Oracle® Insurance IStream

IStream Writer Administrator's Guide

Release 1.3

E14880-01

May 2008

ORACLE®

INSURANCE

Copyright

CONTENTS

Chapter 1

Overview

Welcome to the IStream Writer Administrator's Guide. This guide is for advanced technical users (such as system and database administrators) who need to install, configure and troubleshoot IStream Writer.

This chapter describes:

- *Document Conventions* on page 8
- *IStream Writer* on page 9
- *IStream Writer Documentation* on page 10
- *Contacting Skywire Software for Help* on page 11

Document Conventions

Tips, Notes, Important Notes and Warnings

Tip: A **Tip** provides a better way to use the software.

Note: A **Note** contains special information and reminders.

Important: An **Important** note contains significant information about the use and understanding of the software.

Warning: A **Warning** contains critical information that if ignored, may cause errors or result in the loss of information.

Other Document Conventions

- Microsoft Window names, buttons, tabs and other screen elements are in bold, for example: Click **Next**.

- paths, URLs and code samples are in the Courier font, for example: `C:\Windows`

- values that you need to enter or specify are indicated in the italicized Courier font, for example, *server_name*

- values that are optional are indicated with square brackets, for example `[reserved]`

IStream Writer

IStream Writer streamlines the processes for product development, state filings and policy production. It allows product designers to specify how wordings apply to defined groupings, such as states, lines of business and products, and then automates the creation of all state-specific documents. Wording experts use the existing IStream Writer document that already contains the information defined by product development. An author can then import the IStream Writer document into IStream Author, automatically transforming it into a model document.

IStream Writer Documentation

IStream Writer includes the following documents on the installation CD. If you need a copy of any of these documents, please contact your system or product administrator.

- The *IStream Writer User Guide* describes how to use IStream Writer. It contains overviews, step-by-step procedures and descriptions of the screens and fields.

- The *IStream Writer Administrator's Guide* includes system requirements, technical information about the Data Elements File and configuration files, and installation and configuration information.

- The *IStream Writer Release Notes* contain general product information, product enhancements and new features, supported platforms and third-party software, and known limitations.

- The *IStream Writer Online Help* contains the same information as the User Guide, but in an online help format with a search tool. Both contain an index and a table of contents.

Contacting Skywire Software for Help

Customer Support hours are 8:00 AM to 8:00 PM. (Eastern Time), Monday through Friday. Outside of these hours, send us a detailed e-mail message and you will be contacted during regular business hours. Please provide detailed information, as described in the *Support Checklist*.

Contact Information

Mail:	Customer Support Skywire Software 19 Allstate Parkway, Suite 400 Markham, Ontario, L3R 5A4
Phone:	1-905-513-7466
Fax:	1-905-513-1419
Email:	directsupport@skywiresoftware.com
Web:	www.skywiresoftware.com

Support Checklist

When contacting Skywire Software Customer Support, please provide the following information:

- Your name, company name, email address, and phone number
- The exact version number, including any service pack numbers of
  - all your Skywire Software products
  - your network software
  - your database, including the type
  - Microsoft Windows and Microsoft Word
  - Microsoft .NET Framework
  - your DMS (Document Management System)
- Error messages and the circumstances of their occurrence
- A full description of the problem:
  - What happened? What were the sequence of events that preceded the problem?
  - In which screen or window did the problem occur?
  - Was the problem the result of pressing a key?
  - Did the screen freeze? What functions of the software are affected?
  - How many people are affected?

Chapter 2

Installing IStream Writer

This chapter describes how to install IStream Writer, and how to configure IStream Author for use with IStream Writer.

The installation stages are:

- *Stage 1: Check Your System Requirements* on page 14
- *Stage 2: Install IStream Writer* on page 16
- *Stage 3: Configure Author for IStream Writer* on page 17
- *Stage 4: Configure IStream Writer to Access the DMS* on page 18

If you are upgrading IStream Writer, refer to this section:

- *Upgrading to IStream Writer 1.3* on page 20

Stage 1: Check Your System Requirements

Before installing IStream Writer, you need to ensure that your system meets or exceeds the software and hardware requirements.

Step A: Check the Software Requirements

Ensure that you have the required operating systems and other software required by or supported with IStream Writer.

Note: **For the specific version numbers, please see the *IStream Writer Release Notes*.**

Operating Systems

• Microsoft Windows XP or Windows Vista

Software

• **Microsoft Word**

   After installing IStream Writer 1.3, if you upgrade your version of Microsoft Word (that is, from Word 2003 to Word 2007), you will need to reinstall IStream Writer, otherwise it may not function properly.

   IStream Author – required only if users will be transforming IStream Writer specification documents into model documents

• **Livelink Filter Pack** – required on the DMS server only if users will be using IStream Writer from the DMS

• **Microsoft .NET Framework 2.0 –** if this is not already installed, then run `Dotnetfx.exe` from the `\Microsoft .NET Framework 2.0` folder on the IStream Writer installation CD

• **Microsoft .NET Programming Support** is also required. To add this feature, from the Control panel select **> Add or Remove Programs > Microsoft Office > Change > Add or Remove Features.** Under the **Microsoft Word** entry, select to add **.NET Programming Support**.

• **Visual Studio 2005 Tools for Office Second Edition Runtime** (version 8.0.50727.940 or higher) is needed for all environments and Microsoft Word versions. To check the version of your VSTO, navigate to the folder `Program Files\Common Files\Microsoft Shared\VSTO` and display the properties of `VSTOEE.dll`.You can download version 8.0.50727.940 from `http://tinyurl.com/5nrvag`. However, you should check Microsoft's website for the current version.

• **Livelink Explorer** – required only if users will be saving IStream Writer documents to the DMS from Microsoft Word

Step B: Check the Hardware Requirements

Ensure that your hardware meets or exceeds these requirements:

Component	Minimum	Recommended
Processor	Pentium 3	Pentium 4
Memory (RAM)	128 MB	256 MB
Free Disk Space	500 MB	1 GB

Note: The free disk space listed in the table does not include space for temporary files created while adding and working with documents.

Screen Resolution

• The minimum supported screen resolution is 1024 x 768.

• The maximum supported screen resolution is 1280 x 1024.

• Ensure that the screen font size is set to **Normal**.

Stage 2: Install IStream Writer

Note: You need to upgrade Microsoft Word to Word 2003 or above before installing IStream Writer.

1. Close Microsoft Word and any applications that may use Microsoft Word, for example, Microsoft Outlook, Internet Explorer, Customizer and IStream Author.

2. From the **Task Manager**, ensure that no `WinWord.exe` processes are running.

3. From the `Writer Install` folder on the IStream Writer installation CD, run `setup.exe.`(Accept prompts from Vista.)

   The **IStream Writer** install screen opens, followed by the **InstallShield Wizard** screen.

4. Click **Next** to proceed with the installation.

   The **License Agreement** screen opens.

5. Review the license agreement, choose to **Accept** it, then click **Next**.

   The **Choose Destination Location** screen opens.

6. To accept the default installation location for IStream Writer, click **Next**, or click **Browse** to select a different location.

   The **Start Copying Files** screen opens.

7. If you want to review or change any settings, click **Back**, otherwise click **Next** to start copying the files.

   The **Setup Status** screen opens. A progress bar indicates the progress of the installation.

8. After all the files have been copied, the **InstallShield Wizard Complete** screen opens.

9. Click **Finish** to close the installation program.

   IStream Writer is now installed on your system.

Note: If you will be installing IStream Document Manager after installing IStream Writer, first stop the `ITLogServer` service.

Stage 3: Configure Author for IStream Writer

To be able to transform IStream Writer documents into model documents, you need to install Author on the workstation. Complete one of the following procedures depending on which version of Author you are using:

- *Configure Calligo Author 5.4 for IStream Writer* on page 17

- *Configure IStream Author 6.2 for IStream Writer* on page 17

Note: Throughout this chapter, the default installation folders listed are:

- for IStream Author 6.2:

  ```
  C:\Program Files\Whitehill Technologies\IStream
  ```

- for Calligo Author 5.4:

  ```
  C:\Program Files\Insystems Corporation\Calligo
  ```

If you have installed Author to a different folder, please change the path accordingly.

Configure Calligo Author 5.4 for IStream Writer

Important: If you are using Calligo 5.4, certain files need to be updated: please contact *Customer Support*: see *Contacting Skywire Software for Help* on page 11.

Configure IStream Author 6.2 for IStream Writer

1. On the workstation with IStream Writer installed, install IStream Author if it is not already installed.

2. Back up the following file:

   ```
   C:\Program Files\Whitehill Technologies\IStream\
   Components\AuthorM.exe
   ```

3. Copy the following file from the IStream Writer installation CD:

   ```
   \IStream DM 6.2 Updates for Writer\AuthorM.exe
   ```

   to:

   ```
   C:\Program Files\Whitehill Technologies\IStream\
   Components\
   ```

4. Register `AuthorM.exe`:

   a. Click **Start > Run...**

   b. In the **Run** dialog, enter:

      ```
      "C:\Program Files\Whitehill Technologies\IStream\
      Components\AuthorM.exe" /RegServer
      ```

Stage 4: Configure IStream Writer to Access the DMS

Note: You only need to complete these steps if you will use the DMS to store Data Elements Files (DEFs), filing documents or specification documents.

1. In **Microsoft Word**, open an IStream Writer document.

2. Click the **Writer** menu and select **Writer Options…**

3. Click the **DMS Configuration** tab.

4. Select the **Enable DMS Access** checkbox.

   If DMS access is disabled, you will not be able to save filing documents or Data Elements Files to the DMS directly from IStream Writer.

5. Type in the **Server Name**. This is the name of the computer that hosts the DMS.

6. Enter the **Port** number (usually 2099).

7. Enter the **Connection** name:
   - for Microsoft SQL Server, the **Connection** is the database name where the DMS tables are stored
   - for Oracle, the **Connection** is the name of the Oracle user who owns the database tables

Note: If you leave the **Connection** field blank, IStream Writer will use the default connection assigned to the dftConnection variable in the [general] section of the opentext.ini configuration file. This .ini file was created on the DMS server when Livelink was installed.

8. Enter a **User Name** and **Password**. This is the login information that you use to access the DMS.

Note: If you change the DMS **User Name** and/or **Password** information later, you must update the **User Name** and **Password** fields within IStream Writer's **DMS Configuration** settings.

9. Optionally, you can specify a **Location**. The folder that you specify as the **Location** will be the highest level that users can save to in the DMS. If you do not select a **Location**, the Enterprise level will be used by default.

   a. To specify a **Location**, click next to the **Location** field.

   b. If you need to add a new folder, navigate to the area in the DMS for the new folder and click **Add Folder...**Then, enter a name for the folder and click **OK**.

   c. Select the folder that you want to specify as the **Location**, then click **OK**.

10. Click the **Test Connection** button to ensure that you have properly established a connection to DMS. If you receive a message indicating that the connection test was unsuccessful:

    a. Click **OK**. Verify that the information entered in each field is correct.

    b. Click the **Test Connection** button again.

11. Once the connection has been verified, you will receive a message telling you that the connection test was successful. Click **OK**.

12. In the **IStream Writer Options** dialog, click **OK**.

13. A prompt indicates that you need to close Microsoft Word and reopen it to apply your changes. Click **OK**.

14. Save any documents that you have open, and **Exit** Microsoft Word.

15. Reopen Microsoft Word.

    Your changes to the **IStream Writer Options** are applied.

Upgrading to IStream Writer 1.3

This section describes how to upgrade to the current version of IStream Writer.

Step A: Upgrade Microsoft Word

Since IStream Writer 1.3 does not support Microsoft Word 2000, you may need to upgrade your installation of Microsoft Word. To upgrade Microsoft Word, follow the instructions provided by Microsoft.

After installing IStream Writer 1.3, if you upgrade your version of Microsoft Word (that is, from Word 2003 to Word 2007), you will need to reinstall IStream Writer, otherwise it may not function properly.

Step B: Upgrade Calligo 5.4 to IStream Document Manager 6.2

You may optionally upgrade from Calligo 5.4 to IStream Document Manager 6.2. Follow the instructions in the *IStream Document Manager Installation Guide*.

Step C: Back Up Your Current Configuration

If you will be reinstalling IStream Writer after uninstalling it, you will need to back up the following files:

- all `.config` files in each user's `Application Data` folder and in the IStream Writer installation folder

- `logger.icf` in the IStream Writer installation folder

Step D: Modify the Configuration File

If you want to migrate your user configuration or other settings, you need update the `UserPreferences.config` file as described in the following sections.

Location of UserPreferences.config

In IStream Writer 1.2, `UserPreferences.config` is usually located in:

```
C:\Documents and Settings\username\Application
Data\InSystems Corporation\IStream Writer
```

In IStream Writer 1.3, the file is usually located in:

Windows XP

```
C:\Documents and Settings\username\Application Data\Skywire
Software\IStream Writer
```

Windows Vista

```
C:\Users\username\AppData\Roaming\Skywire Software\IStream
Writer
```

Warning: Do not just copy `UserPreferences.config` unmodified from the IStream Writer 1.2 folder into the IStream Writer 1.3 folder, because this will cause

IStream Writer to stop working. You should either edit or transform the configuration file as described in the next two sections.

Editing the Configuration File

If you have made a small number of changes to `userpreferences.config` in Writer 1.2, you can preserve these changes in Writer 1.3 by manually editing the Writer 1.3 userpreferences.config file.

For example, if you have added a custom variable element under `<Variables>` in the `UserPreferences.config` file, then you can copy the text with the custom variable element from the Writer 1.2 `UserPreferences.config` file to directly under the `<UserPreferences>` element in the Writer 1.3 `UserPreferences.config` file.

By default, the `<Variables>` element at the end of the Writer 1.3 `UserPreferences.config` file is commented. Remove the comment tags (so it is no longer commented) and overwrite this element with your custom variables from the Writer 1.2 `UserPreferences.config` file.

Note: In addition to `UserPreferences.config`, IStream Writer 1.3 includes another configuration file, `WriterConfig.xml`. Note that this file does not contain any user data and that therefore you should never edit it.

Transforming the Configuration File

Complete this procedure if you want to preserve *all* the settings in the Writer 1.2 `UserPreferences.config` file. (These include variables and other settings.)

Use an XSLT processor or XML editor that supports XSL transformation to apply `UserPreferences12.xsl` (in the `Writer` folder) to the IStream Writer 1.2 `UserPreferences.config` file. This transforms the IStream Writer 1.2 file to the current IStream Writer 1.3 format.

Tip: You can also configure the DMS settings using the IStream Writer **Options** menu.

Step E: Uninstall IStream Writer 1.2

1. From the **Control Panel**, select **Change/Remove Programs**.

    The **InstallShield Wizard** screen opens.

2. When you are asked if you want to uninstall IStream Writer, click **OK**.

    The **Setup Status** screen opens, displaying the progress of the uninstallation.

3. On the final screen, click **Finish**.

    IStream Writer is uninstalled.

Step F: Install IStream Writer 1.3

Complete the steps in all the stages for a new installation, beginning with *Stage 1: Check Your System Requirements* on page 14.

Chapter 3

Data Elements File Overview

This chapter describes:

- *The Data Elements File* on page 24
- *Extracting the DEF* on page 25
- *Mapping a DEF Configuration* on page 27
- *Using Values and Codes with Filing Information Groupings* on page 28

Note: For a detailed description of the structure of the DEF, see *Data Elements File Structure* on page 29.

The Data Elements File

A Data Elements File (DEF) is a structured XML file that is used to transform IStream Writer documents (also called *specification documents*) into IStream Author model documents.
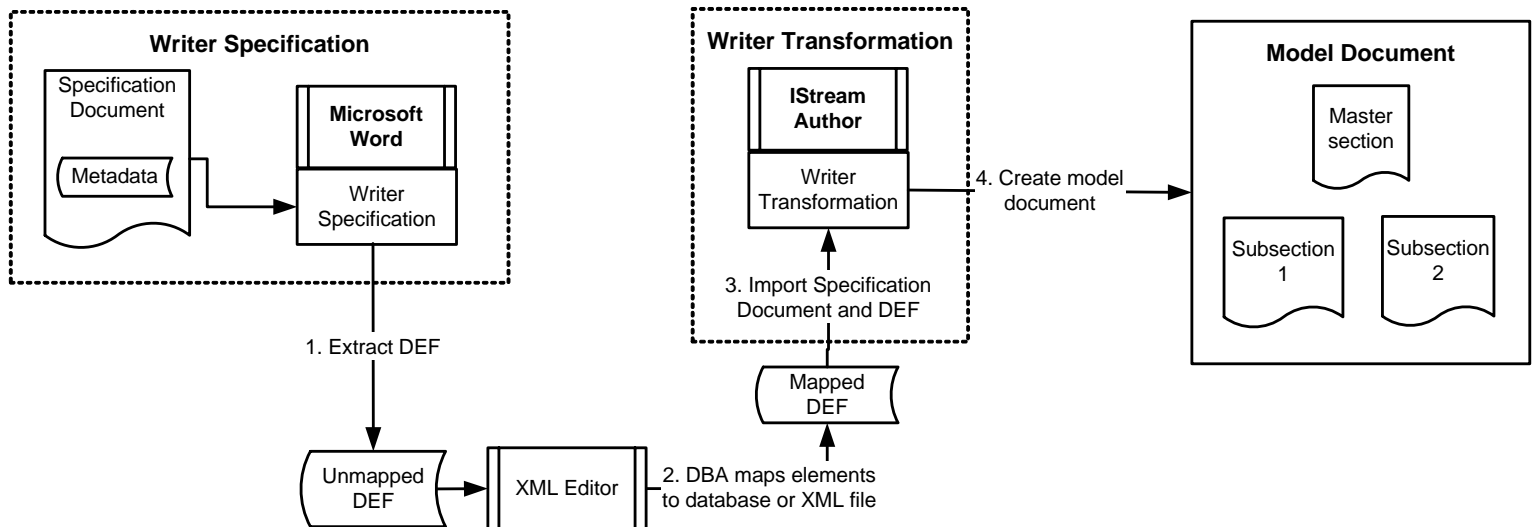
The DEF works with IStream Writer and IStream Author in the following way:

1. A *specification designer* extracts the IStream Writer specification document into an unmapped DEF using the **Extract DEF** function in the IStream Writer Specification module.

2. Using an XML editor, a *database administrator* edits the unmapped DEF in order to map the data elements to a database or XML file.

Note: A DEF can be quite large, depending on the number of unique variable names.

3. When creating a new model document, the author selects the IStream Writer document and the mapped DEF. IStream Author transforms the IStream Writer document using the DEF, to create a model document.

The following diagram illustrates this transformation process:

Extracting the DEF

When the specification designer extracts the specification information into a new DEF, all of the extracted data elements are listed based on the content of the document. After the extraction, the unmapped DEF contains the following sections, in this order:

1.   Three empty `<Query>` tags – all attributes and sub-tags are empty.

2.   A `<TableElement>` tag for each table markup using the table's description and variable elements.

      A `<ListElement>` tag for each list markup using the list's description and variable elements.

3.   An `<Element>` tag for each specified variable element. In each `<Element>` tag, the `SpecName` and `SpecType` attributes contain values from the specification document. `<Element>` tags appear in the DEF in the order given in the specification document.

Note: Do not add comments to the DEF, because they will be removed the next time you extract it.

Important: If you add or change the value for `VarName` attributes in the DEF, they will be overwritten by the automatically generated names the next time you extract and merge the DEF.

Creating a New DEF vs. Merging an Existing DEF

When you a extract a DEF, you can create a new DEF or merge data into an existing DEF. If you select an existing DEF, the data will always be merged.

After merging a DEF, variables that are no longer in a specification document, but which are present in the original DEF, will still be included as `<Elements>` in the merged DEF. These `<Elements>` will be located immediately after the final `<Query>` section, and will be in the same order as the original DEF.

Filing Information Elements

This section describes how filing information elements are transformed to DEF elements.

From the **Main Filing Info** tab in IStream Writer, a user can select five options to apply to text. Based on this selection, IStream Writer elements are transformed to DEF entries in the following ways:

Filing Information Option Selected	Resulting DEF Entry
Apply square brackets only	none
Insert the *drop-down option* based on the current Filing Document	only a DEF entry for the variable is created

Filing Information Option Selected	Resulting DEF Entry
Apply the text according to the options below	creates DEF entries of the grouping value and group code variables in each grouping
DOI Instruction with the **Apply the text according to the options below** option	creates DEF entries of the grouping value and group code variables in each grouping
Only the **DOI Instruction**	none

Note: If the same variable is used several times in IStream Writer, only one DEF entry will be created.

Mapping a DEF Configuration

When a DEF is extracted, it contains the basic information required to start the data mapping for the transformation process.

A DBA needs to complete the following steps to configure the DEF:

1. Determine all data queries that are required to retrieve data for the specified data elements.

2. Complete each `Query` tag for one query. The `Query` tag contains the following attribute tags: `Ref`, `InfoSrc`, `InfoSrcType`, `Ordinal`, `Repeatable` and `SQL`.

    `Ref` is a unique name for the specific query; for descriptions of other tags, see *<Query>* on page 32.

3. Complete each unmapped data element by entering:

    *   `QueryRef` – specify the `Ref` name from a specific query

    *   `Field` – enter a database table field name

Note: Field names with spaces are not supported.

    *   `Type` – the `Field` type: specify a valid type for the field: see *Type* on page 39

    *   `KeyData` – indicates if this data element is a key data variable

    You may need multiple fields to fully specify the variable. For example, you have a variable name of `InsuredName` and the fields `First_Name`, `Initial` and `Last_name` are required to make up `InsuredName`. If so, you will need to specify the `Separator` and `Ordinal` attributes.

Note: You do not need a separator if you just want to concatenate fields. Also, you can only concatenate if the first field (the field with the lowest ordinal) is a string, because only strings can be concatenated.

4. Validate the mapped DEF XML file with any XML parser. To do this, open the DEF in Internet Explorer or any XML viewer, and check it for errors.

Using Values and Codes with Filing Information Groupings

When you transform an IStream Writer document in IStream Author, filing information group items included in the IStream Writer document can appear as either *names* or *codes* in the generated model document. Whether an item appears as a name or code depends upon the setting for its filing information grouping. The setting for each grouping is stored in a `UseFilingInfoGroupName_x` property key in the `Transform.config` IStream Writer configuration file.

State is one example of a possible filing information grouping. Alaska, an item within this grouping, could appear in the model document as **Alaska** (its name) or **AK** (its code). You can modify the `UseFilingInfoGroupName_x` entry for the **State** Filing Information grouping to configure whether the **State** Name or **State** Code is used in transformation.

Note: The appearance of filing information group items in the IStream Writer user interface (as names or codes) does *not* affect how these items will appear in the generated code. If you want to change whether items from a certain filing information group appear as names or codes in the model document, you must modify the appropriate `UseFilingInfoGroupName_x` property key.

For a detailed example, and for more information on editing the `UseFilingGroupName_x` entries of `Transform.config` see: *Use Filing Info Group Name Keys* on page 49.

Chapter 4

Data Elements File Structure

This chapter describes:

- *The Data Elements File Structure* on page 30
- *<DEF>* on page 31
- *<Query>* on page 32
- *<TableElement>* on page 36
- *<ListElement>* on page 37
- *<Element>* on page 38

The Data Elements File Structure

The following table lists the tags, attributes, and subtags in the data elements file. Each tag contains attributes, and can also contain subtags.

Tip: You can click any of the items in this table to quickly go to its detailed description.

Tag	Attributes	Subtags	Subtag Attributes
`<DEF>`	• `Description`	*(none)*	*(none)*
`<Query>`	• `Ref` • `InfoSrc` • `InfoSrcType` • `Ordinal` • `Repeatable`	`<SQL>`	*(none)*
`<TableElement>`	• `Description`	`<Element>`	*(none)*
`<ListElement>`	• `Description`	`<Element>`	*(none)*
`<Element>`	• `SpecName` • `SpecType` • `QueryRef` • `VarName` • `Keydata`	`<Field>`	• `Type` • `Separator` • `Ordinal`

<DEF>

The `<DEF>` tag defines the entire data elements file. There is only one set of `<DEF>` tags in a DEF.

Within the `<DEF>`...`</DEF>` tags, there will be multiple `<Query>` tags, `<Element>` tags, `<ListElement>` tags and `<TableElement>` tags.

Attribute

Description

The `<DEF>` tag contains one attribute: `Description`.

The `Description` attribute contains a description of the DEF.

Note: The `Description` attribute is an informational attribute only and is not processed during the transformation.

Syntax

```
<DEF Description="..."> ...
</DEF>
```

\<Query\>

You use the `<Query>` tag to create a proper `Query/Next` statement and contains a reference name. The reference name is identified in each `<Element>` tag so that the fields for the `<Element>` will be read from the appropriate query.

There will usually be multiple `<Query>` tags within a `<DEF>` tag. By default, there are three sets of `<Query>` tags. To add more queries, add additional `<Query>` tags.

To relate a `<Query>` to an `<Element>` tag, the `QueryRef` attribute is used to match to the `Ref` attribute in a `<Query>` tag.

Attributes

The `<Query>` tag contains the following attributes:

Ref

The `Ref` attribute is the name referred to in an `<Element>` tag to associate fields to the correct data query. Any string is a valid value.

InfoSrc

The `InfoSrc` attribute is the InfoSource name. You use it to form a valid `Query` statement.

InfoSrcType

The `InfoSrcType` attribute is the InfoSource type. Valid values are `XML` or `ODBC`. It is not case-sensitive.

Ordinal

The optional `Ordinal` attribute determines the order of the `Query` statement within the `CONTROL.CDS` section. The lower the value, the earlier the position of Query statement within the section. If omitted, the order will depend on the XML parser. The ordinal must be a whole number.

Repeatable

The optional `Repeatable` attribute controls the result set from the query. The valid values are 1 or 0, as follows:

- 1 – all of the records for a query will be processed; the query will be translated into a `Query/Next` statement without a `Break` statement

- 0 – only the first record for a query will be processed; the query will be translated into a `Query/Next` statement with a `Break` statement

   If omitted, 0 is the assumed value.

Notes

If a Table or List element is defined, you must set Repeatable to 1 in the Query definition that the table or list element refers to. Otherwise, a warning will appear during the transformation. For more information, see *Example: Table or List Query* on page 34.

A repeatable query will not appear in the CONTROL.CDS file. Instead it will be placed around a table or list element inside an actual section where the table or list occurs. Alternatively, the repeatable query can be used to construct a variable that defines an inline list and would be placed at the beginning of the section that uses that list.

Subtag

<SQL>

The <SQL> subtag defines a SQL query statement. The query statement will be translated into a valid Query statement.

The SQL Query statement must be a valid SQL SELECT statement. If the SQL SELECT statement is to be formed during execution with the variable elements, then the variable names must be embedded as "${VarName}", where VarName is the value from the VarName attribute, for example nPolicy_ID.

Example: SELECT * from Insured WHERE Insured.pol_num=${nPolicyNo}

SELECT * from Driver WHERE
driver.driver_name=${nCustomer_Name}

SELECT * from Driver WHERE
driver.driver_num=${nDriver_Number}

Important: A variable must be defined as an element when using the variable name in a SQL statement.

Example Queries

This section contain various `<Query>` examples.

Example: Query with an Optional Variable

The following `Query` example shows how to use an optional variable in a SQL statement. Note that the syntax is `${VarName}` and not `SpecName`.

```
<Query Ref="CustomerRef" InfoSrc="Customer Data" InfoSrcType="ODBC"
Ordinal="1" Repeatable="0">

  <SQL>SELECT * FROM CustomerInfo WHERE PolicyID= ${nPolicy_ID}</SQL>

</Query>
```

Example: Table or List Query

The following `Query` is for a table or list of information, therefore, the `Repeatable` value is set to 1.

```
<Query Ref="DependentsRef" InfoSrc="Customer Data" InfoSrcType="ODBC"
Ordinal="1" Repeatable="1">

  <SQL>SELECT * FROM Dependents WHERE PolicyID= ${nPolicy_ID}</SQL>

</Query>
```

Example: Query with an XML InfoSource

The following `Query` example shows how the InfoSource type can be XML:

```
<Query Ref="AutomobileRef" InfoSrc="Automobile Data" InfoSrcType="XML"
 Ordinal="1" Repeatable="0">

  <SQL>SELECT * FROM Automobile WHERE PolicyID= ${nPolicy_ID}</SQL>

</Query>
```

Example: Controlling the Order of Query Statements

The following query example shows how to control the order of the `Query` statements by setting the `Ordinal` value to be higher than another query.

Note how `cCompany_Name` is used in the SQL statement, but this variable is not available until the `CustomerRef` query statement (shown previously) is executed. By changing the `Ordinal` value to `2`, the `Query` statement will appear after the `Ordinal=1` queries.

```
<Query Ref="CompanyRef" InfoSrc="Customer Data" InfoSrcType="ODBC"
 Ordinal="2" Repeatable="0">

<SQL>SELECT * FROM CompanyInfo WHERE Company_Name= ${cCompany_Name}
</SQL>

</Query>
```

<TableElement>

The <TableElement> tag contains the description of the table defined in the IStream Writer document. There will usually be multiple <TableElement> tags in a DEF.

Note: The <TableElement> tag is an informational tag only and is not processed during the transformation.

Attribute

Description

The <TableElement> tag contains one optional attribute: Description.

The Description attribute contains the description of the table from the IStream Writer document.

Subtag

<Element>

Each <TableElement> contains at least one <Element> subtag.

The <Element> subtag stores a variable element name from within the table. There will an <Element> subtag for each variable used in the table.

Syntax

```
<DEF>
    <TableElement Description=…>…
        <Element>
        </Element>
    </TableElement>
</DEF>
```

Note: For a detailed <TableElement> example, see *Examples: Table and List Elements* on page 37.

<ListElement>

The `<ListElement>` tag defines a list. There will usually be multiple `<ListElement>` tags within a `<DEF>` tag.

Note: The is an informational tag only and is not processed during the transformation.

Attribute

Description

The optional `Description` attribute contains the description of the list from the IStream Writer document.

Subtag

<Element>

The `<Element>` subtag stores a variable element name from the list. There will an `<Element>` tag for each variable used in the list.

Examples: Table and List Elements

The `TableElement` and `ListElement` information below indicates to the DBA which elements are part of the repeatable queries previously listed. This section does not need to be modified. It is for informational purposes only.

```
<TableElement Description="Table of Dependents">
  <Element SpecName="Dependent Name" />
  <Element SpecName="Dependent Age" />
</TableElement>

<ListElement Description="Inline list of dependents">
  <Element SpecName="Dependent Name" />
</ListElement>

<ListElement Description="Inline list of dependents">
  <Element SpecName="Dependent Name" />
  <Element SpecName="Dependent Age" />
</ListElement>

<ListElement Description="Multi line list of coverage">
  <Element SpecName="Coverage" />
  <Element SpecName="CoverageAmt" />
</ListElement>
```

<Element>

The `<Element>` tag corresponds to a variable in the IStream Writer document. Elements are transformed into variables in a model document.

There will be multiple `<Element>` tags within the DEF tag.

Attributes

SpecName

The `SpecName` attribute defines the variable name used in the IStream Writer document.

Important: The `SpecName` attribute must be unique across the entire DEF.

SpecType

The `SpecType` attribute is the data type of the variable. Valid values are:

- Text
- Numeric
- Date
- Boolean

Important: These values are case-sensitive. Enter them exactly as they appear here.

If the `SpecType` is `Boolean`, then the `Type` attribute in the `<Field>` tag must not be `Boolean`, because not all databases support boolean types. For these variables, use a `<Field>` tag `Type` attribute of `Numeric`, where `0` indicates `false`, and `1` indicates `true`.

QueryRef

The `QueryRef` attribute indicates which `<Query>` definition is referenced. It can be any name defined in the `Ref` attribute of a `<Query>` tag.

VarName

The `VarName` attribute is a variable name created automatically and used in the transformation.

Important: You should edit this name only if there are duplicate `VarNames` in the DEF. If you edit this name, you must follow the same naming restrictions that apply for model document variable names.

Note: Each `VarName` attribute must be unique throughout the DEF.

Keydata

The optional Keydata attribute indicates whether the data element is key data. Valid values are 1 or 0 as follows:

• 1 – the data element is key data:

  the element's VarName will be added as key data to the model document created during the transformation

  the transformation program will ignore the <Field> tags within the <Element> tag.

• 0 – the data element is *not* key data.

If the value is omitted, then 0 is the assumed value.

Subtag

<Field>

The <Field> subtag defines the field name in the database table. You use it to create a valid DEFINE statement so that the element is mapped to the data field in a query. Multiple <Field> tags may be required if a number of fields need to be concatenated to properly create the variable setup in IStream Writer.

Note: Database field names with spaces are not supported.

<Field> Attributes

The <Field> subtag contains the following attributes:

Type

The Type attribute is the data type of the database or XML data field. You used it to create a valid DEFINE statement. If the Type is different from the SpecType, a function will be applied so that the variable will be the type specified by the SpecType value.

Valid values are Text, Numeric and Date. Note that these values are case-sensitive.

A DBA needs to map the database or XML file field types to these general types.

If the type in the <Field> tag differs from that set in the SpecType attribute, a function will be applied as follows:

<ELEMENT> Tag SpecType Attribute	<FIELD> Tag Type Attribute	Function Applied During Transformation
Text	Date	STR(*field name*)
Text	Numeric	STR(*field name*)
Numeric	Text	VAL(*field name*)

<ELEMENT> Tag SpecType Attribute	<FIELD> Tag Type Attribute	Function Applied During Transformation
Date	Text	CTOD(*field name*)
Boolean	Text	!EMPTY(*field name*)
Boolean	Date	!EMPTY(*field name*)
Boolean	Numeric	!EMPTY(*field name*)

Separator

The optional `Separator` attribute is only required for multiple `<Field>` tags. It defines a proceeding string as separator. If omitted, it indicates there is no separator. The separator must be a string.

Ordinal

The `Ordinal` attribute is only required for multiple `<Field>` tags, and defines the `<Field>` tags' order. If omitted, the order depends on the XML parser. This value is ignored if there is only a single `<Field>` tag element. Valid values are 1 to 100.

Examples of Elements and Fields

This section contains various examples of `<Element>` and `<Field>` definitions.

Note that in these examples:

- `SpecName` is the variable name used in the Spec Doc.

- `SpecType` is the `Variable` type that was set in the specification document and should not be changed in the DEF; possible values are `Text`, `Numeric`, `Date`, or `Boolean`

- `QueryRef` is set to one of the `Query` statements defined above

- `VarName` is automatically generated and should not be changed

- key data is used to automatically create a Key Data variable in the model document, meaning that the information will be determined at generation time

Example: Element Definition

```
<Element SpecName="Company Name" SpecType="Text" QueryRef="CustomerRef"
 VarName="cCompany_Name" Keydata="0">

<Field Type="Text" Separator="" Ordinal="1">Company_Name</Field>

</Element>


<Element SpecName="Years Insured" SpecType="Numeric"
 QueryRef="CustomerRef" VarName="nYears_Insured" Keydata="0">

  <Field Type="Numeric" Separator="" Ordinal="1">Num_Insured_Years
  </Field>

</Element>


<Element SpecName="More Than 100 Employees Yes/No" SpecType="Boolean"
 QueryRef="CustomerRef" VarName="bMore_Than_100_Employees_Yes_No"
 Keydata="0">

  <Field Type="Numeric" Separator="" Ordinal="1">Over100</Field>

</Element>


<Element SpecName="Insured Since" SpecType="Date" QueryRef="CustomerRef"
 VarName="dInsured_Since" Keydata="0">

  <Field Type="Date" Separator="" Ordinal="1">StartDate</Field>

</Element>
```

Example: Unique SpecType and Field Type Elements

The following element shows how the `SpecType` and `Field Type` do not need to be the same. IStream Writer will automatically create the IStream Author code needed to transform the data into the correct type.

```
<Element SpecName="License ID" SpecType="Text" QueryRef="CustomerRef"
 VarName="cLicense_ID" Keydata="0">

  <Field Type="Numeric" Separator="" Ordinal="1">LicenseID</Field>

</Element>
```

Example: Element Used as Key Data Value

The following element has a key data value of 1, meaning that the end user will be prompted for the data before the document is generated. `QueryRef` and all `Field` data therefore do not apply.

```
<Element SpecName="Policy ID" SpecType="Numeric" QueryRef=""
 VarName="nPolicy_ID" Keydata="1">

  <Field Type="Numeric"
   Separator=""Ordinal="1">database_table_column_name_here</Field>

</Element>
```

Example: Variables with Multiple Fields

The following example illustrates how one variable in the specification document can be made up of several fields in the database or XML file. A `Separator` is the text that you want to put *before* the field.

For example, the following definition will produce this formatting:

    Smith, John Q

If the `Separator` is left empty, the result would be:

    SmithJohnQ

The `Ordinal` values control the order of the values.

```
<Element SpecName="Customer Full Name" SpecType="Text"
 QueryRef="CustomerRef" VarName="cCustomer_Full_Name" Keydata="0">

  <Field Type="Text" Separator=", " Ordinal="2">CustomerFN</Field>
  <Field Type="Text" Separator="" Ordinal="1">CustomerLN</Field>
  <Field Type="Text" Separator=" " Ordinal="3">CustomerInitial</Field>

</Element>
```

<**ELEMENT**>

Example: State, Product and LOB Codes and Values

State, product and LOB codes and values are automatically added for filing information elements when these variables are used in the specification document or in rules.

Important: The code generated for filing information elements will use the **State Value** field by default. For example, if a piece of text has a filing information element to show only if the state is Alabama, the resulting code in IStream Author will start with:

```
"IF cState_Value == "Alabama""
```

even if you only use state codes in IStream Writer.

If you want your IStream Author code to appear as follows (with the state code instead of the state value):

```
"IF cState_Value == "AL""
```

then you need to modify the appropriate `UseFilingInfoGroupName_x` property key in the `transform.config` file. For more information, see *Use Filing Info Group Name Keys* on page 49.

State, LOB, product codes and values always appear as pairs in the DEF. For example, if only a state value is used, then `State Code` will also appear in the DEF.

```
<Element SpecName="State Code" SpecType="Text" QueryRef="GroupDataRef"
 VarName="cState_Code" Keydata="0">

  <Field Type="Text" Separator="" Ordinal="1">StateCode</Field>

</Element>


<Element SpecName="State Value" SpecType="Text" QueryRef="GroupDataRef"
 VarName="cState_Value" Keydata="0">

  <Field Type="Text" Separator="" Ordinal="1">StateValue</Field>

</Element>
```

Chapter 5

Configuring IStream Writer

This chapter describes:

- *IStream Writer Configuration Files* on page 46
- *Template Files* on page 53
- *Working with IStream Writer Log Files* on page 54

IStream Writer Configuration Files

IStream Writer uses several configuration (`.config`) files. These files contain various settings and functions that affect the setup and use of IStream Writer.

Warning: **Before editing any configuration file, please note the following important information:**

- Please contact customer support to assess your specific configuration requirements and to instruct you on implementing the specific changes you require. For contact information, see *Contacting Skywire Software for Help* on page 11.

- Always back up a `.config` file *before* editing it.

- Only edit the `.config` files described in this chapter. Do *not* edit any other files.

- For the `.config` files that are described in this chapter, only edit the parameters as indicated. Do *not* edit any other parameters.

- All `.config` files are structured XML files. Do not change the structure of these files or move or delete any elements. It is critical that all elements remain in their *original* nested order.

Important: **Failure to follow these guidelines can cause IStream Writer to stop working.**

UserPreferences.config

This file contains various settings for each user that control the operation and formats for various rules and variables. Most of the settings affect the IStream Writer user interface, while others affect the IStream Writer transformation process.

The settings in `UserPreferences.config` include:

- settings for certain IStream Writer options including the **Highlight Elements** option, **Show Bookmark** option, and **DMS Configuration** settings

- the default IStream Writer list element separators available

- certain system-wide variables

Important: Some of these settings can be set using the IStream Writer interface. These settings should not be changed in `UserPreferences.config` but in the user interface instead.

Location of UserPreferences.config

A separate `UserPreferences.config` file is created for each IStream Writer user.

The location of this file depends on which version of Microsoft Windows you are using, as follows:

Windows XP:

`C:\Documents and Settings\`*user_name*`\Application Data\Skywire Software\IStream Writer\`

Note: The `Application Data` folder is a hidden folder.

Windows Vista:

`C:\Users\`*username*`\AppData\Roaming\Skywire Software\IStream Writer\`

If this file is missing, it will be copied from the IStream Writer installation folder with the default values when IStream Writer is launched. By default, the IStream Writer installation folder is located at:

`C:\Program Files\Skywire Software\IStream Writer`

Important: The `UserPreferences.config file` that is used when transforming IStream Writer documents into model documents must match the one used when creating the specification document.

Sections in UserPreferences.config

`UserPreferences.config` contains the following sections and settings:

\<Properties\>

This section contains various IStream Writer settings.

Confirmation Properties

The confirmation properties are:

- `PasteConfirmation` – if `true`, a warning message will appear after a user tries to paste data over existing data

- `RemovalConfirmation` – if `true`, a warning message will appear after a user tries to delete data

By default, the value for these properties is set to `true`. However, the user may turn off this setting if they select **"Don't show this warning again"** when they are prompted with the warning message. If the user selects this option, the property setting changes to `false`. To enable the warning message to be displayed again, you can change this setting back to `true`.

These properties appear as:

```
<Property key="PasteConfirmation" value="true" />
<Property key="RemovalConfirmation" value="true" />
```

Validation Property

This property controls whether IStream Writer documents will be validated each time they are opened. You should leave the `value` set to `true`.

This property appears as:

```
<Property key="ValidationOnOpenDocument" value="true" />
```

<GroupDataArray>

This section contains all the grouping names and elements.

Warning: Do not edit these settings in `UserPreferences.config`. The settings should be updated using the IStream Writer user interface only.

<Variables>

You can add variable elements to this section. These elements apply to all specification documents.

The syntax for a variable is:

```
<Variables>
    <Variable name="variable name" type="variable type" />
</Variables>
```

The variable type must be one of:

- Text
- Numeric
- Date
- Boolean

Important: These values are case-sensitive. Enter them exactly as they appear here.

By default, the `<Variables>` section is empty.

When adding new variables, please note that:

- there should only be one pair of `<Variables>` tags in `UserPreferences.config`
- variable names cannot contain any leading or trailing spaces
- variable names should be a maximum of 30 characters

<element_separators> and <final_separators>

These two sections apply to inline lists.

`<element_separators>` contains the `<string>` values that are used by IStream Writer to separate elements in a list.

`<final_separators>` contains the string values that are used by IStream Writer *before* the final element in a list.

You can add `<string>` values to these sections as required.

Updating UserPreferences.config for Multiple Users

You may want to standardize the configuration settings for all IStream Writer users by applying the same `UserPreferences.config` file for each user. Ideally, this should be done before any users have started using IStream Writer, but it can be done at any time.

Method: Update or apply the same IStream Writer user preferences to all users

---

1. Ensure Microsoft Word and any application that uses Microsoft Word is closed.

2. In the IStream Writer installation folder, back up `UserPreferences.config`.

3. Edit `UserPreferences.config` as necessary.

4. If any users have already used IStream Writer, back up their `UserPreferences.config` from their respective `Application Data` folders. For details, see *Location of UserPreferences.config* on page 47.

5. For each workstation running IStream Writer, copy the `UserPreferences.config` file that you edited in step 1 to the users' `Application Data` folders, replacing the existing versions.

Note: When replacing a user's `UserPreferences.config` file, settings such as **Highlight Element** will be reset. Contents of the **User Name** and **Password** fields under the **DMS Configuration** tab will also be reset and must be reentered by the user. You should therefore advise all users to review their IStream Writer configuration settings.

Transform.config

This file contains settings that control the process of transforming IStream Writer documents to model documents.

You can change the following settings in this file:

DefaultInfosourceType

Description: When IStream Writer creates a DEF, the `InfoSrcType` value will assume the value of this key.

Possible values: `ODBC` (default) or `XML`

Use Filing Info Group Name Keys

Description: The "Use Filing Info Group Name" keys are:

- `UseFilingInfoGroupName_0`
- `UseFilingInfoGroupName_1`
- `UseFilingInfoGroupName_2`

Each one of these keys corresponds to one of the three filing information groups, and controls whether names or codes appear in the generated code when a specification document is transformed into a model document. `_0`, `_1`, and `_2` correspond to the filing information groups in the order that they appear in the `<GroupDataArray>` section of `UserPreferences.config`.

Each key appears as:

```
<Property key="UseFilingInfoGroupName_x" value="yes" />
```

When the property value is `yes`, a name as defined in `UserPreferences.config` (`cX_Name = <Name>`) will be used in the generated code.

When the property value is `no`, a code as defined in `UserPreferences.config` (`cX_Code = <Code>`) will be used in the generated code.

The appearance of filing information group items in the IStream Writer user interface (as names or codes) does *not* affect how these items will appear in the generated code. If you want to change whether items from a certain filing information group appear as Names or Codes in the model document, you must change the appropriate `UseFilingInfoGroupName_x` property key.

Example: In this example, `UseFilingInfoGroupName_0` represents the state filing information grouping. With `<Property key="UseFilingInfoGroupName_0" value="yes" />`, an author selects the words `FilingInfo` in the document, and creates an IStream Writer element using the following settings:

The IStream Writer Transformation program will produce the following code based on these settings:

```
TEXT
<IF( cState_Value = "Alaska", "Filing info","")>
ENDTEXT
```

If items from the **State** filing information group appear as *codes* in the user interface (see the following image) instead of *names*, the setting `<Property key="UseFilingInfoGroupName_0" value="yes" />` would still produce the same code.

If you want the transformation to produce code based on state grouping codes instead of state grouping names, you must change the `UseFilingInfoGroupName_0` property key so that it appears as:

```
<Property key="UseFilingInfoGroupName_0" value="no" />
```

This change will produce the following code when you transform your IStream Writer specification document:

```
TEXT
<IF(cState_Code = "AK", "Filing info","")>
ENDTEXT
```

Notice that the State code is used instead of the State Name once you have modified the property key.

Possible values: `yes` (default) or `no`

Template Files

IStream Writer uses the following Microsoft Word template files:

- Empty.dot

- Empty.dotm

Empty.dot

`Empty.dot` is a Microsoft Word template file that is required by IStream Writer.

Warning: Do not edit or change this file, because this can cause IStream Writer to stop working.

Location of Empty.dot

The location of this file depends on which version of Microsoft Windows you are using, as follows:

Windows XP:

```
C:\Documents and Settings\user_name\Application Data\Skywire
Software\IStream Writer\
```

Note: The `Application Data` folder is a hidden folder.

Windows Vista:

```
C:\Users\username\AppData\Roaming\Skywire Software\IStream
Writer\
```

If this file is missing, it will be copied from the IStream Writer installation folder with the default values when IStream Writer is launched. By default, the IStream Writer installation folder is located at:

```
C:\Program Files\Skywire Software\IStream Writer
```

See also *UserPreferences.config* on page 46.

Working with IStream Writer Log Files

IStream Writer uses the IT Log application to create logs.

Backing Up the Current Log Files

If you experience a problem when using IStream Writer, you will need to back up your current log files before they are overwritten. Customer support requires the most current log file if you are experiencing transformation errors.

Method: Back up the current IStream Writer log files

1. Locate the `Logs` folder; this folder contains your log files. It may be under a hidden folder, so ensure that you have configured Windows Explorer to show all hidden files and folders.

    The location of the `Logs` folder depends on whether Calligo or IStream Document Manager is installed:

    • If Calligo or IStream Document Manager is installed, then the `Logs` folder is located in under the Calligo or IStream Document Manager installation folder.

    • If Calligo or IStream Document Manager is *not* installed, then the `Logs` folder is located in under the IStream Writer installation folder.

2. Under the `Logs` folder, locate the current log folder. Folders are named by year and date in the following format: `LOG_YYYY_MM_DD`, for example, `LOG_2008_11_18`.

3. Within the above folder, locate the current log folder. Folders are named by year, date and time in the following format: `RUN_YYYYMMDDHHMM`, for example: `RUN20081118140439`.

4. Within the above folder, locate the most current log file in this folder. Files are named by year, date and time in the following format: `YYYYMMDDHHMMSS.ILF`, for example: `20081118140857.ILF`.

5. Copy this `.ILF` file to another location, and have it ready to send to customer support.

Appendix A

Troubleshooting IStream Writer

This chapter describes:

- *IStream Writer Applications* on page 56
- *IStream Writer Implementation* on page 57
- *Troubleshooting Specific Problems* on page 59

IStream Writer Applications

This chapter describes the design of IStream Writer and includes various troubleshooting procedures.

Troubleshooting IStream Writer is a complex process, because it works with other applications including:

- Microsoft Windows

- Microsoft Word

- Microsoft Visual Studio Tools for Office (VSTO)

- IStream Document Manager

Each of these applications can be customized depending on your requirements. Also, only certain versions are supported with each version of IStream Writer. Because of this, the information given in this chapter is at a general level only. You may need to modify the steps depending on how you have configured these applications, the versions you are using, and the particular problem you have are having. However, these procedures should still help you in most cases.

IStream Writer Implementation

IStream Writer works with Microsoft VSTO. An IStream Writer document (also called an IStream Writer *customized document*) is a Microsoft Word document with IStream Writer attached to it. This document is linked to the `WriterTemplate.dll` .NET assembly file, which acts an entry point to IStream Writer.

When a user issues the **Attach Writer** command through the user interface, the system runs a macro in the `Writer.dot` template and creates a link from the document to IStream Writer. The **Detach Writer** command removes the link from the document.

The **Attach Writer** and **Detach Writer** commands are also available as macros. Therefore, if the IStream Writer user interface is not working, you can run these commands as macros instead.

Verifying the IStream Writer Link

You can quickly verify if a Microsoft Word document contains the link to `WriterTemplate.dll` using Microsoft Word or Microsoft Windows Explorer.

Method: Verifying the IStream Writer link using Microsoft Word

1. In Microsoft Word, display the properties of the Word document:

    •   In **Word 2003**: Click **File** > **Properties**, then click the **Custom** tab.

    •   In **Word 2007**:

    a. Click the Microsoft Office Button > **Prepare** > **Properties**.

    b. Click **Document Properties** > **Advanced Properties**.

    c. Click the **Custom** tab.

2. If the document is linked to IStream Writer, the following **Properties** will appear:

    •   `_AssemblyName`

    •   `_AssemblyLocation`

    •   `IStream_Writer`

    •   `Solution ID`

Method: Verifying the IStream Writer link using Windows Explorer

1. In Windows Explorer, right-click the file and select **Properties**, then click the **Custom** tab.

2. If the document is linked to IStream Writer, the following **Properties** will appear:

    *   IStream_Writer
    *   Solution ID

The link to IStream Writer contains a path to the IStream Writer folder, but it is not an absolute path. Instead the path contains an environmental variable named WRITER_HOME. This variable is resolved dynamically when the document is opened in Microsoft Word.

Tip: You can view the information associated with an IStream Writer document using the VSTO Application Manifest Editor, which you can download from Microsoft.

IStream Writer Ribbon

In Microsoft Word 2007, the interface was completely redesigned from the previous versions. Menus and toolbars were replaced with ribbons – these ribbons were implemented as Microsoft Word add-ins.

Loading the IStream Writer Add-In

Add-ins usually load when Microsoft Word starts and remain loaded until Microsoft Word is closed. However, the user may not always require an add-in, therefore loading the add-ins can be inefficient.

Ideally, the IStream Writer ribbon would appear only with IStream Writer documents. Unfortunately, the Microsoft Word ribbon cannot be selectively loaded for individual documents. Therefore, like all add-ins, the IStream Writer add-in is loaded *every* time Microsoft Word starts. If there is no document open, or if the opened document is not a IStream Writer document, a single **Attach Writer** button appears.

Note that the IStream Writer ribbon add-in is simply a "thin layer" with very limited functionality that forwards most of its requests to IStream Writer when an IStream Writer document is open. The **Attach Writer** macro from the Writer.dot global template is the only other object that receives IStream Writer ribbon add-in commands. This macro is invoked when a user clicks **Attach Writer** on the IStream Writer ribbon.

Troubleshooting Specific Problems

This section contains troubleshooting procedures for the following problems:

- *Writer Menu or Ribbon Missing from Word* on page 59

- *IStream Writer Document Fails to Open – General Troubleshooting* on page 63

- *IStream Writer Document Opens with Warnings* on page 66

- *IStream Writer Functionality is Unavailable* on page 67

- *Only 'Attach Writer' Button Appears in Writer Ribbon* on page 67

- *System Does Not Select DEF Checkbox in IStream Author* on page 67

- *Transformation Failure* on page 69

Writer Menu or Ribbon Missing from Word

Problem

The **Writer** menu does not appear in Microsoft Word 2003.

- or-

The **Writer** ribbon does not appear in Microsoft Word 2007.

Solution

There can be many different causes for this problem. Therefore, you need to complete each of the following steps *in the order they are listed here.*

Note: **Step A** below applies to Microsoft Word 2003 only. If you are using Microsoft Word 2007, skip to *Step B: Exit and Reopen Microsoft Word* on page 60.

Step A: Verify IStream Writer Template

Word 2003 only

The IStream Writer file template file (`WriterAttach.dot`) adds the **Writer** menu to Microsoft Word 2003.

If the **Writer** menu does not appear in Microsoft Word 2003, you need to verify that this `.dot` file is in the correct location and that it has loaded:

1. Ensure that the IStream Writer file template file is at the following location:

Microsoft Office folder`\OFFICE11\STARTUP\`
WriterAttach.dot

By default, the Microsoft Office folder is:

`C:\Program Files\Microsoft Office`

2. In Microsoft Word 2003, click **Tools** > **Templates and Add-Ins**.

3. In the **Global templates and add-ins** section, ensure that `Writer.dot` and `WriterAttach.dot` are listed and checked.

    If the **Writer** menu still does not appear in Microsoft Word, continue to the next step.

Step B: Exit and Reopen Microsoft Word

1. Completely exit Microsoft Word,

2. From the **Windows Task Manager**, ensure that `Winword.exe` process is not running. If the process does not end after one minute, end the process manually.

3. Restart Microsoft Word.

4. If the problem persists, continue to the next step.

Step C: Check Office Security Settings

Word 2003

1. In Microsoft Word 2003, click **Tools > Macro > Security**.

2. Ensure that the **Security Level** is set to **High** (the default setting) or lower.

3. If the **Security Level** is set to **High**, click the **Trusted Publishers** tab and ensure that **Trust all installed add-ins and templates** is selected.

4. Open Microsoft Word again. If the problem persists, continue to *Step D: Check IStream Writer Registry Setting* on page 61.

Word 2007

1. Click the Microsoft Office Button [icon], **Word Options** > **Trust Center** > **Trust Center Settings** button.

2. Click the **Trusted Publishers** tab.

3. Ensure **Skywire Software Canada** is listed with a certificate issued by **Thawte Code Signing CA**.

4. Click the **Add-Ins** tab.

5. Ensure **Disable all Application Add-ins** is *not* **selected**. This is the default setting.

6. Click the **Macro Settings** tab.

7. In the **For macros in documents not in a trusted location** option, ensure that **Disable all macros with notification** is selected. This is the default setting.

8. Open Microsoft Word again. If the problem persists, continue to the next step.

Step D: Check IStream Writer Registry Setting

Word 2007 only

1. In the Windows Registry, navigate to the following registry key:

   `HKEY_CURRENT_USER\SOFTWARE\Microsoft\Office\Word\Addins`
   `\WriterRibbonAddIn`

2. Ensure that the value for `LoadBehavior` is 3, as follows:

   `"LoadBehavior"=dword:00000003`

3. Open Microsoft Word again. If the problem persists, and you are using Microsoft Word 2007, continue to the next step, otherwise go to *Step F: Check IStream Writer Directory Access* on page 61.

Step E: Check for Disabled Add-ins

Word 2007 only

1. In Microsoft Word 2007, click the Microsoft Office Button , then click **Word Options**.

2. Click the **Add-Ins** tab.

3. In the **Manage** drop-down list, select **Disabled Items**, then click **Go...**.

4. If the `WriterRibbonAddin` is listed in the disabled items, click to select it, then click **Enable** to re-enable it.

5. Close Microsoft Word.

6. Restart your computer.

7. Open Microsoft Word again. If the problem persists, continue to the next step.

Step F: Check IStream Writer Directory Access

• Ensure that the user has read access to the IStream Writer application directory. If the problem persists, continue to the next step.

Step G: Check the Word Startup Switch

Ensure that Microsoft Word has not been launched using the `/a` startup switch, because this prevents IStream Writer from loading. If you have been using this

switch, remove it, then open Microsoft Word again. If the problem persists, continue to the next step.

Step H: Back Up and Restore IStream Writer AppData Folder

1. Back up the user's IStream Writer `Application Data` folder, usually located at:

`C:\Documents and Settings\`*user_name*`\Application Data\Skywire Software\IStream Writer\`

2. Delete this folder.

3. If the problem persists, continue to the next step.

Step I: Uninstall and Reinstall IStream Writer

1. Uninstall then reinstall IStream Writer: for detailed instructions, see:

- *Installing IStream Writer* on page 13
- *Upgrade Microsoft Word* on page 20

2. If the problem persists, continue to the next and final step.

Step J: Back Up Log File and Contact Customer Support

By default, IStream Writer logs only warnings and errors. To help troubleshoot this problem, change the logging level so that *all* messages are logged.

1. In the Windows Registry, navigate to the following registry key:

`HKEY_LOCAL_MACHINE\SOFTWARE\Skywire Software\IStream Writer\Log`

2. Change the `Logging Level` value to `0`. (To restore the default logging level, change it back to 2.)

3. Open Microsoft Word in order to attempt to load IStream Writer.

4. Immediately back up your last current log file. For details, see *Backing Up the Current Log Files* on page 54.

5. Contact *Customer Support*: see *Contacting Skywire Software for Help* on page 11.

IStream Writer Document Fails to Open – General Troubleshooting

Problem

An IStream Writer document fails to open.

Solution

Complete each of the following steps in this order.

Step A: Check System Prerequisites

For an IStream Writer document to open successfully, Microsoft Word needs to call the VSTO loader. This requires that certain applications be installed and configured. Complete the following steps to ensure your system is set up correctly:

1. Ensure the VSTO runtime component, `vstor.exe`, version 8.0.50727.940 or greater is installed. To check the version of this component, locate the following registry key:

   `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\vsto runtime Setup\v2.0.50727`

   Ensure the `Update` value is 3 or greater.

2. Ensure .NET 2.0 Framework is installed.

3. Ensure .NET programmability support has been configured for Microsoft Word 2003.

To check if your system has these prerequisites, you can run the Microsoft PSS VSTO 2005 Client Troubleshooter.

The following screenshots are examples of the results that should appear when IStream Writer is configured correctly for Microsoft Word 2003 and Microsoft Word 2007 systems.

Sample Results (Microsoft Word 2003)

Sample Results (Microsoft Word 2007)

Note that because the troubleshooting tool was mainly designed for Microsoft Word 2003, it incorrectly reports that Microsoft Word is missing.

Step B: Check the Path and Permissions

For the VSTO loader to load IStream Writer, it needs to resolves the path to IStream Writer. This path is defined as:

```
%WRITER_HOME%\WriterTemplate.dll
```

1. Ensure that this path is correct, otherwise an *Office document customization is not available* error message appears. The **Details** section of this message states: *The system cannot find the file specified*.

2. If this path is correct, the loader checks the .NET security settings that have been provided for IStream Writer.

    Ensure that the permissions are correct, otherwise an *Office document customization is not available* error message appears. However, the **Details** section indicates states: *Failed to grant permission to execute*. If you see this error, continue to the next step.

Step C: Verify the Assembly Security Policy Configuration

The .NET security policies for the IStream Writer assembly files are automatically configured during the IStream Writer installation. However, sometimes there may be a problem with this configuration.

To verify that the configuration is correct, you can run a .NET utility (`caspol.exe`) that manages the assembly files' security policies. For .NET 2.0, this utility is usually located in:

```
C:\Windows\Microsoft.NET\Framework\v2.0.50727\
```

1. To verify that the .NET security policies have been configured correctly, run `caspol.exe` against one of IStream Writer's assembly files as follows:

    ```
    caspol.exe -resolvegroup "%WRITER_HOME%\
    WriterTemplate.dll"
    ```

2. If the security policy is configured correctly, then text similar to the following example will appear. (As indicated below, some of the excess text has been removed for clarity.)

```
Mcrosoft (R) .NET Framework CasPol 2.0.50727.312
Copyright (c) Microsoft Corporation.  All rights reserved.
Level = Enterprise

Code Groups:

1.  All code: FullTrust

Level = Machine

Code Groups:
1.  All code: Nothing
1.1.  Zone - MyComputer: FullTrust
1.1.4.  StrongName - 0024000004800...(text removed)...: FullTrust

1.1.8.  StrongName - 00240000048000...(text removed)...: FullTrust
```

```
Level = User

Code Groups:
1.  All code: FullTrust
```

IStream Writer Assembly Entries

You can see from this output that there are two `StrongName` entries: `1.1.4. StrongName` and `1.1.8 StrongName`. These entries relate to the IStream Writer assembly files.

The first entry is configured by IStream Writer. The second entry is added when the IStream Author add-in is installed, if both IStream Writer and IStream Document Manager are installed.

Incorrect Security Policy Configuration

If the .NET security policy is *not* configured correctly, the following text appears:

```
Microsoft (R) .NET Framework CasPol 2.0.50727.312
Copyright (c) Microsoft Corporation.  All rights reserved.
The assembly at C:\Program Files\Skywire Software\IStream
Writer\WriterTemplate.dll cannot be loaded. Caspol can make a
partial determination of what evidence would be associated with
this assembly. If this evidence is used, the results are not
necessarily accurate or complete.  Would you like to continue this
operation using partial evidence? (yes/no)
```

Success or Failure Loading IStream Writer

If the VSTO successfully loads `WriterTemplate.dll`, the Microsoft Word status bar will indicate: *IStream Writer initializing...*

However, if the VSTO cannot load this IStream Writer `.dll` file, you will see a *Writer cannot start* message. If this happens, check the IStream Writer configuration in the `UserPreferences.config` file.

IStream Writer Document Opens with Warnings

Word 2007 only

Problem

In Microsoft Word 2007, you see one or both of the following messages when trying to open an IStream Writer document:

*   `Choose XML Expansion pack`
*   `Security Warning: Some active content has been disabled`

Solution

This IStream Writer document may be in a non-trusted location. See the solution in the following section.

IStream Writer Functionality is Unavailable

Word 2007 only

Problem

An IStream Writer opens document successfully in Microsoft Word 2007. The Writer ribbon appears to be displayed correctly, but clicking the ribbon button has no effect or causes the following error message to appear:

```
Custom UI Runtime Error in WriterRibbonAddin.
```

In addition, an error message may appear in the IStream Writer log.

Solution

Verify that the IStream Writer document is in a trusted location.

1. Click the Microsoft Office Button > **Word Options > Trust Center > Trust Center Settings > Trusted Locations**.

2. If the document location is not listed in the trusted locations, add the folder (which the document is located in) to this list, or move the document to the one of existing trusted locations.

Only 'Attach Writer' Button Appears in Writer Ribbon

Word 2007 only

Problem

An IStream Writer document opens in Microsoft Word 2007, but only the **Attach Writer** button appears in the Writer ribbon. That is, the ribbon appears as it would for a non-IStream Writer document.

Solution

Ensure that the document is actually an IStream Writer customized document, in other words, that the document is *linked* to IStream Writer: see *Verifying the IStream Writer Link* on page 57.

System Does Not Select DEF Checkbox in IStream Author

Problem

In IStream Author, when creating a new model document or new model document outline, you will need to select the IStream Writer document to import. The system does not select the **Data Elements File** check box.

Solution

1. Ensure that IStream Writer is installed on the computer.

2. Ensure that a valid IStream Writer document has been selected. To do this:

    a. View the documents properties:

- in Word 2003: click **File** > **Properties**

- in Word 2007: click the Microsoft Office Button , then click **Prepare** > **Properties**

    b. Click the **Custom** tab.

    c. In the **Properties** section, verify that

- an **IStream_Writer** property appears (if you are using IStream Writer 1.2 or 1.3)

      -OR-

- an **InSystems_Writer** property appears (if you are using Writer 1.1.)
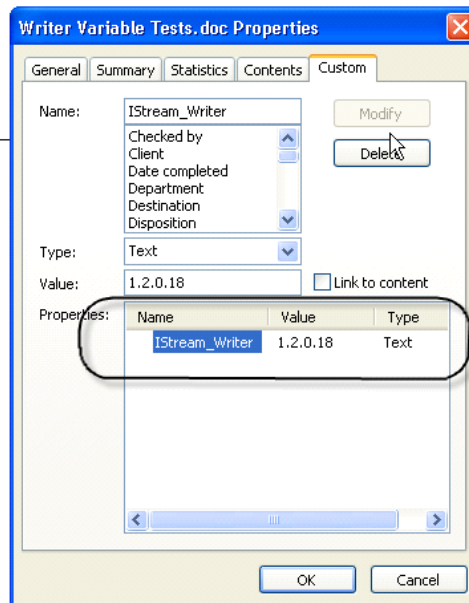
If this property has been manually deleted, then IStream Author will not recognize this document as a valid IStream Writer document. As a result, the **Data Elements File** checkbox in IStream Author will be grayed out.

Transformation Failure

Problem

A *Transformation failed with exception* message appears when transforming an IStream Writer document:

Solution

1. Close IStream Author (if it is open), then open IStream Writer.

2. Open the document that you are trying to transform.

3. Try to transform the document again. If this fails, continue to the next step.

4. Validate the IStream Writer document that you are trying to transform. To do this, on the **Writer** ribbon, click **Validate Writer Elements**.

5. If no errors are found during validation, immediately back up your most current log files: see *Backing Up the Current Log Files* on page 54, then contact customer support.

INDEX

A

Attach Writer button only appears, 67

B

backing up log files, 54

C

contacting Skywire Software for help, 11

D

DEF
    checkbox, unable to select, 67
    creating new vs. merging existing, 25
    extracting, 25
    mapping, 27
    tag, 31
DMS, configuring, 18
document conventions, 8
document fails to open, 63
documentation, IStream Writer, 10

E

Element
    subtag, 36
element samples, 41

F

Field subtag, 39
filing information elements, 25
functionality of Writer unavailable, 67

H

hardware requirements, 15
help, contacting Skywire Software, 11

I

implementation, 57
installation, 16
InSystems_Writer.dot, 53
IStream Writer
    documentation, 10

L

link to Writer, verifying, 57
list element sample, 37
ListElement tag, 37
log files, 54

M

mapping a DEF, 27
menu, Writer missing from Word, 59
missing ribbon, 59

O

opening documents
    fails, 63
    warnings, 66
operating systems, 14

Q

Query tag, 32, 33

R

resolution, screen, 15
ribbon
    missing, 59
    overview, 58

S

screen resolution, 15
Skywire Software, contacting for help, 11
software requirements, 14
SQL tag, 33
support checklist, 11
system requirements, 14

T

table element sample, 37
TableElement tag, 36
technical support, 11
Transform.config, 49
transformation failures, 69
troubleshooting, 56

U

UserPreferences.config
    for multiple users, 49
    overview, 46
    sections, 47

V

values and codes, using with groupings, 28
verifying link to Writer, 57

W

warnings, when opening document, 66

Writer
    functionality unavailable, 67
    menu missing, 59