker Tagname Dictionary when defining the tags for the window being converted.

- 8 Click Next.** The **New Project Information** dialog box appears.
- 9** Verify that the information is correct. Click **Finish**.
- 10** The project you created now appears in the left pane of the Win-XML Exporter with the default name of WindowSet001, regardless of whether you're creating a WindowSet or a SymbolSet. You must create the SymbolSet name manually. To change the name of the set, see "Renaming a WindowSet" on page 27 and "Renaming a TableWeaverSet" on page 29.

## Opening an Existing Project

**To open an existing project**

- 1** On the **File** menu, click **Open Project**. The **Open** dialog box appears.
- 2** Browse to the project file (.wxe) to open and then click **Open**.
- 3** Click **OK**.

## Viewing or Editing Project Properties

The properties at the project level are the default values assigned to each WindowSet or SymbolSet when it is created.

**To view or edit properties for a project**

- 1** In the browser, right-click on the project name and then click **Properties**. The **Project Properties** dialog box appears.

The options on the **Project Properties** tab are read-only at the project level. However, you can change these values at the WindowSet or SymbolSet level. For more information, see "Editing Properties for a WindowSet" on page 27 and "Editing Properties for a TableWeaverSet" on page 30.

- 2 Click the **Publishing Properties** tab.
- 3 In the **Wonderware Information Server URL** box, type the URL where the Wonderware Information Server is located. Converted windows are published to this server. For example, `http://MyWebServer/InformationServer`. You must include the `/InformationServer` virtual directory as part of the URL.  
  
Click the **Data Source Properties** tab to configure the data source. For more information, see "Creating a New Project" on page 23.
- 4 Click **OK**.

## Deleting a Project

You cannot delete a project from within the Win-XML Exporter. Delete the project files manually from your computer using Windows Explorer.

## Working with WindowSets

WindowSets are used to group converted InTouch windows so that you can operate on them as a unit. For more information, see "About WindowSets and TableWeaverSets" on page 12.

## Creating a WindowSet

WindowSets do not inherit any properties from the WindowSet or project from which they were created. Instead, the WindowSet inherits the properties of the project it is under.

### To create a new WindowSet

- 1 In the browser, right-click on either an existing WindowSet or project on which to base the new WindowSet and then click **New WindowSet**. A new WindowSet is created in the project.
- 2 If desired, change the WindowSet name. For more information, see "Renaming a WindowSet" on page 27.
- 3 If desired, edit the default WindowSet properties. For more information, see "Editing Properties for a WindowSet" on page 27.

## Renaming a WindowSet

### To rename a WindowSet

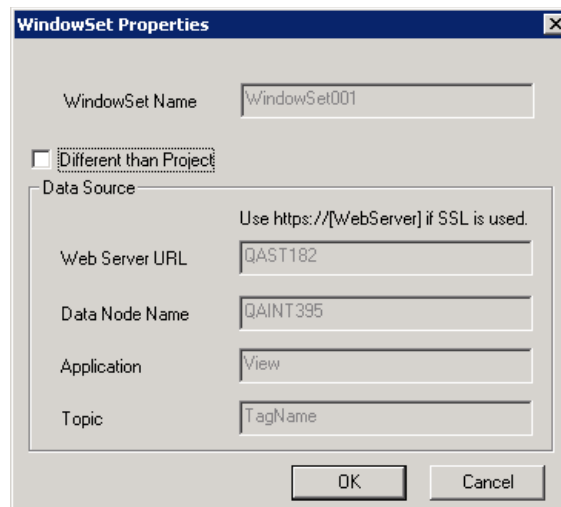
- 1 In the browser, right-click on the WindowSet name and then click **Rename**. A box appears around the name.
- 2 Type the new name.
- 3 Press **Enter** on your keyboard.

## Editing Properties for a WindowSet

**Note:** If you change the properties for a WindowSet, you must re-drag and then republish the windows in the WindowSet to propagate the property changes throughout the WindowSet.

### To edit WindowSet properties

- 1 In the browser, right-click an existing WindowSet and then click **Properties**. The **WindowSet Properties** dialog box appears.



- 2 To change the data source information, click **Different than Project**. When you change the data source for a WindowSet, all windows in that WindowSet use the new data source. Options are as follows:

#### Web Server URL

Used to specify the URL for the web server that serves as the data source. This server can be a different web server than the one you are publishing the converted windows to, allowing converted windows to be hosted on one portal and retrieve data from another. If you are using secure socket layers (SSL), remember to type "https:" instead of "http:." If this host name is not static in your environment (for example, if clients will access the web server

across a NAT router), then you can use "localhost" as the URL, as long as the Runtime DB Handler service is on the same Wonderware Information Server hosting the graphics.

For information on using multiple web servers, see "Using Multiple Web Servers for WindowSets and TableWeaverSets" on page 36.

**Data Node Name**

The name of the computer to receive data from. This can be either the computer on the plant floor that is running the InTouch application, an I/O Server, Wonderware Historian, and so on. This node will supply the data for the converted window.

**Application**

The executable name of the application on the data node from which to receive data.

For example, for WindowViewer, specify VIEW.

**Topic**

The specific subgroup of data elements for the specified application. Examples of topics are Tagname for InTouch, or the name of an Excel spreadsheet.

- 3 Click **OK**.

## Adding Files to a WindowSet

To add files to a WindowSet, simply drag-and-drop them from the pane of InTouch windows.

Be careful when adding windows from different InTouch applications to the same WindowSet. For example, you can accidentally overwrite same-named graphics in the images folder for the WindowSet.

## Deleting a WindowSet

**To delete a WindowSet**

- ◆ Right-click on the WindowSet in the browser and then click **Delete**.

## Working with TableWeaverSets

TableWeaverSets are used to group converted InTouch windows so that you can operate on them as a unit within TableWeaver. For more information, see "About WindowSets and TableWeaverSets" on page 12.

### Creating a TableWeaverSet

TableWeaverSets do not inherit any properties from the TableWeaverSet or project from which they were created. Instead, the TableWeaverSet inherits the properties of the project it is under.

#### To create a new TableWeaverSet

- 1 In the browser, right-click on either an existing TableWeaverSet or project on which to base the new TableWeaverSet and then click **New TableWeaverSet**. A new TableWeaverSet is created in the project.
- 2 If desired, change the TableWeaverSet name. For more information, see "Renaming a TableWeaverSet" on page 29.
- 3 If desired, edit the default TableWeaverSet properties. For more information, see "Editing Properties for a WindowSet" on page 27.

### Renaming a TableWeaverSet

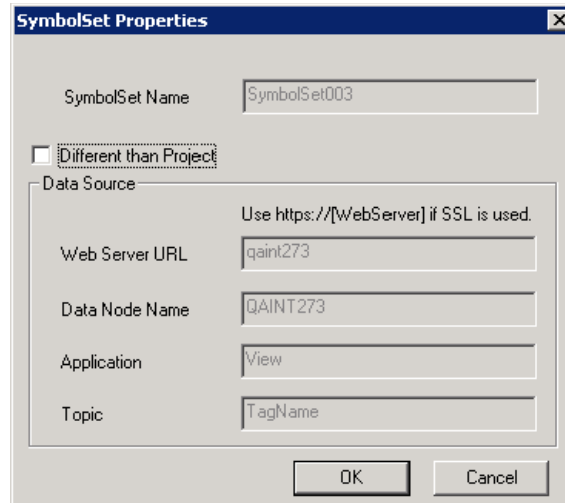
#### To rename a TableWeaverSet

- 1 In the browser, right-click on the TableWeaverSet name and then click **Rename**. A box appears around the name.
- 2 Type the new name.
- 3 Press **Enter** on your keyboard.

## Editing Properties for a TableWeaverSet

### To edit TableWeaverSet properties

- 1 In the browser, right-click an existing TableWeaverSet and then click **Properties**. The **TableWeaverSet Properties** dialog box appears.



- 2 To change the data source information, click **Different than Project**. When you change the data source for a TableWeaverSet, all windows in that SymbolSet use the new data source. Options are as follows:

#### **Web Server URL**

Used to specify the URL for the web server that serves as the data source. This server can be a different web server than the one you are publishing the converted windows to, allowing converted windows to be hosted on one portal and retrieve data from another. If you are using secure socket layers (SSL), remember to type "https:" instead of "http:" If this host name is not static in your environment (for example, if clients will access the web server across a NAT router), then you can use "localhost" as the URL, as long as the Runtime DB Handler service is on the same Wonderware Information Server hosting the graphics.

For information on using multiple web servers, see "Using Multiple Web Servers for WindowSets and TableWeaverSets" on page 36.

**Data Node Name**

The name of the computer to receive data from. This can be either the computer on the plant floor that is running the InTouch application, an I/O Server, Wonderware Historian, and so on. This node will supply the data for the converted window.

**Application**

The executable name of the application on the data node from which to receive data.

**Topic**

The specific subgroup of data elements for the specified application. Examples of topics are Tagname for InTouch, or the name of an Excel spreadsheet.

- 3 Click **OK**.

## Adding Files to a TableWeaverSet

To add files to a TableWeaverSet, simply drag-and-drop them from the pane of InTouch windows.

Be careful when adding windows from different InTouch applications to the same TableWeaver. For example, you can accidentally overwrite same-named graphics in the images folder for the TableWeaverSet.

## Deleting a TableWeaverSet

**To delete a TableWeaverSet**

- ◆ Right-click on the TableWeaverSet in the browser and then click **Delete**.

## Working with InTouch Windows

Most InTouch windows can be converted into a web page viewable in Internet Explorer. However, not all InTouch functionality is compatible with web technology. For more information on how InTouch windows are converted, see Chapter 3, "Designing InTouch Application Windows."

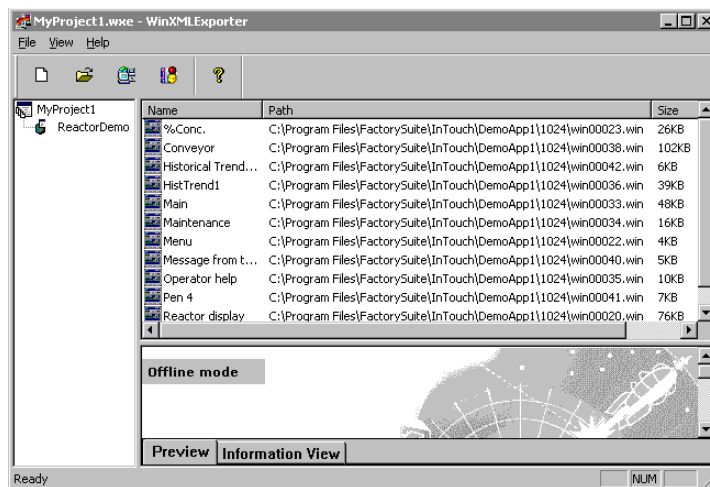
For general information about converted InTouch windows, including possible data sources, see Chapter 1, "Introduction to the Win-XML Exporter."

## Opening an InTouch Application

### To open an InTouch application

- 1 On the **File** menu, click **Open InTouch Application**. The **Browse For Folder** dialog box appears.
- 2 Navigate to the InTouch application folder containing the window(s) to convert. Click **OK**.

The available InTouch windows from the folder you selected now appear in the top right pane of Win-XML Exporter.



For each window, the pane shows the caption name, the path and converted file name, and the converted size in KB.

## Converting InTouch Windows

### To convert an InTouch window

- 1 In the pane of available InTouch windows, select the window to convert.

**Tip** To select multiple windows, hold down the CTRL key while making your selections. To select multiple consecutive windows, hold down the SHIFT key, select the first window, and then select the last window.

- 2 Drag the window on the WindowSet or SymbolSet name in the browser.

The conversion to XML is performed automatically during the drag-and-drop operation.



By default, the bottom right pane of the Win-XML Exporter only shows the status of the LAST window converted. Therefore, if you convert twelve windows at one time, and the first eleven have errors, you still see a successful conversion message as long as the twelfth window converts with no errors. For information about viewing the status for a particular window, see "Viewing the Window Conversion Status" on page 33.

For information about previewing the converted window as it will appear in a browser, see "Previewing a Converted Window" on page 34.

You can change the WindowSet properties but you have to re-drag and drop the InTouch Windows again to the WindowSet and then Re-Publish it. Then the WindowSet properties change will take the latest changes. Once you re-drag the InTouch windows to the WindowSet then the XML files will get changed with the latest WindowSet properties. After publishing the windows again takes the latest changes.

## Viewing the Window Conversion Status

### To view the conversion status for a particular window

- 1 In the browser, select the desired window.
- 2 Access the information view by performing any of the following:
  - Right-click on the window and select **Information View**.
  - In the bottom right pane of the exporter, click the **Information** tab.
- 3 Expand the **Information** item. If all elements were successfully converted in the last window you selected, you see **Successful Conversion!!!**



- 4 Expand the different sections to see information about warnings or errors.

## Publishing Windows to Wonderware Information Server

### To publish a project or individual WindowSet or TableWeaverSet

- ◆ Right-click on the project, WindowSet, or TableWeaverSet in the browser and then click **Publish**.

## Previewing a Converted Window

The preview window has two modes: Online and Offline. The default mode is Offline. In Offline mode, the preview window shows you how your converted window will look; however, you cannot preview animations because the window is not currently receiving any data.

Popup windows are listed in both Online and Offline mode. Popup windows are converted and can be accessed by the user at run time (in the browser). However, popup windows do not appear in the list of windows that the user sees in the browser. Therefore, no user can directly access a popup window.

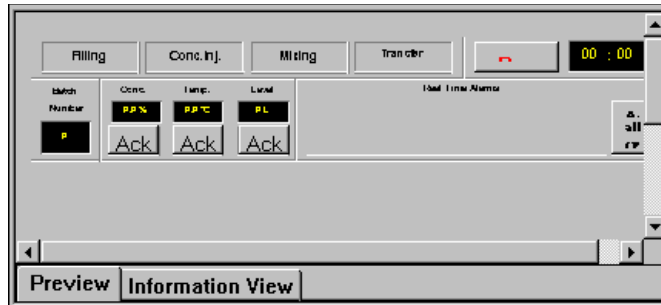
To preview a run-time version of the window, you must first publish the window. After that, if you specify the Online mode, an attempt is made to establish a connection to the Wonderware Information Server that you have specified for the window. If the server has access to the data source, animations function as if you were viewing the window in a browser over the Internet/intranet. If the animations are not working, the data source may not be operating, the data source specifications may be incorrect, or the scripting that drives the animation may need to be modified.

You must be assigned to the Administrator Role by your Wonderware Information Server administrator to use the Online mode. Even then, the preview pane only provides Read Only access; you can not test TouchLink input dialogs or any features that require write-back permissions.

### To preview a converted window

- 1 In the browser, select the desired window.
- 2 Access the information view by performing any of the following:
  - Right-click on the window and select **Preview**.
  - In the bottom right pane of the exporter, click the **Preview** tab.

- 3 A static preview of the converted window appears in the bottom pane.



- 4 To preview a run-time version of the window, first publish the window. Then, on the **File** menu, click **Go Online**.

## Disabling Automatic Window Resizing

By default, the converted window is scaled in the user's browser so that it fits in the display area. However, you can turn this feature off before you publish the window. Disabling the automatic window resizing opens the window full size as it was drawn in InTouch. Scroll bars appear, if necessary, to provide the user access to the hidden areas of the screen.

---

**Note:** The Fit to Window option does not appear on the context menu when you right-click a SymbolSet.

---

### To disable window resizing

- 1 Right-click on the window to resize in the navigation tree and then click **Fit to Window** so that a check mark no longer appears.
- 2 Re-publish the window.

## Deleting a Window

If you are in Online mode, you can delete windows from the site with the proper security clearance from the administrator. If you are offline, you can delete windows from the project workspace.

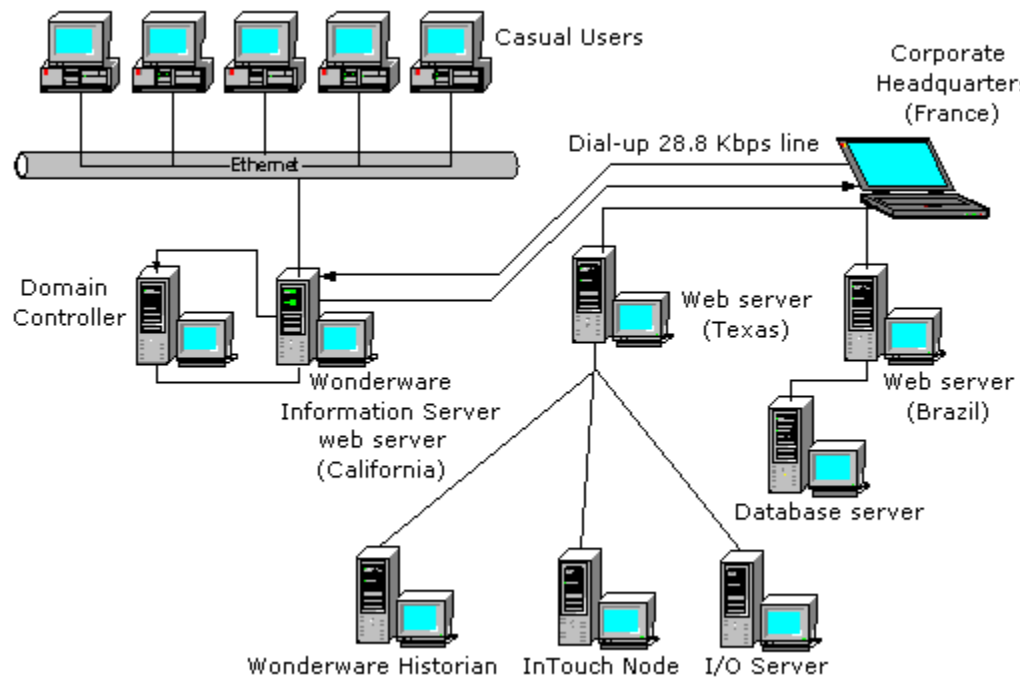
### To delete a window

- ◆ Right-click on the window in the browser and then click **Delete**.

## Using Multiple Web Servers for WindowSets and TableWeaverSets

Corporate sites usually have different requirements, and you want to show only the information that is important to the corporate casual users. Standard corporate InTouch windows can be created and exported to multiple plants to publish on their web servers, allowing the corporate casual user to view the same information from multiple plants without having to know the names for all the different WindowSets and TableWeaverSets at each company. This saves the corporate user a lot of time when trying to access pertinent information about individual plants, as well as combining information company wide.

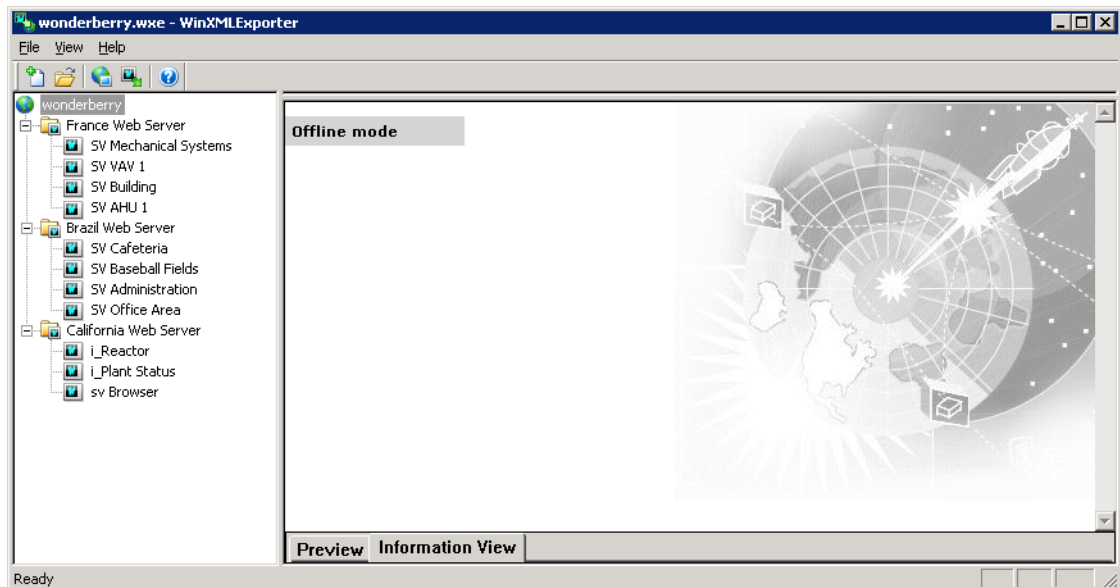
When a casual user accesses the Wonderware Information Server for information, the first thing that happens is that they are authenticated on the domain controller. After authentication, the web server automatically connects the user to the information they want, even if it is being hosted on another web server. This is a good way to design a corporate site. It still has all the security features built into Windows and the site, and the data retrieval is transparent to the casual user. In the following example, the corporate computer is retrieving data from a web server in Texas, but it only had to call up the Wonderware Information Server web server in California. This computer can also access information from Brazil during the same session, without any additional steps, configuration, or security logons.



The client user never needs to worry about changing the connection to the data source, because the administrator has this configured before the client attempts to access the information. The Win-XML Exporter is used to easily configure the data to come from different web servers, as well as different nodes.

You can configure the data source for a WindowSet or WindowSet by modifying the properties in the **WindowSet Properties** or **TableWeaverSet Properties** dialog boxes. For more information, see "Editing Properties for a WindowSet" on page 27 and "Editing Properties for a TableWeaverSet" on page 30.

For example, a WindowSet titled **France Web Server** is configured to retrieve information from Wonderware Information Server titled **France**, which has a computer name (node name) of **Demo**. The windows listed under the France Web Server name are all receiving information from the Wonderware Information Server France and node Demo.



Likewise, the windows listed under the WindowSet name **Brazil Web Server** are receiving their information from the Wonderware Information Server web server Brazil, which has a node name of **Demo**.



---

## Chapter 3

# Designing InTouch Application Windows

Go through the InTouch windows and modify them so that they convert with no errors and show the information that the user is looking for. You must consider several factors when designing or reviewing your InTouch windows so that they can be converted successfully and can be downloaded and viewed with optimum performance by end users.

---

**Tip** Before you edit an existing InTouch window, copy the entire application to another location to preserve the original.

---

## Technologies Used by Converted Windows

Wonderware Information Server introduces a new generation in process graphics designed specifically for use by thin browser-based clients. With the advent of thin-client computing, it is desirable for packets of information sent out via an intranet/Internet to be as succinct and compact as possible. Traditional graphics have always been large files and taken long periods of time to download.

Wonderware Information Server delivers interactive HTML pages by converting existing InTouch graphic windows and associated animation to XML (Extensible Markup Language), applying an XSL transformation (XSLT), and rendering VML (Vector Markup Language) graphics on the fly at the client machine. These XML-based graphics with client-side graphics rendering reduce the amount of information to be transported between client and server by as much as 80 percent.

## XML

Extensible Markup Language (XML) is a language designed to deliver structured information over the web more effectively than current languages used for web publishing. XML separates the content of a document from its presentation and provides a common format for transferring data across the World Wide Web or a company intranet. The result is a technology that makes data available regardless of the proprietary systems involved. Virtually any kind of data can be encapsulated in XML, moved across networks, processed automatically, and published dynamically. Ultimately, XML opens a new world of possibilities for sharing, managing and publishing information on the web.

For example, consider a simple InTouch two dimensional tank-fill graphic. The XML code needed to generate this is much lighter and simpler than traditional means of generating graphics. Also, this 2-D graphic can then be made 3-D simply by adding a single attribute line of code. Generating a cylinder instead of a rectangle fill is just one more attribute line of code. Thus, the code is much lighter and therefore faster for transferring data across the web to thin clients.

## XSL and XSLT

Extensible Stylesheet Language (XSL) is an web standard that allows stylesheets to be designed for rendering XML content on screen. It allows different styles to be applied to your web content and have them accepted by the browser.

XSL transformations (XSLT) is a language for transforming XML documents into other XML documents. XSLT is used by XSL to describe how the document is transformed.

---

**Note:** The Wonderware Information Server automatically determines the browser's capability to use XSLT. Internet Explorer 8.0 or higher (the recommended browser) supports XSLT.

---

## VML

Vector Markup Language (VML) is an XML-based exchange, editing, and delivery format for high-quality vector graphics on the web that meets the needs of both productivity users and graphics design professionals. XML is a simple, flexible, and open text-based language that complements HTML. VML has been proposed to the World Wide Web Council as a standard for vector graphics on the web. Microsoft Office 2000 also supports VML. Microsoft Word, Excel, and PowerPoint can be used to create VML graphics.



## Client Connection Speeds

When designing your InTouch application windows, consider who the user of your browser-enabled InTouch application windows is and when and where they will be accessing your application from. For example, will they be using a high-speed connection such as a corporate intranet? For this type of user, you may decide that an application can include many bitmaps and other graphics. The high-speed intranet allows for fast download times. Also, the internal user has access to your organization's help desk and support personnel.

In contrast, are your users using a slow dial-up connection from a laptop computer several time zones away? This type of user may have a very slow, even unreliable, connection, depending upon the quality of local phone systems. Also, a user located in a different time zone may not be able to contact anyone for assistance in resolving problems. This user needs a simple, very clear, graphic-light application that provides just the specific data they need.

The relative speed of showing the converted window in the end user's browser is determined by a number of factors:

- The size (in bytes) of the converted window
- The number of tags and/or animated objects to convert.
- The actual speed for a user's Internet connection (the result of several factors):
  - The user's modem speed
  - The quality and speed of the user's Internet access
  - The number of users accessing the Internet
  - The number of users accessing the same Wonderware Information Server

## Text Conversions

The Win-XML converter converts all text objects. When designing windows for Wonderware Information Server, use only TrueType fonts. These fonts are scaled to fit as a user adjusts the size and shape of the browser window that shows the InTouch window.

Non-TrueType fonts do not scale correctly and may be truncated when the browser window is resized.

Keep in mind that the browser relies on fonts installed on the user's computer to show text. Try to use fonts that are widely available. Arial and Times New Roman, for example, are typically installed with Windows.

Add blank spaces to the end of text objects. This keeps the text from being truncated when shown in a converted window. Also, keeping the window size below 800 x 600 pixels is important for windows that include text.

## Graphic Conversions

The Win-XML Exporter converts the following WindowMaker graphic elements to XML:

- All basic primitive shapes, such as lines, circles, polygons, and rectangles
- Symbols
- Cells
- Bitmaps (color depth may be reduced)
- InTouch wizards
- Symbol Factory Wizards (from the FactorySuite Productivity Pack)
- SmartSymbols

---

**Note:** The Alarm Wizard and the History Wizard do not convert. Also, use as few Symbol Factory Wizards as possible for performance reasons.

---

If you are using bitmaps or complex graphics, there are additional design considerations. For more information, see "Rendering Bitmaps" on page 43 and "Rendering Complex Graphic Elements" on page 43.

The ArchestrA Web Exporter converts the following elements into .xap files:

- ArchestrA Symbols contained in one of the following:
  - Graphic Toolbox
  - Automation Object template
  - Automation Object instance
- InTouch Windows of managed InTouch applications. The Windows can contain any of the following:
  - InTouch native graphics
  - SmartSymbols, Wizards
  - Embedded ArchestrA Symbols

For more information about the ArchestrA Web Exporter, see the *ArchestrA Web Exporter Guide*.

## Rendering Bitmaps

The Win-XML Exporter converts all of your bitmap images into Internet standard.gif and .jpeg files.

When using bitmaps, ask yourself the following:

- Can I express the same information using simple InTouch graphics instead of a bitmap? Although .gif and .jpeg graphic formats produce smaller files than the original bitmaps, XML-rendered InTouch graphics download much faster over the Internet. Use these in place of bitmaps whenever possible.
- At what connection speed will the end users be accessing the converted window? If your application is going to be used by remote users with slow modems, keep bitmaps to a minimum. Also, browser and/or ISP time-out limits may prevent large graphics from showing over dial-up connections.

In some instances, however, bitmaps may be a better choice than using a complex InTouch graphic. For more information, see "Rendering Complex Graphic Elements" on page 43.

## Rendering Complex Graphic Elements

InTouch was designed to render polygons (graphical elements). A three-dimensional variable fill tank is rendered in InTouch as a series of polygons with the fill as additional elements, also rendered as a polygons. This simple, three-dimensional object can contain hundreds of polygons, depending upon the detail of the illustration. The Win-XML Exporter faithfully converts all polygons to XML objects. As XML objects can have more intelligence than graphic primitives, the browser may render the polygons less efficiently than WindowViewer. It is therefore best to limit the amount of detailed polygon objects.

In some cases, it may make sense to convert a complex InTouch graphic element to a bitmap to speed up rendering such as if you have a complex valve made up of many polygons. An alternative is to use a bitmap to represent the valve, which is then rendered as a .jpeg file. Depending on the complexity of the XML object and the speed of your end users' connections, the bitmap may be a better solution.

If you have a window that contains numerous complex graphic elements, or possibly even just one complex symbol, a bitmap of the original window may render faster than the window full of complex graphics. If the window is only going to be viewed over an intranet connection at 100 Mbps, then using a bitmap representation of the graphic(s) may provide for quicker performance.

**To create a bitmap of a complex InTouch window**

- 1 In WindowMaker, select the window you want to capture as a bitmap.
- 2 On your keyboard, hold down the **ALT** key and press **Print Screen**.
- 3 Open the Windows Paint program.
- 4 On the **Edit** menu, click **Paste**.
- 5 Crop the graphic to include just the window you want.
- 6 In WindowMaker, create a new window.
- 7 Insert the bitmap and then add any other graphic, text, or animation elements.

## Script Conversions

The following table provides information on the InTouch scripting features that are and are not supported:

<b>Script Type</b>	<b>Conversion Support</b>
Window	OnShow supported.
Button	OnButtonDown supported.
Action	All supported, with some limitations.
ActiveX	Event scripts supported.
Functions	Most are supported, with a few restrictions. For more information, see Script Functions Supported on page 45.
Operands	All are supported but MOD. For more information, see Operators Supported on page 48.
Variables	Supported, with some restrictions. For more information, see Support for Variables on page 48.
QuickFunctions	None. However, for a workaround, see Workaround for QuickFunctions on page 49.

Application logic (scripting) resides on a server connected to the Wonderware Information Server software, and not in the converted window. The scripting used to create an animation effect must be contained in the InTouch application that is the data source for the window being shown. For more information, see "Application Logic for Converted Windows" on page 15.

No equivalent function to “Allow While Window Showing” scripts is available. However, because only one window can show at a time, and all elements in the converted window work, this same result is achieved.

## Script Functions Supported

The following script functions are automatically converted by the Win-XML Exporter and are supported by the Wonderware Information Server software. Except where noted, all parameters behave normally; however, because Wonderware Information Server presents windows, WindowSets, and SymbolSets rather than applications, functions need to be approached differently for the Internet.

Supported script functions are listed alphabetically.

Abs(Number);

ArcCos(Number);

ArcSin(Number);

ArcTan(Number);

Cos(Number);

DText(Discrete\_Tag, OnMsg, OffMsg);

Exp(Number);

GetPropertyD(“ControlName.Property”, Tagname);

See "Using Script Functions that Call Windows Controls" on page 47.

GetPropertyI(“ControlName.Property”, Tagname);

See "Using Script Functions that Call Windows Controls" on page 47.

GetPropertyM(“ControlName.Property”, Tagname);

See "Using Script Functions that Call Windows Controls" on page 47.

HideSelf;

Int(Number);

Log(Number);

LogN(Number, Base);

Pi();

PrintWindow(“Window”, Left, Top, Width, Height, Options);

Only the current window/web page is printed.

Round(Number, Precision);

SetPropertyD("ControlName.Property", DiscreteTag);  
See "Using Script Functions that Call Windows Controls" on page 47.

SetPropertyI("ControlName.Property", Integer);  
See "Using Script Functions that Call Windows Controls" on page 47.

SetPropertyM("ControlName.Property", "MessageTag");  
See "Using Script Functions that Call Windows Controls" on page 47.

Sgn(Number);

Show "Window";

ShowAt("Window", Horiz, Vert);  
The Horiz and Vert parameters are ignored because window positioning is not supported.

ShowTopLeftAt("Window", Horiz, Vert);  
The Horiz and Vert parameters are ignored because window positioning is not supported.

Sin(AngleNumber);

Sqrt(Number);

StringASCII("Char");  
Returns a Unicode value.

StringChar(ASCII);  
Returns a Unicode character.

StringFormatGeneral()

StringFromIntg(Number, Base);

StringFromReal(Number, Precision, "Type");  
The Type parameter is not supported. Numbers are returned as decimal values.

StringFromTime()

StringInString()

StringLeft("Text", Chars);

StringLen("Text");

StringLower("Text");

StringMid("Text", StartChar, Chars);

StringReplace()

StringRight("Text", Chars);

StringSpace(NumSpaces);

StringTest()

```
StringToIntg("Text");
StringToReal("Text");
StringTrim()
StringUpper("Text");
Tan(AngleNumber);
Trunc(Number);
wcAddItem("ControlName", "MessageTag");
    See "Using Script Functions that Call Windows Controls" on
    page 47.
wcClear("ControlName");
    See "Using Script Functions that Call Windows Controls" on
    page 47.
wcDeleteItem("ControlName", ItemIndex);
    See "Using Script Functions that Call Windows Controls" on
    page 47.
wcDeleteSelection("ControlName");
    See "Using Script Functions that Call Windows Controls" on
    page 47.
wcErrorMessage(ErrorNumber);
wcFindItem("ControlName", "MessageTag", DiscreteTag, Tagname);
wcGetItem("ControlName", ItemIndex, Tagname);
wcGetItemData("ControlName", ItemIndex, Tagname);
wcInsertItem("ControlName", ItemIndex, "MessageTag");
wcSetItemData("ControlName", ItemIndex, Number);
```

## Using Script Functions that Call Windows Controls

When a Windows control is used dynamically at run time, changes made by the user reflect the added items in the Windows control only in that user's instance of the InTouch window. If the user adds items to a listbox in Internet Explorer, it is not reflected in the InTouch WindowViewer window on the View node. Therefore, the changes are not available if another user opens the same window on his/her computer.

## Operators Supported

All InTouch operators are supported and work as expected, except for MOD. For more information on InTouch operators, see your InTouch documentation.

The behavior of the MOD operator with real (that is, floating point) operands is different in Wonderware Information Server than in InTouch. In InTouch, the real operands are truncated to their integer values before the MOD operator is applied. Consequently, you always get an integer as a result. In Wonderware Information Server, this truncation step is not performed, so the result of the MOD operation more accurately reflects a result value as a real instead of an integer.

If you want the InTouch behavior, you can simply truncate the operand values that surround the MOD operator. The following are examples of how to do this:

```
integer_tag = Trunc(real_tag1) MOD Trunc(real_tag2);  
integer_tag = Trunc(real_tag1/real_tag2) MOD Trunc(real_tag3);
```

## Support for Iterative Statements

When writing InTouch scripts containing iterative statements, use local variables; do not use tagnames. An example of an iterative statement is:

```
For Index = 1 to 100  
  Next;
```

## Support for Variables

Regardless of what InTouch allows a DIM variable to be named, it must adhere to the following rules imposed by JavaScript (which is what InTouch scripts are converted to) if it is to be used in Wonderware Information Server.

The first character must be a letter (either uppercase or lowercase) or an underscore ( \_ ), or a dollar sign ( \$ ). Subsequent characters can be letters, numbers, underscores, or dollar signs.

Also, the variable name cannot be any of the following reserved JavaScript keywords:

break	false	in	this	void
continue	for	new	true	while
delete	function	null	typeof	with
else	if	return	var	case
debugger	export	super	catch	default
extends	switch	class	do	finally
throw	const	enum	import	try



## Workaround for QuickFunctions

The Win-XML Exporter does not directly support QuickFunctions, in that it does not convert the QuickFunction code used by converted windows. However, you can mimic InTouch QuickFunctions by creating user defined JavaScript functions in the CustomAPI.js file. The functions in the CustomAPI.js file are globally available to all converted windows.

The CustomAPI.js file is typically located on Wonderware Information Server computer in the following folder:

```
\inetpub\wwwroot\Wonderware Information Server\_library\scripts
```

For access to the CustomAPI.js file, contact your Wonderware Information Server system administrator.

---

**Important:** When creating user-defined functions, DO NOT use names that match Wonderware Information Server function names. If the name matches a Wonderware Information Server function, then the function in the CustomAPI.js file overrides the Wonderware Information Server function.

---

When the Win-XML Exporter encounters a QuickFunction in an action script or animation expression, it passes it through just like any other function. The only difference is that you receive a warning that the function is not supported.

For example, you have an InTouch window that uses a QuickFunction called **CubeIt()** in a button action script. The code for the **CubeIt()** QuickFunction in InTouch looks like this.

```
<<<< Script Start >>>>
    return (nValue ** 3);
<<<< Script End >>>>
```

Where nValue is the real parameter value passed into the **CubeIt()** QuickFunction that is to be cubed.

Your InTouch window has something like the following calling the **CubeIt()** QuickFunction in a button action script.

```
<<<< Script Start >>>>
    MyRealTag = CALL CubeIt($Second);
<<<< Script End >>>>
```

To make the **CubeIt()** function available in any Wonderware Information Server converted window, you must create a **CubeIt()** function in the CustomAPI.js file.

In Notepad, open the CustomAPI.js file and add the following **CubeIt()** JavaScript function:

```
<<<< Script Start >>>>
    function CubeIt( nValue )
    {
        return nValue = Math.pow(nValue, 3);
    }
<<<< Script End >>>>
```

After you have saved this addition to the CustomAPI.js file, you can now run converted InTouch windows in Internet Explorer that call the **CubeIt()** function.

For information on JavaScript, see the Microsoft JScript documentation available on Microsoft's website.

## Animation Conversions

Appropriate animation for the Internet is used to provide visual status information. Including animations that do not represent meaningful data is distracting to the user and needlessly adds to download times. Simple ON/OFF, OPEN/CLOSE or variable fill information is well-suited to a browser's capability to render graphics, as well as the size constraints for windows that appear within the browser.

Although current Internet technology does not provide for all of the animation features available within InTouch, most animation effects can be converted. The Win-XML Exporter converts all InTouch animation, with the exception of rotated text, and the HideWindow push button animation, and bitmap fills. All expressions are converted normally unless the expression contains an InTouch function call that is not supported by Wonderware Information Server.

In order for animation to function correctly, the scripting used to create the animation effect must be contained in the InTouch application that is the data source for the window being shown. For more information, see "Application Logic for Converted Windows" on page 15.

Also, be sure that the script that contains your animation is supported by the site. In many InTouch applications, the scripting to create the animation effects is contained in the Window WhileShowing Script. A simple example is changing the size of a tank agitator paddle to create the appearance of movement. Because WhileShowing Scripts do not apply to Wonderware Information Server windows, the agitator paddle does not change. To resolve this problem, use a WhileTrue Condition script instead to update the animation tag data. The converted window then shows the animation.

Animation motion (position or size) in converted windows may not appear as smooth as the original animation. The smoothness of the motion is affected by the speed of the client connection and the number of animation effects in the current window. To achieve the smoothest rendering of motion in the browser, limit the amount of moving animation objects in any one window.

Wonderware Information Server does not support the InTouch Tooltip function.

## Window Conversions

When planning for window conversions, consider the following:

- Are there so many elements on the window that it will take too long to download?
- Is the window so large that users will have to scroll excessively to see the information?
- Can the window be scaled up and down in the browser without affecting the display?
- Does your application require that the user see more than one window at a time?
- Are any included objects extending beyond the border of the window?
- Does your application include pop-up windows?

For information on window scripts and window properties, see your InTouch documentation.

## Limiting Window Complexity

Limit the complexity of the converted windows so that they can show in the end users browser quickly. A window's complexity contributes to the file size that needs to be transmitted to the user's computer before a window can be rendered. A user who has a typical dial-up connection may need several minutes to download a window that appears almost instantly over a LAN. You need to know what sort of connections your users have and adapt your graphic elements accordingly. For more information on using graphics within windows, see "Graphic Conversions" on page 42.

## Dynamic Scaling of InTouch Windows

The InTouch windows converted and published to the site are configurable, at publishing time, to either be scaled down to the size of the display area or to be shown full size and accessible with a scroll bar, if necessary.

## Using Multiple Windows for a Single InTouch Screen

When creating an InTouch application, you can design several windows that appear together to represent a single screen visible to the user. However, in Wonderware Information Server, only one window can be shown at a time within the browser. If you want to convert an InTouch screen into a Wonderware Information Server window, you must combine all of the elements of the various windows that comprise that screen into a single window, which is then converted. When a user activates a Show function “window name”, that window appears in place of the current window, provided that the window type is a replace or overlay. In other words, overlay windows work like replace windows in Wonderware Information Server.

For the browser to show two windows at one time, one window must have the properties set to Popup. For the other window, create a Window Script as follows:

Select **On Show**, and type “show “windowname” where windowname is the name of the Popup window.

## Support for Popup Windows

When a user activates a Show “window name” script, and that window is a popup type, then the window appears in a new instance of Internet Explorer. This instance of Internet Explorer does not have the normal menus and toolbars. It simply shows the contents of the window.

Keep in mind that this is not an “always on top” or foreground display, and that if you change focus to the previous instance of Internet Explorer or another Windows application, then this window is hidden. You can access the window from the Windows taskbar or by holding down Alt key and pressing the TAB key to select the window.

## Support for ActiveX Controls

Because Internet Explorer is a good ActiveX container, ActiveX controls are well suited for converted windows. Using ActiveX controls in place of InTouch controls (for example, a list box) is a good design practice.

Exported Wonderware and Wonderware Partner ActiveX controls reside on the Wonderware Information Server as downloadable cab files. If an ActiveX control is included in an exported window, it is downloaded to the client computer. For third-party controls, the Win-XML Exporter creates the .cab file on the fly and then publishes it to the site. The Win-XML Exporter does not sign the .cab file for third-party controls, nor does it ensure that the control works over the Internet. This is the responsibility of the InTouch application developer. Also, third-party ActiveX controls may require a license on the client computer.

The Win-XML Exporter must be used on the same node as the InTouch application with the ActiveX controls in order for the cabling function to work. Also, the inclusion of supporting DLLs in the cab is not supported.

When a user first shows a converted window that uses an ActiveX control(s), the control is automatically downloaded and registered on the client computer. ActiveX event scripts used in an ActiveX control are converted to Java Script and runs as expected on the client.

ActiveX methods can be executed in converted windows using action scripts or window OnShow scripts.

ActiveX properties linked to InTouch tagnames work normally, as they are linked to the business logic (for example a View node).

ActiveX objects that retrieve data from a database operate correctly on an over the Internet/intranet.

---

**Note:** Many third-party ActiveX controls require separate licenses. Contact your vendor for details.

---

## Unsupported InTouch Elements

The following InTouch elements are either not converted by the Win-XML Exporter or not supported by the Wonderware Information Server software:

- Trends created using the Historical Trend Wizard.
- Alarm displays created using the InTouch Alarm Display Wizard. However, alarms can be configured and viewed using the Alarms page on the Wonderware Information Server software.

**Note:** Unlike the InTouch HMI, where all the alarms in the sub-areas are also captured and shown along with the alarms for an area, in Wonderware Information Server the alarms reported in the sub-areas are not shown in the current alarm display for an area.

---

- Any Statistical Process Control (SPC) elements.
- QuickScripts, Application scripts, and Window While Showing, Data Change, and Condition scripts.
- Expression support in Min/Max:  
In the InTouch window, if the Analog User Input graphic element is configured with the expression for Min and/or Max, the run-time display behaves as if the object did not exist on the InTouch window and the following warning is shown in the Win-XML Exporter Information view:  
The <Analog User Input > at location <X,Y> has an expression for <Min/Max> limit, hence this object is being deleted during conversion.
- Intelligent format support:  
In the InTouch window, if the Analog Value Display or the Analog User Input graphic element is configured with the format of **Text String**, the conversion takes place without warnings. If it is configured with the format of **Real**, **Fixed Decimal**, **Integer** or **Exponential**, the run-time display behaves as if the object is set with the format of **Text String** and the following warning is shown during conversion:  
The <Analog User Input / Analog Value Display > at location <X,Y> has a format of <format name>, but its format is being reset to Text String.  
  
If the configured format is **Hex** or **Binary**, the run-time display behaves as if the object did not exist on the InTouch window and the following warning is shown during conversion:  
The <Analog User Input / Analog Value Display > at location <X,Y> has a format of <format name>, hence this object is being deleted during conversion.
- Window Name Property:  
In the InTouch window, if the **Window Name** property is used during conversion, the following warning is shown in the Information View:  
Unsupported function: GetWindowName  
  
At run-time, if the **GetWindowName** function is used, the following warning is shown:  
The Intouch function GetWindowName() is not supported.

---

# Index

## A

action scripts 44  
ActiveX controls 53  
ActiveX scripts 44  
animations 50  
application logic 15, 44  
ArchestrA 11  
ArchestrA Symbols 11  
ArchestrA Web Exporter 11

## B

bitmaps 42, 43  
button scripts 44

## C

circles 42  
client connection speeds 41

## D

data source 15  
  for projects 24  
  for windows 13, 14

## E

Extensible Markup Language 39, 40

Extensible Stylesheet Language 40

## F

fonts 41  
functions 44, 45, 47

## G

graphic conversions 42  
graphics 40  
  complex 43

## I

installation 19  
Internet access 17  
InTouch application  
  opening 32  
intranet access 17  
iterative statements 48

## J

JavaScript 48, 49

## L

lines 42

**O**

Offline mode 34  
Online mode 34  
operands 44  
operators 48

**P**

polygons 42, 43  
popup windows 34, 52  
primitive shapes 42  
projects  
  about 12, 23  
  creating 23  
  deleting 26  
  editing properties 25  
  opening 25  
  publishing 34  
publishing windows 16

**Q**

QuickFunctions 44, 49

**R**

rectangles 42

**S**

script conversions 44  
script functions 45, 47  
scripts 15, 48  
  action 44  
  ActiveX 44  
  animation 50  
  button 44  
  functions 44  
  operands 44  
  QuickFunctions 44  
  variables 44  
  window 44  
security 34  
  for publishing windows 16  
  for viewing windows 16  
SymbolSets 12  
  about 29  
  creating 29  
  editing 30  
  publishing 34  
  renaming 29

**T**

TableWeaverSet  
  deleting 31  
  editing properties 30  
  renaming 29  
TableWeaverSets 12  
  deleting 31  
text conversions 42  
toolbar buttons 22

**V**

variables 44, 48  
Vector Markup Language 39, 40  
VML 39, 40

**W**

web servers 36  
Window scripts 44  
windows  
  about 31  
  about converted 12  
  converting 32, 51  
  data sources for 13  
  deleting 35  
  limiting complexity 51  
  previewing 34  
  publishing 16  
  scaling 35, 52  
  technologies used 39  
  using multiple 52  
  viewing 16  
  viewing conversion status 33  
WindowSets 12  
  about 12, 26  
  adding files 28  
  creating 26  
  deleting 28  
  editing 27  
  publishing 34, 37  
  renaming 27  
  using multiple web servers 36  
WindowViewer 14  
Win-XML Exporter  
  installing 19  
  starting 21  
  toolbar buttons 22  
  using 22



Wizards 42

X

XML 39, 40

XSL 40

XSL transformations 39, 40

XSLT 39, 40

