IBM InfoSphere DataStage and Quality Stage
Version 9 Release 1

Guide to Extracting Unstructured Data

IBM
IBM InfoSphere DataStage and Quality Stage
Version 9 Release 1

Guide to Extracting Unstructured Data
Note

Before using this information and the product that it supports, read the information in “Notices and trademarks” on page 39.
Chapter 1. Unstructured Data Stage

Use the Unstructured Data stage to extract information from unstructured data sources and integrate the information with your jobs.

Unstructured data is information that does not have a predefined data model or does not fit well into relational tables. Unstructured data can be text from books, journals, metadata, audio, video files, the body of word processor documents, web pages, and presentation charts. In this release, the Unstructured Data stage supports only Microsoft Excel files as data sources.

Prerequisites

Before you start the installation and configuration of Unstructured Data stage, make sure that you meet the system requirements and that you have installed all the prerequisite software.

Before you begin

Procedure

1. Install Information Server of language that matches the language of Microsoft Excel file that you want to extract.
2. Ensure that the Microsoft Excel Viewer program that shows the content of Microsoft Excel spreadsheets (for example, Microsoft Excel, Microsoft Excel Viewer or IBM Lotus Symphony) is installed on your client machine.
3. Ensure that the file extension .xls and .xlsx are properly associated to your Microsoft Excel Viewer program.

Supported data sources

The Unstructured Data stage supports only Microsoft Excel files as the source file.

The Unstructured Data stage supports the following file formats:
- Microsoft Excel 97-2003 OLE2 (.xls), including support of password-encrypted file.
- Microsoft Excel 2007-2010 OOOXML (.xlsx), including support of password-encrypted file.

The Unstructured Data stage does not support Microsoft Excel files that are created by Microsoft Excel for Mac.

Extracting the data from Microsoft Excel

You can use the Unstructured Data stage to extract several types of data from a Microsoft Excel file.

Data ranges

When you use the Unstructured Data stage, you can extract data from a specified data range in a Microsoft Excel spreadsheet.
Data range represents a cell, a row, a column, or a selection of cells that contain one or more continuous blocks of cells. A data range is specified by the range expression. In the Unstructured Data stage, you can use a range expression to specify the data range to extract.

For example, Employee_Salary!A1:G8 describes a data range in which the first cell is A1 and the last cell is G8 in the Employee_Salary spreadsheet.

Table 1. Example of Microsoft Excel file: Employee_Salary spreadsheet

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>EMPNO</td>
<td>FIRSTNAME</td>
<td>LASTNAME</td>
<td>DEPT</td>
<td>JOB</td>
<td>SALARY</td>
<td>BONUS</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>MICHAEL</td>
<td>THOMPSON</td>
<td>B01</td>
<td>MANAGER</td>
<td>94250</td>
<td>800</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>SALLY</td>
<td>Kwan</td>
<td>C01</td>
<td>MANAGER</td>
<td>98250</td>
<td>800</td>
</tr>
<tr>
<td>4</td>
<td>60</td>
<td>IRVING</td>
<td>STERN</td>
<td>D11</td>
<td>MANAGER</td>
<td>72250</td>
<td>500</td>
</tr>
<tr>
<td>5</td>
<td>70</td>
<td>EVA</td>
<td>PULASKI</td>
<td>D21</td>
<td>MANAGER</td>
<td>96170</td>
<td>700</td>
</tr>
<tr>
<td>6</td>
<td>50</td>
<td>JOHN</td>
<td>GEYER</td>
<td>E01</td>
<td>MANAGER</td>
<td>80175</td>
<td>800</td>
</tr>
<tr>
<td>7</td>
<td>90</td>
<td>ELEEN</td>
<td>HENDERSON</td>
<td>E11</td>
<td>MANAGER</td>
<td>89750</td>
<td>600</td>
</tr>
<tr>
<td>8</td>
<td>100</td>
<td>THEODORE</td>
<td>SPENSER</td>
<td>E21</td>
<td>MANAGER</td>
<td>86150</td>
<td>500</td>
</tr>
</tbody>
</table>

The Unstructured Data stage maps the Microsoft Excel row and column in the specified data range to InfoSphere® DataStage® row and column, and extracts the records.

The following table describes the records extracted by the Unstructured Data stage when the range expression is Employee_Salary!A2:G8.

Table 2. Example of DataStage row and column

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>MICHAEL</td>
<td>THOMPSON</td>
<td>B01</td>
<td>MANAGER</td>
<td>94250</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>SALLY</td>
<td>Kwan</td>
<td>C01</td>
<td>MANAGER</td>
<td>98250</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>IRVING</td>
<td>STERN</td>
<td>D11</td>
<td>MANAGER</td>
<td>72250</td>
<td>500</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>EVA</td>
<td>PULASKI</td>
<td>D21</td>
<td>MANAGER</td>
<td>96170</td>
<td>700</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>JOHN</td>
<td>GEYER</td>
<td>E01</td>
<td>MANAGER</td>
<td>80175</td>
<td>800</td>
<td></td>
</tr>
<tr>
<td>90</td>
<td>ELEEN</td>
<td>HENDERSON</td>
<td>E11</td>
<td>MANAGER</td>
<td>89750</td>
<td>600</td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>THEODORE</td>
<td>SPENSER</td>
<td>E21</td>
<td>MANAGER</td>
<td>86150</td>
<td>500</td>
<td></td>
</tr>
</tbody>
</table>

The **Range option** property of Unstructured Data stage allows you to specify the data range either by selecting the **Specify the start row** option or the **Specify the entire data range** option. If you select the **Specify the start row** option, then identify the start row. Unstructured Data stage then identifies the end row of the data range. If you select the **Specify the entire data range** option, then you must specify the start and end rows of the data range to be extracted.

If you want to use the value of cells in the first row as IBM® InfoSphere DataStage column name, then you can use the **Column header** property. If the **Column header property** is set to **First row of data ranges**, and if you specify the range expression as Employee_Salary!A1:G8, the first row is treated as header, and the value of the cells in the first row is used as default DataStage column name in the job. You can generate range expression at design time by using Unstructured Data stage.
Types of data that can be extracted from Microsoft Excel

You can use the Unstructured Data stage to extract several types of data from a
Microsoft Excel file.

File properties

The following table lists the information that can be extracted as file
properties:

Table 3. Data that can be extracted as file properties

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>File name</td>
<td>Name of the file. For example: Workbook1.xls</td>
</tr>
<tr>
<td>File path</td>
<td>Path of the file. For example: C:\excel\Workbook1.xls</td>
</tr>
<tr>
<td>File size</td>
<td>Size of the file in bytes.</td>
</tr>
<tr>
<td>Last modified date</td>
<td>The date and time that the file was last modified.</td>
</tr>
</tbody>
</table>

Document properties

The following table lists the information that can be extracted as document
properties:

Table 4. Data that can be extracted as document properties

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authors</td>
<td>Authors of the document.</td>
</tr>
<tr>
<td>Document comments</td>
<td>Comments of the document.</td>
</tr>
<tr>
<td>Content creation date</td>
<td>The date and time that the document was created.</td>
</tr>
<tr>
<td>Key words</td>
<td>Key words of the document.</td>
</tr>
<tr>
<td>Revision number</td>
<td>Revision number of the document.</td>
</tr>
<tr>
<td>Subject</td>
<td>Subject of the document.</td>
</tr>
<tr>
<td>Title</td>
<td>Title of the document.</td>
</tr>
<tr>
<td>Company</td>
<td>Company property value of the document.</td>
</tr>
<tr>
<td>Category</td>
<td>Category of the document.</td>
</tr>
<tr>
<td>Manager</td>
<td>Manager of the document.</td>
</tr>
<tr>
<td>Custom properties</td>
<td>Custom properties of the document. You must specify the name of the custom property to extract.</td>
</tr>
</tbody>
</table>

Sheet information

The following table lists the information that can be extracted as sheet
information:

Table 5. Data that can be extracted as sheet information

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sheet name</td>
<td>Name of the Microsoft Excel sheet.</td>
</tr>
<tr>
<td>Header (left, center, right)</td>
<td>Header of the specified position.</td>
</tr>
<tr>
<td>Footer (left, center, right)</td>
<td>Footer of the specified position.</td>
</tr>
</tbody>
</table>
Row information
The following table lists the information that can be extracted as row information:

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row number</td>
<td>Microsoft Excel row number within the sheet. The first row number is 1.</td>
</tr>
<tr>
<td>Is hidden</td>
<td>Whether the row is hidden or not. Writes true if the row or the sheet to which this row belongs is hidden.</td>
</tr>
</tbody>
</table>

Cell information
You can extract the cell information based on the Microsoft Excel column or the cell position. You can specify the source Microsoft Excel column based on the relative position within the data range when extracting the cell information based on the Microsoft Excel column.

The following table lists information that can be extracted as cell information:

<table>
<thead>
<tr>
<th>Data</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>Value of a cell. If the cell has a formula, the stage extracts the value from the cache.</td>
</tr>
<tr>
<td>Comment</td>
<td>Comment of a cell.</td>
</tr>
<tr>
<td>Author of Comment</td>
<td>Author of the comment of a cell.</td>
</tr>
<tr>
<td>Formula</td>
<td>Formula of a cell in text.</td>
</tr>
<tr>
<td>Hyperlink Type</td>
<td>Type of hyperlink of a cell.</td>
</tr>
<tr>
<td>Hyperlink Address</td>
<td>The address this hyperlink points to. The format depends on type of this hyperlink.</td>
</tr>
<tr>
<td>Hyperlink label</td>
<td>Text label for this hyperlink.</td>
</tr>
</tbody>
</table>

Extracting information from unstructured data sources
You can use Unstructured Data stage to design jobs that read unstructured data from Microsoft Excel files.

About this task
The following figure shows an example of using the Unstructured Data stage to read data. In this example, the Unstructured Data stage reads data from the Microsoft Excel files and passes the rows to a Transformer stage. The Transformer stage transforms the data and then loads the data into the ODBC connector. When you configure the Unstructured Data stage to read data from the Microsoft Excel files, you create only one output link.

Procedure
1. "Adding an Unstructured Data stage to a job” on page 5.
2. "Configuring the Unstructured Data stage“ on page 5.
Adding an Unstructured Data stage to a job

Before you can read data from Microsoft Excel files, you must create a job that includes the Unstructured Data stage, add any required additional stages, and create the necessary links.

Procedure

1. From the Designer client, click File > New.
2. In the New window, click the Parallel Job icon, and then click OK.
3. From the Palette, click File.
4. Drag the Unstructured Data stage icon to the canvas.
5. Create the next stage in the job.
6. On the left side of the Designer client in the Palette menu, select the General category, and then create an output link from the Unstructured Data stage to the next stage.
7. (Optional) Double click the Unstructured Data stage icon to enter or modify the following attributes:
   - **Stage**: Modify the default name of the Stage. You can enter up to 255 characters. Alternatively, you can modify the name of the stage in the job design canvas.
   - **Description**: Enter an description of the stage.
8. Click Save.

Configuring the Unstructured Data stage

When you create an Unstructured Data stage job, you must configure the Unstructured Data stage so that it extracts the data and generates the output in the data type that the user requires.

Procedure

1. On the parallel canvas, double-click the Unstructured Data stage.
2. From the Document Type list, select Excel
3. Click Configure to configure additional properties, and define mapping between Microsoft Excel items and DataStage columns.
4. In the Data source pane:
   a. Specify the name of the file from which you want to read the data, in the **File name** field. Job compilation fails if this field is empty. If the file is password protected, specify password in the **Password** field.
   b. Optional: If you specify wildcard characters in the file name, select **Use template file for design time** and specify a template file name. Template file is used for subsequent configuration steps, and not used at runtime. Specify a value for **Template password** if the specified template file is password protected.
   c. Optional: Specify a value for **Range option**. If you select Specify the start row, you only need to specify the first row. Unstructured Data stage finds the last row at runtime. If you select **Specify the entire data range**, you need to specify both start row and end row.
   d. Optional: Specify **Range expression**. **Range expression** is a required property at runtime, but it can be empty when clicking **Load** button. Unstructured Data stage searches the entire document and lists the
candidates of data range in the Template data range list box in the Import pane. Range expression property is set with the appropriate value when you click Import in the Import pane.

e. Optional: If you want to skip any sheet names from range expression, then specify the name in the Specify the Sheet names to skip field. Use this field when the sheet names are omitted from the range expression.

f. Optional: Select First row of data ranges to skip to skip the first row at runtime. At design time, if you select None, Microsoft Excel column names are expressed in the format: "Column#column number(ColumnExcel column label)" in the Map pane. If you select First row is header, then the first row value is displayed in the Map pane.

g. Optional: Click View to launch the external Microsoft Excel viewer program. You can confirm the content of Microsoft Excel file you are working with.

h. Click Load.

5. In the Import pane:
   a. Select one data range from Template data range
   b. Optional: If you want to extract additional Microsoft Excel items such as document properties, select Property tab and select items to be extracted.
   c. Click Import.

6. In the Map pane:
   a. Define the mapping between DataStage columns and imported Microsoft Excel items. You can add DataStage column mappings or change the column order by clicking Up, Down, Insert, or Delete buttons. In DataStage Column, specify the DataStage column name for each Microsoft Excel item. In Microsoft Excel Item, you can select the item you want to map to the DataStage column. All items that can be selected in the Import pane are listed in each cell. In Import Option, you can select the Microsoft Excel item if there is any import options available. For example, If you select Microsoft Excel column in Excel Item, Value, Comment, Author of Comment, Formula, Hyperlink Type, and Hyperlink Address options are available.
   b. Click OK.

7. Specify required details in the Properties tab and the Advanced tab.

8. Click OK to save the settings that you specified. to save the settings that you specified.

Modifying the column definition on the link

You can modify the column definition such as SQL Type, Length, Scale, and Nullable on the link. If you want to change the column name that was imported by Configuration window, launch the Configuration window again to specify the name.

Procedure

1. On the parallel canvas, double-click the Unstructured Data stage.
2. Select the Output tab, then select the output link from Output name (downstream stage).
3. Edit the SQL type, Length, and Scale of each column.
4. Click OK to save the changes.
Using job parameters

Unstructured Data stage does not have the ability to create new job parameters in Configuration window. However, you can use the job parameters in the Configuration window. You must create job parameters in the Job Properties window before or after you work on the Configuration window, by selecting Edit > Job Properties from IBM InfoSphere DataStage and QualityStage® Designer client. For more information about creating job parameters, see Lesson 2.4: Adding parameters in the IBM InfoSphere DataStage Parallel Job Tutorial.

A job parameter is specified in the Configuration Window with a # character. For example, job parameter FileName, is specified as #FileName# in the Configuration window. For String type field such as File name property, you can directly type the name of job parameter within #.

If you want to use job parameter for the List type property such as Range option, you must create a List type parameter that contains a list of string variables. The String variables must match with the label text of the corresponding property in the Configuration window. For example, if you want to use job parameter for Range option property, you must create a List type job parameter that contains the string variable Specify the start row and Specify the entire data range. After creating a job parameter, select <Parameterize...> from the Configuration window, and specify the job parameter name within the # character in the Input Parameter dialog box. Click Load to edit or select variables in the Resolve job parameters panel.

Compiling and running Unstructured Data stage jobs

You can compile the Unstructured Data stage jobs into executable scripts that you can schedule and run.

Procedure

1. In the InfoSphere DataStage and QualityStage Designer client, open the job that you want to compile.
2. Click the Compile button.
3. If the Compilation Status area shows errors, edit the job to resolve the errors. After resolving the errors, click the Re-compile button.
4. When the job compiles successfully, click the Run button, and specify the job run options:
   a. Enter the job parameters as required.
   b. Click the Validate button to verify that the job will run successfully without actually extracting, converting, or writing data.
   c. Click the Run button to extract, convert, or write data.
5. To view the results of validating or running a job:
   a. In the Designer client, select Tools > Run Director to open the Director client.
   b. In the Status column, verify that the job was validated or completed successfully.
   c. If the job or validation fails, select View > Log to identify any runtime problems.
6. If the job has runtime problems, fix the problems, recompile, validate (optional), and run the job until it completes successfully.
Runtime column propagation

In InfoSphere DataStage, you can configure a job to propagate extra columns that are not defined in the metadata through the rest of the job. This process is known as runtime column propagation (RCP).

When runtime column propagation is enabled, the Unstructured Data stage propagates Microsoft Excel columns based on the first data range. If wildcard characters are used in the file name, the first file that matches the expression is used. The setting of the hidden columns property determines whether a hidden column is propagated. For each propagated Microsoft Excel column, only cell values are extracted. To extract information such as the file name, sheet name, or the row number, you can define the additional columns in the configuration window.

Column naming rules

InfoSphere DataStage columns are named based on the Microsoft Excel column letter of the first data range. The column name is prefixed by "Column_" followed by the Microsoft Excel column letter. For example, Column_A, Column_B, Column_C, and so on.

If the job already has a column with the name, the job aborts.

Data types

All columns that are added by the Unstructured Data stage are in Unicode Varchar type with undefined length.

Options to read data from Microsoft Excel files

Use the following options to modify how the Unstructured Data stage reads data.

Null row handling

You can configure the Unstructured Data stage to skip rows with null values in its cells that are being extracted.

You can set the Skip null rows property to Yes or No. The default value is No. You can set the Error action property to Fail or Skip. The default value is Fail.

Error handling

You can specify whether to log an error message and stop the job when an error occurs while extracting data from the file.

you can set the Error action property to Fail or Skip. The default value is Fail.

- If you select Fail, the Unstructured Data stage logs an unrecoverable error and stops the job when an error occurs while extracting data.
- If you select Skip, the Unstructured Data stage logs a warning message and continues to process the remaining input fields and records when an error occurs while extracting data.
Extracting the value of a particular cell or custom properties

You can specify the value of a particular cell or the custom properties to be extracted.

**Procedure**

1. On the job design canvas, double-click the Unstructured Data stage icon.
2. Click Configure.
3. From the Import pane, select Advanced tab.
4. To import the value of a particular cell, select Particular Cell in the Type column and specify the cell position in the Value column. For example, if you want to import the value of cell A1, enter A1 in the Value column.
5. To import custom property, select Custom Property in the Type column and specify the property value in the Value field. For example, if you want to import the custom property named as Prop1, specify Prop1 in the Value field.
6. Click Import.
7. In the Map pane, define mapping between InfoSphere DataStage columns and Microsoft Excel items.
8. Click OK.

Propagating hidden columns

You can specify the action that needs to be taken for the hidden columns during runtime column propagation.

You can set the Hidden columns property in the Runtime Column Propagation category to either extract or skip the hidden columns during runtime column propagation.

- If you select Extract, the hidden columns are extracted.
- If you select Skip, the hidden columns are skipped.

The default value is Extract.

Examples of extracting data from Microsoft Excel files

You can build sample jobs that extract data from Microsoft Excel files.

To get the files for the examples, extract the IS_install\Clients\Samples\Connectors\UnstructuredData_Samples.zip file.

Example 1: Extracting data from a range in an Microsoft Excel file

Create a job that uses the Unstructured Data stage to retrieve data from a range in an Microsoft Excel spread sheet.

About this task

This example uses the sample Microsoft Excel file, Employee1.xls, which contains details of employees working in an organization. This sample file has three spread sheets, Sheet1, Sheet2 and Sheet3. Sheet1 contains information about the employees in every department in the organization. Sheet2 and Sheet3 are blank. In this example, you extract business information about only the employees who work for department B01.
**Step 1: Creating the job**
Create an example job that includes one Unstructured Data stage and one Sequential File stage.

**Procedure**
1. Start the IBM InfoSphere DataStage and QualityStage Designer client.
2. In the Repository pane, right-click the Jobs folder, and then click New > Parallel job.
3. From the File section of the palette, drag the Unstructured Data stage to the canvas.
4. Drag a Sequential File stage to the canvas, then position the stage to the right of the Unstructured Data stage.
5. Create a link from the Unstructured Data stage to the sequential file stage.
6. Rename the stage and the link.
7. Select File > Save, and name the job.

**Step 2: Configuring the Unstructured Data stage**
Configure the Unstructured Data stage to extract data from a range in an Microsoft Excel file.

**Procedure**
1. Double-click Unstructured Data stage.
2. Click Configure.

   **Note:** Do not configure any stage properties in this step because you can configure all the required configurations in the Configuration window.
3. In the Configuration window, specify the full file path of the Microsoft Excel input file Employee1.xls.
4. From the Range option list, select Specify the entire data range, to extract the data in a specific range.
5. In the Range expression field, specify Sheet1!A16:K28.
6. From the Column header field, select First row of data ranges. When First row of data ranges is selected, first row is regarded as the header and the Unstructured Data stage starts extracting from the second row.
7. Click Load, then make sure that check boxes next to the Microsoft Excel columns to be extracted with the job are selected. The Unstructured Data stage accesses the specified file and lists the Microsoft Excel columns in the specified data range in the Import pane. By default, all Microsoft Excel columns are selected.
8. Clear the check box next to the Microsoft Excel columns E (PHONE NO) and I (BIRTH DATE).
9. Click Import. When Import is clicked, the Map pane at the lower right of the Configuration window is updated.
10. Click OK.
11. Confirm that the values that you specified in the Configuration window are saved on the property tab of the stage editor.
12. In the Output > Column page, change the type of the EMP_NO column to Integer, and then click OK.

**Step 3: Configuring the Sequential File stage**
Configure the Sequential File stage.
Procedure
1. Double-click the Sequential File stage.
2. On the Properties page, specify the path where you want the output file to be created, followed by the file name OutputOfExample1.txt.
3. Click OK.

Step 4: Viewing the output of the job
After you run the job, open the file, and verify the output.

About this task
For example, if a Microsoft Excel input file contains the employee information for different departments in an organization, the job can extract data from the specified department.

Procedure
1. Save the job.
2. Compile and run the job.

The following table displays the information in a Microsoft Excel input file that contains the employee information for different departments.

<table>
<thead>
<tr>
<th>EMP NO</th>
<th>FIRST NAME</th>
<th>MID INIT</th>
<th>LAST NAME</th>
<th>PHONE NO</th>
<th>HIRE DATE</th>
<th>JOB</th>
<th>SEX</th>
<th>BIRTH DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CHRISTINE</td>
<td>J</td>
<td>HAAS</td>
<td>3979</td>
<td>1/1/1995</td>
<td>PRES</td>
<td>F</td>
<td>8/24/1963</td>
</tr>
<tr>
<td>20</td>
<td>MICHAEL</td>
<td>L</td>
<td>THOMSON</td>
<td>3476</td>
<td>10/10/2003</td>
<td>MANAGER</td>
<td>M</td>
<td>2/2/1976</td>
</tr>
<tr>
<td>50</td>
<td>JOHN</td>
<td>B</td>
<td>GEYER</td>
<td>6789</td>
<td>8/17/1979</td>
<td>MANAGER</td>
<td>M</td>
<td>9/15/1955</td>
</tr>
</tbody>
</table>

Employees in DEPT_B01
50    IRVING    F          S  6423  9/14/2003  MANAGER    M    7/7/1975
90    EILEEN    W          HENDERSON 3498  8/15/2000  MANAGER    F    5/15/1971
100   THEODORE  Q          SPENSER  742   6/19/2000  MANAGER    M    12/18/1980
120   SEAN      O'CONNELL  2267  12/5/1997  CLERK      M    10/18/1972
130   DELORES   M          QUINTANA 4578  7/28/2001  ANALYST    F    9/15/1955
140   HEATHER   A          NICHOLLS 1793  12/15/2006  ANALYST    F    1/19/1976
150   BRUCE     ADAMSON    4510  12/2/2002  DESIGNER   M    5/17/1972
180   MARILYN   S          SCOUTTEN 1682  7/7/2003  DESIGNER   F    2/21/1978

After the job runs, open the OutputOfExample1.txt file to view the result.

The OutputOfExample1.txt file displays data for DEPT_B01 except the PHONE NO and the BIRTH DATE columns:

*60", "IRVING", "F", "STERN", "2003-09-14", "MANAGER", "M", "72250", "500"
*70", "EVA", "D", "PULASKI", "2005-09-30", "MANAGER", "F", "96170", "700"
*90", "EILEEN", "W", "HENDERSON", "2000-08-15", "MANAGER", "F", "89750", "600"
*100", "THEODORE", "Q", "SPENSER", "2000-06-19", "MANAGER", "M", "86150", "500"
*120", "SEAN", "O'CONNELL", "1993-12-05", "CLERK", "M", "49250", "600"
*140", "HEATHER", "A", "NICHOLLS", "2006-12-15", "ANALYST", "F", "68420", "600"
*150", "BRUCE", "ADAMSON", "2002-02-12", "DESIGNER", "M", "55280", "500"
*160", "ELIZABETH", "R", "PIANKA", "2006-10-11", "DESIGNER", "F", "62250", "400"
Example 2: Extracting data from multiple Microsoft Excel sheets

Create a job that uses the Unstructured Data stage to extract data from multiple Microsoft Excel sheets.

About this task

This example uses the sample Microsoft Excel file, Employee2.xls. This sample file has the following sheets: DEPT A00, DEPT B01, DEPT C01, and DEPT D01. Each sheet contains information about the employees in the department.

The data structure of each sheet is similar. Each sheet has the EMP NO, FIRST NAME, MID INIT, LAST NAME, PHONE NO, HIRE DATE, JOB, and ADDRESS columns, and the third row is the header. But each sheet has a different number of rows.

Step 1: Creating the job

Create an example job that includes one Unstructured Data stage and one Sequential File stage.

Procedure

1. Start the IBM InfoSphere DataStage and QualityStage Designer client.
2. In the Repository pane, right-click the Jobs folder, and select New > Parallel job.
3. From the File section of the palette drag an Unstructured Data stage to the canvas.
4. From the File section of the palette drag a Sequential File stage to the canvas. Position the stage to the right of the Unstructured Data stage.
5. Create a link from the Unstructured Data stage to the sequential file stage.
6. Rename the stages and links.
7. Select File > Save, and name the job.

Step 2: Configuring the Unstructured Data stage

Configure the Unstructured Data stage to extract the data from the multiple Microsoft Excel sheets.

Procedure

1. Double-click the Unstructured Data stage to open the stage properties.
2. Click Configure.

   Note: Do not configure any stage properties in this step because you can configure all the required configurations in the Configuration window.
3. In the Configuration window, specify the full file path of the Microsoft Excel input file Employee2.xls.
4. From the Range option, select Specify the start row.
5. In the Range expression field, specify A3:H3. When the stage runs with Specify the first row option and no specific sheet name is specified in the Range expression, the job finds the last row dynamically and extracts rows to the last row at runtime.
6. In Column header, select First row of data ranges.
7. Click Load. The Excel columns that exist in the specified data range are listed in the Import pane.
8. On the **Property** tab, select the checkbox next to the property, to extract the property value. In this example, select the **SheetName** as the property.

9. Click **Import**. The column mappings are generated by the stage.

10. To make the SheetName column the first column in the list:
   a. Select the SheetName column.
   b. Click **Up** until the SheetName column is the first column in the list.

11. In the mapping table, insert a row for ADDRESS column in the input file that has hyperlink.
   a. Click **Insert**.
   b. In the **Excel item** option, select **Column ADDRESS**.
   c. In the **Import** option cell in the new row, select **Hyper link address**.
   d. Specify the DataStage column name EMAIL_ADDRESS for the new row.

12. Click **OK**.

13. Confirm that the values that you entered on the Configuration window are saved on the **Property** tab of the stage editor.

14. Click **Output > Column** tab to change the data type or other attributes. Change the type of **EMP_NO** column to **Integer**.

15. Click **OK**.

### Step 3: Configuring the Sequential File stage

Configure the Sequential File stage.

**Procedure**

1. Double-click the Sequential File stage.

2. On the **Properties** page, specify the path where you want the output file to be created, followed by the file name *OutputOfExample2.txt*.

3. Click **OK**.

### Step 4: Viewing the output of the job

After you run the job, open the file, and verify the output.

**Procedure**

1. Save the job.

2. Compile and run the job.

An example input Microsoft Excel files that contains the employee information for each department in the different sheets. The job extracts of employee data from all sheets are displayed in the form of following tables:

**Table 9. Information of employees in DEPT_A00**

<table>
<thead>
<tr>
<th>EMP NO</th>
<th>FIRST NAME</th>
<th>MID INIT</th>
<th>LAST NAME</th>
<th>PHONE NO</th>
<th>HIRE DATE</th>
<th>JOB</th>
<th>SEX</th>
<th>BIRTH DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>CHRISTINE</td>
<td>I</td>
<td>HAAS</td>
<td>978</td>
<td>1/1/1995</td>
<td>PRES</td>
<td>F</td>
<td>8/24/1963</td>
</tr>
<tr>
<td>20</td>
<td>MICHAEL</td>
<td>L</td>
<td>THOMSON</td>
<td>3476</td>
<td>10/10/2003</td>
<td>MANAGER</td>
<td>M</td>
<td>2/2/1976</td>
</tr>
<tr>
<td>50</td>
<td>JOHN</td>
<td>R</td>
<td>GEYER</td>
<td>6789</td>
<td>8/17/1979</td>
<td>MANAGER</td>
<td>M</td>
<td>9/15/1955</td>
</tr>
</tbody>
</table>

**Table 10. Details of employees in Employees in DEPT_B01**

<table>
<thead>
<tr>
<th>EMP NO</th>
<th>FIRST NAME</th>
<th>MID INIT</th>
<th>LAST NAME</th>
<th>PHONE NO</th>
<th>HIRE DATE</th>
<th>JOB</th>
<th>SEX</th>
<th>BIRTH DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>IRVING</td>
<td>F</td>
<td>STERN</td>
<td>6423</td>
<td>9/14/2003</td>
<td>MANAGER</td>
<td>M</td>
<td>7/7/1975</td>
</tr>
<tr>
<td>70</td>
<td>EVA</td>
<td>D</td>
<td>PULASKI</td>
<td>7831</td>
<td>9/30/2003</td>
<td>MANAGER</td>
<td>F</td>
<td>5/26/2003</td>
</tr>
<tr>
<td>90</td>
<td>EILEEN</td>
<td>W</td>
<td>HENDERSON</td>
<td>5498</td>
<td>8/15/2000</td>
<td>MANAGER</td>
<td>F</td>
<td>5/15/1971</td>
</tr>
<tr>
<td>110</td>
<td>THEODORE</td>
<td>Q</td>
<td>SPENSER</td>
<td>742</td>
<td>8/19/2000</td>
<td>MANAGER</td>
<td>M</td>
<td>12/18/1980</td>
</tr>
</tbody>
</table>
Table 10. Details of employees in Employees in DEPT_B01 (continued)

<table>
<thead>
<tr>
<th>EMP NO</th>
<th>FIRST NAME</th>
<th>MIDINIT</th>
<th>LAST NAME</th>
<th>PHONE NO</th>
<th>HIRE DATE</th>
<th>JOB</th>
<th>SEX</th>
<th>BIRTH DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>120</td>
<td>SEAN</td>
<td>O'CONNEEL</td>
<td>2167</td>
<td>12/5/1993</td>
<td>CLERK</td>
<td>M</td>
<td>10/18/1972</td>
<td></td>
</tr>
<tr>
<td>130</td>
<td>DELORES</td>
<td>M</td>
<td>QUINTANA</td>
<td>4578</td>
<td>7/28/2001</td>
<td>ANALYST</td>
<td>F</td>
<td>9/15/1955</td>
</tr>
<tr>
<td>140</td>
<td>HEATHER</td>
<td>A</td>
<td>NICHOLLS</td>
<td>1795</td>
<td>12/15/2006</td>
<td>ANALYST</td>
<td>F</td>
<td>1/19/1976</td>
</tr>
<tr>
<td>150</td>
<td>BRUCE</td>
<td></td>
<td>ADAMSON</td>
<td>4510</td>
<td>2/12/2002</td>
<td>DESIGNER</td>
<td>M</td>
<td>5/17/1972</td>
</tr>
<tr>
<td>180</td>
<td>MARILYN</td>
<td>S</td>
<td>SCOUTTEN</td>
<td>1682</td>
<td>7/7/2003</td>
<td>DESIGNER</td>
<td>F</td>
<td>2/21/1978</td>
</tr>
</tbody>
</table>

3. After the job runs, open the OutputOfExample2.txt file contains the following result

"DEPT A00", "10", "CHRISTINE", "I", "HAAS", "3978", "1995-01-01", "PRES ", "CHRISTINE HAAS", "mailto:CHRISTINE%20HAAS@abc.com"
"DEPT A00", "20", "MICHAEL", "L", "THOMPSON", "3476", "2003-10-10", "MANAGER ", "MICHAEL THOMPSON", "mailto:MICHAEL%20THOMPSON@abc.com"
"DEPT A00", "30", "SALLY", "A", "KWAN", "4738", "2005-04-05", "MANAGER ", "SALLY KWAN", "mailto:SALLY%20KWAN@abc.com"
"DEPT A00", "50", "JOHN", "B", "GEYER", "6789", "1979-08-17", "MANAGER ", "JOHN GEYER", "mailto:JOHN%20GEYER@abc.com"
"DEPT B01", "60", "IRVING", "F", "STERN", "6423", "2003-09-14", "MANAGER ", "IRVING STERN", "mailto:IRVING%20STERN@abc.com"
"DEPT B01", "70", "EVA", "D", "PULASKI", "7831", "2005-09-30", "MANAGER ", "EVA PULASKI", "mailto:EVA%20PULASKI@abc.com"
"DEPT B01", "90", "EILEEN", "W", "HENDERSON", "5498", "2000-08-15", "MANAGER ", "EILEEN HENDERSON", "mailto:EILEEN%20HENDERSON@abc.com"
"DEPT B01", "100", "THEODORE", "Q", "SPENSER", "972", "2000-06-19", "MANAGER ", "THEODORE SPENSER", "mailto:THEODORE%20SPENSER@abc.com"
"DEPT B01", "110", "VINCENTO", "G", "LUCCHESSI", "3490", "1988-05-16", "SALESRP ", "VINCENTO LUCCHESI", "mailto:VINCENTO%20LUCCHESSI@abc.com"
"DEPT B01", "120", "SEAN", "O'CONNELL", "2167", "1993-12-05", "CLERK ", "SEAN O'CONNELL", "mailto:SEAN%20O'CONNELL@abc.com"
"DEPT B01", "130", "DELORES", "M", "QUINTANA", "4578", "2001-07-28", "ANALYST ", "DELORES QUINTANA", "mailto:DELORES%20QUINTANA@abc.com"
"DEPT B01", "140", "HEATHER", "A", "NICHOLLS", "1793", "2006-12-15", "ANALYST ", "HEATHER NICHOLLS", "mailto:HEATHER%20NICHOLLS@abc.com"
"DEPT B01", "150", "BRUCE", "R", "ADAMSON", "4510", "2002-02-12", "DESIGNER ", "BRUCE ADAMSON", "mailto:BRUCE%20ADAMSON@abc.com"
"DEPT B01", "160", "ELIZABETH", "R", "PIANKA", "3782", "2006-10-11", "DESIGNER ", "ELIZABETH PIANKA", "mailto:ELIZABETH%20PIANKA@abc.com"
"DEPT B01", "170", "MASATOSHI", "J", "YOSHIMURA", "2890", "1999-09-15", "DESIGNER ", "MASATOSHI YOSHIMURA", "mailto:MASATOSHI%20YOSHIMURA@abc.com"
"DEPT B01", "180", "MARILYN", "S", "SCOUTTEN", "1682", "2003-07-07", "DESIGNER ", "MARILYN SCOUTTEN", "mailto:MARILYN%20SCOUTTEN@abc.com"
"DEPT C01", "190", "JAMES", "H", "WALKER", "2986", "2004-07-26", "DESIGNER ", "JAMES WALKER", "mailto:JAMES%20WALKER@abc.com"
"DEPT C01", "200", "DAVID", "J", "BROWN", "4501", "2002-03-03", "DESIGNER ", "DAVID BROWN", "mailto:DAVID%20BROWN@abc.com"
"DEPT C01", "210", "WILLIAM", "T", "JONES", "942", "1998-04-11", "DESIGNER ", "WILLIAM JONES", "mailto:WILLIAM%20JONES@abc.com"
"DEPT C01", "220", "JENNIFER", "K", "LUTZ", "672", "1998-08-29", "DESIGNER ", "JENNIFER LUTZ", "mailto:JENNIFER%20LUTZ@abc.com"
"DEPT C01", "230", "JAMES", "J", "JEFFERSON", "2094", "1996-11-21", "CLERK ", "JAMES JEFFERSON", "mailto:JAMES%20JEFFERSON@abc.com"
"DEPT C01", "240", "SALVATORE", "M", "MARINO", "3780", "2004-12-05", "CLERK ", "SALVATORE MARINO", "mailto:SALVATORE%20MARINO@abc.com"
"DEPT C01", "260", "SYBIL", "P", "JOHNSON", "8953", "2005-09-11", "CLERK ", "SYBIL JOHNSON", "mailto:SYBIL%20JOHNSON@abc.com"
"DEPT D01", "270", "MARIA", "L", "PEREZ", "9601", "2006-09-30", "CLERK ", "MARIA PEREZ", "mailto:MARIA%20PEREZ@abc.com"
"DEPT D01", "280", "ETHEL", "R", "SCHNEIDER", "8997", "1997-03-24", "OPERATOR ", "ETHEL SCHNEIDER", "mailto:ETHEL%20SCHNEIDER@abc.com"
These topics describe supported Microsoft Excel types and Microsoft Excel-type-to-DataStage-type mappings and describe job abort conditions.

**Data type conversions from Microsoft Excel to InfoSphere DataStage**

Before the Unstructured Data stage writes data that is extracted from Microsoft Excel to the output link, the data is converted to InfoSphere DataStage data types.

The following table shows the mapping between Microsoft Excel data types and InfoSphere DataStage data types.

**Table 11. Mapping between Microsoft Excel cell value data types InfoSphere DataStage data types**

<table>
<thead>
<tr>
<th>Microsoft Excel cell data type</th>
<th>DataStage data type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blank</td>
<td>Integer data types</td>
</tr>
<tr>
<td></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td>Text data types</td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data</td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Fraction data types</td>
<td>Double Float Real</td>
</tr>
<tr>
<td>Decimal data types</td>
<td>Decimal Numeric</td>
</tr>
<tr>
<td>Date and time data type</td>
<td>Date Time Timestamp</td>
</tr>
</tbody>
</table>
Table 11. Mapping between Microsoft Excel cell value data types InfoSphere DataStage data types (continued)

<table>
<thead>
<tr>
<th>Microsoft Excel cell data type</th>
<th>DataStage data type</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Boolean</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Integer data types</strong></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: 1, FALSE: 0</td>
<td></td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: &quot;true&quot;, FALSE: &quot;false&quot;</td>
<td></td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: &quot;true&quot;, FALSE: &quot;false&quot;</td>
<td></td>
</tr>
<tr>
<td><strong>Error</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> String expression of the error. For example, #NAME?</td>
<td></td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> String expression of the error. For example, #NAME?</td>
<td></td>
</tr>
<tr>
<td><strong>Numeric</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Integer data types</strong></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Fraction data types</strong></td>
<td>Double Float Real</td>
</tr>
<tr>
<td><strong>Decimal data types</strong></td>
<td>Decimal Numeric</td>
</tr>
<tr>
<td><strong>Date and time data type</strong></td>
<td>Date Time Timestamp</td>
</tr>
<tr>
<td><strong>String</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Date and time data type</strong></td>
<td>Date Time Timestamp</td>
</tr>
</tbody>
</table>
Table 12. Microsoft Excel other cell information data types and InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Microsoft Excel other cell information data types</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formula</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Comment</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Author of comment</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Hyperlink type</td>
<td>Integer data types &lt;br&gt;BigInt Integer SmallInt TinyInt &lt;br&gt;Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Hyperlink address</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Hyperlink label</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
</tbody>
</table>

Table 13. Mapping between Microsoft Excel cell value data types and InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Microsoft Excel cell value data types</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>File name</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>File Path</td>
<td>Text data types &lt;br&gt;Char VarChar LongVarChar &lt;br&gt;National language text data types &lt;br&gt;NChar NVarChar LongNVarChar</td>
</tr>
</tbody>
</table>
Table 13. Mapping between Microsoft Excel cell value data types and InfoSphere DataStage data types (continued)

<table>
<thead>
<tr>
<th>Microsoft Excel cell value data types</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Size</td>
<td></td>
</tr>
<tr>
<td>Integer data types</td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td>Text data types</td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Last Modified Date</td>
<td></td>
</tr>
<tr>
<td>Text data types</td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>Note: String expression in yyyy-mm-dd format</td>
<td></td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Note: String expression in yyyy-mm-dd format</td>
<td></td>
</tr>
<tr>
<td>Date and time data type</td>
<td>Date Time Timestamp</td>
</tr>
</tbody>
</table>

Table 14. Mapping between Microsoft Excel document properties and InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Microsoft Excel document properties</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authors</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Document Comments</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Content Creation Date</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>Note: String expression in yyyy-mm-dd format</td>
<td></td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Note: String expression in yyyy-mm-dd format</td>
<td></td>
</tr>
<tr>
<td>Date and time data type</td>
<td>Date Time Timestamp</td>
</tr>
<tr>
<td>Key Words</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Microsoft Excel document properties</td>
<td>InfoSphere DataStage data types</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>Revision Number</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Subject</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Title</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Company</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Category</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Manager</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
</tbody>
</table>

Table 15. Mapping between Microsoft Excel custom property and InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Mapping between Microsoft Excel custom property</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Date</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td>National language text data types</td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td></td>
<td>Date and time data type</td>
</tr>
<tr>
<td></td>
<td>Date Time Timestamp</td>
</tr>
</tbody>
</table>
### Table 15. Mapping between Microsoft Excel custom property and InfoSphere DataStage data types (continued)

<table>
<thead>
<tr>
<th>Mapping between Microsoft Excel custom property</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number</td>
<td></td>
</tr>
<tr>
<td><strong>Integer data types</strong></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td><strong>Note:</strong> If the value is an integer.</td>
<td></td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Fraction data types</strong></td>
<td>Double Float Real</td>
</tr>
<tr>
<td><strong>Decimal data types</strong></td>
<td>Decimal Numeric</td>
</tr>
<tr>
<td><strong>Boolean</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Integer data types</strong></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: 1, FALSE: 0</td>
<td></td>
</tr>
<tr>
<td><strong>Text data types</strong></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: &quot;true&quot;, FALSE: &quot;false&quot;</td>
<td></td>
</tr>
<tr>
<td><strong>National language text data types</strong></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> Maps TRUE: &quot;true&quot;, FALSE: &quot;false&quot;</td>
<td></td>
</tr>
</tbody>
</table>

### Table 16. Mapping Microsoft Excel sheet information with InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Microsoft Excel sheet information</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sheet Name</td>
<td><strong>Text data types</strong></td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td><strong>National language text data types</strong></td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td>Header</td>
<td><strong>Text data types</strong></td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td></td>
<td><strong>National language text data types</strong></td>
</tr>
<tr>
<td></td>
<td>NChar NVarChar LongNVarChar</td>
</tr>
<tr>
<td><strong>Note:</strong> For both text data types and National language text data types, Microsoft Excel supports special commands represented by single letter with a leading ampersand &quot;&amp;&quot; in Microsoft Excel header and footer. The Unstructured Data stage does not convert those letters and just preserve them in the extracted text. Refer to <a href="http://msdn.microsoft.com/en-us/library/dd773041%28v=office.12%29.aspx">http://msdn.microsoft.com/en-us/library/dd773041%28v=office.12%29.aspx</a> for more information about the special commands.</td>
<td></td>
</tr>
</tbody>
</table>
Table 16. Mapping Microsoft Excel sheet information with InfoSphere DataStage data types (continued)

<table>
<thead>
<tr>
<th>Microsoft Excel sheet information</th>
<th>InfoSphere DataStage data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Footer</td>
<td>Text data types</td>
</tr>
<tr>
<td></td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Note:</td>
<td>For both text data types and National language text data types, Microsoft Excel supports special commands represented by single letter with a leading ampersand ”&amp;” in Microsoft Excel header and footer. The Unstructured Data stage does not convert those letters and just preserve them in the extracted text. Refer to <a href="http://msdn.microsoft.com/en-us/library/dd773041%28v=office.12%29.aspx">http://msdn.microsoft.com/en-us/library/dd773041%28v=office.12%29.aspx</a> for more information about the special commands.</td>
</tr>
</tbody>
</table>

Table 17. Mapping between Microsoft Excel row information and their equivalent InfoSphere DataStage data types

<table>
<thead>
<tr>
<th>Microsoft Excel row information</th>
<th>InfoSphere DataStage Data types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Row Number</td>
<td>Integer data types</td>
</tr>
<tr>
<td></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td>Text data types</td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Note:</td>
<td>Maps TRUE: 1, FALSE: 0</td>
</tr>
<tr>
<td>Is Hidden</td>
<td>Integer data types</td>
</tr>
<tr>
<td></td>
<td>BigInt Integer SmallInt TinyInt</td>
</tr>
<tr>
<td>Text data types</td>
<td>Char VarChar LongVarChar</td>
</tr>
<tr>
<td>National language text data types</td>
<td>NChar NVChar LongNVChar</td>
</tr>
<tr>
<td>Note:</td>
<td>Maps TRUE: ”true”, FALSE: “false”</td>
</tr>
</tbody>
</table>

Job abort conditions in Microsoft Excel

The tables describe the different job abort conditions in Microsoft Excel files.

File name (wildcard character is not used)

When file name is used without wildcard character, the following job abort conditions can occur:

Table 18. File name (wildcard is not used)

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>The specified file does not exist.</td>
<td>Job aborts.</td>
</tr>
</tbody>
</table>
### Table 18. File name (wildcard is not used) (continued)

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>User does not have permission to read the specified file.</td>
<td>Job aborts.</td>
</tr>
<tr>
<td>The specified file is not a valid Microsoft Excel file.</td>
<td>Job aborts.</td>
</tr>
<tr>
<td>The file cannot be opened by specified password.</td>
<td>Job aborts.</td>
</tr>
</tbody>
</table>

### File name (wildcard character is used)

When file name is used with wildcard character, the following job abort conditions can occur:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>There is no file with the specified name.</td>
<td>The job continues with a warning message (no output row)</td>
</tr>
<tr>
<td>User does not have permission to read a matched file.</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
<tr>
<td>The matched file is not a valid Microsoft Excel file.</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
<tr>
<td>The matched file cannot be opened by the specified password.</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
</tbody>
</table>

### Sheet Name

The following job abort conditions can occur for sheet name:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sheet name is specified in data range, and the sheet does not exist</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
</tbody>
</table>

### Column header

The following job abort condition can occur for column header:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>First row is column header and the value of first row does not match the value of the first row of the template data range.</td>
<td>The job continues and an informational message is logged.</td>
</tr>
</tbody>
</table>

### Data type

The following job abort conditions can occur for data type:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>The data type is not supported to extract the Microsoft Excel object type mapped to the DataStage column.</td>
<td>The job aborts.</td>
</tr>
</tbody>
</table>
Table 22. Data type (continued)

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>The specified property does not exist.</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
</tbody>
</table>

Custom property

The following job abort condition can occur for custom property:

Table 23. Custom property

<table>
<thead>
<tr>
<th>Condition</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>The specified property does not exist.</td>
<td>The job continues with a warning message if the Error action property is set to Skip. Otherwise, the job aborts.</td>
</tr>
</tbody>
</table>

Troubleshooting

Use the information in this section to help you understand, isolate, and resolve issues with the InfoSphere DataStage Unstructured Data stage.

Messages displayed at the bottom of the configuration window are truncated

If you notice that the messages displayed at the bottom of the configuration window are truncated, then move the mouse over the message area to view the entire message.

Unable to get the expected template data area

If you click Load without specifying the Range expression or specify only the sheet name in the Range expression, then a list of template data area for column mapping are displayed. However, with some Microsoft Excel files, you might not see the template data area that is required by you.

To get a complete list of Range expression for the template data area, specify the complete Range expression for the expected template data area, then click Load. If you specify the start cell for the expected template data area, then you can get the template data area starting from the specified cell.

Warning message is displayed when modifying configuration for Range expression

The Range expression field is updated when you select one of the template data areas from the list box and click Map. You can view a relevant range expression that is associated with your selected template data area. However, you might want to change the range expression. For example, by default, the Unstructured Data stage returns the range expression information including the sheet name. However, you can modify the configuration to display the range expression without the sheet name. When you click OK to save the modified configuration, you get the following warning message:

The data source has been changed since the column mapping was created. The changes might cause a runtime error. Do you want to save your changes?
If the changes are not required for column mapping, click **OK** to complete the configuration and confirm that the updated range expression is consistent with the mapping.

**Timeout error**

The following error might occur when you try to load a large Microsoft Excel file on the Configuration window and the operation did not complete within the time specified for CAS service:

Failed to process the request:
Failed to receive the response from the handler: Request Timed Out.

When this error is displayed, modify the timeout value specified with the **PropertyAdmin** command.

For example, to change the timeout value to 180 seconds, specify the following command on the services tier:

```
InformationServer/ASBServer/bin/PropertyAdmin -set -key cas.agent.timeout -value 180
```

Where, **InformationServer** is the installation directory for InfoSphere Information Server.

**Out of memory error when loading a large Microsoft Excel file**

You might encounter the following error when you try to load a large Microsoft Excel file on the Configuration window:

The file `{0}` cannot be loaded because there is not enough memory.
The file might be too large (its size is `{1}` bytes).
Specify a smaller file as the template.

Where, `{0}`: indicates the file name specified by the user and `{1}`: indicates the size of the specified file, size.

When you load a large Microsoft Excel file, a large amount of Java heap memory is used. As a result, the Connection Access Service, which is a Java process that the Unstructured Data stage uses, might hang. If the process hangs, the request is cancelled and the error message is displayed.

To avoid the error, do one of the following:
- Specify a smaller Microsoft Excel file.
- Create a smaller file by copying the original Microsoft Excel file. Delete the rows and sheets that are not used for column mapping.

**Error related to changing the heap size of ASBAgent**

You might encounter the following error when you try to load a large Microsoft Excel file:

It was not found how to change the heap size of ASBAgent

If you enable the log view for the **ISF-CAS-NATIVE** category, you might see the following message is logged:
Warning message received from the native (C++) layer:
The file {0} cannot be loaded because there is not enough memory.
The file might be too large (its size is {1} bytes).
The JVM maximum heap size is {2}. The consumed heap size is {3}. {4}

Where,
• {0} indicates the file name specified by the user
• {1} indicates the size of the specified file
• {2} indicates the maximum heap size for JVM
• {3} indicates the heap size currently consumed

To workaround the issue, increase the Java heap memory size of the ASBAgent.

Java runtime exception error

You might encounter the following fatal error when a large Microsoft Excel file is being processed:
Unstructured_Data_0,0: Java runtime exception occurred: java.lang.OutOfMemoryError
(java.util.Arrays::copyOfRange, file Arrays.java, line 4,138)

The error occurs as there is not enough Java heap memory size.

To workaround, increase the available Java heap size by setting the environment variable CC_UNST_JAVA_HEAP. The value of the environment variable is the integer value of Java heap size in MB. For example, to set Java heap size to 512 MB, set CC_UNST_JAVA_HEAP=512. The default heap size is 256 MB.

Unable to find a proper range to specify
• You might not be able to find a proper range expression to specify when you want to extract data ranges from multiple sheets or multiple files in the following situation.
  – When each data range starts from different positions
  – When end of data ranges are not the last row in the sheets

In such cases, define name of the data ranges in the source Microsoft Excel files and specify the name as range expression.
Chapter 2. Environment variables: Unstructured Data stage

The Unstructured Data stage uses these environment variables.

**CC_JNI_EXT_DIRS**

Set this environment variable to add a prefix to the class path of `java.ext.dirs` system property.

When the value of this environment variable is set, a prefix is added to the class path of `java.ext.dirs` system property.

**CC_JVM_OPTIONS**

Set this environment variable to specify the JVM arguments that are used when a job is run.

When this variable is specified, it takes precedence over the value of the default JVM arguments for the Java-based connectors. For example, if you set `CC_JVM_OPTIONS` as `-Xmx512M`, the maximum heap size is set to 512 MB when JVM instances for the connector are created.

**CC_JVM_OVERRIDE_OPTIONS**

Set this environment variable to override the JVM options for the conductor node so that you can avoid or fix a possible conflict.

In the conductor node in a parallel job, Java connectors are initialized for schema reconciliation. Therefore, all Java connectors in a job are initialized in the same JVM. In a single job, multiple stages might be developed in Java. Each of these stages might define JVM options such as class path, system property, heap size and so on. If two stages are run in the same physical JVM, the JVM options might conflict with each other.

**CC_IGNORE_TIME_LENGTH_AND_SCALE**

Set this environment variable to change the behavior of the connector on the parallel canvas.

When this environment variable is set to 1, the connector running with the parallel engine ignores the specified length and scale for the timestamp column. For example, when the value of this environment variable is not set and if the length of the timestamp column is 26 and the scale is 6, the connector on the parallel canvas considers that the timestamp has a microsecond resolution. When the value of this environment variable is set to 1, the connector on the parallel canvas does not consider that the timestamp has a microsecond resolution unless the microseconds extended property is set even if the length of the timestamp column is 26 and the scale is 6.

**CC_MSG_LEVEL**

Set this environment variable to specify the minimum severity of the messages that the connector reports in the log file.
At the default value of 3, informational messages and messages of a higher severity are reported to the log file.

The following list contains the valid values:

- 1 - Trace
- 2 - Debug
- 3 - Informational
- 4 - Warning
- 5 - Error
- 6 - Fatal

**CC_UNST_JAVA_HEAP**

Set this environment variable to control the size of the Java heap that can be used by the Unstructured Data stage.

Set the variable to an integer value that represents the Java heap size in MB. For example, to set the Java heap size to 512 MB, set `CC_UNST_JAVA_HEAP` to 512. The default Java heap size is 256 MB.
Appendix A. Product accessibility

You can get information about the accessibility status of IBM products.

The IBM InfoSphere Information Server product modules and user interfaces are not fully accessible. The installation program installs the following product modules and components:

- IBM InfoSphere Business Glossary
- IBM InfoSphere Business Glossary Anywhere
- IBM InfoSphere DataStage
- IBM InfoSphere FastTrack
- IBM InfoSphere Information Analyzer
- IBM InfoSphere Information Services Director
- IBM InfoSphere Metadata Workbench
- IBM InfoSphere QualityStage

For information about the accessibility status of IBM products, see the IBM product accessibility information at [http://www.ibm.com/able/product_accessibility/index.html](http://www.ibm.com/able/product_accessibility/index.html).

**Accessible documentation**

Accessible documentation for InfoSphere Information Server products is provided in an information center. The information center presents the documentation in XHTML 1.0 format, which is viewable in most Web browsers. XHTML allows you to set display preferences in your browser. It also allows you to use screen readers and other assistive technologies to access the documentation.

The documentation that is in the information center is also provided in PDF files, which are not fully accessible.

**IBM and accessibility**

See the [IBM Human Ability and Accessibility Center](http://www.ibm.com/able/) for more information about the commitment that IBM has to accessibility.
Appendix B. Reading command-line syntax

This documentation uses special characters to define the command-line syntax.

The following special characters define the command-line syntax:

[ ] Identifies an optional argument. Arguments that are not enclosed in brackets are required.

... Indicates that you can specify multiple values for the previous argument.

| Indicates mutually exclusive information. You can use the argument to the left of the separator or the argument to the right of the separator. You cannot use both arguments in a single use of the command.

{} Delimits a set of mutually exclusive arguments when one of the arguments is required. If the arguments are optional, they are enclosed in brackets ([ ]).

Note:

- The maximum number of characters in an argument is 256.
- Enclose argument values that have embedded spaces with either single or double quotation marks.

For example:

\texttt{wsetsrc[-S server] [-l label] [-n name] source}

The \texttt{source} argument is the only required argument for the \texttt{wsetsrc} command. The brackets around the other arguments indicate that these arguments are optional.

\texttt{wlsc [-l] [-f format] [key...] profile}

In this example, the -l and -f format arguments are mutually exclusive and optional. The \texttt{profile} argument is required. The \texttt{key} argument is optional. The ellipsis (…) that follows the \texttt{key} argument indicates that you can specify multiple key names.

\texttt{wrb -import [rule_pack | rule_set]…}

In this example, the rule_pack and rule_set arguments are mutually exclusive, but one of the arguments must be specified. Also, the ellipsis marks (…) indicate that you can specify multiple rule packs or rule sets.
Appendix C. How to read syntax diagrams

The following rules apply to the syntax diagrams that are used in this information:
- Read the syntax diagrams from left to right, from top to bottom, following the path of the line. The following conventions are used:
  - The >>--- symbol indicates the beginning of a syntax diagram.
  - The ---> symbol indicates that the syntax diagram is continued on the next line.
  - The >--- symbol indicates that a syntax diagram is continued from the previous line.
  - The --->< symbol indicates the end of a syntax diagram.
- Required items appear on the horizontal line (the main path).

```
  >---required_item
```

- Optional items appear below the main path.

```
  >---required_item
    ---optional_item
```

If an optional item appears above the main path, that item has no effect on the execution of the syntax element and is used only for readability.

```
  >---required_item
     ---optional_item
```

- If you can choose from two or more items, they appear vertically, in a stack. If you must choose one of the items, one item of the stack appears on the main path.

```
  >---required_item
    ---required_choice1
    ---required_choice2
```

If choosing one of the items is optional, the entire stack appears below the main path.

```
  >---required_item
    ---optional_choice1
    ---optional_choice2
```

If one of the items is the default, it appears above the main path, and the remaining choices are shown below.

```
  >---required_item
    ---default_choice
    ---optional_choice1
    ---optional_choice2
```

- An arrow returning to the left, above the main line, indicates an item that can be repeated.
If the repeat arrow contains a comma, you must separate repeated items with a comma.

A repeat arrow above a stack indicates that you can repeat the items in the stack.

- Sometimes a diagram must be split into fragments. The syntax fragment is shown separately from the main syntax diagram, but the contents of the fragment should be read as if they are on the main path of the diagram.

**Fragment-name:**

```
|---required_item---| fragment-name
```

- Keywords, and their minimum abbreviations if applicable, appear in uppercase. They must be spelled exactly as shown.
- Variables appear in all lowercase italic letters (for example, `column-name`). They represent user-supplied names or values.
- Separate keywords and parameters by at least one space if no intervening punctuation is shown in the diagram.
- Enter punctuation marks, parentheses, arithmetic operators, and other symbols, exactly as shown in the diagram.
- Footnotes are shown by a number in parentheses, for example (1).
Appendix D. Contacting IBM

You can contact IBM for customer support, software services, product information, and general information. You also can provide feedback to IBM about products and documentation.

The following table lists resources for customer support, software services, training, and product and solutions information.

Table 24. IBM resources

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description and location</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM Support Portal</td>
<td>You can customize support information by choosing the products and the topics that interest you at <a href="http://www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server">www.ibm.com/support/entry/portal/Software/Information_Management/InfoSphere_Information_Server</a></td>
</tr>
<tr>
<td>Software services</td>
<td>You can find information about software, IT, and business consulting services, on the solutions site at <a href="http://www.ibm.com/businesssolutions/">www.ibm.com/businesssolutions/</a></td>
</tr>
<tr>
<td>My IBM</td>
<td>You can manage links to IBM Web sites and information that meet your specific technical support needs by creating an account on the My IBM site at <a href="http://www.ibm.com/account/">www.ibm.com/account/</a></td>
</tr>
<tr>
<td>Training and certification</td>
<td>You can learn about technical training and education services designed for individuals, companies, and public organizations to acquire, maintain, and optimize their IT skills at <a href="http://www.ibm.com/software/sw-training/">http://www.ibm.com/software/sw-training/</a></td>
</tr>
</tbody>
</table>
Appendix E. Accessing and providing feedback on the product documentation

Documentation is provided in a variety of locations and formats, including in help that is opened directly from the product client interfaces, in a suite-wide information center, and in PDF file books.

The information center is installed as a common service with IBM InfoSphere Information Server. The information center contains help for most of the product interfaces, as well as complete documentation for all the product modules in the suite. You can open the information center from the installed product or from a Web browser.

Accessing the information center

You can use the following methods to open the installed information center.

- Click the Help link in the upper right of the client interface.

  Note: From IBM InfoSphere FastTrack and IBM InfoSphere Information Server Manager, the main Help item opens a local help system. Choose Help > Open Info Center to open the full suite information center.

- Press the F1 key. The F1 key typically opens the topic that describes the current context of the client interface.

  Note: The F1 key does not work in Web clients.

- Use a Web browser to access the installed information center even when you are not logged in to the product. Enter the following address in a Web browser: http://host_name:port_number/infocenter/topic/com.ibm.swg.im.iis.productization.iisinfsv.home.doc/ic-homepage.html. The host_name is the name of the services tier computer where the information center is installed, and port_number is the port number for InfoSphere Information Server. The default port number is 9080. For example, on a Microsoft® Windows® Server computer named iisdocs2, the Web address is in the following format: http://iisdocs2:9080/infocenter/topic/com.ibm.swg.im.iis.productization.iisinfsv.nav.doc/dochome/iisinfsrv_home.html.

A subset of the information center is also available on the IBM Web site and periodically refreshed at http://pic.dhe.ibm.com/infocenter/iisinfsv/v9r1/index.jsp.

Obtaining PDF and hardcopy documentation

- A subset of the PDF file books are available through the InfoSphere Information Server software installer and the distribution media. The other PDF file books are available online and can be accessed from this support document: https://www.ibm.com/support/docview.wss?uid=swg27008803&wv=1

- You can also order IBM publications in hardcopy format online or through your local IBM representative. To order publications online, go to the IBM Publications Center at http://www.ibm.com/e-business/linkweb/publications/servlet/pbi.wss
Providing comments on the documentation

Your feedback helps IBM to provide quality information. You can use any of the following methods to provide comments:

- To comment on the information center, click the Feedback link on the top right side of any topic in the information center.
- Send your comments by using the online readers’ comment form at [www.ibm.com/software/awdtools/rcf/](http://www.ibm.com/software/awdtools/rcf/)
- Send your comments by e-mail to comments@us.ibm.com. Include the name of the product, the version number of the product, and the name and part number of the information (if applicable). If you are commenting on specific text, include the location of the text (for example, a title, a table number, or a page number).
Notices and trademarks

This information was developed for products and services offered in the U.S.A.

Notices

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785 U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web
sites. The materials at those Web sites are not part of the materials for this IBM
product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it
believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose
of enabling: (i) the exchange of information between independently created
programs and other programs (including this one) and (ii) the mutual use of the
information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003 U.S.A.

Such information may be available, subject to appropriate terms and conditions,
including in some cases, payment of a fee.

The licensed program described in this document and all licensed material
available for it are provided by IBM under terms of the IBM Customer Agreement,
IBM International Program License Agreement or any equivalent agreement
between us.

Any performance data contained herein was determined in a controlled
environment. Therefore, the results obtained in other operating environments may
vary significantly. Some measurements may have been made on development-level
systems and there is no guarantee that these measurements will be the same on
generally available systems. Furthermore, some measurements may have been
estimated through extrapolation. Actual results may vary. Users of this document
should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of
those products, their published announcements or other publicly available sources.
IBM has not tested those products and cannot confirm the accuracy of
performance, compatibility or any other claims related to non-IBM products.
Questions on the capabilities of non-IBM products should be addressed to the
suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or
withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to
change before the products described become available.

This information contains examples of data and reports used in daily business
operations. To illustrate them as completely as possible, the examples include the
names of individuals, companies, brands, and products. All of these names are
fictitious and any similarity to the names and addresses used by an actual business
enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which
illustrate programming techniques on various operating platforms. You may copy,
modify, and distribute these sample programs in any form without payment to
IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

**Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)

The following terms are trademarks or registered trademarks of other companies:

Adobe is a registered trademark of Adobe Systems Incorporated in the United States, and/or other countries.

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

The United States Postal Service owns the following trademarks: CASS, CASS Certified, DPV, LACS\textsuperscript{link}, ZIP, ZIP + 4, ZIP Code, Post Office, Postal Service, USPS and United States Postal Service. IBM Corporation is a non-exclusive DPV and LACS\textsuperscript{link} licensee of the United States Postal Service.

Other company, product or service names may be trademarks or service marks of others.
Index

C
CC_UNST_JAVA_HEAP environment variable 27
command-line syntax
  conventions 31
  commands
    syntax 31
customer support
  contacting 35

command-line syntax 31
commands syntax 31

data types
DataStage 15
loading data 15
Unstructured Data stage 15
writing data 15

environment variables
  Unstructured Data stage 27
Example 2: Creating the job 12
Extracting data from a range in an Microsoft Excel file
  Creating the job 10
Extracting the data
  Data range 2
  DataStage 1, 3

jobs
  compiling and running 7

legal notices 39

mapping
data types 15

product accessibility
  accessibility 29
  product documentation
    accessing 37

software services
  contacting 35

special characters
  in command-line syntax 31
support customer 35
syntax command-line 31

troubleshooting
  Unstructured Data stage 23

unstructured data stage
  using in a DataStage job 4
Unstructured Data stage 12
  Adding Unstructured Data stage to a job 5
  compiling and running jobs 7
  configuring the data source 5
  error handling 8, 9
  Example 1: Configuring the Sequential File stage 11, 13
  Example 1: Configuring the Unstructured Data stage 10
  Example 2: Configuring the Unstructured Data stage 12
  Examples: Extracting data from Microsoft Excel files 9
  extracting the value of a cell or customer properties 9
  installing and configuring 1
  Job abort conditions 21
  Job parameters in Configuration Window 7
  modifying the column definition 6
  null row handling 8
  options for reading data from Microsoft Excel files 8
  overview 1
  runtime column propagation 8
  Viewing the output of the job 11, 13

validation
  running 7

web sites
  non-IBM 33