CHAPTER 1

Introduction to Data Integration Hub

This chapter includes the following topics:

Data Integration Hub Overview

Data Integration Hub is an application integration solution that your organization can use to share and synchronize data between different applications in the organization.

To publish data to Data Integration Hub, first define the data set that you want to manage, for example, sales, customers, or orders. You define a data set by defining a topic. A topic defines the structure of the data that Data Integration Hub stores in the publication repository and the type of publication repository where data is stored. You can manage multiple topics that represent different data sets in Data Integration Hub. Applications publish data to topics and subscribe to data sets that are represented by topics.

Multiple applications can publish to the same topic, for example, applications at different stores can publish their orders to the same Orders topic. Multiple subscribers can consume the data from a topic. Different subscribing applications can consume the data in different formats and in different latencies based on a defined schedule.

Data Integration Hub stores the data that applications publish to topics in the Data Integration Hub publication repository. Data Integration Hub keeps the data in the publication repository until the retention period expires, and then deletes the data from the publication repository.

Applications can use PowerExchange connectors and Informatica Cloud connectors to share data from different sources, such as database tables, files, or any sources that Informatica supports. Each application can be a publisher and a subscriber to different topics.

Publications publish to a specific topic. A publication defines the data source type and the location from where Data Integration Hub retrieves the data that the application publishes. Subscriptions subscribe to one

or more topics. A subscription defines the data target type and the location in the subscribing application to where Data Integration Hub sends the published data.

When you create a publication or a subscription, you can choose to use either an automatic Data Integration Hub mapping or a custom Data Integration Hub mapping. Data Integration Hub creates automatic mappings based on the data structure that you define in the topic. Custom Data Integration Hub mappings are based either on PowerCenter workflows or on Informatica Cloud data synchronization tasks that the developer creates and maintains for the publication or for the subscription.

Examples

You run a data center for a major retail chain. The main office has multiple applications. Some of the applications are located on premise and some are located on the cloud. Each retail branch has a point-of-sale (POS) application and an inventory application. Your applications and branches require the following data:

Customer service applications

Require up-to-date customer order data.

Sales applications

Require up-to-date product sales data.

Marketing application

Requires a weekly deals report.

Accounting application

Requires a monthly deals report.

Branch applications

Require up-to-date inventory and pricing data.

With Data Integration Hub, you can address the following use-cases:

Share product catalog and prices.

You can share product price updates from the sales department with each branch.

1. Create a Products topic.
2. For the Product Information Management (PIM) application, define a publication that publishes product details and prices to the Products topic and set the schedule to publish the data daily.
3. For each branch application, define a subscription to the Products topic and set the subscription to consume the published data when it is available in Data Integration Hub.

Share daily sales details.

You can share the daily sales details that you receive from the stores with your central sales application and your customer service applications.

1. Create a Sales topic.
2. For each branch application, define a publication to the Sales topic, and set the schedule to publish daily.
3. For the sales application, define a subscription to the Sales topic, and set the schedule to consume the data when it is published.
4. For the customer service application, define a subscription to the Sales topic, and set the schedule to consume the data once a week.

Share deal details from Salesforce.

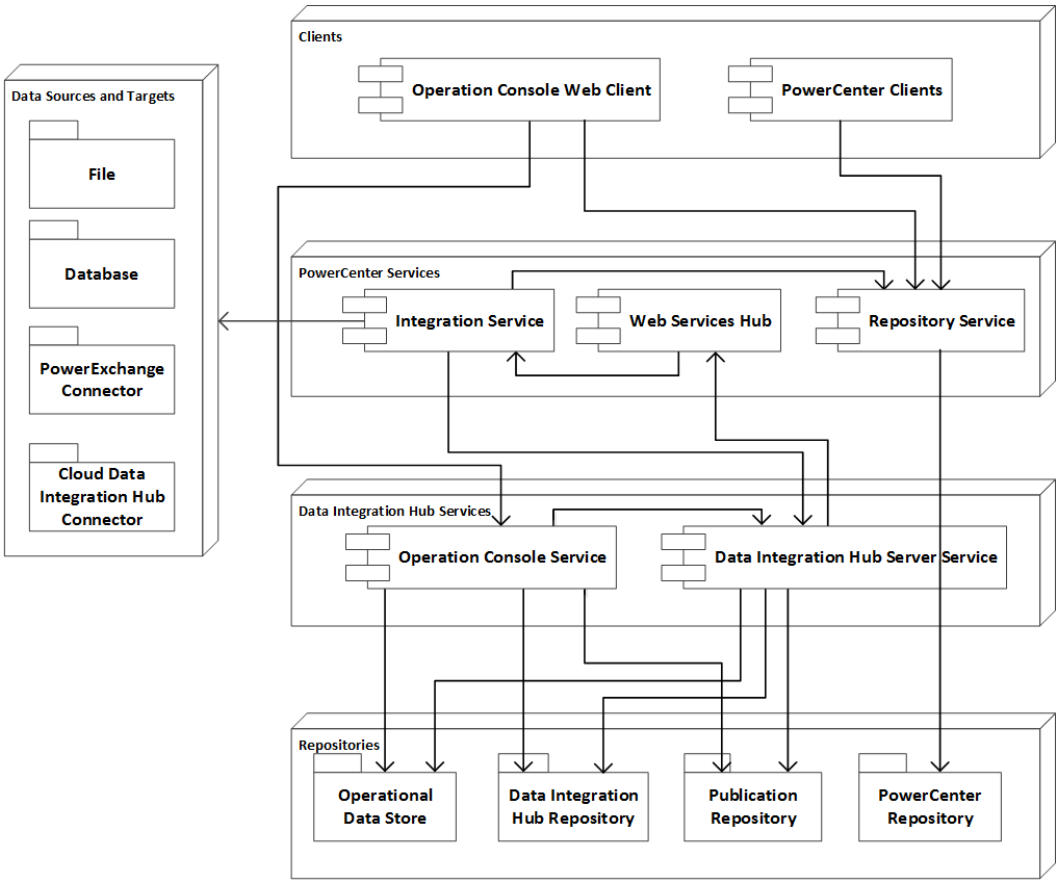
You can share deal details from a Salesforce cloud application with the marketing and accounting applications.

1. Create a Deals topic.
2. For the Salesforce application, define a cloud publication to the Deals topic, and set the schedule to publish weekly.
3. For the marketing application, define a subscription to the Sales Deals, and set the schedule to consume the data once a week.
4. For the accounting application, define a subscription to the Sales Deals, and set the schedule to consume the data once a month.

Data Integration Hub Architecture

The Data Integration Hub environment consists of user interface clients, PowerCenter services, Data Integration Hub services, and repositories.

The following image shows the Data Integration Hub components:



Data Integration Hub contains the following components:

Data Integration Hub Operation Console Web client

User interface to manage applications, topics, publications, and subscriptions, and to monitor publications, subscriptions, and events. Administrators also use the Operation Console to manage users and system settings. Developers use the Operation Console to manage workflows and connections.

This component sends commands to the Operation Console service.

PowerCenter Clients

User interfaces to define sources and targets, build custom mappings, and create workflows. Use the PowerCenter Clients if you use workflows in custom mappings.

This component writes metadata to the PowerCenter Repository Service.

PowerCenter Integration Service

Service that retrieves data from publication sources and sends the data to subscription targets.

This component sends data to the data targets, receives data from the data sources, receives commands from the PowerCenter Web Services Hub, and sends commands to the Data Integration Hub server.

PowerCenter Web Services Hub

Gateway that connects Data Integration Hub to PowerCenter through web services.

This component sends commands to the PowerCenter Integration Service and receives commands from the Data Integration Hub server.

PowerCenter Repository Service

Service that creates and modifies repository metadata for the PowerCenter Integration Service to use when a workflow runs.

This component receives commands from the PowerCenter Client and sends commands to the PowerCenter repository.

Note: Each Data Integration Hub instance requires a dedicated PowerCenter Repository Service.

Data Integration Hub Operation Console service

Service that processes actions that users perform on the Operation Console and creates the structure for published data sets in the publication repository.

This component receives commands from the Operation Console client and sends commands to the Data Integration Hub repository and the publication repository.

Data Integration Hub Server service

Service that starts and monitors PowerCenter batch workflows for publications and subscriptions.

This component receives commands from the PowerCenter Integration Service and the Operation Console service, and sends commands to the PowerCenter Web Services Hub and to the Data Integration Hub repository.

Data Integration Hub repository

Database that stores metadata for applications, topics, publications, subscriptions, and events.

This component receives data from the Operation Console service and the Data Integration Hub server.

Data Integration Hub publication repository

Database that stores published data until the subscribers consume the data. After the data retention period ends, Data Integration Hub deletes the data from the publication repository.

This component receives data from the PowerCenter Integration Service and the Operation Console service.

PowerCenter repository

Database that stores metadata for PowerCenter mappings, workflows, and transformations.

This component receives data from the PowerCenter Repository Service.

Operational data store

A repository that contains aggregated information for reporting purposes. When you install the Data Integration Hub Dashboard and Reports component of Data Integration Hub, Data Integration Hub creates the operational data store repository based on the database connection details that you supply.

This component receives data from the Data Integration Hub server service and the operational data store event loader. The component sends data to the Data Integration Hub server service, the operational data store event loader, and the Operation Console service. The Data Integration Hub server service, the operational data store event loader, and the Operation Console service read data from this component.

Data sources and targets

Sources and targets that you use to publish and consume data. You can use the following types of sources and targets:

- Database. Tables and columns.
- File. Binary, text, or unstructured files.
- PowerExchange connector. Connection objects for PowerExchange adapters. Available when you use a custom mapping.
- Cloud Data Integration Hub connector. Connection objects for Informatica Cloud adapters. Available when you use a custom mapping.

Operation Console

Use the Operation Console user interface to manage applications, topics, publications, and subscriptions, and to monitor publications, subscriptions, and events. Administrators also use the Operation Console to manage users and system settings. Developers use the Operation Console to manage workflows and connections.

You can view the Operation Console in English or in Japanese. You can switch between the display languages.

The Operation Console contains two areas:

Navigator

Use the navigator to navigate between tasks that you can perform in the Operation Console. The navigator shows in the left pane of the Operation Console.

Current page

Main work area in which you perform the tasks that you select in the Navigator. The current page shows in the right pane of the Operation Console.

Changing the Operation Console Language

You can view the Operation Console in English or in Japanese. You can switch between the display languages.

1. In the browser from where you access Data Integration Hub, set the language to the required language.
2. The **Help** link opens the online help in English. To view the Japanese online help access the following URL:

   `http(s)://<host>:<port>/dih-help-ja`

   Where:

   - `<host>` is the host name or the IP address of the Data Integration Hub server.
   - `<port>` is the port number of the Data Integration Hub server.

   For example:

   `https://dih-releases:19443/dih-help-ja/`

Data Integration Hub Topics

A Data Integration Hub topic is an entity that represents a data domain that is published and consumed in Data Integration Hub. A topic defines the canonical data structure and additional data definitions such as the data retention period.

For example, a Sales topic that represents sales data. Applications from all the stores in the organization publish sales data to the Sales topic. The accounting application subscribes to the Sales topic and consumes published sales data from all stores, or, if a filter is applied, from specific stores.

Before you define publications and subscriptions for the data that is published and consumed in Data Integration Hub, you need to define the canonical structure that will hold the data that is published to Data Integration Hub in the Data Integration Hub publication repository. You define the canonical structure when you define the topic. You can define multiple topics that represent different source data sets.

Publication and Subscription Process

Publications and subscriptions are entities that define the type, format, and schedule of data flow between applications in your organization. Publications publish data to a defined topic and subscriptions subscribe to topics.

The publication process starts on the schedule that you define in the publication, when you manually run the publication, or when an external process triggers the publication.

When data transfer is complete, the topic data set is ready for subscribers when one of the following conditions exist, based on the configuration of data consumption in the subscriptions:

- When data is published to the topic.
- When all publishers that publish to the topic finish publishing.

If the topic to which the data is published has subscribes, Data Integration Hub triggers a Data Integration Hub subscription workflow for each subscriber, to consume the data.

Data Integration Hub generates events to track the progress of the publication and subscription process. When an application publishes data, Data Integration Hub creates a parent publication event. When the publication process ends and the published data is ready to consume, Data Integration Hub generates a child event for each subscription.

The events change status as the publication and subscription process progresses, and reach a completed status after the process ends successfully. You also use events to monitor and troubleshoot issues that might occur during the process.

During the publication or the subscription process Data Integration Hub communicates either with PowerCenter or with Informatica Cloud, based on the type of publication or subscription:

- For publications and subscriptions that use automatic mappings or custom mappings and that publish from and subscribe to on-premise applications, Data Integration Hub uses PowerCenter mappings.
  - When an on-premise application publishes a data set, the Data Integration Hub server triggers the PowerCenter workflow that is defined for the publication and instructs the PowerCenter Integration Service to retrieve the data from the publishing application.
  - The PowerCenter Integration Service runs the workflow that transfers the source data to the Data Integration Hub publication repository.
  - When data transfer is complete, the PowerCenter Integration Service notifies the Data Integration Hub server that the topic data set is ready for subscribers.
  - If the topic to which the data is published has subscribes, Data Integration Hub triggers the subscription workflows to consume the data.
  - For on-premise subscriptions, the PowerCenter Integration Service runs the workflow that transfers the published data from the Data Integration Hub publication repository to the subscribing application.
- For publications and subscriptions that publish from and subscribe to cloud applications, you use custom mappings with Informatica Cloud tasks.
  - When a cloud application publishes a data set, the Data Integration Hub server triggers the Informatica Cloud task that is defined for the publication through an Informatica Cloud REST API.
  - For cloud publications, the target is defined using a Data Integration Hub cloud connector. The publication process uses the connector to write the data to Data Integration Hub.
  - If the topic to which the data is published has subscribes, Data Integration Hub triggers the subscription workflows to consume the data.
  - For cloud subscriptions, the source is defined using a Data Integration Hub cloud connector. The subscription process uses the connector to read data from Data Integration Hub.
  - Data Integration Hub monitors the task for processing status.

Publication Process

The publication process includes retrieving the data from the publisher, running the publication mapping, and writing the data to the relevant topic in the publication repository. After the publication process ends, each subscriber consumes the published data according to the schedule and the filter that you define when you create the subscription.

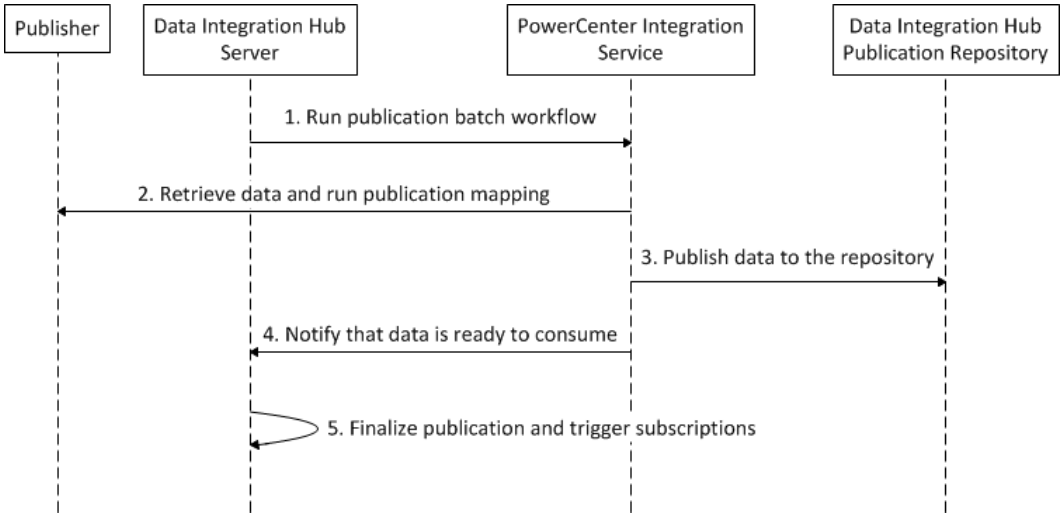
The publication process depends on the type of workflow that the publication runs. Publications with an automatic mapping run a batch workflow. Publications with a custom mapping can run either a batch workflow or a real-time workflow.

Batch Workflow Process

The publication process for publications that run a batch workflow includes the following stages:

1. When the publisher is ready to publish the data, the Data Integration Hub server runs the publication batch workflow and sends a request to the PowerCenter Integration Service.
2. The PowerCenter Integration Service extracts the data from the publisher and runs the automatic or custom mapping on the data.
3. The PowerCenter Integration Service writes the data to the Data Integration Hub publication repository.
4. The PowerCenter Integration Service notifies the Data Integration Hub server that the published data is ready for subscribers.
5. The Data Integration Hub server changes the status of the publication event to complete and triggers subscription processing.

The following image shows the main stages of the publication process for publications that run a batch workflow:



Real-time Workflow Process

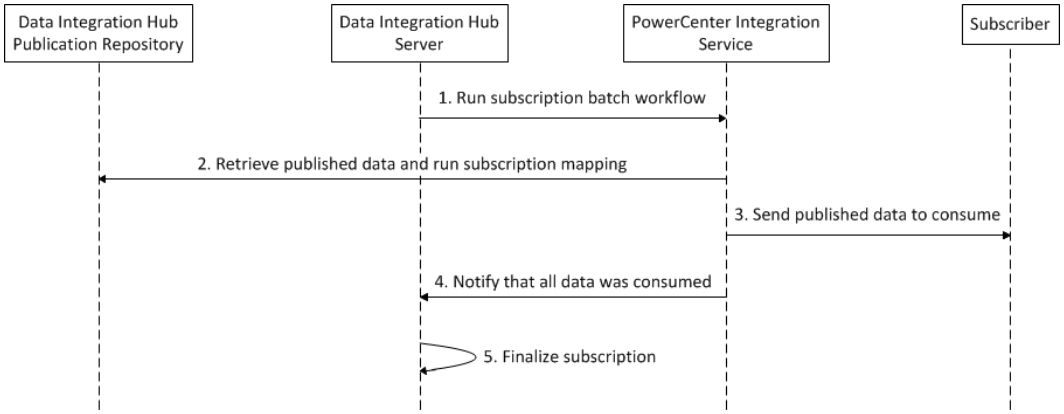
The publication process for publications that run a real-time workflow includes the following stages:

1. The developer runs the real-time workflow. The workflow writes the data to the relevant tables in the Data Integration Hub publication repository.
2. The Data Integration Hub server triggers a scheduled process and checks for new data in the relevant tables in the Data Integration Hub publication repository.
3. If new data is found, Data Integration Hub updates the publication ID and the publication date of the data to indicate that the data is ready for consumption and creates a publication event in the Data Integration Hub repository.
4. The Data Integration Hub server changes the status of the publication event to complete and triggers subscription processing.

Subscription Process

The subscription process includes retrieving the required data from the Data Integration Hub publication repository, running the subscription mapping, and writing the data to one or more subscriber targets. Data Integration Hub keeps the data in the publication repository until the retention period of the topic expires.

The following image shows the main stages of the subscription process for each subscription:



The subscription process includes the following stages:

1. When the publication is ready for subscribers, the Data Integration Hub server runs the subscription batch workflow and sends a request to the PowerCenter Integration Service.
2. The PowerCenter Integration Service extracts the data from the Data Integration Hub publication repository, and runs the automatic or custom mapping on the data.
3. The PowerCenter Integration Service sends the required data to the subscriber.
4. The PowerCenter Integration Service notifies the Data Integration Hub server after the subscriber consumed the published data that they require.
5. The Data Integration Hub server changes the status of the subscription event to complete.

Developer User Role

The Data Integration Hub developer is responsible for developing workflows for custom mappings and customizing workflow parameters for the workflows with the Forms Designer. The developer also manages source and target connections for publications and subscriptions in the Operation Console.

CHAPTER 2

PowerCenter Integration

This chapter includes the following topics:

PowerCenter Integration Overview

PowerCenter is a tool the you use to process Data Integration Hub publications and subscriptions. You use the PowerCenter Client tools to develop workflows that you use in publications and subscriptions with custom mappings.

Automatic Data Integration Hub Mappings

When you create a publication or a subscription with an automatic mapping, Data Integration Hub creates PowerCenter components that process the publication or subscription according to the definitions that you set in the Publication or Subscription wizard.

Data Integration Hub creates the following PowerCenter entities for a publication or a subscription with an automatic mapping:

- Batch workflow. Scheduled batch workflow that the Data Integration Hub triggers to process the data.
- Data processing session. Session with transformations that extract the data from the publication source or write the data to the subscription target. The session also includes a mapping, source and target objects.
- Reporting session. Session that sends and receives notifications about the publication or subscription process. This session updates the publication or subscription events after processing. The session also includes a mapping, source and target objects.
- Metadata folders in the PowerCenter repository. The folders contain workflow, mapping, and session information. A separate folder contains publication or subscription metadata. When you create an automatic publication mapping, PowerCenter creates subscription source shortcuts based on the publication target.

Automatic Data Integration Hub Mappings Rules and Guidelines

When you publish or consume data sets with automatic mappings, consider the following rules and guidelines:

- Data Integration Hub creates PowerCenter entities for each automatic mapping. Each PowerCenter entity name starts with the prefix **DIH__**. Do not manually change or delete the PowerCenter entities.
- You cannot use high precision processing in PowerCenter to run automatic mappings. To use high precision, use a custom mapping.
- Do not manually change or delete tables in the publication repository that store data sets from publications with automatic mappings.
- Automatic mappings run with the DIH__STAGING and the DIH__REPO PowerCenter connections. To prevent publication or subscription failure, do not change or delete the connections.
- When you publish varbinary or nvarchar datatypes from a Microsoft SQL Server database source, PowerCenter converts the data size limit to a fixed size of 100 MB. If you create a subscription with an automatic mapping that writes the published data to a Microsoft SQL Server database target, you must change the data size limit to MAX when you create the target tables before you run the subscription workflow.
- When you use tables from a default schema to publish from a relational database or to subscribe to a relational database, the schema name is not persistent.
- When you use tables from a non default schema to publish from a relational database or to subscribe to a relational database, the schema name is persistent.
- Schema names are saved in the Mapping tab of the PowerCenter session in the following locations:
  - For publications, the schema name is saved in the Source properties in the **Owner Name** field.
  - For subscriptions, the schema name is saved in the Target properties in the **Target Name Prefix** field.

Automatic Data Integration Hub Mappings Logs

When you publish or consume data sets with automatic mappings, the PowerCenter Integration Service creates the following logs:

Publication workflow logs

The PowerCenter Integration Service creates publication workflows with the following settings:

Parameter	Value
Save Workflow Log By	Runs
Save Workflow Log for These Runs	$PMWorkflowLogCount

The value of *$PMWorkflowLogCount* in the PowerCenter Integration Service determines the number of logs that the Integration Service creates for each workflow run.

Publication session logs

The PowerCenter Integration Service creates publication workflow sessions with the following settings:

Parameter	Value
Save Session Log By	Runs
Save Session Log for These Runs	$PMSessionLogCount

The value of *$PMSessionLogCount* in the PowerCenter Integration Service determines the number of logs that the Integration Service creates for each session run.

Subscription logs

The PowerCenter Integration Service creates subscription workflows with a concurrent run set. The number of logs is unlimited with timestamps.

Custom Data Integration Hub Mappings

If you need to prepare data before you publish it, or if you require additional processing or transformation to the data set to publish or consume, you can develop PowerCenter workflows and import them to Data Integration Hub.

You develop the workflows for publication pre-processing and for publications and subscriptions with custom mappings in a similar way that you develop other PowerCenter workflows. You then select the workflow to use in a publication pre-process, or in a publication or a subscription with a custom mapping.

Supported Datatypes

When you publish data sets to the publication repository, you can publish datatypes based on the database type of the publication repository.

The following table describes the supported datatypes you can publish to the publication repository:

Database Type	Datatypes
Oracle	- blob - clob - number - number($p, $s) - timestamp up to 29 bit - varchar2($l char)
Microsoft SQL Server	- bigint - datetime - decimal($p,$s) - float - int - nvarchar($l) - nvarchar(MAX) - varbinary(MAX)

Custom Mappings Rules and Guidelines

When you develop PowerCenter workflows to use in publications and subscriptions with custom mappings, consider the following rules and guidelines:

- Name and store PowerCenter entities for custom mappings in different folders with a different naming convention from the naming convention of PowerCenter entities for custom mappings.
- Data Integration Hub uses a separate reporting session to update the status of publication and of subscription events that use an automatic mapping. It is recommended to create separate sessions for data processing and reporting, similar to automatic mappings. You can use a workflow for an automatic mapping as a reference.
- You can use user-defined session parameters in custom workflows and define their values in Data Integration Hub or in a parameter file. You can manage the values of user-defined session parameters in Data Integration Hub in the Forms Designer. You cannot manage the values of built-in session parameters in Data Integration Hub. For more information about session parameters, see the section "Working with Session Parameters" in the *PowerCenter Advanced Workflow Guide*.

  Note: Data Integration Hub does not support parameters in the format `$InputFile_$$CustomVariable`.
- If you publish from a database source, you cannot use the following special characters in table names and in column names of a publication target: space ( ), dash (-), and period (.). The publication process replaces the characters with underscores (_).
- When you develop a publication pre-process workflow, call the DX_Start_Publication transformation at the end of the pre-processing workflow, for example in a separate mapping. The transformation instructs the Data Integration Hub server to trigger the publication process. When you configure the DX_Start_Publication transformation, consider the following guidelines:
  - When a publication pre-process starts a single publication, use the DXEventId port. The event ID ensures that Data Integration Hub uses the same event for both the publication pre-process workflow and the publication workflow and changes the event status accordingly.

    Note: If you do not define a DXEventId port you must define a DXPublicationName port.
  - When a publication pre-process starts multiple publications, do not use the event ID in the DX_Start_Publication transformation. In this case, you can use the Event Details PowerCenter transformation to change the event status.

  Note: Do not call the DX_Start_Publication transformation more than once in a workflow. If you do, Data Integration Hub starts the publication multiple times.
- To prevent naming conflicts, do not use **_DIH__** in the parameter names, and do not use workflow and mapping parameters with the same names as workflow and mapping parameters that Data Integration Hub uses in workflows for publications and subscriptions with automatic mappings.
- If you publish from a database source or write to a database target with a different database type from the publication repository database type, Data Integration Hub converts the data to a data type that the publication repository database supports. Therefore, if you consume the published data from the publication repository to a different target database, verify that the data type conversion does not create run-time errors during processing. For example, if you publish data from a Microsoft SQL Server database source to an Oracle publication repository, and then consume the published data to a Microsoft SQL Server database target, MIN or MAX values might be converted to a value that is higher or lower than values that the Microsoft SQL Server database target supports.
- When you develop a workflow for a compound subscription, define the behavior if the compound subscription starts manually before all published data sets are ready to consume. For example, you can instruct the mapping to fail the workflow or to ignore empty tables. Published data sets that are not ready to consume have the publication instance ID 0.
- When you develop a workflow for a publication with a file source, if the path of the source file is parameterized, Data Integration Hub picks up the file and moves it to the Data Integration Hub document

store. If the path of the source file is hard coded, a PowerCenter source picks up and processes the file. For source files with a parameterized file path, the following rules apply:

- For flat file sources, the source file type must be indirect.
- For pass-through file sources, the source file type must be direct.

- When you develop a workflow for a subscription with a file target, you can parameterize the target file path. The following rules and guidelines apply when you parameterize the file path:
  - For flat file targets, the target file parameter must start with `$OutputFile`.
  - For pass-through file targets, the target file parameter must start with `$OutputFile_DIHRepoFile_`.
  - When the Data Integration Hub operator creates the subscription in the Data Integration Hub Operation Console, they must specify the target output file name as the value for the output file parameter.
  - The value of the output file parameter can contain a pattern that ensures that the name is unique for each file, for example `($sequence)`.

Developing Custom Mappings

You develop PowerCenter workflows for Data Integration Hub custom mappings in the same way that you develop other PowerCenter workflows. Data Integration Hub transformations interact directly with Data Integration Hub.

Before you develop Data Integration Hub workflows in PowerCenter, verify that the Data Integration Hub PowerCenter client and server plug-ins are installed and registered to the PowerCenter repository. For details, see the *Data Integration Hub Installation and Configuration* Guide.

To use the workflow in Data Integration Hub as a publication workflow or a subscription workflow, create a Data Integration Hub workflow in the Data Integration Hub Operation Console by selecting the workflow in the PowerCenter repository or by selecting the exported workflow definition file. For more information, see "Managing Workflows in the Operation Console" on page 45.

When you select a PowerCenter workflow to use in a publication with a custom mapping, Data Integration Hub creates the structure of the published data set in the publication repository based on the target definitions of the workflow.

When you add PowerCenter transformations to a PowerCenter mapping, you can add workflow parameters that the corresponding Data Integration Hub workflow can use. You can use the Forms Designer, in the Data Integration Hub Operation Console, to customize the layout and behavior of the workflow parameters that appear when the operator creates or edits a publication or a subscription with a custom mapping.

If you edit the PowerCenter workflow or mapping, you must update the Data Integration Hub workflow by re-selecting the PowerCenter workflow or the XML file in the Data Integration Hub Operation Console. During the update process, you can resolve changes to workflow parameters and parameter types. When you change the type of a workflow parameter type, the import process deletes the values for the imported parameters and you must manually enter the parameter values.

Developing Custom Mappings Process

To develop a workflow in PowerCenter that processes Data Integration Hub data, perform the following steps:

1. Create the source and target definitions.
2. Create the mapping and add transformations.

3. Create the workflow and the session.
4. Save the PowerCenter workflow to the PowerCenter repository.

Step 1. Create the Source and Target Definitions

When you develop a publication workflow, you create the source based on the structure of the database or file that you want to publish and set the target to the publication repository.

If you publish from a database source, verify that you publish datatypes that the publication repository can store.

When you develop a subscription workflow, you create a source based on the topic structure from the publication repository. You can copy the source from the publication metadata folder in the PowerCenter repository.

You create the source and target definitions in the PowerCenter Designer in the same way that you create source and target definitions for other mappings. For general information about source and target definitions, see the *PowerCenter Designer Guide*.

Creating the Source Definition for a Subscription Workflow

In the PowerCenter Designer, create the source and define the source properties of a subscription workflow. When you develop a publication workflow, you define the source based on the application from which you want to publish data.

The source definition process includes the following steps:

1. Create the source object. Set the source connection to DIH__STAGING. You create a source based on the topic structure from the publication repository. You can copy the source from the publication metadata folder in the PowerCenter repository.
2. Add variables to filter the published data to consume. For example, you can select to consume data from a specific table in the publication repository.
3. Add the required fields that determine the data set to consume. You can define multiple data sets to consume, similar to an aggregated or a compound subscription with an automatic mapping.

The following table describes the fields to add to the source object of a subscription workflow:

Field	Description
DIH__PUBLICATION_INSTANCE_ID	Required. Identifiers of one or more published data sets in a comma-separated list. Each data set that an application publishes has a unique identifier. To filter the data to consume, use the value from the **$$<topicName>__PublicationInstanceIDs** workflow parameter. The parameter datatype must be number(19) if you write to an Oracle database target or number(19,0) if you write to a Microsoft SQL Server database target.

DIH__PUBLICATION_INSTANCE_DATE	Date and time that each application started publishing the data sets, in a comma-separated list. If you use database partitions, you can filter the data to consume by using the value from the **$$<topic_name>__PublicationInstanceDatesSQL** workflow parameter. The value format depends on the publication repository database type.
	On an Oracle database, the datatype must be date and the value must be in the following format:
	`YYYY-MM-DD HH24:MI:SS`
	On a Microsoft SQL Server database, the datatype must be datetime and the value must be in the following format:
	`yyyy-mm-dd hh:mi:ss (24h)`
	Note: If you want to filter the data to consume with a different transformation, you can use the **$$<topic_name>__PublicationInstanceDates** parameter instead.

You can filter data to consume in the Source Filter attribute of the Source Qualifier transformation in subscription workflow. The following example shows the field syntax to filter by ID and date range in a single line:

```
MY_TABLE.DIH__PUBLICATION_INSTANCE_ID in ($$myTopic__PublicationInstanceIDs)
AND MY_TABLE.DIH__PUBLICATION_INSTANCE_DATE in ($$myTopic__PublicationInstanceDatesSQL)
```

Creating the Target Definition

In the PowerCenter Designer, create the target and add ports to store properties for running the workflow. You can also add properties to store other Data Integration Hub values that you want to send back from PowerCenter.

The target definition process includes the following steps:

1. Create the target object. If you develop a publication workflow, use the DIH__STAGING connection in the target definition. The database target type must match the publication repository database type.
2. If you develop a publication workflow, add the required fields that determine the data set to publish.

The following table describes the required fields to add to the target object of a publication:

Parameter	Description
DIH__PUBLICATION_INSTANCE_ID	Required. Identifier of the published data set. Each data set that an application publishes has a unique identifier. The field uses the value from the $$publicationInstanceID workflow parameter.

DIH__PUBLICATION_INSTANCE_DATE	Required. Date and time that the application started publishing the data set. The field uses the value from the $$publicationInstanceDate workflow parameter. The value format depends on the publication repository database type.
	On an Oracle database, the value must be in the following format:
	`to_date('<DATE>', 'YYYY-MM-DD HH24:MI:SS')`
	On a Microsoft SQL Server database, the value must be in the following format:
	`convert(datetime,'<DATE>',120)`

Step 2. Create the Mapping

Create a mapping that contains the source definition, target definition, and transformations that you want to use to process data for Data Integration Hub.

You create a mapping for Data Integration Hub in the same way you build other PowerCenter mappings. Use Data Integration Hub transformations and Unstructured Data transformations to add product-specific functionality to the mapping. You can also add other PowerCenter transformation to the mapping.

Step 3. Create the PowerCenter Workflow and Session

Create the workflow and the session that runs the mapping. You create the workflow in the same way you create other PowerCenter workflows.

1. In the Workflow Designer, create the workflow and the session object. For general information about creating workflows and sessions, see the *PowerCenter Advanced Workflow Guide*.
2. Add the transformations that notify the Data Integration Hub server that the publication or that the subscription process ended.
   - If you develop a publication workflow, add the DX_Notification transformation to trigger the subscription workflows and change the publication event status.
   - If you develop a subscription workflow, add the DX_Event_Details to change the subscription event status.
3. Save the session object.
4. Test the workflow to ensure that it works correctly.

Step 4. Save the PowerCenter Workflow

Save the workflow to the PowerCenter repository. After you save the workflow, you can export it from PowerCenter to an XML file that will be used as the workflow definition file. You must use the Repository Manager to export the workflow.

Note: Do not save the workflow to a folder where the folder name is prefixed by `DIH__pub` or by `DIH__sub`.