

Spectrum™ Technology Platform

Version Technical Preview

Flow Designer Guide



Table of Contents

1 - Welcome to our technical preview of Flow Designer

About this preview version	4
----------------------------	---

2 - Getting starting with the Flow Designer user interface

Browser behavior	6
Accessing Flow Designer	6
Using mouse and cursor actions	7
Field data formats	8
Data Types	10
Flow Designer Home page	10
Flow Designer Manager page	14
Edit Job page	17
New Flow page	21
Profile page	22
Introduction to flow types	22
Introduction to stages	24

3 - Using Flow Designer - Building, testing, and running flows

Designing and building flows	37
Deleting flows	65
Exposing flows	66
Importing flows	66
Exporting flows	67
Running an External Program	69

Using the template browser to create flows and jobs	70
---	----

4 - File properties reference

Supported character encoding methods	73
Field separators	73
Record separators	74

5 - Stage and transform reference

Flow Designer stages and transforms	76
Transformer stage transform types	80

6 - Supported data types reference

Data types	86
------------	----

7 - Date and time patterns reference

Date and time patterns	88
------------------------	----

8 - About Spectrum™ Technology Platform

What Is Spectrum™ Technology Platform?	92
Enterprise Data Management Architecture	93
Spectrum™ Technology Platform Architecture	96
Modules and Components	101

1 - Welcome to our technical preview of Flow Designer

In this section

About this preview version

4

About this preview version

This is a technical preview version of Flow Designer.

This preview version allows you to test-drive the fundamental capabilities that we believe you will use most often. We are providing this preview version now so that you can help us make sure that future releases will have the features that you find most useful.

As you explore this preview version, keep these things in mind:

- Custom transforms are supported in this preview version of Flow Designer. Custom transforms remain configurable through Enterprise Designer for use in your Flow Designer jobs. Note that multiple Custom Transforms are not supported in this preview version.
- This version does not include all stages currently supported in Enterprise Designer. You will see a descriptive message if you try to open an unsupported stage in Flow Designer.
- If you encounter a roadblock, or if something isn't working as you'd expect, we ask that you let us know through our Technical Support team.
- Although we encourage you to use this preview version of Flow Designer in your production environment, know that we cannot provide standard service level agreement (SLA) for preview versions of our software. Additionally, we cannot guarantee that the work you do with the preview version will migrate to future releases.

2 - Getting starting with the Flow Designer user interface

Flow Designer is a visual tool for creating work flows. Using this client, you can:

- Create and modify jobs, services, subflows, and process flows
- Inspect and validate flows for correctness
- Expose and hide services
- Generate reports

In this section

Browser behavior	6
Accessing Flow Designer	6
Using mouse and cursor actions	7
Field data formats	8
Data Types	10
Flow Designer Home page	10
Flow Designer Manager page	14
Edit Job page	17
New Flow page	21
Profile page	22
Introduction to flow types	22
Introduction to stages	24

Browser behavior

Flow designer supports current versions of Microsoft Internet Explorer (IE), Mozilla Firefox, and Google Chrome.

Note: Flow Designer opens new pages in new (separate) tabs.

Note these unique browser limitations.

- Firefox will open new tabs when opening a flow or when you select the **Home** button on the workflow editor page.
- IE version 11 "flashes" the tab for the page you select if that page is currently open, but does not bring that page into the front focus of the display.

Note: Downloaded files are saved to a default location defined by your browser settings. Browser settings are global, applying to *all* downloads, and cannot be set specifically for Flow Designer.

Accessing Flow Designer

Use your browser to access Flow Designer from the Spectrum™ Technology Platform Welcome Page.

To start Flow Designer:

1. Open a web browser and go to the Spectrum™ Technology Platform Welcome Page:

`http://server:port.`

where *server* is the server name or IP address of your server, and *port* is the HTTP port. By default, the HTTP port is 8080.

2. Click **Platform Client Tools**.
3. Select the **Web** drop-down arrow to expand the list of selections.
4. Click **Open Flow Designer**.
5. Enter a valid username and password.

The default user name is "admin" and the default password is "admin".

Important: You should change the admin password immediately after installing Spectrum™ Technology Platform to prevent unauthorized administrative access to your system.

Using mouse and cursor actions

The user interface uses familiar Windows™ mouse and cursor actions to navigate its pages.

Scroll page content

Mouse actions allow you to scroll through the content of any page.

Use the mouse to click the scroll arrow at either end of the scroll bars, click on an empty portion of the slider bar, or click-and-drag the scroll box.

Use the mouse wheel to scroll up and down in a page view. Alternatively, click anywhere inside the scroll bar, then use the mouse wheel to move up and down through the display.

Resize window - cursor dragging method

Follow these steps to resize the window in focus using your mouse cursor.

1. Hover the mouse pointer over any corner or side of the page in focus.
2. Click and hold down the left mouse button.
3. Drag the arrow out or in to make the window larger or smaller.
4. Release the mouse button when the window is the size you want.

Resize window - keyboard method

Follow these steps to resize the window in focus using your keyboard.

1. Click the mouse pointer on the edge of the window that you want to resize.
2. Release the mouse button when the window is the size you want.
3. Hold down the left mouse button. The mouse pointer changes into a double-headed arrow.
4. While continuing to hold down the left mouse button, press the arrow keys in any direction to resize the window.

Resize columns

In tabular displays, use the mouse to make columns wide or narrow.

1. Hover over a column boundary until the cursor changes to a plus sign with arrows.
2. Hold down the left mouse button and drag the cursor to resize a column.

Field data formats

Automatic Data Type Conversion

When your flow presents data to a stage for processing, but the data is not in an acceptable format (data type), Spectrum™ Technology Platform can, in some cases, automatically convert the data to allow processing.

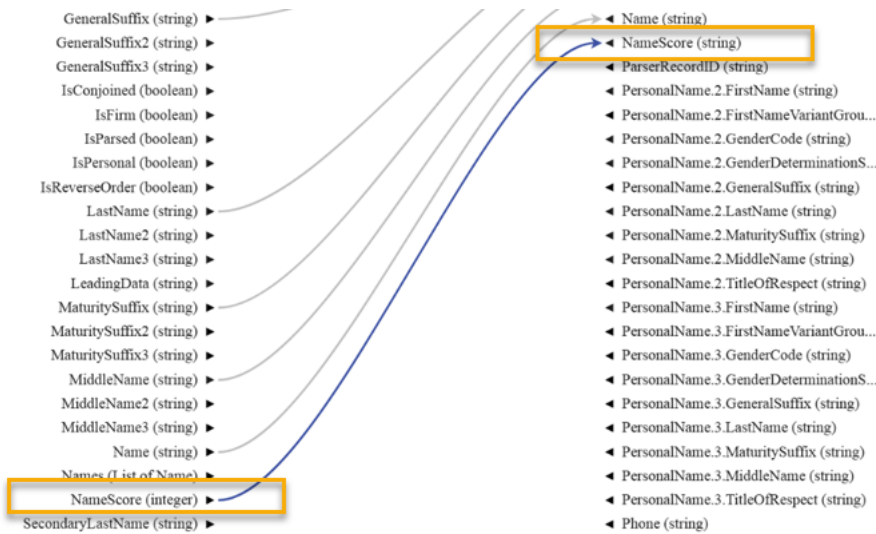
Note: This feature will be available for all stages in a future release.

For example, Validate Address accepts only string data as input. If the PostalCode input field is of type integer, Spectrum™ Technology Platform can automatically convert the field to string and successfully process the PostalCode field. Likewise, the Math stage requires numeric data. If Math detects incoming string data, Spectrum™ Technology Platform can convert the data to the data as defined in the Math stage's **File** tab.

Automatic data type conversions occur in the channels of a flow. If a channel is successfully converting a data type, there will be a blue dot in the middle of the channel:

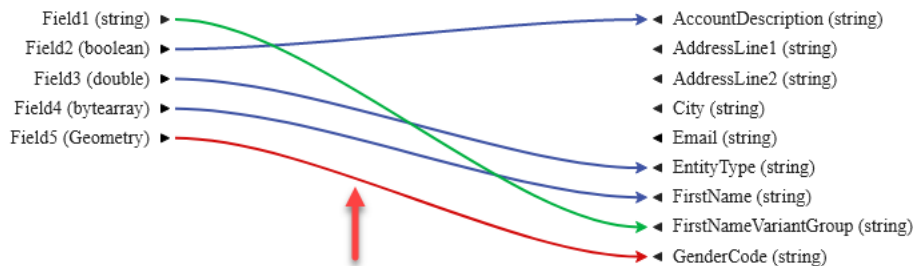


If you double-click the channel you can see the data type conversions for mapped fields. In the example below, the Open Name Parser output field **NameScore**, which is automatically mapped to the Write to File input field **NameScore**, and converted from integer to string format.



You cannot change the data type in this dialog box for automatic data type conversions. The output data type is determined by settings in the downstream stage.

Fields that do not contain valid values or that cannot be converted result in a red map channel.



You can specify the processing of conversion/mapping errors using type conversion options.

Reserved Field Names

Flow designer reserves these field names, so do not use these names in your flows:

- Status
- Status.Code
- Status.Description

Data Types

Spectrum™ Technology Platform supports a variety of numeric, string, and complex data types. Depending on the type of processing you want to perform you may use one or more of these. For an address validation flow you might only use string data. For flows that involve the mathematical computations you may use numeric or Boolean data types. For flows that perform spatial processing you may use a complex data type. For flows that combine these, you may use a variety of data types.

Specifying a Field's Data Type

You can specify the data type for a field in these situations:

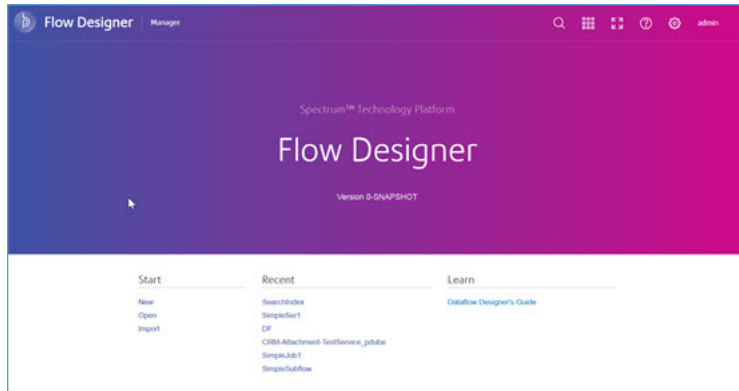
- **Source stages:** Specifying data types allows you to set the data type at the beginning of a flow, eliminating the need for data type conversions later in the flow. Note that for Read from DB, the data type is selected automatically and cannot be changed.
- **Sink stages:** Specifying data types allows you to control the data format returned by the flow. Note that for Write to DB, the data type is selected automatically and cannot be changed.
- **Transformer stage:** You can specify data types in this stage if you use a custom script.
- **Math stage and Group Statistics stage:** Since these stages perform mathematical calculations, choosing to use a particular numeric data type can have an effect on the results of the calculations, such as the precision of a division operation. If you specify a data type for a field that is different than the data type of the field coming into the stage, the downstream channel will automatically convert the field to the data type you specify, as described in [*Automatic Data Type Conversion](#).

Note: Each stage supports different data types. For a description of the supported data types for each stage, see the documentation for a specific stage.

Flow Designer Home page

The **Home** page is the first page you see when you start Flow Designer.

This page is the initial access point for Flow Designer features.




Application toolbar - Home page

The **Application toolbar** is at the top right corner of the **Home** page.




The Application toolbar offers these tools and tasks:

Search tool


Use the **Search** tool  to open an input box that allows you to search for flows by name or by key string.

Note: You can also use the keyboard shortcut **CTRL+F** to search.


Application Switcher

Use the **Application Switcher**  to start another application in the same browser page.


Full-Screen Mode tool

Use the **Full-Screen Mode** tool  to expand the display. Press **ESC** to exit full-screen mode.

Help tool

Use the **Help**  tool to access documentation, the keyboard shortcuts reference, or to contact Pitney Bowes support.

Application Setting tool

- Use the **Application Setting** tool  to clear the history of recently opened flows.
- Click the user name (in our example, "admin") to display the Manager tools or to sign out.

Version information

Find the installed version information in the middle of the Flow Designer **Home** page.

Start, Recent, and Learn access points

- Under **Start**:
 - Click **New** to display the **New Flow page** on page 21, in a new tab, which allows you to create a new empty flow or a new flow from a template.
 - Click **Open** to display the **Flow Designer Manager page** on page 14, in a new tab, with a list of saved and exposed flows.
 - Click **Import** to display a file selector. Go to an existing data flow *.df file to open in Flow Designer.
- Under **Recent**, click any flow to open it on the **Edit Job page** on page 17.
- Under **Learn**, click any help resource. Under **Learn**, click any help resource to open or access it.

The canvas

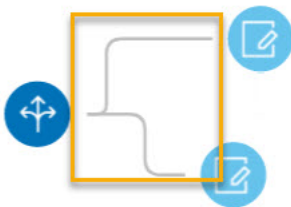
The canvas is the main work area, located on the Manager page. To access the canvas, click **Open** or **Manager** from the **Home** page title bar.

Using the canvas, you can you can:

- Edit saved flows.
- Save an existing flow with a new name to create a starting point for a new flow.

Learn more about the controls on the Manager page and its controls: **Flow Designer Manager page** on page 14.

Channel A channel is a connection between two or more stages through which records are passed from one stage to another.



Port If you look closely at the stage icons you will notice small round port indicators on one or more sides of a stage. A port sends data to stages or reads data from a channel. Stages that read data into the flow ("sources") have output ports since they are at the start of a flow. Stages that send data out of the flow ("sinks") have input ports since they are at the end of a flow. All other stages have both input and output ports.



Some stages have error ports, which output records that cause errors during the stage's processing. Some stages have report ports, which generate reports about the stage's output.

You can hover over a port indicator to view its name.



You can double-click a port indicator to open a page that allows you to configure that port's details. The configuration settings vary depending on the type of operation being defined.

List and work with saved flows

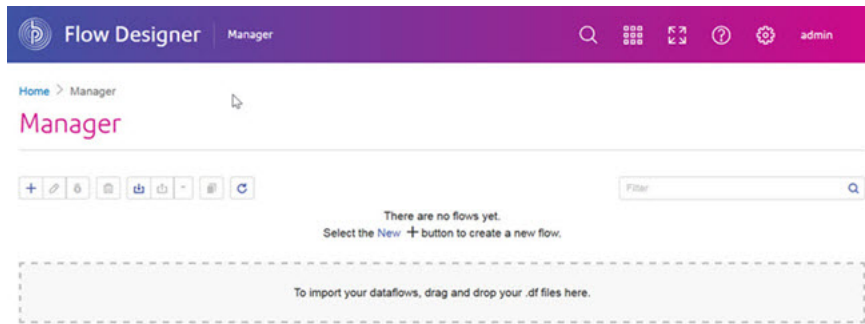
Flow Designer displays a list of saved flows, providing quick and easy access to those flows.

To open the list view: From the [Flow Designer Home page](#) on page 10, click **Open** or **Manager** from the title bar. From the list view, you can:

- **View details** – See the locked status, flow type, exposed version, and last date and time the flow was modified.
- **Open a flow** - Click on any name to open the flow.
- **Select multiple flows**– Check any box to edit, delete, export, or unlock a locked flow.
- **Global search** - Specify a keyword in the entry field. Lists any flow name or type that contains the string you enter; clear the search using "x"
- **Sort** – Use the up and down sort arrows next to any column heading to sort on that field. Or, click on the column heading to toggle the sort order.

Flow Designer Manager page

The **Manager** page displays and allows access to flows. To work on the Manager page, click **Open** or **Manager** from the Home page title bar.



From this page you can:

- Edit saved flows.
- Import or export saved flows.
- Save an existing flow with a new name to create a starting point for a new flow.
- Return to the Home page.

Icons, toolbar, and controls - Manager page

This topic provides a reference to the controls on the Manager page.

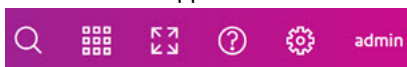
Table 1: Controls on the Manager Page

Home button








Click the **Home** button to return to the Home page.

Application



The **Application toolbar** is at the top right corner of the page and supports these tools and tasks:

- Search
- Application Switcher
- Full-screen mode
- Display Help
- Application setter
- Display Version information
- Provide **Start**, **Recent**, and **Learn** access points







Search tool		Use the Search tool to open an input box that allows you to search for flows by name or by key string. Note: You can also use the keyboard shortcut CTRL+F to search.
Application Switcher		Use the Application Switcher to start another application in the same browser page.
Full-Screen Mode tool		Use the Full-Screen Mode tool to expand the display. Press ESC to exit full-screen mode.
Help button		Use the Help button to access documentation, access the keyboard shortcuts reference, or to contact Pitney Bowes support.
Application Setting tool		<ul style="list-style-type: none"> Use the Application Setting tool to clear the history of recently opened flows. Click the user name (in our example, "admin") to display the Flow Designer Tools or to to sign out.
Version information		Find the installed version information in the middle of the Flow Designer Home page.
Start, Recent, and Learn access points		<ul style="list-style-type: none"> Under Start: <ul style="list-style-type: none"> Click New to display the New Flow page, in a new tab, which allows you to create a new empty flow or a new flow from a template. Click Open to display the Manager page, in a new tab, with a list of saved and exposed flows. Click Import to display a file selector. Go to an existing data flow *.df file to open in Flow Designer. Under Recent, click any flow to open it on the Edit Job page. Under Learn, click any help resource. Under Learn, click any help resource to open or access it.

Paging controls - Manager page

This topic provides a reference to the paging and display controls on the Manager page. The Manager page has a group of buttons that let you change the content and number of flows displayed in the table.


The Manager page has a group of buttons that let you change the content and number of flows displayed.




Table 2: Controls - Manager page






	Scroll to the top of the list
	Scroll up one page
	Select a page to display from the series of pages. The highlighted page is the currently displayed page in the series.
	Scroll down one page
	Scroll to the bottom of the list
	Use the drop-down to define the number of rows display in the flow list. If the number of flows exceeds the number of rows defined, Flow Designer shows the page count in the series of pages display.

Task buttons - Manager page

This topic provides a reference to the Task buttons on the Manager page.


The **Manager** page task buttons  support tasks that allow you to work with new and existing flows.

New flow		Opens the New Flow dialog, allowing you to select a template for a new flow, or to open the blank job workspace to define a new flow on a blank canvas.
Edit selected flow		Edit the selected flow using the canvas and its available tools and sources. Edit opens one or more selected flows in the flow editor.
Lock or unlock selected flow		When you lock flows, you prevent others from making changes.

Delete flows		Deletes the selected flow.
Import flows		
Export flows		
Copy selected flows		Copying is useful for applying existing elements to create a new flow.
Refresh the list of flows		Refresh the flow list after you make changes. Click this button, periodically, to capture changes submitted by other users in a shared environment.

Edit Job page

The **Edit Job** page contains a canvas with a palette and workflow task buttons that you use to modify existing flows.

To access this page, select a flow on the Manager page and click the **Edit**  button. From this page you can:

- Work with a new or existing flow
- Import or export flows
- Perform last actions
- Edit flow details
- Run a flow
- Change the display

Toolbar and controls - Edit Job page

Home button and flow name

These items display at the top left corner of the page and support these tasks:

Return to the Home page



Displays the current flow name: You can click and rename *Current flow name* the current flow to save it as a new file.

Application toolbar

The **Application toolbar** is at the top right corner of the page and supports these tasks:

Enter full screen mode



Access help resources: View documentation, View the keyboard shortcuts reference, or contact Pitney Bowes support.



Sign out of Flow Designer



Canvas task buttons

The canvas task buttons allow you to interact with the workflow space.

Open the New Flow page to select a flow, job, or service to add to the flow currently displayed on the canvas.



Open the Manager page, and display the flow list to select and open one of the available flows.

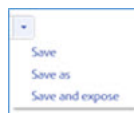


Save the flow displayed on the canvas.



Select a Save method:

- Save - Perform a standard save.
- Save as - Save a copy of the flow with a new name.
- Save and expose - Save the flow, creating a versioned copy.




Note: All saved flows are visible (exposed) to other Flow Designer users.


Import a flow 


Export a flow 


Undo the last action 

Repeat the last action 

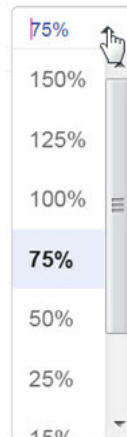
Cut the selected item from the canvas 


Copy the selected item to the clipboard 


Paste the item on the clipboard to the canvas 

Permanently delete the selected item 

Select the workflow canvas display size as a percentage of the actual size



Fit the canvas display to the page (screen) 

Run the flow that is currently displayed 

Toggle the grid line display



Toggle overview display: Click to open a small overview display that shows you the current focus of a workflow display. This is useful for larger, complicated workflows.



Refresh the current display



Panel display and view settings

These controls change the display of the palette panel.

Show the Validation Errors panel or Hide the Validation Errors panel. This panel evaluates the accuracy of the flow on the canvas and reports the errors in a concise view. The errors will contain the stage and a description of the error.



Show the Inspection panel or Hide the Inspection panel. This panel provides a brief but expandable view of the selected items.



Show the Execution History panel or hide the Execution History panel. This panel provides a quick way for you to run flows and obtain processing results.



Filter

Immediately below the palette view settings is the filter entry box. You can narrow the stages displayed by entering a string, such as "read," and see only the stages containing that description.



Palette display and view settings

The palette has a robust set of display settings that you can change as needed.

Toggle to hide or show the palette panel (**CTRL+C**)



List mode display - lists all of the stages available in a compact view with small icons. With this view, you can quickly scan available stages.



Icon mode – lists all of the stages with larger icons. Since this display emphasizes icons, more scrolling is required to see all the stages in a grouping.



Expand all – shows all of the stages in all groupings as indicated by the selected (List or Icon) display mode.



Collapse all – Shows a list of available Sources, Sinks, and Control stages, deployable modules, and user-defined stages in a list format. Click on any label to expand the display and see the stages under each label as indicated by the selected (List or Icon) display mode.



Stages

The palette shows the stages you can apply to your flows.

New Flow page

The **New Flow** page is the starting point for building new flows. Flow Designer provides a template browser that allows you to select templates to use in creating new flows.




To access this page, click **New** or the new flow button  from the Manager page.

Filter and find


Use the Filter input box to quickly find a template based on a keyword.



Show templates

Use the list on the left to show the Job, Service, or Subflow templates. Click on any help tool  next to any flow label to display a description of the types of flows.

Inspect templates

Hover over any template, and you will see a zoom tool . Click the zoom icon to inspect the template content before selecting it.

Profile page

The profile page allows you to perform administrative tasks for the current user. To access this page, select **Admin** from the Application toolbar on the Home page or the Manager page.

The Profile page contains these fields and selections:

Display user name	The user name display comes from the user's login credentials. You cannot change this field.
Specify an e-mail address	Use this input field to specify an email address for the current user.
Change the password	Display the Change Password dialog box to update those settings.
Change the language	Select one of the supported languages from the Language drop-down, and the user interface will change the language display settings.
Define the country (locale)	Further tune your language settings using the Country drop-down. For example, if you choose Spanish as your language, you can select a specific Spanish-speaking locale, such as Chile, to localize to that dialect.

Introduction to flow types

A flow is a series of operations that takes data from some source, processes that data, then writes the output to a destination. The processing of the data can be take the form of simple sorting to more complex data quality and enrichment actions.

While the concept of a flow is simple, you can design very complex flows with branching paths, multiple sources of input, and multiple output destinations. The [New Flow page](#) on page 21 is the starting point to build a job flow, service flow, or subflow.

Job

A job is a flow that performs batch processing. A job reads data from one or more files or databases, processes that data, and writes the output to one or more files or databases. Run jobs manually in Flow Designer or from a command line using the job executor.

The job example, below, uses the Read from File stage for input and two Write to File stages as output.



Service

A service is a flow that you can access as web services or using the Spectrum™ Technology Platform API. You pass a record to the service and optionally specify the options to use when processing the record. The service processes the data and returns the data.

Some services become available when you install a module. For example, when you install the Universal Addressing Module the service ValidateAddress becomes available on your system. In other cases, you must create a service in Flow Designer, then expose that service on your system as a user-defined service. For example, the Spatial Module's stages are not available as services unless you first create a service using the module's stages.

You can also design your own custom services. For example, you can create a flow that helps to determine whether an address is at risk for flooding.

Note: Since the service name, option name, and field name ultimately become XML elements, they may contain characters that are invalid in XML element names (for example, spaces are not valid). Services not meeting rules of well-formed XML will function but will not be exposed as web services.

Subflow

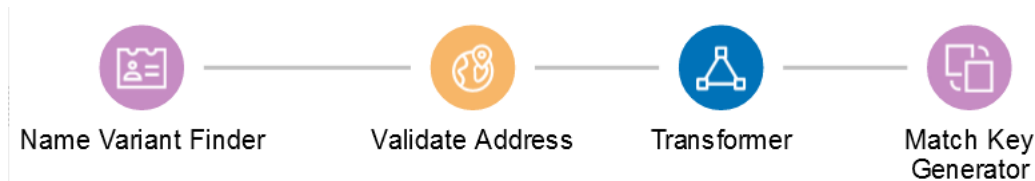
A subflow is a flow that can be reused within other flows. Subflows are useful when you want to create a reusable process that can be easily incorporated into flows. For example, you might want to create a subflow that performs deduplication using certain settings in each stage so that you can use the same deduplication process in multiple flows.

You could then use this subflow in a flow. For example, you could use the deduplication subflow within a flow that performs geocoding so that the data is deduplicated before the geocoding operation.

In this type of flow, data would be read in from a database then passed to the deduplication subflow, where it would be processed through Match Key Generator, then Intraflow Match, then Best of Breed, and finally sent out of the subflow and on to the next stage in the parent flow, in this case Geocode US Address.

Process flows

A process flow runs a series of activities such as jobs and external applications. Each activity in the process flow runs after the previous activity finishes. Process flows are useful if you want to run multiple flows in sequence or if you want to run an external program. For example, a process flow could run a job to standardize names, validate addresses, then invoke an external application to sort the records into the proper sequence to claim postal discounts. Such a process flow would look like this:



In this example, the jobs Standardize Names and Validate Addresses are exposed jobs on the Spectrum™ Technology Platform server. Run Program invokes an external application, and the Success activity indicates the end of the process flow.

Introduction to stages

The palette on the edit job page, adjacent to the canvas, shows the stages you can apply to your flows.

Note: The stages and sinks displayed in the **Stages** palette are grouped and color-coded by type.

Sources	Define the input for a flow. A source is the first stage in a flow, and defines the input data to process.
Sinks	Define the output from a flow. A sink is the last stage in a flow, and defines what to do with the output from the flow. A sink can also perform other actions at the end of a flow, such as executing a program.
Control stages	Move data along different paths in a flow, to split or group records, and to perform basic data transforms and mathematical operations. Custom transforms, a type of control stage, are supported in this preview release of Flow Designer. You also have the option of configuring custom transforms through Enterprise Designer for use in your Flow Designer jobs.
Module-specific stages	Provide functions that are available through the modules you have installed on the Spectrum Technology Platform.

Note: We will add module-specific stages in a future release.

User-defined stages Provide a means to develop custom operation stages for your flows.

Input and Read from File stages

The input stage type for a flow depends on the type of flow.

Note: This technical preview version of Flow Designer allows you to use input files and to write to output files that are on the Spectrum Server location, only. At this time, Flow Designer does not support *local* input or output files.

Input stage types

Input data for a job can come from a file, database, or cloud service, depending on the modules you have licensed. Each module supports input from different sources, and the procedure for configuring each type of source varies.

- Jobs use the Read from File input stage. For more information, see [Configuring the Read from File Stage](#) on page 37.
- Services use the Input stage. For more information, see [Configuring the Input stage](#) on page 39.
- Subflows use the Read from File, Input, or Read from XML stage since they accept input from Jobs and Services. For more information, see [Configuring the Read from XML Stage for Subflows](#) on page 41.

When you specify a flow's input file, Flow Designer examines the input file to determine as much of the file information as possible using that information to populate the fields on this page. The information it returns includes:

- Record type:
 - Line Sequential - Records are separated by an end-of-line (EOL) character such as a carriage return or line feed (CR or LF) and each field has a fixed starting and ending character position.
 - Fixed Width - Each record is a specific number of characters in length and each field has a fixed starting and ending character position.
 - Delimited - Records are separated by an end-of-line (EOL) character such as a carriage return or line feed (CR or LF), and each field is separated by a designated character such as a comma.
- Character Encoding ([Supported character encoding methods](#) on page 73)
- Field Separator ([Field separators](#) on page 73)
- Text Qualifier – single or double quotes
- Record Separator ([Record separators](#) on page 74)
- File Schema (layout)

Input for a Job

Input data for a job can come from a file or a database. Spectrum™ Technology Platform has the ability to read data from many file formats and database types. The types of data sources you can

read from depend on which modules you have licensed. The Enterprise Data Integration Module provides access to the most data sources of any module.

Note: When designing a job, you should anticipate the possibility of malformed input records. A malformed record is one that cannot be parsed using one of the parser classes provided by Spectrum™ Technology Platform. For information about handling malformed input records, see [Managing malformed input records](#) on page 26.

Input for a Service

Input data for a service is defined in an Input stage. This stage defines the fields that the service will accept from a web service request or an API call.

See the solution guide for your modules available at support.pb.com/spectrum.

File tab

The file tab defines relevant content and format information.

- Display details about the file's format and layout.
- Define the presence of a header record.
- Mark records with fewer fields than defined as "malformed."
- Use the convenient File Schema workspace to change record details using the user interface.

Sort tab

The Sort tab describes a flow's sort key fields and sort order.

- Accept, delete, or change the sort fields used.
- Accept or change the order of the sort fields.
- Accept or change the sort direction for any sort field: ascending or descending.

Runtime tab

The Runtime tab defines a flow's input file processing options.

- Define the starting record to process.
- Define whether to process all input records or specify a maximum number of records to process.

Managing malformed input records

A malformed record is one that Spectrum™ Technology Platform cannot parse. When Spectrum™ Technology Platform encounters a malformed record, it can do one or more of these tasks:

- Terminate the job
- Continue processing
- Continue processing until a certain number of bad records are encountered
- Continue processing but write bad records to a log file (via an optional sink stage)

Note: Malformed records functionality is limited to sources configured to read from files local to the server and that do not have sorting configured. When a source is configured with either a remote file or with sort fields and the source encounters a malformed record, the job will terminate regardless of the configuration for malformed records.

To manage malformed records,

1. Open the flow on the canvas.
2. Add a malformed records sink in your flow.
 - a) Create your job by defining your input file and source stage and adding services and subflows to your flow.
 - b) You can:
 - Connect a sink stage to the optional output port on the source stage in your flow. The optional port is the clear output port just beneath the black output port on your source stage. If you mouse over this port, you will see a tool tip that says, "error_port." Malformed records go to this sink.
 - Connect nothing to the optional output port on the source stage in your flow, ignoring all malformed records.
3. By default, processing stops at malformed records. This default behavior can be changed in your Advanced configuration options or in Management Console. Regardless of your system's default behavior, you can override the default behavior for a job by following these steps:
 - a) Open the job in Flow Designer.
 - b) Within an open job, go to **Edit > Job Options**.
 - c) Select either **Do not terminate the job on a malformed record** or select **Terminate the job after encountering this many malformed records** and enter the number of malformed records you will allow a job to encounter before terminating.

Control Stages

Use control stages to move data along different paths in a flow, to split or group records, and to perform basic data transforms and mathematical operations.

Note: In this technical preview, some stages cannot be configured in Flow Designer. You can configure your stages in Enterprise Designer, and expose the flows to Flow Designer for trial purposes.

Broadcaster

A Broadcaster takes a stream of records and splits it into multiple streams, allowing you to send records to multiple stages for simultaneous processing.

Broadcaster has no settings to configure.

Record Combiner

Record Combiner is a Control stage that uses one or more input ports to merge multiple records from multiple streams into a single survivor record based on a commonality.

For example, you have one group of records from input port number one and another group from input port number two. Record Combiner examines the groups and merges similar records into a single record.

Note: If you delete a middle stage, the ports do not renumber consecutively. Record Combiner will not release a record as output until each of its input ports has received a record to process. This stage attempts to combine as many records as it has input ports before releasing output records.

You can specify the port to preserve in cases where input streams have the same name. For example, if you combine records from two streams, and both streams contain a field named AccountNumber, you can specify the stream whose AccountNumber field prevails by choosing the input port that corresponds to the stream that you preserve. Flow Designer discards the data from the non-surviving AccountNumber field.

Record Joiner

Note: You cannot configure this stage in the technical preview version of Flow Designer.

Record Joiner performs a SQL-style `JOIN` operation to combine records from different streams based on a relationship between fields in the streams. You can use Record Joiner to join records from multiple files, multiple databases, or any upstream channels in the flow. You must connect at least two input channels to Record Joiner. The results of the `JOIN` operation are then written to one output channel. Optionally, records that do not match the join condition can be written to a separate output channel.

Note: Before using Record Joiner you should have a good understanding of the SQL `JOIN` operation. For more information, see [wikipedia.org/wiki/Join_\(SQL\)](https://wikipedia.org/wiki/Join_(SQL)).

Join Definition

Option	Description
Left port	<p>The port whose records you want to use as the left table in the <code>JOIN</code> operation. All other input ports will be used as right tables in the <code>JOIN</code> operation.</p> <p>Note: "Left" table and "right" table are SQL <code>JOIN</code> concepts. Before using Record Joiner you should have a good understanding of the SQL <code>JOIN</code> operation. For more information, see wikipedia.org/wiki/Join_(SQL).</p>

Option	Description
Join type	<p>The type of <code>JOIN</code> operation you want to perform. One of the following:</p> <p>Left Outer Returns all records from the left port even if there are no matches between the left port and the other ports. This option returns all records from the left port plus any records that match in any of the other ports.</p> <p>Full Returns all records from all ports.</p> <p>Inner Returns only those records that have a match between the left port and another port. For example, if you have four input sources and port 1 is the left port, an inner join will return records that have matching fields between port 1 and port 2, port 1 and port 3, and port 1 and port 4.</p>
Join Fields	<p>The field or fields from the left port that must match the data in a field from another port in order for the records to be joined.</p> <p>Note: The valid data types for join fields are integer, string, datetime, date, long, float, double, and big decimal.</p>
Data from the left port is sorted	<p>Specifies whether the records in the left port are already sorted by the field specified in Join Fields. If the records are already sorted, checking this box can improve performance. If you do not check this box, Record Joiner will sort the records according to the field specified in Join Fields before performing the join operation.</p> <p>If you have specified multiple join fields, then the records must be sorted using the order of the fields listed in Join Fields. For example, if you have two join fields:</p> <p style="padding-left: 40px;">Amount Region</p> <p>Then the records must be sorted first by the Amount field, then by the Region field.</p> <p>Important: If you select this option but the records are not sorted, you will get incorrect results from Record Joiner. Only select this option if you are sure that the records in the left port are already sorted.</p>

Option	Description
Join Definitions	<p data-bbox="535 325 1433 430">Describes the join conditions that will be used to determine if a record from the left port should be joined with a record from one of the other ports: <code>port1.Name = port2.Name</code></p> <p data-bbox="535 441 1433 504">This indicates that if the value in the Name field of a record from port1 matches the value in the Name field of a record from port2, the two records will be joined.</p> <p data-bbox="535 514 1433 703">To modify a join condition, click Modify. Select a field from the right port whose data must match the data in the join field from the left port in order for the records to be joined. If you want to change the left port field, click Cancel and change it in the Join Fields field. If the records in the right port are sorted by the join field, check the box Data from the right port is sorted. Checking this box can improve performance.</p> <p data-bbox="535 714 1433 800">Important: If you select Data from the right port is sorted but the records are not sorted, you will get incorrect results from Record Joiner. Only select this option if you are sure that the records in the right port are already sorted.</p>

Field Resolution

This tab specifies which port's data to use in the joined record in cases where the same field name exists in more than one input port. For example, if you are performing a join on two sources of data, and each source contains a field named DateOfBirth, you can specify which port's data to use in the DateOfBirth field in the joined record.

If there are fields of the same name but with different data, and you want to preserve both fields' data in the joined record, you must rename one of the fields before the data is sent to Record Joiner. You can use the Transformer stage to rename fields.

Handling Records That Are Not Joined

In order for a record to be included in the Record Joiner output it must meet the join condition, or a join type must be selected that returns both joined records and those that did not meet the join condition. For example, a full join will return all records from all input ports regardless of whether a record meets the join condition. In the case of a join type that does not return all records from all ports, such as a left outer join or an inner join, only records that match the join condition are included in the Record Joiner output.

To capture the records that are not included in the result of the join operation, use the **not_joined** output port. The output from this port contains all records that were not included in the regular output port.

Records that come out of this port have the field **InputPortIndex** added to them. This field contains the number of the Record Joiner input port where the record came from. This allows you to identify the source of the record.

Note:

- For optimal performance of this stage, ensure two independent streams of records are joined to generate a consolidated output.
- If a single path is first branched using either a broadcaster or conditional router then re-joined back using a Record Joiner, the flow may hang. In case multiple stages are used between branching and joining, use the Sorter as close to the Record Joiner as possible.

Math

Note: You cannot configure this stage in the technical preview version of Flow Designer.

The Math stage handles mathematical calculations on a single data row and allows you to conduct a variety of math functions using one or more expressions. Data is input as strings but the values must be numeric or Boolean, based on the type of operation being performed on the data.

1. Under Control Stages, click the Math stage and drag it to the canvas, placing it where you want on the flow.
2. Connect the stage to other stages on the canvas.
3. Double-click the Math stage. The **Math Options** dialog box appears, with the Expressions tab open. This view shows the input fields, the Calculator, and the Expressions canvas. Alternately, you can click the Functions tab to use functions instead of the Calculator.

The Input fields control lists the valid fields found on the input port. Field name syntax is very flexible but has some restrictions based on Groovy scripting rules. If you are not familiar with Groovy scripting, see this website for complete information about Groovy: groovy-lang.org.

Sorter

Use the Sorter stage to sort records using fields you specify.

You can configure these details for a Sorter stage:

- Order – Displays the sort fields in sort sequence
- Sort Field – Name of the sort field
- Direction – Sort direction: Ascending or Descending
- Sort Type – The type of content in the sort field, such as integer, string, and others
- Trim Blanks – Indicate whether to remove blank space before or after the field content
- Treat Nulls as – Select smallest or largest to direct the placement of null values in the sorted list
- Advanced – Advanced sort options that define overrides for sort performance. We suggest that you not change advanced sort performance options without consulting your system administrator.

How Flow Designer sorts data

Flow Designer applies the ASCII standard to sort fields, strings, and characters into ascending and descending order.

Ascending order

The smallest or first or earliest items appear at the top of the list.

- Numbers or amounts: Sorts smallest to largest; lower numbers or amounts are at the top of the list.
- Letters or strings: From A to Z
- Combination strings, such as address lines: 0-9 then A-Z

Dates: The oldest dates are at the top of the list.

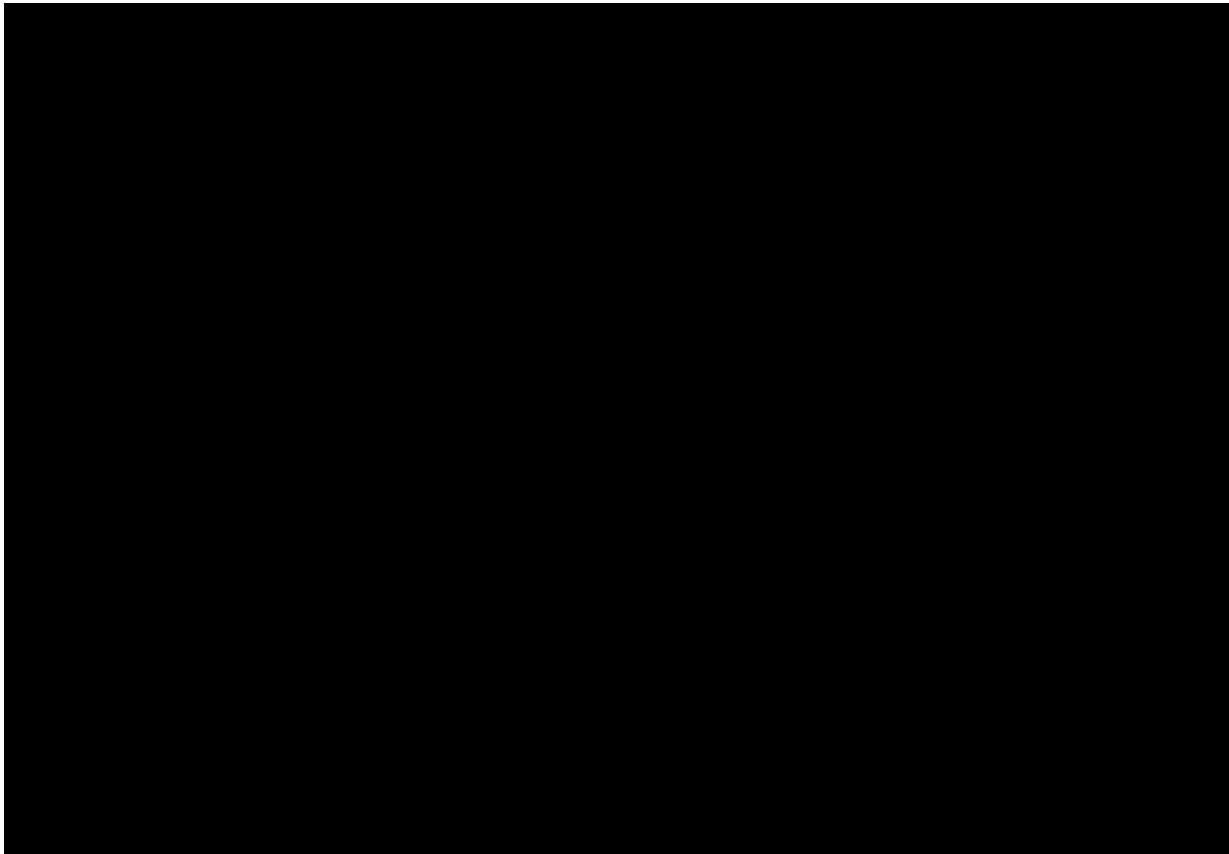
Descending order

The largest or last items appear at the top of the list:

- Numbers or amounts: Sorts largest to smallest. Higher numbers or amounts are at the top of the list.
- Letters or strings: From Z to A
- Combination strings, such as address lines: Z-A then 9-0
- Dates: Most recent dates are at the top of the list.

Sort order precedence

This chart (*source: ecisolutions.com*) shows characters processed in ascending order. Reverse the characters to determine the order for descending sorts.



Output stages (Sinks)

Output stages are also known as Sinks and Write-to stages.

To define the output from a flow, use a "sink" stage. A sink is the last stage in a flow. It defines what to do with the output from the flow. A sink can also perform other actions at the end of a flow, such as executing a program.

Note: This version of Flow Designer allows you to use input files and to write to output files that are on the Spectrum Server location, only. At this time, Flow Designer does not support *local* input and output files.

Output from a Job

Output from a job can be written to a file or a database. Spectrum™ Technology Platform has the ability to write data to many file formats and database types. The types of sinks you can write to depend on which modules you have licensed. See the solution guide for your modules available at support.pb.com/spectrum.

Output from a Service

Output data from a service is defined in an Output stage. This stage defines the fields that the service will return in response to a web service request or an API call.

Other output stages

Additional output Sink stages include:

- Terminate Job – Add to a flow to stop a job at a specified point or for a specific reason.
- Write to Null – Counts records, then discards the records that you choose to not keep according to flow processing criteria you specify. Use this stage if there are records that you do not want to preserve after the dataflow finishes.
- Write to XML – Sends output to an XML-format output file that can be consumed by other processes or flows.

These additional Sink stages are not configurable.

Output details

Output or "Write to" stages define the output fields that a flow returns.

When you specify an output file for your flow, Flow Designer examines the incoming records to determine as much as it can about the output information, populating the File tab/page. The information it returns includes:

- [Output] File Name
- Record type
 - Line Sequential - Records are separated by an end-of-line (EOL) character such as a carriage return or line feed (CR or LF) and each field has a fixed starting and ending character position.
 - Fixed Width - Each record is a specific number of characters in length and each field has a fixed starting and ending character position.
 - Delimited - Records are separated by an end-of-line (EOL) character such as a carriage return or line feed (CR or LF), and each field is separated by a designated character such as a comma.
- Character Encoding - For more information, see [Supported character encoding methods](#) on page 73.
- Field Separator - For more information, see [Field separators](#) on page 73 for a list of valid characters.
- Text Qualifier - Use single or double quotes.
- Record Separator - For more information, see [Record separators](#) on page 74 for a list of valid formats.

The Write to File stage configuration has three tabs: File, Sort, and Runtime.

File tab

The file tab defines relevant content and format information for your output.

Using the File tab, you can:

- Display details about the output file's format and layout.
- Define the presence of a header record.
- Mark records with fewer fields than defined as "malformed."
- Use the convenient File Schema workspace to change record details in the user interface.

Sort tab

The Sort tab describes a flow's sort key fields and sort order.

Using the Sort tab, you can:

- Accept, delete, or change the sort fields used.
- Accept or change the order of the sort fields.
- Accept or change the sort direction for any sort field: ascending or descending.

Runtime tab

The Runtime tab defines a flow's input file processing options.

With the Runtime tab, you can:

- Define the starting record to process.
- Define whether to process all input records or specify a maximum number of records to process.

3 - Using Flow Designer - Building, testing, and running flows

In this section

Designing and building flows	37
Deleting flows	65
Exposing flows	66
Importing flows	66
Exporting flows	67
Running an External Program	69
Using the template browser to create flows and jobs	70

Designing and building flows

Configuring input stages

Configuring the Read from File Stage

Follow these steps to configure the Read from File stage.

1. Open an existing flow or start with a blank workspace.
2. Select an Input stage from the palette and drag the stage to the canvas.
3. Double-click the input stage to open the configuration view. Note that the input configuration has three tabs: **File**, **Sort**, and **Runtime**.
4. Select an input file for this stage.

You are ready to [Configure the File tab details - Read from File](#) on page 37.

Configure the File tab details - Read from File

1. At the **File Name** field, click the file name button to display the **Choose File** selection dialog. Flow Designer displays a list of available files at the default location.
2. To select an input file, you can:
 - Go to another location using the navigation character (<).
 - Filter the list of files using a letter or string then clicking the filter button.
 - Click a to select a file name in the scroll box, and click **OK** to confirm your choice.

Flow Designer returns to the Read From File configuration page File tab, displaying your input file in the File **Name** field.

3. Review the populated File tab fields to ensure correctness and make adjustments if needed.
4. 5. Check the boxes that provide more detail on the file content:
 - First row is header record – Toggle to **YES** if the first row in the input file is a header record.
 - Treat records with fewer fields than defined as malformed – Toggle to **YES** to implement this global record handling option. You can learn more about handling malformed records here: [Managing malformed input records](#) on page 26.
5. Use the **File Schema** preview workspace to:
 - a) Change the order of the input fields:
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.

3. Release the mouse button when the new position is correct.
 4. Click **Regenerate** to save the change.
- b) Rename a field – Double-click a field name to open a text input box, allowing you to change the name.
 - c) Change the field content type – Double-click the **Type** column for any field to display the drop-down, and select a valid format for the field content.
 - d) Selectively trim fields to their used length.
 - e) Selectively define a field's locale or format – If available, double-click the **Locale/Format** column for any field, and change the definition.
 - f) Selectively delete fields – Click the trash can: You will see a confirmation message before you can proceed.

Tip: Use the **Regenerate** button to re-examine and re-display the file schema after you make changes.


6. Click **Apply** so that your changes take effect, or click **Cancel** to return to the workflow canvas without saving changes.

You are ready to [Configure the Sort tab details - Read from File](#) on page 38.

[Configure the Sort tab details - Read from File](#)

The default view for the Sort tab assumes that your flow's sort order will copy the order in which those fields appear in the input file. You can change the defaults on this page.

Tip: You can click **Cancel** to abandon changes at any time.

1. Change the order of the sort fields.
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Release the mouse button when the new position is correct.
 4. Click **Regenerate** to save the change.
2. Change the sort fields used.
 1. Double-click the name of any sort field. Alternatively, click the field name once to select it and click the drop-down arrow to show a list of available sort fields.
 2. Select a new field from the list.
 3. The new field name will appear indented until you click **Apply** to save your changes.
3. Delete a sort field: Click the trash can button  next to any field to delete it from the list.
4. Define advanced sort options.

Important: We suggest that you not change advanced sort performance options without consulting your system administrator. If you change these options, you may negatively affect sort performance.

1. Click **Advanced**.

2. Use the **Override sort performance options** toggle to change your sort memory and temporary file settings.
3. Use the **In memory record limit** field to define the caching and memory to keep before writing to disk.
4. Define the **Maximum number of temporary files** to use for sorting.
5. Use the **Compression** toggle to change your option for compressing files stored on disk.
6. Click **OK** to save your changes.

You are ready to [Configure the Sort tab details - Read from File](#) on page 38.

[Configure the Runtime tab details - Read from File](#)

Use the Runtime tab to define record processing options for your flow.

1. Set the first record to process.
 - Use the **Starting Record** up and down arrows to select a record other than record 1 as the first record to process. You can use the arrows to set the starting record back to 1.
 - Alternatively, you can click the **Starting Record** field and enter the starting record number.
2. Define the number of records to process.
 - Select **All** records to process all input records (this is the default).
 - Select **Max Records** and define a maximum number of input records to process. Use the arrows to select a number or type a value in the entry field.
3. Click **Apply** to save your definitions.

you can now configure a Sorter stage if your flow is using one.

[Configuring the Input stage](#)

Follow these steps to configure the Input stage.

1. Open an existing flow or start with a blank workspace.
2. Select an Input stage from the palette and drag the stage to the canvas.
3. Double-click the input stage to open the configuration view. Note that the input configuration has three tabs: **File**, **Sort**, and **Runtime**.
4. Select an input file for this stage.

You are ready to [Configure the File tab details - Read from File](#) on page 37.

[Configure the File tab details - Input stage](#)

1. At the **File Name** field, click the file name button to display the **Choose File** selection dialog. Flow Designer displays a list of available files at the default location.
2. To select an input file, you can:
 - Go to another location using the navigation character (<).
 - Filter the list of files using a letter or string then clicking the filter button.

- Click a to select a file name in the scroll box, and click **OK** to confirm your choice.

Flow Designer returns to the Read From File configuration page File tab, displaying your input file in the File **Name** field.

3. Review the populated File tab fields to ensure correctness and make adjustments if needed.
4. 5. Check the boxes that provide more detail on the file content:
 - First row is header record – Toggle to **YES** if the first row in the input file is a header record.
 - Treat records with fewer fields than defined as malformed – Toggle to **YES** to implement this global record handling option. You can learn more about handling malformed records here: [Managing malformed input records](#) on page 26.
5. Use the **File Schema** preview workspace to:
 - a) Change the order of the input fields:
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Release the mouse button when the new position is correct.
 4. Click **Regenerate** to save the change.
 - b) Rename a field – Double-click a field name to open a text input box, allowing you to change the name.
 - c) Change the field content type – Double-click the **Type** column for any field to display the drop-down, and select a valid format for the field content.
 - d) Selectively trim fields to their used length.
 - e) Selectively define a field's locale or format – If available, double-click the **Locale/Format** column for any field, and change the definition.
 - f) Selectively delete fields – Click the trash can: You will see a confirmation message before you can proceed.

Tip: Use the **Regenerate** button to re-examine and re-display the file schema after you make changes.
6. Click **Apply** so that your changes take effect, or click **Cancel** to return to the workflow canvas without saving changes.


You are ready to [Configure the Sort tab details - Input stage](#) on page 40.

[Configure the Sort tab details - Input stage](#)

The default view for the Sort tab assumes that your flow's sort order will copy the order in which those fields appear in the input file. You can change the defaults on this page.

Tip: You can click **Cancel** to abandon changes at any time.

1. Change the order of the sort fields.
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Release the mouse button when the new position is correct.

4. Click **Regenerate** to save the change.
2. Change the sort fields used.
 1. Double-click the name of any sort field. Alternatively, click the field name once to select it and click the drop-down arrow to show a list of available sort fields.
 2. Select a new field from the list.
 3. The new field name will appear indented until you click **Apply** to save your changes.
3. Delete a sort field: Click the trash can button  next to any field to delete it from the list.
4. Define advanced sort options.

Important: We suggest that you not change advanced sort performance options without consulting your system administrator. If you change these options, you may negatively affect sort performance.

1. Click **Advanced**.
2. Use the **Override sort performance options** toggle to change your sort memory and temporary file settings.
3. Use the **In memory record limit** field to define the caching and memory to keep before writing to disk.
4. Define the **Maximum number of temporary files** to use for sorting.
5. Use the **Compression** toggle to change your option for compressing files stored on disk.
6. Click **OK** to save your changes.

You are ready to configure the **Runtime tab** on page 26 details.

Configure the Runtime tab details - Input stage

1. Set the first record to process.
 - Use the **Starting Record** up and down arrows to select a record other than record 1 as the first record to process. You can use the arrows to set the starting record back to 1.
 - Alternatively, you can click the **Starting Record** field and enter the starting record number.
2. Define the number of records to process.
 - Select **All** records to process all input records (this is the default).
 - Select **Max Records** and define a maximum number of input records to process. Use the arrows to select a number or type a value in the entry field.
3. Click **Apply** to save your definitions.

You can now configure a Sorter stage if your flow is using one.

Configuring the Read from XML Stage for Subflows

Important: We suggest that you not change advanced sort performance options without consulting your system administrator. If you change these options, you may negatively affect sort performance.

1. Open an existing flow or start with a blank workspace.
2. Select a **Read from XML** stage from the palette and drag the stage to the canvas.
3. Double-click the stage to open the configuration view.
4. Use the **Override system-wide default options** toggle so that options defined for this service will not affect any other services.
5. Click **OK** to save your changes.

Configuring Control stages

Creating and configuring data transformations - Transformer stage

Flow Designer functions provide the building blocks that help you to process, change, and convert data for your sites's specific requirements. These custom transformations are configurable through the Transformer stage.

Note: This section provides a preview of the transforms available in Flow Designer. Note that only *custom* transforms are supported in this release of Flow Designer. Additionally, this preview version does not support multiple custom transforms. You can create Transforms in Enterprise Designer to use in this version of Flow Designer. For more information, see [Creating a Custom Transform](#) in the *Dataflow Designer Guide*.

The Transformer stage has predefined transforms that perform a variety of common data transformations. If the predefined transforms do not meet your needs you can write a custom transform script using Groovy scripting. For more information about Apache Groovy, see groovy-lang.org.

[Transformer Options page](#)

Use the Transformer stage to perform common data transformations.

When you add a Transformer stage to your flow and double-click the stage icon, you will see the Transformer Options page. The Transformer Options page is a workspace that allows you to construct transforms from a variety of functions that come with Flow Designer. This page consists of a palette on the left side, a scripting workspace, and **Apply** and **Cancel** buttons.

Table 3: Fields and settings on the Transformer Options page

Field or setting	Description	Options
------------------	-------------	---------

Field and function display	This drop-down, at the top left of the page, displays all field and function options, or filters the display based on your selection.	<p>Select the fields and functions to display:</p> <ul style="list-style-type: none"> • All (no filtering) - Display all fields and functions • Input - Display all available input fields for the current input file • Output - Display all Output fields resulting from the current flow • Function - Display all available processing functions based on flow input and output
Filter entry field	<p>For the selection in the Field and function display, apply a filter to narrow the fields or functions displayed.</p> <p>Click X in this field to clear the filtering and display all items in the category you selected in the Field and function display.</p>	<p>Enter a character or string to use for filtering the display.</p> <p>For example, if you chose Input in the Field and function display, then type "addr" in the filter entry field, the display might show "AddressLine1" and "AddressLine2".</p>
Input fields	Use the arrow to expand and collapse the list of input fields available to process using a custom transform.	<ul style="list-style-type: none"> • Name - Field name • Type - Field data type • List - Select a field to use in a List transform.
Output fields	<p>Use the arrow to expand and collapse the list of output fields to process using a custom transform.</p> <p>To add a field:</p> <ol style="list-style-type: none"> 1. Click NEW to display a new row and input field. 2. Add the new field name; for example, "OutputAddr1". 3. Click the content of the Type field to apply the field format. 4. Check the List selection box to include this field in output transform processing. 	<ul style="list-style-type: none"> • Name - Field name • Type - Field data type • List - Select a field to use in a List transform. <p>You can click the trash can next to any output field to delete it.</p>

Functions	<p>Use the arrow to expand and collapse the list of functions. You can expand any function parent category to see the functions available under it.</p> <p>Hover over any function in an expanded category to see a description of the function, a list of applicable parameters, and a description of the data returned.</p> <p>For any selected field, double-click a function to add it to the scripting workspace.</p>	<p>The function categories are:</p> <ul style="list-style-type: none"> • DateTime - Processes data to derive a DATETIME object using INT64 values representing the year, month, day, hour, minute, and second. • Math - Performs mathematical/trigonometric calculations. • String - Performs processing on a defined data string producing a non-numerical result. • Miscellaneous - Includes random, roundTo, truncate, and truncateTo functions.
Scripting workspace	This is the area where you will construct your data transformation scenarios.	For any selected field, double-click a function to add it to the scripting workspace.
Apply and Cancel buttons	Save or discard your data transformation.	<ul style="list-style-type: none"> • Apply - Apply the function/transformation definition to the selected field. • Cancel - Discard the transformation scenario for the selected field.

Creating a Custom Transform

The Transformer stage has predefined transforms that perform multiple common data transformations. If the predefined transforms do not meet your needs, you can write a custom transform script using Groovy.

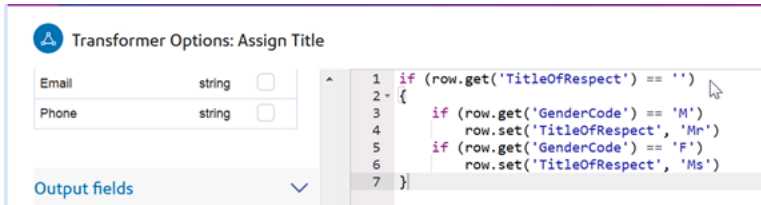
This procedure describes how to create basic custom transforms using Groovy. For complete documentation on Groovy, see groovy-lang.org.

Note: Review this topic to learn how to select fields for processing: [Creating a Custom Transform](#).

1. In Flow Designer, add a Transformer stage to the flow.
2. Double-click the Transformer stage to display the **Transformer Options** page.
3. Using the palette on the left side of the page:
 - a) Select one or more **Input fields** to which you will apply the transform.
 - b) In the **Output fields** field, specify the field to which you want to write the output from the transform. If necessary, you can define a new field: by clicking **NEW** in the field list.
 1. Click NEW in the field list.
 2. Type the field name in the text entry field, then use the drop down to select the field's format.
 3. Click away from the new field name to add it to the list of output fields.

- Click in the **Script Editor** workspace.

If you are working with a template, you may see scripting in the editor window:



The editor provides an easy-to-use workspace to develop your custom transforms, which includes type-ahead and selections. As you type in the workspace, Flow Designer provides applicable parameters based on your entries.

- Use this reference as a guideline for constructing your custom transformation scenarios.

Task	Instructions
------	--------------

To add a function	In the Functions pane, double-click the function you want to add.
--------------------------	--

Note: The functions listed in the editor are functions provided to make writing custom transform scripts easier. They perform functions that would otherwise require multiple lines of Groovy code to accomplish. They are not standard Groovy functions.

To get the value from a dataflow field	In the Input Fields pane, double-click the input field you want. The following will be added to your script:
---	---

```
data['FieldName']
```

For example, if you want to get the value from the field CurrentBalance, the following would be added:

```
data['CurrentBalance']
```

To set the value of a dataflow field	Enter this code in the script editor:
---	---------------------------------------

```
data['FieldName']=NewValue
```

For example, to set the field Day to the day of the week contained in the field PurchaseDate:

```
data['Day']=dayOfWeek(data['PurchaseDate'])
```

Task	Instructions
------	--------------

In this example, the function `dayOfWeek()` is used to get the day from the date value in the `PurchaseDate` field, and the result is written to the `Day` field.

Tip: You can double-click the name of the output field in the **Output Fields** pane to add the field reference to the script.

To change the scope of a script variable in a dataflow

To change the scope of a script variable from a single input record to all the input records in a dataflow, use the `@Field` annotation in your script as shown:

```
import groovy.transform.Field;
@Field ['data type']['VariableName']= Value;
```

For example, to set the scope of a variable `RecordNumber` to a single input record, specify this:

```
int recordNumber = 1;
data['Record_Number']= recordNumber;
recordNumber++;
```

To change the scope of this variable to all input records, specify this:

```
import groovy.transform.Field
@Field int recordNumber = 1;
data['Record_Number']= recordNumber;
recordNumber++;
```

To create a new field using a numeric data type

Enter this code in the script editor:

```
data['FieldName'] = new constructor;
```

Where `constructor` is one of these:

java.lang.Double(*number*)

Creates a field with a data type of Double.

java.lang.Float(*number*)

Creates a field with a data type of Float.

java.lang.Integer(*number*)

Creates a field with a data type of Integer. You can also create a new integer field by specifying a whole number. For example, this will create an integer field with a value of 23:

```
data['MyNewField'] = 23;
```

Task	Instructions
------	--------------

java.lang.Long(*number*)

Creates a field with a data type of Long.

For example, to create a new field named "Transactions" with a data type of Double and the value 23.10, you would specify the following:

```
data['Transactions'] = new com.java.lang.Double(23.10);
```

To create a new field using a date or time data type Enter this code in the script editor:

```
data['FieldName'] = new constructor;
```

Where *constructor* is one of these:

com.pb.spectrum.api.datetime.Date(*year,month,day*)

Creates a field with a data type of date. For example, December 23, 2013 would be:

```
2013,12,23
```

com.pb.spectrum.api.datetime.Time(*hour,minute,second*)

Creates a field with a data type of time. For example, 4:15 PM would be:

```
16,15,0
```

com.pb.spectrum.api.datetime.DateTime(*year,month,day,hour,minute,second*)

Creates a field with a data type of DateTime. For example, 4:15 PM on December 23, 2013 would be:

```
2013,12,23,16,15,0
```

For example, to create a new field named "TransactionDate" with a data type of Date and the value December 23, 2013, you would specify this:

```
data['TransactionDate'] = new
com.pb.spectrum.api.datetime.Date(2013,12,23);
```

Task	Instructions
------	--------------

To create a new field with a data type of Boolean	<p>Enter this code in the script editor:</p> <pre>data['FieldName'] = true or false;</pre> <p>For example, to create a field named <code>IsValidated</code> and set it to <code>false</code>, you would specify this:</p> <pre>data['IsValidated'] = false;</pre>
To create a new list field	<p>Use the <code>factory.create()</code> method to create new fields in a record then use the leftShift operator <code><<</code> to append the new record to the list field.</p> <pre>NewListField = [] NewRecord = factory.create() NewRecord['NewField1'] = "Value" NewRecord['NewField12'] = "Value" ... NewListField << NewRecord NewRecord = factory.create() NewRecord['NewField1'] = "Value" NewRecord['NewField12'] = "Value" ... NewListField << NewRecord data['ListOfRecords'] = NewListField</pre> <p>For example, this creates a new list field called "addresses" consisting of two "address" records.</p> <pre>addresses = [] address = factory.create() address['AddressLine1'] = "123 Main St" address['PostalCode'] = "12345" addresses << address address = factory.create() address['AddressLine1'] = "PO Box 350" address['PostalCode'] = "02134" addresses << address data['Addresses'] = addresses</pre>

Task Instructions

You can also create a new list field that contains a list of individual fields rather than a list of records. For example, this creates a new list field called PhoneNumbers containing home and work phone numbers:

```
phoneNumbers = []
phoneNumbers << data['HomePhone']
phoneNumbers << data['WorkPhone']
data['PhoneNumbers'] = phoneNumbers
```

To concatenate fields Use the + symbol. For example, this example concatenates the FirstName field and the LastName field into a value and stores it in the FullName field

```
String fullname = data['FirstName'] + ' ' + data['LastName'];
data['FullName']=fullname;
```

In this example there are two input fields (AddressLine1 and AddressLine2) which are concatenated and written to the output field Address.

```
address1 = data['AddressLine1'];
address2 = data['AddressLine2'];
data['Address']=address1+ ',' + address2;
```

To parse a field Identify a separation character then use `substring` to parse the field. In this example, if the PostalCode field is greater than five characters, it separates the five-character ZIP Code and the +4 portion and writes them to separate fields in the output record.

```
if (data['PostalCode'].length() > 5)
{
    String postalCode = data['PostalCode'];
    int separatorPosition = postalCode.indexOf('-');
    String zip = postalCode.substring(0, separatorPosition);
    String plusFour = postalCode.substring(
        separatorPosition + 1,
        postalCode.length());
    data['Zip']=zip;
    data['PlusFour']=plusFour;
}
```

Task	Instructions
To perform conditional processing	<p>Use an <code>if</code> or <code>switch</code> statement. These are the most common conditional processing constructs. For more information see groovy-lang.org.</p> <p>This example sets the field <code>AddressCity</code> to the first address line and city name if the city is Austin.</p> <pre>city = data['City']; address1 = data['AddressLine1'] if(city.equals('Austin')) data['AddressCity']=address1 + ',' + city;</pre>
To perform looping	<p>Use the <code>for</code> loop. This is the only looping construct you should need. For more information about looping or syntax see groovy-lang.org.</p>
To augment data	<p>Define a constant and use the concatenation character <code>+</code>. For example, this script appends the word "Incorporated" to the <code>FirmName</code> field.</p> <pre>firmname = data['FirmName']; constant = 'Incorporated'; if(firmname.length() > 0) data['FirmName']=firmname + ' ' + constant;</pre>
To access an option specified at runtime	<p>If the dataflow has runtime options enabled, you can access settings passed to the dataflow at runtime by using this syntax:</p> <pre>options.get("optionName")</pre> <p>For example, to access an option named <code>casing</code>, you would include this in your custom transform script:</p> <pre>options.get("casing")</pre>

- After you are done entering your script, click **Apply** to save your work.

Channel Mapping






This section describes the preview version of the channel mapping feature.

Channel mapping is the process of designating which data to move from an input source field into an output target field. With Flow Designer, you can:

- Map fields
- View field mapping details
- Un-map fields

Flow Designer indicates map and selection status in these ways:

Table 4: Channel and map status

Description	What it shows
If a channel contains no mapping, it appears as a solid gray line if it is not selected.	
If a channel is selected, it changes to a solid blue line.	
If a channel contains mapping, but is not selected, it appears as a gray line containing a dot to show that mapping applies to that channel.	
If a selected channel contains mapping, it appears as a solid blue line with a dot.	
Hover on any channel to see the port names on each endpoint.	

View mapping details

Flow Designer displays mapping controls and status on the Mapping details page.

Double-click the mapping indicator (dot) in the channel connection



to display this page.

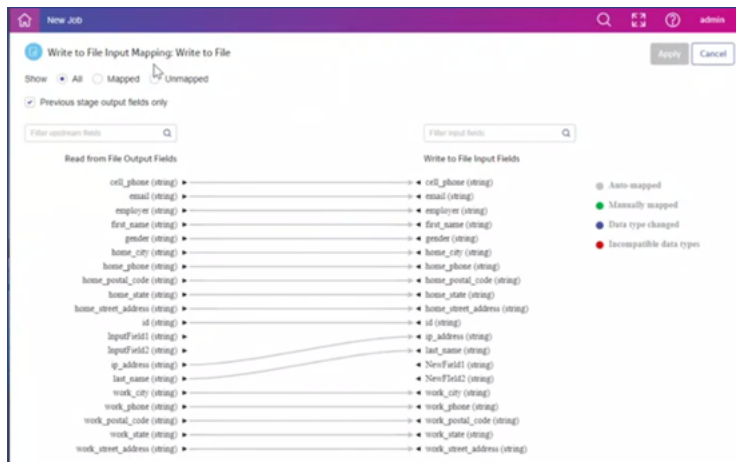


Table 5: Contents of the Mapping Detail page

Entity	What it does
Show options	Filters the fields displayed: <ul style="list-style-type: none"> • All (default) • Mapped only • Unmapped only
Previous stage output fields only	Controls whether the output field list contains only fields from the previous stage (default) or all upstream fields
Filter input field names	Depending on the display you selected, shows a filter field that displays names based on a string you enter here
[stage name] Output fields	Lists the output fields upstream of this channel
[stage name] Input fields	Lists the input fields of the next stage in the dataflow

Entity

What it does

Mapping type

The channel color, indicated by the legend on the right side of the page:

- Auto-mapped
- Manually mapped
- Data type changed
- Incompatible data types

- Auto-mapped fields – Fields that Flow Designer could map between output and input
- Manually mapped fields – Fields that you mapped for this flow
- Data type changed – Shows mapped fields that require some data conversion between output and input
- Incompatible data types – Fields that have incompatible data types: This will cause an error during dataflow execution

Mapping hierarchy

It is essential to understand how mapping hierarchy works in Flow Designer, since multiple mapping types may apply to any mapped field.

In cases where more than one mapping scenario applies to a mapped field, Flow Designer uses a color-coded hierarchy to represent the mapping type. This precedence, shown in a legend on the Mapping details page, shows the hierarchy in ascending order:

- Auto-mapped
- Manually mapped
- Data type changed
- Incompatible data types

For example, if a manual mapping (green) also contains a data type change (blue), the channel between the mapped fields will show as blue, since blue takes a higher precedence than green.

Manually map fields

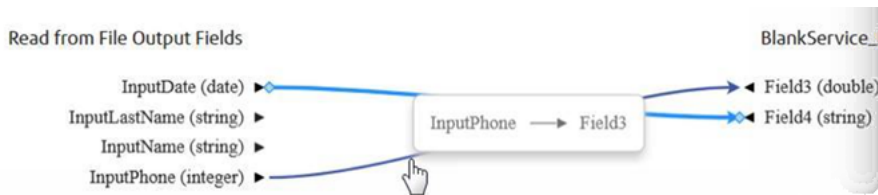
Flow Designer auto-maps fields, by default, based on its investigation of applicable fields.

You have the option to manually map fields to customize your output format and content.

1. With the Input Mapping page open, left-click the field name to map on the left/source field side.
2. Hold down the left mouse button and a line appears. Drag this channel (line) to the right-side target field to connect.
3. If the data types are compatible, Flow Designer displays a green line to show the manual field map you defined.



For successfully mapped fields, with data type conversions, you can inspect the conversion details from a tooltip. Hover on any mapping channel to see the mapping conversion tooltip.



If the data types are incompatible, the channel is red.



- If the channel is red, You must un-map or remap to a compatible field.

Note: For more information data type conversions, see [Defining Data Type conversion for mapped fields](#) on page 54.

- Once you have correctly mapped your fields, click **Apply** to save your changes.

Change (re-map) fields

To re-map an auto-mapped field:

- Click the channel between two auto-mapped fields. Flow Designer changes the channel to blue and adds a diamond handle on the channel, showing that the field can be remapped.
- Drag the arrow to the new field, and release the mouse.

Un-map fields

Click once on any map channel (connector line), and press **DELETE**.

Defining Data Type conversion for mapped fields

Spectrum™ Technology Platform automatically changes field data types as needed using the type conversion settings specified in Management Console, or through the dataflow type conversion options specified in Flow Designer.

In most situations you do not need to manually change field data types because any necessary data type conversions are handled automatically. However, in cases where a stage is unable to convert incoming data to the necessary data type, you may need to manually change the data type in the upstream channel.

There are a limited number of type conversions that you can perform manually:

- Polygon and MultiPolygon types can be converted to and from a geometry type.
- Date, time, and datetime data types can be converted to and from a string type.

To manually change a field's data type, follow this procedure.

- In Flow Designer, double-click the channel where you want to change the field's data type. A channel is the line that connects two stages on the canvas.

If type conversion options are configurable for the current fields, you will see the Data Type Conversions dialog.

Note: Only the appropriate options will be displayed depending on the data type chosen.

2. Specify the formats that you want to use for date and time data that is converted to a string. When the data or time is converted to a string, the string will be in the format you specify here.
 - a) In the **Locale** field, select the country whose format you want to use for dates converted to a string. Your selection will determine the default values in the **Date**, **Time**, and **DateTime** fields. Your selection will also determine the language used when a month is spelled out. For example, if you specify English the first month of the year would be "January" but if you specify French it would be "Janvier."
 - b) In the **Date** field, select the format to use for date data when it is converted to a string. A list of the most commonly used formats for the selected locale is provided.

For example, if you choose the format **M/D/YY** and a date field contains 2020-3-2, that date data would be converted to the string `3/2/20`.
 - c) In the **Time** field, select the format to use for time data when it is converted to a string. A list of the most commonly used formats for the selected locale is provided.

For example, if you choose the format **h:mm a** and a time field contains 23:00, that time data would be converted to the string `11:00 PM`.
 - d) In the **DateTime** field, select the format to use for fields containing the DateTime data type when converted to a string. A list of the most commonly used formats for the selected locale is provided.

For example, if you choose the format **M/d/yy h:mm a** and a DateTime field contains 2020-3-2 23:00, that DateTime data would be converted to the string `3/2/20 11:00 PM`.
 - e) In the **Whole numbers** field, select the formatting you want to use for whole numbers (data types float and double).

For example, if you choose the format **#,###** then the number 4324 would be formatted as `4,324`.

Note: If you leave this field blank, numbers will be formatted in the same way they were in Spectrum™ Technology Platform 8.0 and earlier. Specifically, no thousands separator is used, the dot (".") is used as the decimal separator, numbers less than 10^{-3} or greater than or equal to 10^7 are shown in scientific notation, and negative numbers have a minus sign ("-") in front of them. Also note that if you leave this field blank, numbers that use the bigdecimal data type will always be in the format **#,###.000**.
 - f) In the **Decimal numbers** field, select the formatting you want to use for numbers that contain a decimal value (data types integer and long).

For example, if you choose the format **#,##0.0#** then the number 4324.25 would be formatted as 4,324.25.

Note: If you leave this field blank, numbers will be formatted in the same way they were in Spectrum™ Technology Platform 8.0 and earlier. Specifically, no thousands separator is used, the dot (".") is used as the decimal separator, numbers less than 10^{-3} or greater than or equal to 10^7 are shown in scientific notation, and negative numbers have a minus sign ("-") in front of them. Also note that if you leave this field blank, numbers that use the bigdecimal data type will always be in the format **#,###.000**.

You can also specify your own date, time, and number formats if the ones available for selection do not meet your needs. To specify your own date or time format, type the format into the field using the notation described in [Date and time patterns](#) on page 88. To specify your own number format, type the format into the file using the notation described in [Number Patterns](#).

3. Click **OK**.
The color of the data type name changes to green.
4. Click **OK** again to save the change.

Configuring Record Combiner stage

Configuring the Sorter stage

These configuration steps apply to the Technical Preview version of Flow Designer. We will update the documentation for this stage when the full version of Flow Designer is available.

You must configure an input file before you can apply a sorter stage.

1. Configure the input stages for your flow.
2. Locate the **Sorter** stage in the Stage Palette under Control Stages.
3. Click the **Sorter** stage and drag it to the canvas.
4. Connect the **Input** file and **Sorter** stages with a channel.
5. Double-click the **Sorter** stage to display the Sorter Options configuration page.
6. If not sort fields are listed, click Select to add a sort field to the sort options list.

Change order of sort fields

1. Change the order of the sort fields.
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Release the mouse button when the new position is correct.
 4. Click **Regenerate** to save the change.
2. Change the sort fields used.

1. Double-click the name of any sort field. Alternatively, click the field name once to select it and click the drop-down arrow to show a list of available sort fields.
 2. Select a new field from the list.
 3. The new field name will appear indented until you click **Apply** to save your changes.
3. Change the sort order for a field.
 - a) Double-click the sort direction for any sort field. Alternatively, click the Direction definition once to select it and click the drop-down arrow.
 - b) Select an option from the list.
 - c) The selection appears indented until you click **Apply** to save your changes.
 4. Add and configure a sort field.
 - a) Click **Select**, which appears below the list of sort fields. You will see a list of available fields.
 - b) Select a field to add to the list.
 - c) Configure the sort direction: Ascending or Descending order.

Note: To learn more about how Flow Designer sorts data, review [How Flow Designer sorts data](#) on page 32

- d) Click the **Sort Type** column to select the field's data type. This column is disabled if incoming data is not in string format (see [Data Types](#) on page 10 for more information).
- e) To remove blank space from before and after the value before sorting, check the box in the Trim column.

Note: The trim option does not modify the value of the field. It only trims the value for sorting. Note that if your incoming data is not in string format, the Trim column is disabled.

- f) Use the **Treat Null As** column define the placement of null values in the sorted list: largest or smallest. Placement depends on the combination of options selected in the **Order** and **Treat Null As** fields:

Order	Treat Null As	Placement of null values in sorted list
Ascending	Largest	Bottom of list
Ascending	Smallest	Top of list
Descending	Largest	Top of list
Descending	Smallest	Bottom of list

5. Repeat the above steps to add sort fields.

6. See Step 1 to rearrange the sort order. change the order to sort first by one field, then sort the resulting order using another field.
7. Click **Apply** to save your changes.
8. To delete a sort field, click the trash can button next to any field to delete it from the list.
9. Define advanced sort options.

Important: We suggest that you not change advanced sort performance options without consulting your system administrator. If you change these options, you may negatively affect sort performance.

1. Click **Advanced**.
2. Use the **Override sort performance options** toggle to change your sort memory and temporary file settings.
3. Use the **In memory record limit** field to define the caching and memory to keep before writing to disk.
4. Define the **Maximum number of temporary files** to use for sorting.
5. Use the **Compression** toggle to change your option for compressing files stored on disk.
6. Click **OK** to save your changes.

Configuring output stages

Configuring the Output/Write to File stage

1. From the Stage palette, drag a Write to File (output) stage to the canvas.
2. Double-click the output stage to open the configuration view.

Configure the File tab to define the output file details.

Configure the File tab details - Write to File

Use the File tab to define details about your processing output.

1. At the **File Name** field, click the file name button to display the **Save File** selection dialog. Flow Designer displays a list of available files at the default location.
You can select an existing file to overwrite or define a new output file.
2. To define a new output file, enter the file name in the **File Name** field.

To select an existing file:

- a) Go to another location using the navigation character (<).
- b) Filter the list of files using a letter or string, then click the **Filter** button.
- c) Click to select a file name in the scroll box, and click **OK** to confirm your choice.

Flow Designer returns to the Write to File configuration page "File" tab, displaying your output file in the **File Name** field.

3. Review the populated File tab fields to ensure correctness and make adjustments if needed.
4. Select the options that provide more detail on the file content:
 - **First row is header record** – Check if the first row in the output file is a header record.
 - **Treat records with fewer fields than defined as malformed** – check to implement this global record handling option. You can learn more about handling malformed records here: [Managing malformed input records](#) on page 26.
5. Use the File Schema preview workspace to:
 - a) Change the order of the output fields:
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Click **Regenerate** to save the change.
 - b) **Rename a field** – Double-click a field name to open a text input box, allowing you to change the field's name.
 - c) **Change the field content type** (format) – Double-click the **Type** column for any field to display the drop-down, and select a valid format for the field content.
 - d) **Selectively trim fields to their used length** - Eliminate unused space in any field.
 - e) **Add a field** – Click **Add** in the File schema view or click the **Quick Add** button above the table. Enter the new field name and click away from the row. You will see the new field name and can make adjustments to it.
 - f) **Selectively delete fields** – Click the trash can: You will see a confirmation message before you can proceed.

Note: Click the **Regenerate** button to re-examine and re-display the file schema after you make changes.
6. Click **Apply** so that your changes take effect, or click **Cancel** to return to the workflow canvas without saving changes.

You are ready to configure the Sort Tab details.


Configure the Sort tab details - Write to File

The default view for the Sort tab assumes that your flow's sort order will copy the order in which those fields appear in the output file. You can change the defaults on this page.

Tip: You can click **Cancel** to abandon changes at any time.

1. Change the order of the sort fields.
 1. Click any gray bar in front of a field name; the pointer changes to a hand.
 2. Hold down the mouse button and drag the field selection to its new position.
 3. Release the mouse button when the new position is correct.
 4. Click **Regenerate** to save the change.

Note: Flow Designer sorts data in ascending or descending order, according to the [ASCII sort order standard](#).

2. Change the sort fields used.
 - a) Double-click the name of any sort field. Alternatively, click the field name once to select it and click the drop-down arrow to show a list of available sort fields.
 - b) Select a new field from the list.
 - c) The new field name will appear indented until you click **Apply** to save your changes.
3. Change the sort order for a field.
 - a) Double-click the sort direction for any sort field. Alternatively, click the **Direction** definition once to select it and click the drop-down arrow.
 - b) Select an option from the list.
 - c) The selection appears indented until you click **Apply** to save your changes.
4. Delete a sort field: Click the trash can button  next to any field to delete it from the list.
5. Define advanced sort options.

Important: We suggest that you not change advanced sort performance options without consulting your system administrator. If you change these options, you may negatively affect sort performance.

- a) Click **Advanced**.
- b) Use the **Override sort performance options** toggle to change your sort memory and temporary file settings.
- c) Use the **In memory record limit** field to define the caching and memory to keep before writing to disk.
- d) Define the **Maximum number of temporary files** to use for sorting.
- e) Use the **Compression** toggle to change your option for compressing files stored on disk.
- f) Click **OK** to save your changes.

You are now ready to configure the Runtime tab options.


Configure the Runtime tab details - Write to File

Use the Runtime tab to define record processing options for your flow.

1. Set the first record to process.
 - Use the Starting Record up/down arrows to select a record other than record 1 as the first record to write. You can use the arrows to set the starting record back to 1.
 - Alternatively, you can click the **Starting Record** field and type the starting record number.
2. Define the number of records to process.
 - Select **All Records** to write all output records (this is the default).
 - Select **Max Records** and define a maximum number of output records to write. Use the arrows to select a number or type a value in the entry field.

Inspecting and testing flows

Checking a Flow for Errors

Flow Designer automatically checks a flow for errors when you run a flow, run inspection, expose a flow, or save an exposed flow. You can also check for errors by clicking the validation button .

When Flow Designer finds an error, it displays the **Validation** pane at the bottom of the window. Click an error to highlight the error on the canvas. Double-click an error to open the options window of the item containing the error.

Inspecting a flow

Use the inspection tool to view how input data is transformed and processed at any point in a flow.

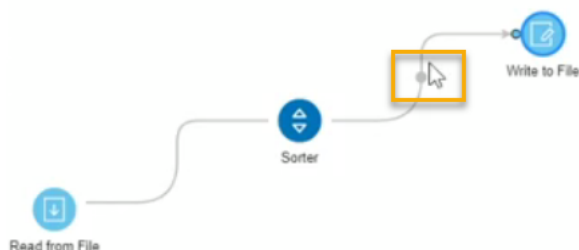
Note: For the preview version of Flow Designer, note these inspection limitations:

- This version supports single-point inspection. You cannot add multiple inspection points.
- Inspection is available for jobs, only. Flow Designer does not support for services.
- The input files must reside on the Spectrum server: Flow Designer does not support the inspection of local files at this time.

Inspection points provide a processing snapshot at any point in a flow. Inspection points allow you to confirm that the flow is producing the results you want, isolating problems, and identifying records that contain errors. The scenario below shows how to inspect data conversion of mapped fields during processing.

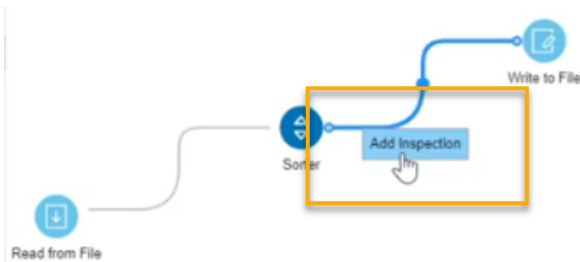
To add inspection points to your flow:

1. Open a flow containing an input file that resides on the server.
2. Perform a high-level check of the flow.
 - Are input and output fields mapped? Field mapping is indicated by a dot (node) on the channel between any two stages.

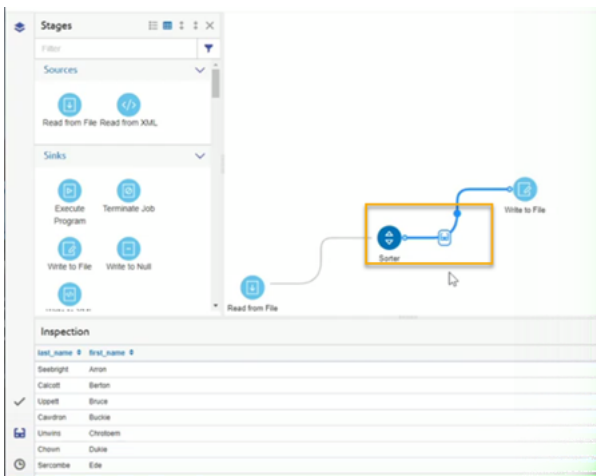


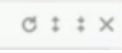
- You can hover over the mapping indicator to see the mapping that is taking place at that point in the flow.

- Double-click the channel with the node, and it turns blue. Click that same channel once more to display the Mapping options page. Learn more about field mapping here: [View mapping details](#) on page 51.
 - Click at any section along the channel to see how the data will change for mapped fields during processing. Data changes include transforms, such as data type conversions. Flow Designer displays a hover message to provide that information.
3. To more closely inspect any point in the flow, right-click on a channel in the position where you will perform the inspection.



4. Click again to insert the inspection point. Notice that this first inspection point occurs before the data conversion, to the left of the mapping indicator. Flow designer adds an eyeglass icon indicating the inspection point and displays the Inspection details panel at the bottom of the page.



The inspection panel includes controls  that refresh, collapse all, and expand all contents, and close the panel. If your input contains hierarchical data, you can extend the display to see parent-child data relationships. You can sort data by clicking any column. The sort order changes as you click on the column name.

When you add the inspection point, Flow Designer runs the job to produce a snapshot of that inspection point. You will see the data as it relates to that specific processing point. The preview provides a picture of how the data conversion takes place during the different phases of processing.

5. To see the state of the data after mapping/conversion, place an inspection point on the other side of the mapping indicator.

Note: Changing the inspection point removes the previous inspection point. If you remove the inspection point, Flow Designer displays a message: There are no inspection points yet. Add an inspection point on a channel to view inspection data.



Once again, Flow Designer runs the job to take a snapshot of the data.

6. Observe the difference in field names this example, of the data before conversion and after conversion:

First inspection point, before processing

Inspection	
last_name	first_name
Seebright	Arron
Calcott	Berton
Uppett	Bruce
Cawdron	Buckie
Unwins	Chrotoem
Chown	Dukie
Sercombe	Ede

Second inspection point, post-conversion

Inspection	
last	first
Seebright	Arron
Calcott	Berton
Uppett	Bruce
Cawdron	Buckie
Unwins	Chrotoem
Chown	Dukie
Sercombe	Ede

Note: Right-click on any inspection point to delete it. You can also right-click on any point of the channel containing the inspection point to delete it. When you delete an inspection point, Flow Designer closes the inspection details panel.

Testing a service with Management Console


Management Console provides a preview feature that allows you to send test data to a service and see the results.

Note: A service must be saved and exposed before it can be tested through Management Console.

1. In a web browser go to this URL:

`http://server:port/managementconsole`

Where *server* is the server name or IP address of your Spectrum™ Technology Platform server and *port* is the HTTP port used by Spectrum™ Technology Platform. By default, the HTTP port is 8080 and the HTTPS port is 8443.


2. Go to the **Services** menu and click the other services containing the service you want to test.
3. Click the service you want to test.
4. Click **Preview**.
5. Enter the input data you want to use for your test. To import data from a file, click the Import button .
6. Click **Run Preview**.

Running flows

Run a job or flow using the Run Jobs button

This release of Flow Designer supports only input and output files that reside on the Spectrum™ Technology Platform server. Local files are not supported at this time.

There are two ways to run a job or flow: using the Run Job button or from the Execution History view.


1. Open the flow (job) to run on the canvas.
2. Save the flow to capture recent changes.
3. Click the run job button .
4. After you start the job, you can observe the progress and these details.
 - **ID** – Job run number for this flow/job
 - **User** – Name of the user who started the flow/job
 - **Start Time** and **End Time**
 - **Duration**
 - **Results** - Green indicates that the job ran without errors. Red indicates errors that you will need to correct.
 - **Status** - Indicates whether the job is actively running or completed

Run jobs or flows from the Execution History panel

This release of Flow Designer supports only input and output files that reside on the Spectrum™ Technology Platform server. Local files are not supported at this time.

Only the jobs containing input and output files on server are supported. The jobs containing client side input and output files are not supported in this release.

There are two ways to run a job or flow: using the Run Job button or from the Execution History view.

1. Open the flow (job) to run on the canvas.
2. Click the Execution History button  to display the history for the current flow.

- If someone ran the flow previously, you will see details for those job runs.
 - If no one has run the flow, you will see this message: There is no execution history for this flow. Select the Run flow button to run the flow.
3. Click the text "Run flow" to start the job.
 4. After you start the job, you can observe the progress and these details.
 - **ID** – Job run number for this flow/job
 - **User** – Name of the user who started the flow/job
 - **Start Time** and **End Time**
 - **Duration**
 - **Results** - Green indicates that the job ran without errors. Red indicates errors that you will need to correct.
 - **Status** - Indicates whether the job is actively running or completed

Deleting flows

This topic provides the steps for deleting flows and describes the confirmation messages returned by the delete function.

1. Select the flow to delete using the check box next to the flow name.

Note: You may select multiple flows to delete at once.

2. Use the **delete flow** button to delete the selected flows from the server.

Flow Designer displays a confirmation message:

Deleting will permanently remove these flows from the server. Are you sure you wish to delete the selected flows?

3. If the flow you want to delete is not bound to any other flows, you can safely click **Yes**.

Some conditions prevent you from deleting flows. In those cases, Flow Designer returns an informational message. The message provides you with information to resolve the condition that prevents you from deleting selected flows.

If you received this message...

These flows cannot be deleted until issues are resolved: *flowname* is in use.

This is what is causing the error...

Other flows are using the named flows, and you cannot delete them.

Use the **View Impact Analysis** link to see more information on the flows that use the selected flow.

If you received this message...

This is what is causing the error...

These flows cannot be deleted until issues are resolved: *filename* is checked out by *user*.

The named user locked the flows.

Coordinate with the named user before deleting the selected flows.

These flows cannot be deleted until issues are resolved: *flowname* does not exist.

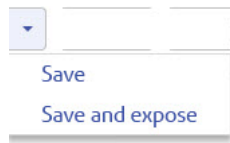
Someone already deleted the named flows from the server.



Refresh your screen to see all available flows.

Exposing flows

Exposing a flow to make it available for other users.

To expose a flow, open an existing flow or create a new flow on the Editor page, select the Save




as drop-down button  next to the save icon  , and select **Save and expose** . This will make the flow available for use in other flows.

Note: If you have more than one version of the flow as a subflow, the version that is used in the parent flow is the exposed version. When you make a change to a subflow, be sure to expose the most recent version so your changes take effect in the flows that use the subflow.

Importing flows

Importing flows - import button method

1. Click the import button  to open Windows Explorer.
2. Go to the location that contains the *.df files to import.
3. Select one or more files to import; click **Open**. Select one or more files to import; click **Open**.

Importing flows - drag-drop method

1. Open Windows Explorer in a separate window.
2. Go to the location that contains the *.df files to import.
3. Import files by dragging them to the canvas and dropping them in to the Import bin.

The Import bin is the area that is labeled:

To import your flows, drag and drop your .df files here.

Note: You can drop any exported flow anywhere on the Flow Designer workspace to import it. When you import a flow, Flow Designer automatically opens the flow in the editor.

Exporting flows

Export any saved selected flow.

Flow designer exports the selected flows to the default browser download folder according to browser and user settings. Some browsers will prompt you to define a location to save the exported file.

You have the option to:

- Export the last saved version for the selected flow, regardless of its exposure status.
- Export the last exposed version of the selected flow.

Exporting the latest saved version of a flow - Manager page

You can export the latest saved version of a flow, regardless of whether the flow is exposed to other users. Follow these steps to export the latest saved version of a flow.

Before you export a flow, you must save or save and expose the flow.

1. To select a flow to export, check the box next to any saved flow listed on the Manager page.

You can select multiple flows.

2. Click the Export drop-down  next to the Export icon .
3. Select **Export latest saved**.

You will see this message:

Successfully exported *flowname*.

Exporting the latest saved version of a flow - Editor page

You can export the latest saved version of a flow, regardless of whether the flow is exposed to other users. Follow these steps to export the latest saved version of a flow.

Before you export a flow, you must save or save and expose the flow.

Click the Export icon .

You will see this message:

Successfully exported *flowname*.



Exporting the exposed version of a flow

You can export the current exposed version of a flow. This allows you to make adjustments to a flow without changing the status of a saved flow, regardless of whether the flow is exposed to other users. Follow these steps to export a flow.

Before you export a flow, you must save or save and expose the flow.

These steps take place on the Manager page.

1. To select a flow to export, check the box next to any saved flow listed on the Manager page. You can select multiple flows.

2. Click the Export drop-down  next to the Export icon .
3. Select **Export exposed version**.

You will see this message:

Successfully exported *flowname*.

Running an External Program

An Execute Program stage invokes an executable entity, such as a program or command line command, when it receives a record. To use an Execute Program stage in your flow:

Options

Option	Description
Command-line	The executable name and arguments (if applicable). The arguments can be data available in the flow. To access that data, click the [...] (Browse) button. You can select from the following three contexts: Current Job ID, Current Job Name, or Current User Name. You can also select from the available fields. For example, JobStatus and JobComment.
Timeout	<p>Specifies whether to cancel the job if the command does not respond within a given amount of time:</p> <p>No timeout Do not cancel the execution if the command fails to respond.</p> <p>Timeout in milliseconds Cancels the execution attempt if the command does not respond in the specified number of milliseconds.</p>
Environment Variables	<p>Optional. Specifies environment variables to use when executing the command. To add an environment variable click Add.</p> <p>Enter the appropriate key word in the Key field, such as "JAVA_HOME".</p> <p>Enter the appropriate value in the Value field, such as C:\Java\jre7.</p> <p>Alternatively, you can select a field from the Field List dialog box by clicking the [...] (Browse) button. You can select from one of these contexts: Current Job ID, Current Job Name, or Current User Name. You can also select from the available fields, such as JobStatus and JobComment.</p>

Using the template browser to create flows and jobs

Flow Designer provides a template browser that provides options for creating new jobs, services, or subflows on a blank canvas or using pre-defined templates.

Flow templates

Flow templates illustrate ways to use Spectrum™ Technology Platform and its modules to meet your business needs. They show how particular modules solve various requirements, such as parsing, standardizing, and validating names and addresses.


Flow templates are delivered with each module that you license. For example, if you are licensed for the Data Normalization Module, you receive the Standardizing Personal Names flow template. If you are licensed for the Universal Addressing Module, you receive the Validating U.S. and Canadian Addresses flow templates.

Depending on the purpose of each template, the job may come with sample data. Or, it may be a service with no sample data. You can use flows in their original state and run those that are delivered as jobs to see how they function. Alternatively, you can change the flows' input and output files or add services into your jobs, adding other input and output files.

Note: Templates are illustrations of various Spectrum™ Technology Platform features. They are intended to be starting points and examples for solutions you can create for your environment.

Flow Designer provides a selection of templates, available when you open the [New Flow page](#) on page 21.

Create a new job, service, or subflow on a blank canvas

1. From the Manager page, click the **New** icon . Flow Designer displays the [New Flow page](#) on page 21 with template selections.
2. Use the **Job**, **Service**, or **Subflow** links at the top left side of the page to display selections. Depending upon the number of templates available, you may have to use the scroll bar on the right side of the display to see all templates.
3. Optionally, you can use the filter feature to find templates, by name, for the type of flow you selected.
Type a letter or a string, such as "b" or "blank."

4. Double-click the blank template (this is the first template displayed at the top of the selection space) to open the flow on the canvas, or click the blank template and click **OK**.

Flow Designer prompts you to specify a file name. The application provides a unique name suggestion. For example, if "New Job 1" is already used, Flow Designer suggests "New Job 2." If you try to use an existing flow name, you will see an error prompting you for a different name.


5. Once you decide on a name, click **OK**.

The flow editor saves the new flow on the server and displays a blank workflow canvas, with your job name displayed at the top left of the workspace.

You can go to the Manager page to see if your flow is displayed. You may have to refresh the view to see the new flow. The initial display may show that the flow is locked by the current user (you), and displays the user's name under the **Locked by** column.

Create a new job, service, or subflow from a template

The template browser provides pre-defined templates for creating jobs, services and subflows.

1. From the Manager page, click the **New** icon . Flow Designer displays the **New Flow page** on page 21 with template selections.
2. Use the **Job**, **Service**, or **Subflow** links at the top left side of the page to display available selections in each category.
Depending upon the number of templates available, you may have to use the scroll bar on the right side of the display to see all templates.
3. Optionally, you can use the filter feature to find templates, by name, for the type of flow you want to build.
Type a letter or a string, such as "p" or "parseaddress."
4. Select a template, and click **OK**.
Flow Designer prompts you to specify a name. The application provides a unique name suggestion. For example, if "ParseAddressFields" is already used, Flow Designer suggests "ParseAddressFields1." If you try to use an existing flow name, you will see an error prompting you for a different name.
5. Once you decide on a name, click **OK**.
The flow editor saves the new flow on the server and displays the selected flow on the canvas, with all of the stages it contains. You will see your flow name displayed at the top left of the workspace.

You can now modify the template job to suit your needs.

4 - File properties reference

In this section

Supported character encoding methods	73
Field separators	73
Record separators	74

Supported character encoding methods

CP1252	This encoding is also known as the Windows-1252 or simply Windows character set. It is a super set of ISO-8859-1 and uses the 128-159 code range to display additional characters not included in the ISO-8859-1 character set.
UTF-8	Supports all Unicode characters and is backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
UTF-16	Supports all Unicode characters but is not backwards-compatible with ASCII. For more information about UTF, see unicode.org/faq/utf_bom.html .
US-ASCII	A character encoding based on the order of the English alphabet.
UTF-16BE	UTF-16 encoding with big endian byte serialization (most significant byte first).
UTF-16LE	UTF-16 encoding with little endian byte serialization (least significant byte first).
ISO-8859-1	An ASCII character encoding typically used for Western European languages. Also known as Latin-1.
ISO-8859-3	An ASCII character encoding typically used for Southern European languages. Also known as Latin-3.
ISO-8859-9	An ASCII character encoding typically used for Turkish language. Also known as Latin-5.
CP850	An ASCII code page used to write Western European languages.
CP500	An EBCDIC code page used to write Western European languages.
Shift_JIS	A character encoding for the Japanese language.
MS932	A Microsoft's extension of Shift_JIS to include NEC special characters, NEC selection of IBM extensions, and IBM extensions.
CP1047	An EBCDIC code page with the full Latin-1 character set.

Field separators

Use one of these characters to separate fields in a delimited file.

- Space
- Tab
- Comma
- Period
- Semicolon

- Pipe

Record separators

Specifies the character used to separate records in line a sequential or delimited file. Note that this selection is not available if you select **Use default EOL**.

Unix (U+000A)	A line feed character separates the records. This is the standard record separator for Unix systems.
Macintosh (U+000D)	A carriage return character separates the records. This is the standard record separator for Macintosh systems.
Windows (U+000D U+000A)	A carriage return followed by a line feed separates the records. This is the standard record separator for Windows systems.

Note: If your file uses a different record separator, select another character as a record separator.

5 - Stage and transform reference

In this section

Flow Designer stages and transforms	76
Transformer stage transform types	80




Flow Designer stages and transforms







This is a quick reference to the stages and transforms available with Flow Designer.




We will document module-specific stages in the next release.

Control stages

Control stages move data along different paths in a flow, to split or group records, and to perform basic data transforms and mathematical operations.



Stage	Icon	Description	For more information
Aggregator		Aggregator converts flat data to hierarchical data. It takes input data from a single source, creates a schema (a structured hierarchy of data) by grouping the data based on fields you specify, then constructs the groups in the schema.	Aggregator is not available in this release.
Broadcaster		Splits a single stream of records into multiple streams, allowing you to send records to multiple stages for simultaneous processing.	Broadcaster is not available in this release.
Conditional Router		Sends records to different paths in the flow depending on the criteria you specify. The stage can have one or more output ports, depending on the defined criteria. Output ports are numbered consecutively, starting with 1 (which displays as <code>port</code>).	Conditional Router is not available in this release.

Stage	Icon	Description	For more information
Group Statistics		The Group Statistics stage allows you to run statistical operations across multiple data rows broken down into groups that you want to analyze. If no groups are defined all rows will be treated as belonging to one group.	Group Statistics is not available in this release.
Math		The Math stage handles mathematical calculations on a single data row and allows you to conduct a variety of math functions using one or more expressions. Data is input as strings but the values must be numeric or Boolean, based on the type of operation being performed on the data.	Math is not available in this release.
Record Combiner		Record Combiner is a Control stage that uses one or more input ports to merge multiple records from multiple streams into a single survivor record based on a commonality	#unique_110
Record Joiner		Record Joiner performs a SQL-style JOIN operation to combine records from different streams based on a relationship between fields in the streams.	Record Joiner is not available in this release.
Sorter		Use the Sorter stage to sort records using fields you specify	Configuring the Sorter stage on page 56
Splitter		A Splitter converts hierarchical data to flat data. Splitters have one input port and one output port that delivers data from the Splitter to the next stage.	Splitter is not available in this release.

Stage	Icon	Description	For more information
Stream Combiner		Joins multiple record streams from multiple stages. Stream Combiner has one or more stage input ports. For example, you can have one group of records from one stage and another group from a second stage, and the records will merge into a single stream.	Stream Combiner
Transformer		Flow Designer provides the building blocks (functions) that allow you to process, change, and convert data to suit your requirements. These transformations are configurable through the Transformer stage. This section provides a preview of the transforms available in Flow Designer. Note that only custom transforms are supported in this release of Flow Designer.	Creating and configuring data transformations - Transformer stage on page 42
Unique ID Generator		The Unique ID Generator stage creates a unique key that identifies a specific record. A unique ID is crucial for data warehouse initiatives in which transactions may not carry all name and address data, but must be attributed to the same record or contact. A unique ID may be implemented at the individual, household, business, and premises level. Unique ID Generator provides a variety of algorithms to create unique IDs.	Unique ID Generator is not available in this release.






Sources

A source is the first stage in a flow, and defines the input data to process.

Stage	Icon	Description	For more information
Read from File		Specifies an input file for a job or subflow; not available for services. Note: If you want to use an XML file as input for your flow, use the Read from XML stage instead of Read from File. If you want to use a variable format file as input, use Read from Variable Format File.	Configuring the Read from File Stage on page 37
Read from XML		Specifies an XML-format input file for a job or subflow; not available for services.	Configuring the Read from XML Stage for Subflows on page 41

Sinks

A sink is the last stage in a flow, and defines what to do with the output from the flow. A sink can also perform other actions at the end of a flow, such as executing a program.

Stage	Icon	Description	For more information
Execute Program		Runs an external application as part of a process flow	Defining Service Output
Terminate Job		Used in combination with Conditional Router to end a job if certain criteria are found within a record	
Write to File		Writes output of a flow to a specified file	
Write to Null		Sends output to an XML-format output file that can be consumed by other processes or flows.	
Write to XML		The Write to XML stage writes output from a flow or subflow to an XML file.	

Transformer stage transform types

The Transformer stage modifies field values and formatting. You can perform more than one transform on a field as long as the input and output field names are identical.

General Transforms

Construct Field Uses values from existing fields and/or constant values to either replace field values or create a new field. For example, say you have a field named City and you want to add the phrase "City of" to the values in the City field. You would create a template like this:

```
City of ${City}
```

In the **To field** field, you would select the City field. This has the effect of replacing the existing values in the City field with a value constructed using the template. For

example, if the value in the City field is Chicago, the new value would be City of Chicago.

Some characters must be preceded by a backslash to produce a valid template. For example, the single quote character must be preceded by a backslash like this: \ '. See groovy-lang.org/syntax.html for a list of characters that must be escaped with a backslash.

Copy	Copies the value from one field to another.
Custom	Allows you to define your own transform using the Groovy language. For more information, see Creating a Custom Transform . For users of the Spatial Module, custom transforms can access spatial datasets. See the Stages section in the <i>Spectrum Spatial Guide</i> on support.pb.com .
Rename	Changes the name of a field. You can select from a list of field names already in the dataflow or you can type the name you want.
Status	Changes the Status field to a value of either Success or Fail. When set to Fail, an optional Description and Code may also be set.

Formatting Transforms

Case	Changes casing upper or lower case.
Mask	Applies or removes characters from a field. For more information, see Using a Mask Transform .
Pad	Adds characters to the left or right of the field value.

String Transforms

Minimize Whitespace	Removes whitespace at the beginning and end of the field. It also replaces any sequence of whitespaces (such as multiple, consecutive spaces) to a single whitespace character.
Remove Substring	Removes all occurrences of a string from a field. For example, you could remove "CA" from the StateProvince field.
Substring	Copies a contiguous sequence of characters from one field to another.
Trim	Removes specified characters from the left, right, or both sides of a field. Note that this transform is case-sensitive.
Truncate	Removes a specified number of characters from the left and right sides of a field.

List Transforms

This feature helps you to create canned transformation that operate on lists, for example input from read from XML.

For defining list transformations, follow these steps:

1. Select a list transformation operation. Input fields appear in a tree view on the right.

2. Select a valid field in the tree to apply the operation on. Properties for the operation show up below the input fields tree view.
3. Specify the operation properties and click add. The transform gets added to the list in the parent window, that is the 'Transformer Options' window.

Create Field	Allows creating a field under the user selected list type field. For example if a list called Football has two clubs, Knitters and Lambs, you user can add a new club called Irons, for a total of three clubs.
Sort	Performs sorting on values present in the selected field. In a complex list, the user needs to specify the key element for sorting while in case of simple list, the sorting takes place on the elements present in the list. The user can select the sort order as either ascending or descending. In the example of Football, when the list has three clubs, the user needs to select field 'name' under 'club' to sort the clubs based on name. The current club entries list as Irons, Knitters and Lambs if sort order is ascending and descending for sort order descending. Now, if the user wants the list of players sorted, the field 'player' needs to be selected and sort order defined for it.
Sum	Performs summation of all the values present in the selected field. The output is stored in a field specified by the user. For example, if the user wants to view the total points gained by each football club, user needs to select field 'points' under 'Tournament' and specify the output field name.
Copy	Performs the copy operation from the selected field to the field specified by the user. When user selects a field to copy, the field and all fields under it (if any) are copied to the new field specified. This operation takes place at the same level of hierarchy.
Rename	Performs the rename operation of the selected field to the new name specified by the user.

This sample XML code provides a reference to the List Transform feature:

```
<?xml version="1.0"?>
<sports_details>
  <sports name="football">
    <clubs>
      <club name="Knitters">
        <player>Samuel</player>
        <player>Messi</player>
        <player>kaka</player>
        <player>Alan</player>
        <coach>Stuart</coach>
      <Tournament name="Football League">
        <result>won</result>
        <points>4</points>
      </Tournament>
      <Tournament name="UEFA">
        <result>draw</result>
        <points>2</points>
      </Tournament>
    </clubs>
  </sports>
</sports_details>
```

```

</club>
<club name="Lambs">
  <player>Ronaldo</player>
  <player>Neymar</player>
  <player>Zlatan</player>
  <player>Mesut</player>
  <coach>Ivan</coach>
  <Tournament name="Airtel League">
<result>draw</result>
<points>2</points>
  </Tournament>
  <Tournament name="Champions League">
<result>lost</result>
<points>0</points>
  </Tournament>
</club>
<club name="Irons">
  <player>Scott</player>
  <player>Paul</player>
  <player>John</player>
  <player>Andrew</player>
  <coach>Jeff</coach>
  <Tournament name="CAF">
<result>won</result>
<points>4</points>
  </Tournament>
  <Tournament name="Copa America">
<result>won</result>
<points>4</points>
  </Tournament>
</club>
</clubs>
</sports>
<sports name="badminton">
<clubs>
  <club name="Shuttlers">
    <player>Saina</player>
    <player>Viktor</player>
    <player>Chen</player>
    <player>Srikanth</player>
    <coach>Jan</coach>
    <Tournament name="Olympic Games">
<result>won</result>
<points>4</points>
    </Tournament>
    <Tournament name="Commonwealth Games">
<result>won</result>
<points>4</points>
    </Tournament>
  </club>
  <club name="Choppers">
    <player>Wang</player>
    <player>Sindhu</player>

```

```
<player>Carolina</player>
<player>Li Xuerui</player>
<coach>Ratchanok</coach>
<Tournament name="World Junior">
<result>draw</result>
<points>2</points>
</Tournament>
<Tournament name="Uber Cup">
<result>draw</result>
<points>2</points>
</Tournament>
</club>
<club name="Lobbers">
<player>Nozomi</player>
<player>Chou</player>
<player>Marc</player>
<player>Lin</player>
<coach>Kevin</coach>
<Tournament name="World Senior">
<result>won</result>
<points>4</points>
</Tournament>
<Tournament name="Thomas Cup">
<result>won</result>
<points>4</points>
</Tournament>
</club>
</clubs>
</sports>
</sports_details>
```

6 - Supported data types reference

Depending on the type of processing you want to perform you may use one or more of . For an address validation flow you might only use string data. For flows that involve the mathematical computations you may use numeric or Boolean data types. For flows that perform spatial processing you may use a complex data type. For flows that combine these, you may use a variety of data types.

In this section

Data types

86

Data types

Spectrum™ Technology Platform supports these data types.

bigdecimal	A numeric data type that supports 38 decimal points of precision. Use this data type for data that will be used in mathematical calculations requiring a high degree of precision, especially those involving financial data. The bigdecimal data type supports more precise calculations than the double data type.
boolean	A logical type with two values: true and false.
bytearray	An array (list) of bytes. Note: Bytearray is not supported as an input for a REST service.
date	A data type that contains a month, day, and year. For example, 2012-01-30 or January 30, 2012. You can specify a default date format in Management Console.
datetime	A data type that contains a month, day, year, and hours, minutes, and seconds. For example, 2012/01/30 6:15:00 PM.
double	A numeric data type that contains both negative and positive double precision numbers between 2^{-1074} and $(2-2^{-52}) \times 2^{1023}$. In E notation, the range of values is -1.79769313486232E+308 to 1.79769313486232E+308.
float	A numeric data type that contains both negative and positive single precision numbers between 2^{-149} and $(2-2^{-23}) \times 2^{127}$. In E notation, the range of values -3.402823E+38 to 3.402823E+38.
integer	A numeric data type that contains both negative and positive whole numbers between -2^{31} (-2,147,483,648) and $2^{31}-1$ (2,147,483,647).
list	Strictly speaking, a list is not a data type. However, when a field contains hierarchical data, it is treated as a "list" field. In Spectrum™ Technology Platform a list is a collection of data consisting of multiple values. For example, a field Names may contain a list of name values. This may be represented in an XML structure as: <pre><Names> <Name>John Smith</Name> <Name>Ann Fowler</Name> </Names></pre> It is important to note that the Spectrum™ Technology Platform list data type different from the XML schema list data type in that the XML list data type is a simple data type consisting of multiple values, whereas the Spectrum™ Technology Platform list data type is similar to an XML complex data type.
long	A numeric data type that contains both negative and positive whole numbers between -2^{63} (-9,223,372,036,854,775,808) and $2^{63}-1$ (9,223,372,036,854,775,807).
string	A sequence of characters.
time	A data type that contains the time of day. For example, 21:15:59 or 9:15:59 PM.

7 - Date and time patterns reference

In this section

Date and time patterns

88

Date and time patterns

When defining data type options for date and time data, you can create your own custom date or time pattern if the predefined ones do not meet your needs. To create a date or time pattern, use the notation described in the table below. For example, this pattern:

`dd MMMM yyyy`

Would produce a date like this:

14 December 2020

Letter	Description	Example
G	Era designator	AD
yy	Two-digit year	96
yyyy	Four-digit year	1996
M	Numeric month of the year	7
MM	Numeric month of the year: If the number is less than 10 a zero is added to make it a two-digit number.	07
MMM	Short name of the month	Jul
MMMM	Long name of the month	July
w	Week of the year	27
ww	Two-digit week of the year: If the week is less than 10 an extra zero is added.	06
W	Week of the month	2
D	Day of the year	189
DDD	Three-digit day of the year: If the number contains less than three digits, zeros are added.	006

Letter	Description	Example
d	Day of the month	10
dd	Two-digit day of the month: Numbers less than 10 have a zero added.	09
F	Day of the week in month	2
E	Short name of the day of the week	Tue
EEEE	Long name of the day of the week	Tuesday
a	AM PM marker	PM
H	Hour of the day, with the first hour being 0 and the last hour being 23	0
HH	Two-digit hour of the day, with the first hour being 0 and the last hour being 23: Numbers less than 10 have a zero added.	08
k	Hour of the day, with the first hour being 1 and the last hour being 24.	24
kk	Two-digit hour of the day, with the first hour being 1 and the last hour being 24: Numbers less than 10 have a zero added.	02
K	Hour hour of the morning (AM) or afternoon (PM), with 0 being the first hour and 11 being the last hour.	0
KK	Two-digit hour of the day, with the first hour being 1 and the last hour being 24: Numbers less than 10 have a zero added.	02
h	Hour of the morning (AM) or afternoon (PM), with 1 being the first hour and 12 being the last hour.	12
hh	Two-digit hour of the morning (AM) or afternoon (PM), with 1 being the first hour and 12 being the last hour: Numbers less than 10 have a zero added.	09

Letter	Description	Example
m	Minute of the hour	30
mm	Two-digit minutes of the hour: Numbers less than 10 have a zero added.	05
s	Second of the minute	55
ss	Two-digit second of the minute: Numbers less than 10 have a zero added.	02
S	Millisecond of the second	978
SSS	Three-digit millisecond of the second: Numbers containing fewer than three digits will have one or two zeros added to make them three digits.	978 078 008
z	Time abbreviation of the time zone name: If the time zone does not have a name, the GMT offset.	PST GMT-08:00
zzzz	The full time zone name. If the time zone does not have a name, the GMT offset	Pacific Standard Time GMT-08:00
Z	The RFC 822 time zone	-0800
X	The ISO 8601 time zone	-08Z
XX	The ISO 8601 time zone with minutes	-0800Z
XXX	The ISO 8601 time zone with minutes and a colon separator between hours and minutes	-08:00Z

8 - About Spectrum™ Technology Platform

In this section

What Is Spectrum™ Technology Platform?	92
Enterprise Data Management Architecture	93
Spectrum™ Technology Platform Architecture	96
Modules and Components	101

What Is Spectrum™ Technology Platform?

Spectrum™ Technology Platform is a system that improves the completeness, validity, consistency, timeliness, and accuracy of your data through data standardization, verification and enhancement. Ensuring that your data is accurate, complete, and up to date enables your firm to better understand and connect with your customers.

Spectrum™ Technology Platform aids in the design and implementation of business rules for data quality by performing the functions described here.

Parsing, Name Standardization, and Name Validation

To perform the most accurate standardization you may need to break up strings of data into multiple fields. Spectrum™ Technology Platform provides advanced parsing features that enable you to parse personal names, company names, and many other terms and abbreviations. In addition, you can create your own list of custom terms to use as the basis of scan and extract operations. The Universal Name Module provides this functionality.

Deduplication and Consolidation

Identifying unique entities enables you to consolidate records, eliminate duplicates and develop "best-of-breed" records. A "best-of-breed" record is a composite record that is built using data from other records. The Advanced Matching Module and Data Normalization Module provide this functionality.

Address Validation

Address validation applies rules from the appropriate postal authority to put an address into a standard form and even validate that the address is a deliverable address. Address validation can help you qualify for postal discounts and can improve the deliverability of your mail. The Universal Addressing Module provides this functionality.

Geocoding

Geocoding is the process of taking an address and determining its geographic coordinates (latitude and longitude). Geocoding can be used for map generation, but that is only one application. The underlying location data can help drive business decisions. Reversing the process, you can enter a geocode (a point represented by a latitude and longitude coordinate) and receive address information about the geocode. The Enterprise Geocoding Module provides this functionality.

Location Intelligence

Location intelligence creates new information about your data by assessing, evaluating, analyzing and modeling geographic relationships. Using location intelligence processing you can verify locations

and transform information into valuable business intelligence. The Spatial Module provides this functionality.

Master Data Management

Master data management enables you to create relationship-centric master data views of your critical data assets. The Data Hub Module helps you identify influencers and non-obvious relationships, detect fraud, and improve the quality, integration, and accessibility of your information.

Tax Jurisdiction Assignment

Tax jurisdiction assignment takes an address and determines the tax jurisdictions that apply to the address's location. Assigning the most accurate tax jurisdictions can reduce financial risk and regulatory liability.

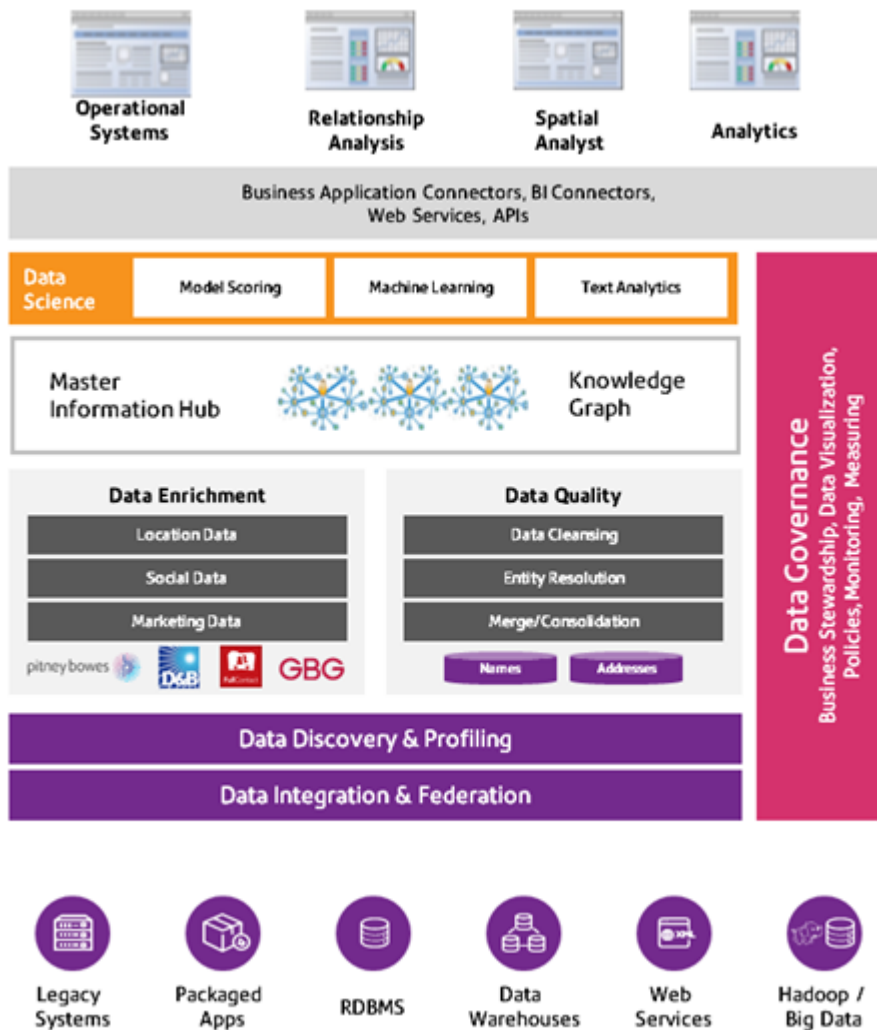
Spectrum™ Technology Platform software from Pitney Bowes integrates up-to-date jurisdictional boundaries with the exact street addresses of your customer records, enabling you to append the correct state, county, township, municipal, and special tax district information to your records. Some example uses of tax jurisdiction assignment are:

- Sales and use tax
- Personal property tax
- Insurance premium tax

The Enterprise Tax Module provides this functionality.

Enterprise Data Management Architecture

With Spectrum™ Technology Platform, you can build a comprehensive enterprise data management process, or you can use it as a more targeted solution. This diagram illustrates a complete solution that takes data from its source, through data enrichment and data quality processes, feeding a master data management hub which makes a single view of the data available to multiple business applications.



Master Information Hub

The Master Information Hub hub allows for rapid modeling of entities and their complex relationships across roles, processes, and interactions. It provides built-in social network analysis capabilities to help you understand influencers, predict churn, detect non-obvious relationships and fraudulent patterns, and provide recommendations.

Spectrum™ Technology Platform supports two approaches to the MDM hub. In the master hub approach, the data is maintained in a single MDM database and applications access the data from the MDM database. In the registry approach, the data is maintained in each business application and the MDM hub registry contains keys which are used to find related records. For example, a customer's record may exist in an order entry database and a customer support database. The MDM registry would contain a single key which could be used to access the customer data in both places.

Data Enrichment

Data enrichment processes augment your data with additional information. You can base enrichment on spatial data, marketing data, or data from other detail sources. For example, if you have a database of customer addresses, you could geocode the address to determine the latitude/longitude coordinates of the address and store those coordinates as part of the record. You can then use your customer data to perform a variety of spatial calculations, such as finding the customer's nearest bank branch. Spectrum™ Technology Platform allows you to enrich your data with a variety of information, including geocoding (with the Enterprise Geocoding Module), tax jurisdiction assignment (with the Enterprise Tax Module), geospatial calculations (with the Spatial Module), and travel directions between points (with the Spatial Module).

Data Quality and Data Governance

Data quality and data governance processes check your data for duplicate records, inconsistent information, and inaccurate information.

Duplicate matching identifies potential duplicate records or relationships between records, whether the data is name and address in nature or any other type of customer information. Spectrum™ Technology Platform allows you to specify a consistent set of business match rules using Boolean matching methods, scoring methods, thresholds, algorithms, and weights to determine if a group of records contains duplicates. Spectrum™ Technology Platform supports extensive customization so you can tailor the rules to the unique needs of your business.

Once duplicate records have been identified, you may wish to consolidate records. Spectrum™ Technology Platform allows you to specify how to link or merge duplicate records so you can create the most accurate and complete record from any collection of customer information. For example, you can build a single best-of-breed record from all of the records in a household. The Advanced Matching Module is used to identify duplicates and eliminate them.

Data quality processes also standardize your data. Standardization is a critical process because standardized data elements are necessary to achieve the highest possible results for matching and identifying relationships between records. While several modules perform standardization of one type or another, the Spectrum™ Technology Platform Data Normalization module provides the most comprehensive set of standardization features. In addition, the Universal Name module provides specific data quality features for handling personal name and business name data.

Standardized data is not necessarily accurate data. Spectrum™ Technology Platform can compare your data to known, up-to-date reference data for correctness. The sources used for this process may include regulatory bodies such as the U.S. Postal Service, third-party data providers such as Experian or Dunn and Bradstreet, or your company's internal reference sources, such as accounting data. Spectrum™ Technology Platform is particularly strong in address data validation. It can validate or standardize addresses in 250 countries and territories around the world. The Universal Addressing Module performs address validation.

To determine which one is right for you, discuss your needs with your account executive.

While Spectrum™ Technology Platform can automatically handle a wide range of data quality issues, there are some situations where a manual review by a data steward is appropriate. To support this, the Business Steward Module provides a way to specify the rules that will trigger a manual review,

and it provides a web-enabled tool for reviewing exception records. It includes integrated access to third-party tools such as Bing maps and Experian data to aid data stewards in the review and resolution process.

Data Discovery and Profiling

Data discovery is the process of scanning your data resources to get a complete inventory of your data landscape. Spectrum™ Technology Platform can scan structured data, unstructured data, and semi-structured data using a wide array of data profiling techniques. The results of the scan are used to automatically generate a library of documentation describing your company's data assets and to create a metadata repository. This documentation and accompanying metadata repository provide the insight you need before beginning data integration, data quality, data governance, or master data management projects.

For more information about the Spectrum™ Technology Platform Data Discovery Module, contact your account executive.

Data Integration and Federation

Once you have an inventory of your data landscape, you need to consider how you will access the data you need to manage. Spectrum™ Technology Platform can connect to data in multiple sources either directly or through integration with your existing data access technologies. It supports batch and real-time data integration capabilities for a variety of business needs, including data warehousing, data quality, systems integration, and migration. Spectrum™ Technology Platform can access data in RDBMS databases, data warehouses, XML files, flat files, and more. Spectrum™ Technology Platform supports SQL queries with complex joins and aggregations and provides a visual query development tool. In addition, Spectrum™ Technology Platform can access data over REST and SOAP web services.

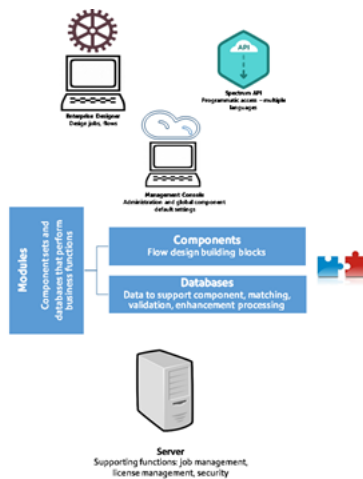
Spectrum™ Technology Platform can trigger batch processing based on the appearance of one or more source files in a specified folder. This "hot folder" trigger is useful for monitoring FTP uploads and processing them as they occur.

Some of these data integration capabilities require a license for the Enterprise Data Integration Module. For more information, contact your account executive.

Finally, Spectrum™ Technology Platform can integrate with packaged applications such as SAP.

Spectrum™ Technology Platform Architecture

Spectrum™ Technology Platform from Pitney Bowes consists of a server that runs a number of modules. These modules provide different functions, such as address validation, geocoding, and advanced parsing, among others. This diagram illustrates the Spectrum™ Technology Platform architecture.



Server

The foundation of the Spectrum™ Technology Platform is the server. The server handles data processing, synchronizes repository data, and manages communication. It provides job management and security features.

Modules

Modules are sets of features that perform a specific function. For example, the Universal Addressing Module standardizes addresses to conform to postal standards. The Enterprise Tax Module determines the tax jurisdictions that apply to a given address. Modules are grouped together to solve common business problems and licensed together as bundles.

Components

Modules are comprised of components which perform a specific function in a flow or as a service. For example, the Enterprise Geocoding module's Geocode US Address component takes an address and returns the latitude and longitude coordinates for that address; the Universal Addressing module's Get City State Province takes a postal code and returns the city and state or province where that postal code is located.

The components that you have available on your system depend on which Spectrum™ Technology Platform bundle you have licensed.

Databases

Some modules depend on databases containing reference data. For example, the Universal Addressing module needs to have access to U.S. Postal Service data in order to verify and standardize addresses in the U.S. Databases are installed separately and some are updated on a regular basis to provide you with the latest data.

Modules have both required and optional databases. Optional databases provide data needed for certain features that can enhance your Spectrum™ Technology Platform process.

Management Console

Management Console is a tool for administering Spectrum™ Technology Platform. You can use Management Console to:

- Define the connections between Spectrum™ Technology Platform and your data
- Specify the default settings for services and flows
- Manage user accounts, including permissions and passwords
- View logs
- View licenses including license expiration information

Management Console | Flows | Services | Resources | System | admin

Home > Resources: Data Sources

Data Sources

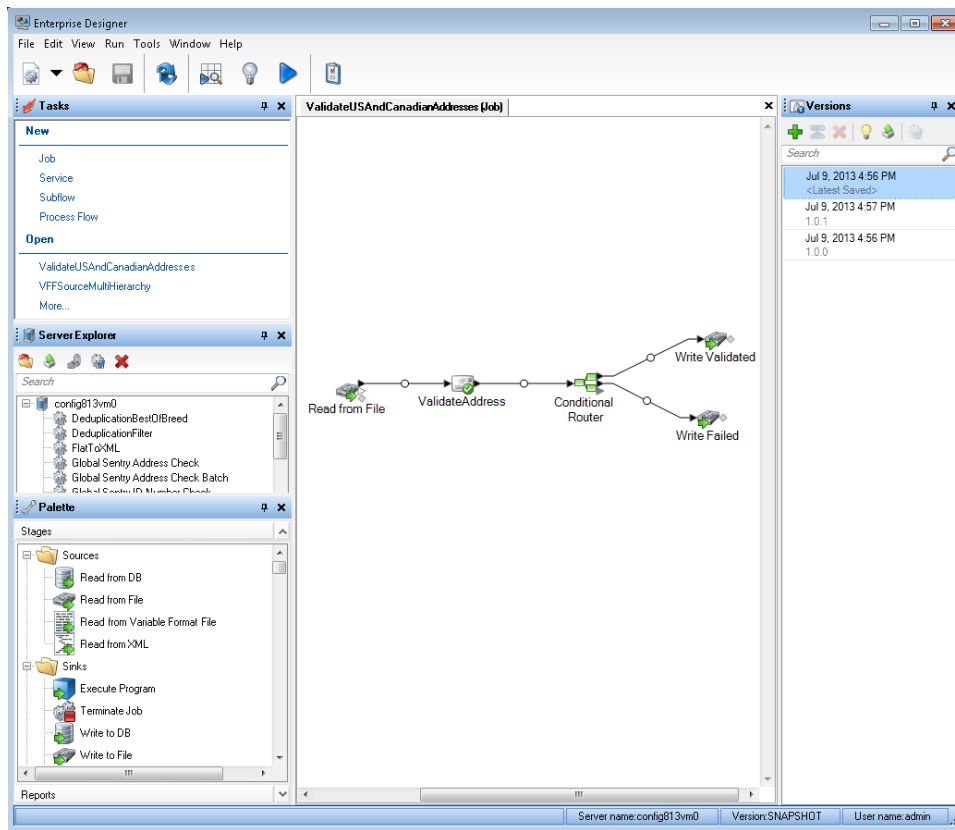
+ [edit] [refresh] [delete] [share] Filter

Name	Type
test1	FTP
test2	FTP
test4	Cloud
test5HDFS	HDFS
mdg1teamcity1	FTP

Showing 5 of 5 records | Rows per page: 10

Enterprise Designer

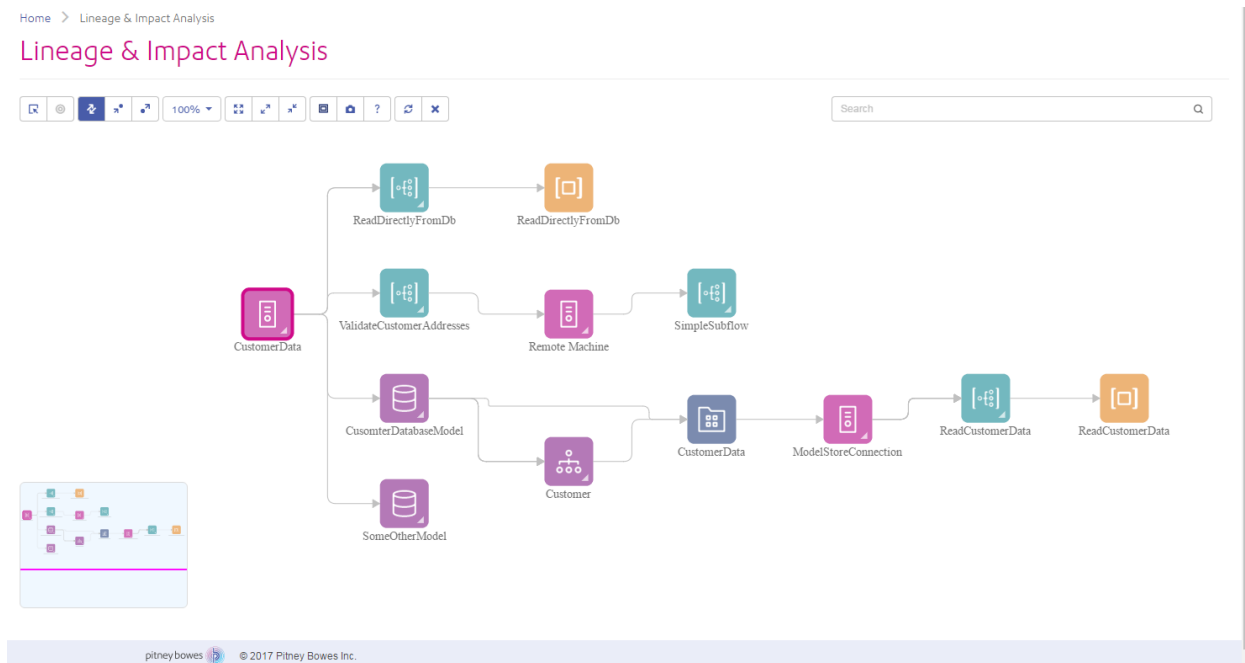
Enterprise Designer is a tool for creating Spectrum™ Technology Platform jobs, services, subflows, and process flows. It provides a familiar drag-and-drop interface to allow you to graphically create complex flows.



Note: Enterprise Designer will be replaced by Flow Designer in a future release. Flow Designer is in Technical Preview status at this time.

Metadata Insights

Metadata Insights gives you the control you need to deliver accurate and timely data-driven insights to your business. Use Metadata Insights to develop data models, view the flow of data from source to business application, and assess the quality of your data through profiling. With this insight, you can identify the data resources to use to answer particular business questions, adapt and optimize processes to improve the usefulness and consistency of data across your business, and troubleshoot data issues.



Web Services and API

You can integrate Spectrum™ Technology Platform capabilities into your applications using web services and programming APIs. These interfaces provide simple integration, streamline record processing, and support backward compatibility of future versions.

The Spectrum™ Technology Platform API is available for these languages:

- C
- C++
- COM
- Java
- .NET

Web services are available via SOAP and REST.

Administration Utility - Command Line Interface (CLI)

The Administration Utility provides command line access to administrative functions. You can run commands interactively or in scripts. Some administrative functions are not available in the Administration Utility. For these functions, you can use Management Console as well as some component applications.

Modules and Components

Table 6: Modules and Components

Module	Description	Components
Advanced Matching Module	Matches records within or between input files.	<ul style="list-style-type: none"> Best Of Breed Candidate Finder Duplicate Synchronization Filter Interflow Match Intraflow Match Match Key Generator Transactional Match
Business Steward Module	Identifies exception records and provides a browser-enabled tool for manually reviewing exception records.	<ul style="list-style-type: none"> Exception Monitor Read Exceptions Write Exceptions
Country Identifier	Takes a country name or a combination of postal code and state-province and returns the two-character ISO country code, the three-character Universal Postal Union (UPU) code, and the English country name.	Country Identifier
Metadata Insights	Gives you the control you need to deliver accurate and timely data-driven insights to your business. Develops data models, gives you a view the flow of data from source to business application, and assesses the quality of your data through profiling. It helps you identify the data resources you should use to answer particular business questions and to optimize processes to improve the usefulness and consistency of data across your business.	<ul style="list-style-type: none"> Models (Logical and Physical) Model Store Profile Lineage and Impact Analysis

Module	Description	Components
Data Hub Module	Links and analyzes data, identifying relationships and trends.	Write to Hub Read From Hub Query Hub Graph Visualization
Data Integration Module	Provides capabilities useful in data warehousing, data quality, systems integration, and migration.	Field Selector Generate Time Dimension Query Cache Write to Cache
Data Normalization Module	Removes inconsistencies in data.	Advanced Transformer Open Parser Table Lookup Transliterator
Enterprise Data Integration	Connects to data in multiple sources for a variety of business needs including data warehousing, data quality, systems integration, and migration.	Call Stored Procedure Field Selector Generate Time Dimension Query Cache Write to Cache
Enterprise Geocoding Module	Determines the geographic coordinates for an address. Also determines the address of a given latitude and longitude.	Geocode Address AUS Geocode Address GBR - deprecated. Use Global Geocoding Module geocoding stage. Geocode Address Global Geocode Address World Geocode US Address GNAF PID Location Search Reverse APN Lookup Reverse Geocode Address Global Reverse Geocode US Location

Module	Description	Components
Enterprise Tax Module	Determines the tax jurisdictions that apply to a given location.	Assign GeoTAX Info Calculate Distance
FCC Screening Module	Helps banks and financial institutions to effectively detect financial crimes, reduce false positives, and maintain robust detection capability as required by the regulators.	Party Groups Lists Screen Alerts
GeoConfidence Module	Determines the probability that an address or street intersection is within a given area.	Geo Confidence Surface CreatePointsConvexHull
Global Addressing Module	Provides enhanced address standardization and validation. Also, automatically suggests addresses as you type and immediately returns candidates based on your input. Splits postal address strings into individual address elements using machine learning techniques.	Global Address Parser Global Address Validation Global Type Ahead
Global Geocoding Module	Determines the geographic coordinates for an address. Also determines the address of a given latitude and longitude. Interactive geocoding is a type-ahead feature in Global Geocoding Module. Key Lookup uses a key to geocode addresses.	Global Geocode Global Reverse Geocode Global Interactive Geocoding Global Key Lookup
Global Sentry	Attempts to match transactions against government-provided watch lists that contain data from different countries.	Global Sentry Global Sentry Address Check Global Sentry ID Number Check Global Sentry Name Check Global Sentry Other Data Check

Module	Description	Components
Spatial Module	Performs point in polygon and radial analysis against a variety of geospatial databases.	<ul style="list-style-type: none"> Closest Site Find Nearest Point In Polygon Query Spatial Data Read Spatial Data Spatial Calculator Spatial Union Write Spatial Data
	Performs routing calculations to obtain directions, calculate drive time and drive distance, and identify locations within a certain time or distance from a starting point.	<ul style="list-style-type: none"> Get Route Data Get Travel Boundary Get Travel Cost Matrix Get Travel Directions Persistent Update
SAP Module	Enables Spectrum™ Technology Platform to interface with SAP Customer Relationship Management Module applications.	<ul style="list-style-type: none"> SAP Generate Match Key SAP Generate Match Score SAP Generate Search Key SAP Generate Search Key Constant SAP Generate Search Key Metaphone SAP Generate Search Key Substring SAP Validate Address With Candidates
Universal Addressing Module	Standardizes and validates addresses according to the postal authority's standards.	<ul style="list-style-type: none"> Get Candidate Addresses Get City State Province Get Postal Codes Validate Address Validate Address Global

Module	Description	Components
Universal Name Module	Parses personal names, company names, addresses, and many other terms and abbreviations.	Name Parser (Deprecated) Name Variant Finder Open Name Parser

Notices

© 2019 Pitney Bowes. All rights reserved. MapInfo and Group 1 Software are trademarks of Pitney Bowes Software Inc. All other marks and trademarks are property of their respective holders.

USPS® Notices

Pitney Bowes Inc. holds a non-exclusive license to publish and sell ZIP + 4® databases on optical and magnetic media. These trademarks are owned by the United States Postal Service: CASS, CASS Certified, DPV, eLOT, FASTforward, First-Class Mail, Intelligent Mail, LACS^{Link}, NCOA^{Link}, PAVE, PLANET Code, Postal Service, POSTNET, Post Office, RDI, Suite^{Link}, United States Postal Service, Standard Mail, United States Post Office, USPS, ZIP Code, and ZIP + 4. This list is not exhaustive of the trademarks belonging to the Postal Service.

Pitney Bowes Inc. is a non-exclusive licensee of USPS® for NCOA^{Link}® processing.

Prices for Pitney Bowes products, options, and services are not established, controlled, or approved by USPS® or United States Government. When utilizing RDI™ data to determine parcel-shipping costs, the business decision on which parcel delivery company to use is not made by the USPS® or United States Government.

Data Provider and Related Notices

Data Products contained on this media and used within Pitney Bowes Software applications are protected by various trademarks and by one or more of these copyrights:

- © Copyright United States Postal Service. All rights reserved.
 - © 2014 TomTom. All rights reserved. TomTom and the TomTom logo are registered trademarks of TomTom N.V.
 - © 2016 HERE Fuente: INEGI (Instituto Nacional de Estadística y Geografía) - Based upon electronic data © National Land Survey Sweden.
 - © Copyright United States Census Bureau
 - © Copyright Nova Marketing Group, Inc.
- Portions of this program are © Copyright 1993-2019 by Nova Marketing Group Inc. All Rights Reserved
- © Copyright Second Decimal, LLC
 - © Copyright Canada Post Corporation - Data is from a compilation in which Canada Post Corporation is the copyright owner.
 - © 2007 Claritas, Inc.

The Geocode Address World data set contains data licensed from the GeoNames Project (www.geonames.org) provided under the Creative Commons Attribution License ("Attribution License") located at <http://creativecommons.org/licenses/by/3.0/legalcode>. Your use of the GeoNames data (described in the Spectrum™ Technology Platform User Manual) is governed by the terms of the Attribution License, and any conflict between your agreement with Pitney Bowes and the Attribution License will be resolved in favor of the Attribution License solely as it relates to your use of the GeoNames data.



3001 Summer Street
Stamford CT 06926-0700
USA

www.pitneybowes.com