Informatica® PowerExchange for DataSift 10.1

User Guide

Informatica PowerExchange for DataSift User Guide
10.1
June 2016

Publication Date: 2018-09-30

Table of Contents

Preface

The *Informatica PowerExchange for DataSift User Guide* provides information about extracting data from DataSift. The guide is written for developers who are responsible for developing mappings that read data from DataSift.

This book assumes you have knowledge of DataSift and Informatica.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

CHAPTER 1

Introduction to PowerExchange for DataSift

This chapter includes the following topics:

PowerExchange for DataSift Overview

You can use PowerExchange for DataSift to extract real-time and historical social data from websites through DataSift.

Use DataSift to access social data from multiple social websites. DataSift augments the social data with information such as sentiment and language analysis. For example, use DataSift to access profile data with providers such as Klout and sentiment information that use salience scoring.

You create an account and configure filters in DataSift. In each filter, you select the social media websites that you want to track and then define the filter and search conditions.

Use PowerExchange for DataSift to read the data from the filter into a DataSift data object through the Data Integration Service. You can load the extracted data to a target and then use the data for data mining and analysis.

Example

Your organization needs to track the Wikipedia data of a retail service. You create a DataSift filter to track the Wikipedia data of a retail service. You define a filter that tracks a retail service such as a chain of stores. You can monitor the social data such as the stores that generate the maximum positive tweets, the sentiments of male and female social users, and the effect of promotional drives. You can load the extracted data from the filters to a database and then use the data for data mining and analysis.

For information about DataSift, see the DataSift documentation.

Real-Time and Historical DataSift Data Extraction

You can extract real-time and historical data from DataSift. Before you extract data, you must configure a DataSift connection, a DataSift data object, and a DataSift object operation.

You can create streams in DataSift. When you specify the hash values of the streams in the data object operation, the Integration Service extracts real-time data from DataSift. If you have not defined streams in DataSift and you want to extract real-time data, you can specify search filters that use Curated Stream Definition Language (CSDL) in a file. The Data Integration Service uses the CSDL at run time to extract data from DataSift.

The Integration Service extracts historical data from DataSift when you specify the hash value and the time period for which you want to extract data. If you do not specify the start and end time, the Integration Service extracts real-time data from DataSift.

CHAPTER 2

PowerExchange for DataSift Configuration

This chapter includes the following topics:

PowerExchange for DataSift Configuration Overview

PowerExchange for DataSift is installed with the Informatica services. You enable PowerExchange for DataSift with the PowerExchange for DataSift license key.

Before you use PowerExchange for DataSift, complete the prerequisite tasks.

Prerequisites

PowerExchange for DataSift requires services and accounts to be available.

Before you use PowerExchange for DataSift, complete the following tasks:

1. Install Informatica services and the Developer Tool.
   - Create a Data Integration Service and a Model Repository Service.
2. Create an account in DataSift.
   - Subscribe to a DataSift plan.
   - The social media websites are listed as data sources in DataSift. Select and activate the data sources you plan to monitor.

Data Extraction Prerequisites

Before you extract data from DataSift, you can perform the following steps:

- If you want to create streams in DataSift, define the streams, and note the hash value for each stream. You can later use the hash value in the Developer Tool to specify the streams in DataSift from which you want to read real-time or historical data.
- If you do not want to create streams in DataSift, but you want to extract real-time data from DataSift, perform the following tasks:
  - Create a file and specify the filter conditions that use the Curated Stream Definition Language (CSDL) code in it. CSDL is a programming language that defines the filtering conditions. For more information about CSDL, see the Language Guide at the DataSift site: http://dev.datasift.com/csdl. You can later specify the location of the file in the properties of the DataSift data object operation in the Developer Tool. The Integration Service uses the CSDL code in the file at run time to extract data from DataSift.

Configuring HTTP Proxy Options

If your organization uses a proxy server to access the internet, you can configure the HTTP proxy server authentication settings for the Developer Tool and the Data Integration Service.

- Verify if the Informatica domain Administrator has configured the HTTP proxy server authentication settings for the Data Integration Service. For more information about configuring HTTP Proxy Options for the Data Integration Service, see the "HTTP Proxy Server Properties" in the *Informatica Application Service Guide*.
- To enable the proxy server settings from the web browser on the machine that runs the Developer Tool, perform the following tasks:
  - Access the `developerCore.ini` file from the following location:

    ```
    <Informatica Installation Location>\clients\DeveloperClient
    ```

  - Add the following properties to the `developerCore.ini` file:

Property	Description
-Dhttp.proxyHost=	Name of the HTTP proxy server.
-Dhttp.proxyPort=	Port number of the HTTP proxy server.
-Dhttp.proxyUser=	Authenticated user name for the HTTP proxy server. This is required if the proxy server requires authentication.
-Dhttp.proxyPassword=	Password for the authenticated user. This is required if the proxy server requires authentication. **Note:** The password is in plain text and not encrypted.

Property	Description
-Dhttp.nonProxyHosts=	List of host names or IP addresses for which you must not use the proxy server. Separate the list of IP addresses or host names with a pipe symbol (|). For example, `localhost:10.20.30.40|myHost` Specify the IP address or name of the machine on which the Informatica gateway node runs so that the Developer tool connects to the domain.
-Dhttps.proxyHost=	Name of the HTTPS proxy server.
-Dhttps.proxyPort=	Port number of the HTTPS proxy server.

CHAPTER 3

DataSift Connections

This chapter includes the following topics:

DataSift Connections Overview

Create a DataSift connection to create data objects, preview data, and run mappings.

Use a DataSift connection to extract data from DataSift filters. You must have a DataSift account before you create a DataSift connection.

DataSift Connection Properties

Use a DataSift connection to extract data from the DataSift streams. A DataSift connection is a social media connection. You can create and manage a DataSift connection in the Administrator tool or the Developer tool.

Note: The order of the connection properties might vary depending on the tool where you view them.

The following table describes DataSift connection properties:

Property	Description
Name	Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Description	The description of the connection. The description cannot exceed 765 characters.
Location	The domain where you want to create the connection.

Property	Description
Type	The connection type. Select DataSift.
Username	User name for the DataSift account.
API Key	API key. The Developer API key that appears in the Dashboard or Settings page in the DataSift account.

Creating a DataSift Connection

Create a DataSift connection before you import physical data objects.

1. Click **Window** > **Preferences**.
2. Select **Informatica** > **Connections**.
3. Expand the domain in the **Available Connections**.
4. Select a connection type in **Social Media** > **DataSift** and click **Add**.
5. Enter a connection name and optional description.
6. Click **Next**.
7. Enter the user name and API key.
8. Click **Test Connection** to verify the connection to the DataSift web site.
9. Click **Finish**.

CHAPTER 4

DataSift Data Objects

This chapter includes the following topics:

DataSift Data Objects Overview

Create a DataSift data object to read data from DataSift filters. A DataSift data object is a physical data object that represents data based on a DataSift filter.

When you create a DataSift data object, you can add a resource to it. The resource is the stream that you create in DataSift. The streams provide data such as salience, klout, and user information.

You can configure the data object read operation properties that determine how data can be read from DataSift. You can use the DataSift data object operation as a source in mappings and mapplets.

DataSift Data Object Views

The DataSift data object contains views to edit the object name and the properties.

After you create a DataSift data object, you can change the data object and data object operation properties in the following data object views:

- **Overview** view. Use the **Overview** view to edit the DataSift data object name, description, and connection.
- **Data Object Operation** view. Use the **Data Object Operation** view to modify the properties that the Data Integration Service uses when it reads data from DataSift.

When you create a mapping that uses DataSift sources, you can view the data object properties in the **Properties** view.

Real-Time Data Extraction

You can extract real-time data from DataSift. After you create an account in DataSift, you can configure streams. You must then specify the default hash value of the streams in the Advanced properties of the data object operation.

You can specify a single hash value or multiple hash values to extract data from multiple streams.

To extract real-time data, set the following properties in the data object read operation:

Default Hash

You can specify the streams from which you want to read data. You can specify a stream name for each stream. In the output, the stream name port indicates the stream that the data belongs to.

CSDL List File Location

If you have not defined streams in DataSift and you want to extract real-time data, you can create a file and specify search filters that use Curated Stream Definition Language (CSDL).

You can specify multiple CSDL codes in a file in the following pipe-separated format:

```
<CSDL_code1>|<stream_name1>||<CSDL_code2>|<stream_name2>
```

For each CSDL code, you can specify a stream name. Use two pipe symbols to delimit each set of CSDL code and stream name. The stream name port in the output indicates which code the data is from.

For example, `interaction.content CONTAINS_ANY "table tennis, Cricket - Twenty20 World Cup, football, hockey, olympics"|Sports||interaction.content CONTAINS_ANY "iphone, ipad, smart phone"|Phone`

When you specify the CSDL List File Location in the advanced properties of the data object operation, the Data Integration Service uses the CSDL at run time to extract data from DataSift. DataSift validates the CSDL code and returns the specified social data.

Note: Every time you modify and save a another version of the stream in DataSift, the hash value of the stream changes. Ensure that you use the hash value for the correct version of the stream.

Historical Data Extraction

You can extract historical data from DataSift. You can extract historical data only for a Twitter source. After you create an account in DataSift, you can configure a stream. When you create a data object operation, you can specify the DataSift stream from which you want to extract the data.

You specify each stream by providing the hash value that DataSift assigns to a stream. To extract historical data, you can specify a single stream. Specify the hash value of the stream that contains Twitter as source and the start and end time for the duration for which you want to extract data.

To extract historical data, set the following properties in the data object read operation:

- Default Hash
- Historics Start Time

- Historics End Time

Note: If you do not specify all the properties to extract historical data, the Data Integration Service generates an error message.

Specify the start and end time as a UNIX time stamp. For example, you want to extract data from January 1, 2012 at 2:00 p.m. through January 1, 2012 at 4:00 p.m. Enter `1325383200` as the Historics Start Time and `1325390400` as the Historics End Time.

The Advanced properties that you set to extract historical data takes precedence over the properties that you set to extract real-time data. For example, you set the Default Hash, Historics Start Time, Historics End Time, and the Hash List File Location. The Data Integration Service runs a query to extract historical data and does not consider the location of the hash file list.

DataSift Data Object Overview Properties

The **Overview** properties include general properties that apply to the DataSift data object. They also include object properties that apply to the resources in the DataSift data object.

General Properties

You can modify the name, description, and the connection of the data object in the general properties.

The following table describes the general properties that you configure for DataSift data objects:

Property	Description
Name	Name of the DataSift data object.
Description	Description of the DataSift data object.
Connection	Name of the DataSift connection.

Object Properties

You can modify the resource name and description in the object properties.

The following table describes the object properties that you can configure for DataSift resources:

Property	Description
Name	Name of the resource.
Type	Type of the resource. This property is read only.
Description	Description of the resource.

DataSift Data Object Read Operation Properties

The data object read operation properties include general, ports, sources, and advanced properties that the Data Integration Service uses to read data from DataSift.

When you create a data object operation, the Developer tool creates a source and output object. The source object is named after the resource and represents the data that the Data Integration Service reads from DataSift. Select the source object to view the General, Column, and Advanced properties.

The output object is named Output and represents the data that the Data Integration Service passes into the mapping pipeline. Select the output object to view the General, Ports, Sources, and Advanced properties.

General Properties

The general properties list the name and description of the source object of the data object operation.

The following table describes the general properties that you can view for a DataSift data object operation:

Property	Description
Name	Name of the source object of the DataSift data object operation.
Description	Description of the source object of the DataSift data object operation.

Column Properties

The column properties represent the data that the Data Integration Service reads from DataSift.

The following table describes the column properties that you can view for a DataSift data object operation:

Property	Description
Name	Metadata name. A DataSift filter includes the metadata such as demographic, feed, and interaction. In addition to the metadata received from DataSift filters, the column called stream_name indicates the user-specified stream name.
Type	Native data type of the metadata.
Precision	Maximum number of characters for string data types.
Scale	Maximum number of digits after the decimal point for numeric values.
Description	Description of the metadata.

Advanced Properties

The advanced properties list the resource physical name of the source object.

Ports Properties

The ports properties represent the data that the Data Integration Service passes into the mapping pipeline. The native data types are mapped to transformation data types and the resulting data properties are listed in the ports properties.

The following table describes the ports properties that you can view for a DataSift data object operation:

Property	Description
Name	Metadata name. A DataSift stream includes metadata such as demographic, feed, and interaction. In addition to the metadata received from DataSift filters, the port called stream_name indicates the user-specified stream name.
Type	Transformation data type of the metadata.
Precision	Maximum number of characters for string data types.
Description	Description of the metadata.

Sources Properties

The sources properties lists the resource of the DataSift data object operation. You can have only one stream as a resource in a DataSift data object operation. You use the Advanced properties to specify multiple DataSift filters.

Advanced Properties

The advanced properties include the run-time properties of the output object. You can specify the hash values of the filters from which you want to extract data. If you want to extract real-time data and you have not created filters in DataSift, you can specify the filter conditions in a file that uses the CSDL code and specify the file location. When you run the mapping, the Data Integration Service uses the hash values or the CSDL code to extract the data. You can also specify how long you want the extraction to continue.

When you extract real-time data and if you specify more than one property, the Data Integration Service extracts the data for the combined list. For example, you specify a default hash and provide a CSDL file location, the output consists of data retrieved from the combined list of hash values and CSDL code.

You can specify the hash value of the filter from which you want to extract historical data along with the start and end times.

The following table describes the advanced properties that you configure for a DataSift data object operation:

Property	Description
Operation Type	The operation type. This field is a read-only parameter.
Default Hash	Hash value and name for a single filter. Enter a hash value and hash name delimited by a pipe symbol in the following format: `<hash_value>|<stream_name>` where hash_value is the hash for the filter you created in DataSift. You can specify a stream_name for the filter. The name is not case sensitive, and it must be unique. It cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? / Configure for real-time and historical data extraction.
Historics Start Time	Start date and time for the time period for which you want to extract data. Specify the start time as a UNIX time stamp. Configure for historical data extraction.

Property	Description
Historics End Time	End date and time for the time period for which you want to extract data. Specify the end time as a UNIX time stamp. Configure for historical data extraction.
Hash List File Location	UNC file location of file that contains list of hash values for multiple streams. You can specify multiple streams in a file in the following pipe-separated format: <Hash_value>|<stream_name> <Hash_value>|<stream_name> Where Hash_value is the hash for the stream you have created in DataSift. You can specify a name for each stream. The name is not case sensitive and must be unique. It cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? / Configure for real-time data extraction.
CSDL List File Location	UNC file location of file that contains list of CSDL codes and user-specified CSDL names. You can specify multiple CSDL codes in a file in the following pipe-separated format: <CSDL_code1>|<stream_name1>||<CSDL_code2>|<stream_name2> For each code, you can specify a name. Use two pipe symbols to delimit each set of CSDL code and stream name. The name is not case sensitive and must be unique. It cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? / Configure for real-time data extraction.

Ends After	Duration for which the Data Integration Service runs the mapping. Enter the duration in the following format:
	`hh:mm`
	For example, specify the following duration to run the mapping for 10 days:
	`240:00`
	If you leave this option blank, the Data Integration Service runs the mapping until you stop it.
	Configure for real-time data extraction.

Subscription ID	Subscription ID of the Historics query.
Cursor	Pointer to the location from where the data retrieval starts. You must use the cursor in combination with Subscription ID.
Historics ID	Playback ID or Historics ID generated for the Historics query. The Historics ID serves as a unique identifier for the Historics query. Use the Historics ID to resume the Historics query.
On cancel	Defines the behavior of the Historics query when the Historics query is canceled. Select one of the following values: - On Cancel. Cancels the Historics query. - On Pause. Pauses the Historics query.

Property	Description
Maximum reconnection attempts	Maximum number of attempts to re-establish a connection to DataSift if a connection fails. Default is 3. Set to -1 for infinite attempts.
Connection retry interval	The number of seconds between two consecutive connection retry attempts. Default is 120.

Parameterization

You can parameterize the DataSift connection and read operation properties to override the properties at run time.

The following table lists the read operation properties that you can parameterize and the type of parameterization supported:

Property	Type of Parameterization Supported
Default Hash	Full
Hash List File Location	Partial
CSDL List File Location	Partial
Ends After	Full
Historics Start Time	Full
Historics End Time	Full
Subscription ID	Full
Cursor	Full
Historics ID	Full
Maximum reconnection attempts	Full
Connection retry interval	Full

Creating a DataSift Data Object

Create a DataSift data object to specify a DataSift filter.

You configure a DataSift connection before you create a DataSift data object.

1. Select a project or folder in the **Object Explorer** view.
2. Click **File** > **New** > **Data Object**.
3. Select **DataSift Data Object** and click **Next**.

   The **New DataSift Data Object** dialog box appears.
4. To select the target project or folder, click **Browse** next to the **Location** option.
5. Click **Browse** next to the **Connection** option and select a connection from which you want to import the DataSift resource.
6. To add a resource to the Data Object, click **Add** next to the **Resource** option.

   The **Add sources to the data object** dialog box appears.
7. Navigate or search for the filter resource to add to the data object and click **OK**.
8. Select **Stream** and click **OK**.

   You can have only one filter as a resource in a DataSift data object operation. Use the Advanced properties of the data object operation to specify multiple filters.
9. Optionally, enter a name for the data object.
10. Click **Finish**.

    The data object appears under Data Object in the project or folder in the **Object Explorer** view.

Creating a DataSift Data Object Operation

Create a data object operation from a data object. You can have one filter as a resource in a DataSift data object operation. Use the Advanced properties of the data object operation to specify multiple filters.

You must create the data object with the resource before you create a data object operation.

1. Select the data object in the Object Explorer view.
2. Right-click and select **New** > **Data Object Operation**.

   The **Data Object Operation** dialog box appears.
3. Enter a name for the data object operation.
4. Select the type of data object operation.
5. Click **Add**.

   The **Select a resource** dialog box appears.
6. Select **Stream**.
7. Click **Finish**.

   The Developer tool creates the data object operation for the selected data object. Use the Advanced properties of the data object operation to specify the filters from which you want to extract data.

Troubleshooting a Historical Data Extraction

What happens when you run a query to extract historical data and the Data Integration Service shuts down unexpectedly or becomes unavailable.

If the Data Integration Service shuts down unexpectedly or if you choose to abort the Data Integration Service, the query either stops or pauses. The pause or stop is based on whether you have set the query to stop or pause in the **On Cancel** settings in the advanced properties. When you pause or stop, a historics query triggers a subscription ID and a historics ID. Find the historics ID from the mapping logs, go to DataSift API console, and stop or pause the query in DataSift.

CHAPTER 5

DataSift Mappings

This chapter includes the following topics:

DataSift Mappings Overview

After you create the DataSift data object operation, you can develop a mapping.

You can create an Informatica mapping that contains objects such as a DataSift data object operation as the input. You can add transformations and a target to load data to. Validate and run the mapping to extract the DataSift data and load it to a target.

Output Data Parsing

You can use transformations to parse the output data of a DataSift filter.

The output data of a DataSift filter is in the JSON format. The Data Integration Service reads the JSON document that contains an array of JSON objects and passes the data into the mapping pipeline. Sample transformations are provided for each of the augmentations.

To use the sample transformations in the Developer Tool, perform the following tasks:

1. Import the sample transformations that are available in the following folder: `<Informatica Services Installation Directory>\clients\DeveloperClient\samples\socialmedia\datasift\transformations`
2. To use the Java transformations, specify the classpath for the JAR files located in the following location: `<Informatica Services Installation Directory>\clients\DeveloperClient\infacmd`
   The Java transformations require the following JAR files:
   - com.fasterxml.jackson.core-jackson-annotations-2.0.6.jar
   - com.fasterxml.jackson.core-jackson-core-2.0.6.jar

- com.fasterxml.jackson.core-jackson-databind-2.0.6.jar

3. To use the feed port, use the sample Router transformation that is provided. You can use the Java transformations for all the other ports.
4. Configure the input and output ports.

For example, you can extract the Facebook demographics, interaction, and Klout data. The data in JSON format for each augmentation is hierarchical, with nested elements such as source, author, and type.

The following code illustrates a sample of the data in JSON format for Facebook interaction data:

```
{
"interaction": {
"type": "facebook",
"author": {
"name": "John Doe",
"avatar": "https://graph.facebook.com/111111111/picture",
"link": "http://www.facebook.com/profile.php?id=111111111",
"id": "777777777"
},
"content": "Presidential Elections!",
"source": "Facebook for iPhone",
"id": "1e150e26da22a780e06635591f6759f4",
 "created_at": "Mon, 06 Feb 2012 16:48:43 +0000"
},
```

Each of the augmentations such as demographics, interaction, and Klout are available in individual ports in the read data object. Link each port to the input port of the sample Java transformation. The output ports of the Java transformation provide each element of the augmentation. You can write all the elements to a single target or select the elements you want to extract.

DataSift API Rate Limits

DataSift API imposes rate limits based on the type and the number of requests to provide appropriate usage of API resources to all users.

For information about rate limiting, see the DataSift documentation at the following web site:

http://dev.datasift.com/docs/rest-api/api-rate-limiting

The following rules and guidelines apply to DataSift rate limits:

- Streaming API is not rate limited but all other API requests have rate limits.
- Historics API have rate limits.
- Every time a hash value is validated at run time, it incurs API rate cost.

DataSift Mapping Example

Your organization, HypoMarket Corporation, wants to monitor and analyze the spectrum of positive sentiments it receives for its new range of designer wear across all its stores in the six countries where it is marketed. HypoMarket intends to analyze the data for future product launches.

Create a filter in DataSift that monitors the sentiments of social users from Tumblr. Create a mapping that reads the filter from DataSift and writes those records to a table.

You can use the following objects in the DataSift mapping:

DataSift input

The mapping source is a DataSift data object.

Create a DataSift connection and a DataSift data object. Specify the hash value of the filter. Specify the duration for which you want to run the mapping. You can begin monitoring the feeds when the promotional sale begins.

Java transformation

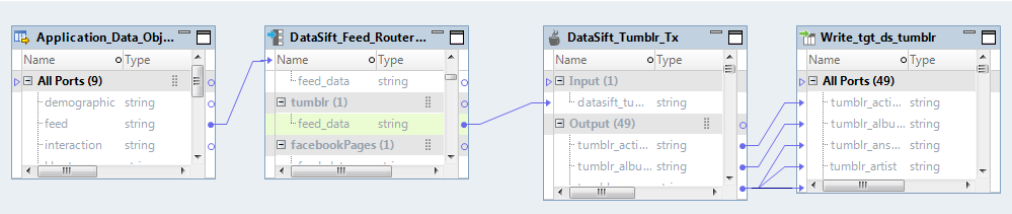
Use the sample Java transformation to parse the output.

Import the transformation, specify the classpath, and connect the ports.

Mapping output

Add a relational data object to the mapping as a target.

The following mapping shows the mapping from the source to the target:



After you run the mapping, the Data Integration Service writes the extracted data to the target table. You can use the data to analyze the customer sentiments.

APPENDIX A

Data Type Reference

This appendix includes the following topics:

Data Type Reference Overview

Informatica Developer uses the following data types in DataSift mappings:

- DataSift native data types. DataSift data types appear in the physical data object column properties.
- Transformation data types. Set of data types that appear in the transformations. They are internal data types based on ANSI SQL-92 generic data types, which the Data Integration Service uses to move data across platforms. Transformation data types appear in all transformations in a mapping.

When the Data Integration Service reads source data, it converts the native data types to the comparable transformation data types before transforming the data. When the Data Integration Service writes to a target, it converts the transformation data types to to the comparable native data types.

DataSift and Transformation Data Types

The following table lists the DataSift data types that Data Integration Service supports and the corresponding transformation data types:

DataSift Data Type	Transformation Data Type	Range and Description
String	String	1 to 104,857,600 characters

INDEX