Informatica®

Informatica® Cloud Data Integration

FHIR Connector

Informatica Cloud Data Integration FHIR Connector
July 2024

© Copyright Informatica LLC 2024

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, the Informatica logo, Informatica Cloud, and PowerCenter are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at https://www.informatica.com/trademarks.html. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

See patents at https://www.informatica.com/legal/patents.html.

DISCLAIMER: Informatica LLC provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica LLC does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2024-07-18

Table of Contents

Preface 4

Chapter 1: Introduction to FHIR Connector 5

FHIR Connector overview 5

Chapter 2: Connections for FHIR 6

Connections for FHIR 6

Connect to FHIR 6

- Connection details 6
- Authentication types 8

Chapter 3: Mappings and mapping tasks with FHIR 11

FHIR objects in mappings 11

- FHIR sources in mappings 11
- Request message 12
- FHIR targets in mappings 13

FHIR mapping example 15

Index 16

Preface

Use *FHIR Connector* to learn how to access resources on a Fast Healthcare Interoperability Resources (FHIR) server in Data Integration. Learn to create a FHIR connection and develop and run mappings and mapping tasks.

CHAPTER 1

Introduction to FHIR Connector

This chapter includes the following topic:

- FHIR Connector overview, 5

FHIR Connector overview

You can use FHIR Connector to access resources on a Fast Healthcare Interoperability Resources (FHIR) server in Data Integration.

You can create a FHIR connection and use the connection in mappings or mapping tasks. You can also use the connection in mapping templates for industry solutions. Create a mapping task to process data based on the data flow logic defined in a mapping.

CHAPTER 2

Connections for FHIR

This chapter includes the following topics:

- Connections for FHIR, 6
- Connect to FHIR, 6

Connections for FHIR

Create a FHIR connection to securely read data from or write data to a FHIR server. You can use FHIR connections to specify sources and targets in mappings and mapping tasks.

You create a FHIR connection on the **Connections** page. You can then use the connection in the Mapping Designer when you create a mapping. You can also create a FHIR connection in a Source or Target transformation in a mapping.

Connect to FHIR

Let's configure the FHIR connection properties to connect to a FHIR server.

Connection details

The following table describes the basic connection properties:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Type	FHIR

Property	Description
Runtime Environment	Name of the runtime environment where you want to run tasks. Select a Secure Agent.
Host	Host name or IP address of the FHIR server, including the port number. Enter the value in the following format, where host_name can be a host name or IP address: `host_name:port_number`
HTTP Method	HTTP method used to send requests. Select one of the following HTTP methods: - HTTP - HTTPS Default is HTTP.
Connection Timeout	Maximum number of seconds to wait when attempting to connect to the server. A timeout occurs if a successful connection doesn't occur in the specified amount of time. If the value is 0 or blank, the wait time is infinite. Default is 30 seconds.
Keep Alive	Indicates whether to keep the connection open for multiple HTTP requests or responses. Default is true.
Follow Redirects	Indicates whether to follow redirect links when creating a connection. Default is true.
Connection Retry Attempts	Number of times to retry connecting to the FHIR server if a successful connection doesn't occur. This setting applies to both the initial connection and any reconnect attempts due to lost connections. Default is 0.
Connection Retry Interval	Number of seconds to wait between each connection retry attempt. For example, to retry to connect up to 10 times with a five-second delay between retries, set **Connection Retry Attempts** to 10 and **Connection Retry Interval** to 5. Default is 0.
Base Path	Base path for the FHIR server. The initial URL segment of the API.
Test Connection Resource Path	Resource path to append to the base path to test the connection.
Content Type	Media type of the request. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json
Accept	Media type of the response. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json

Property	Description
Additional Headers	Additional headers that the connection requires. Define headers in JSON format. For example: `[{"Name":"Content-Type","Value":"application/fhir+json"},{"Name":"accept","Value":"text/xml"}]`
Authentication Type	Authentication method that the connector must use to connect to the REST endpoint. You can use one of the following options: - None - Basic. For more information, see "Basic authentication" on page 9. - OAuth 2.0 authorization code. For more information, see "OAuth 2.0 authorization code authentication" on page 9. - OAuth 2.0 client credentials. For more information, see "OAuth 2.0 client credentials authentication" on page 10. Default is None.
Trust Store File Path	Absolute path of the truststore file that contains the TLS certificate to establish a one-way or two-way secure connection with the REST API. Ensure that the truststore file is in .jks format. Specify a directory path that is available on each Secure Agent machine in the runtime environment.
Trust Store Password	Password for the truststore file that contains the SSL certificate.
Key Store File Path	Absolute path of the keystore file that contains the keys and certificates required to establish a two-way secure communication with the REST API. Ensure that the keystore file is in .jks format. Specify a directory path that is available on each Secure Agent machine in the runtime environment.
Key Store Password	Password for the keystore file required for secure communication.
Proxy Type	Type of proxy. You can select one of the following options: - No Proxy. Bypasses the proxy server configured at the agent or the connection level. - Platform Proxy. Considers the proxy configured at the agent level. - Custom Proxy. Considers the proxy configured at the connection level. Not applicable when you use a serverless runtime environment.
Proxy Config	Host name or IP address of the proxy server, including the port number. Enter the value in the following format, where host_name can be a host name or IP address: `host_name:port_number`

Authentication types

You can configure basic, OAuth 2.0 authorization code, and OAuth 2.0 client credentials authentication to access the FHIR server.

Select the required authentication method and then configure the authentication-specific parameters.

Basic authentication

Basic authentication requires the user name and password from the FHIR server.

The following table describes the connection properties for basic authentication:

Property	Description
Auth User ID	User name to log in to the web service application.
Auth Password	Password associated with the user name.

OAuth 2.0 authorization code authentication

Configure authentication properties in the FHIR connection to use an OAuth 2.0 authorization code.

To use authorization code authentication, register the following Informatica redirect URL in your application:

```
https://<Informatica cloud hosting facility for your
organization>/ma/proxy/oauthcallback
```

If the access token expires and you receive error code 400, 401, or 403 in the response, the Informatica redirect URL tries to connect to the endpoint and retrieve a new access token. Note that the Informatica redirect URL is usually outside the organization firewall.

The following table describes the authentication properties for a FHIR connection that uses an OAuth 2.0 authorization code:

Property	Description
Authorization Token URL	Authorization server URL configured in your application.
Access Token URL	Access token URL configured in your application.
Client ID	Client ID of your application.
Client Secret	Client secret of your application.
Scope	Specifies access control if the API endpoint defines custom scopes. Separate scope attributes using a space. For example: `root_readonly root_readwrite manage_app_users`
Access Token Parameters	Additional parameters to use with the access token URL. Define parameters in JSON format. For example: `[{"Name":"resource","Value":"https://<serverName>"}]`
Authorization Code Parameters	Additional parameters to use with the authorization token URL. Define parameters in JSON format. For example: `[{"Name":"max_age","Value":60},{"Name":"state","Value":"test"}]`
Client Authentication	Select an option to send the client ID and client secret for authorization either in the request body or in the request header. Default is **Send Client Credentials in Body**.

Property	Description
Access Token	Enter the access token value or click **Generate Access Token** to populate the access token value. To generate the access token through a proxy server, configure an unauthenticated proxy server on the Secure Agent. The FHIR connection-level proxy configuration doesn't apply when generating the access token.
Refresh Token	Enter the refresh token value or click **Generate Access Token** to populate the refresh token value. If the access token is not valid or expires, the Secure Agent generates a new access token through the refresh token. If the refresh token expires, you must either enter a valid refresh token or generate a new refresh token by clicking **Generate Access Token**.

OAuth 2.0 client credentials authentication

Configure authentication properties in the FHIR connection to use OAuth 2.0 client credentials.

The following table describes the authentication properties for a FHIR connection that uses OAuth 2.0 client credentials:

Property	Description
Access Token URL	Access token URL configured in your application.
Client ID	Client ID of your application.
Client Secret	Client secret of your application.
Scope	Specifies access control if the API endpoint defines custom scopes. Separate scope attributes using a space. For example: `root_readonly root_readwrite manage_app_users`
Access Token Parameters	Additional parameters to use with the access token URL. Define parameters in JSON format. For example: `[{"Name":"resource","Value":"https://<serverName>"}]`
Client Authentication	Select an option to send the client ID and client secret for authorization either in the request body or in the request header. Default is **Send Client Credentials in Body**.
Access Token	Enter the access token value or click **Generate Access Token** to populate the access token value. To generate the access token through a proxy server, configure an unauthenticated proxy server on the Secure Agent. The FHIR connection-level proxy configuration doesn't apply when generating the access token.

CHAPTER 3

Mappings and mapping tasks with FHIR

This chapter includes the following topics:

- FHIR objects in mappings, 11
- FHIR mapping example, 15

FHIR objects in mappings

When you create a mapping, you can configure a Source or Target transformation to represent a FHIR source or target. In the transformation, select the FHIR connection and then configure the REST API operation and request message.

FHIR sources in mappings

When you configure a Source transformation, select the FHIR connection and use a GET operation to access a FHIR resource. You can configure the request message using the request message template.

The following table describes the FHIR source properties that you can configure in a Source transformation:

Property	Description
Connection	Name of the source connection.
Operation	REST API operation to perform on the FHIR resource. Select a GET operation.

The following table describes the advanced properties that you can configure in a Source transformation:

Property	Description
Path	Path to the FHIR resource, such as /Patient. This path is appended to the base path in the FHIR connection.
Request Body Type	Type of input to send in the request body. Select buffer or file.

Property	Description
Content Type	Media type of the request. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json Overrides the media type of the request specified in the connection.
Accept	Media type of the response. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json Overrides the media type of the response specified in the connection.
Additional Headers	Additional headers that the connection requires. Define headers in JSON format. For example: `[{"Name":"Content-Type","Value":"application/fhir+json"},{"Name":"accept","Value":"text/xml"}]` Overrides the additional headers specified in the connection.
Convert request/ response data format	Converts the data format in the GET response between XML and JSON. If the response is in XML format, this option converts it to JSON format. If the response is in JSON format, this option converts it to XML format.
Response Directory	Directory to store the response.
Response Headers Directory	Directory to store response headers.
Tracing Level	Amount of detail that appears in the log for the source. Use the following tracing levels: - Terse - Normal - Verbose Initialization - Verbose Default is Normal.
Cache Size for Web Service Response (KB)	Memory available for the web service response. If the web service response contains many rows or columns, you might want to increase the cache size. Default is 1024 KB. Maximum is 99999 KB.
Optimize Transformation	Select this option to improve performance for processing records from hierarchical to relational format or from relational to hierarchical format.

Request message

In the Source transformation, you can configure the request message to specify request headers, path parameters, and query parameters to access a resource on a FHIR server.

You can configure the following headers and parameters in the request message:

requestHeaders

Request headers are additional headers that the connection requires.

Define headers in JSON format. For example:

```
[{"Name":"Content-Type","Value":"application/fhir+json"},
{"Name":"accept","Value":"text/xml"}]
```

Note: You can configure request headers in the request message, the FHIR connection properties, and the advanced properties of the Source transformation. When you define a header in the request message, it overrides the request headers in the connection properties and in the advanced properties.

pathParams

Path parameters are a sequence of values that replace the values in the path of the request message.

For example, you might want to access a FHIR resource at the following path:

```
https://hapi.fhir.org/baseR4/Patient/593251/_history/2
```

In the advanced properties, configure the following path:

```
/Patient/{id}/{history}/{version}
```

Then, in the request message, configure the following value for pathParams:

```
<pathParams>59321,_history,2<pathParams/>
```

queryParams

Query parameters are used to query resources on the FHIR server using search criteria that are available for the resource.

For example, you can use the following query parameters to access a patient in `https://hapi.fhir.org/baseR4/Patient`:

```
<queryParams>given=Benjamin;identifier=bcook74;gender=male
<queryParams/>
```

At run time, the query parameters resolve to `https://hapi.fhir.org/baseR4/Patient?given=Benjamin&identifier=bcook74&gender=male`.

You can find examples of search parameters and modifiers in the FHIR Specification documentation. However, not all search criteria and modifiers are available in mappings.

FHIR targets in mappings

When you configure a Target transformation, select the FHIR connection and use a REST API operation to update the FHIR resource.

The following table describes the FHIR target properties that you can configure in a Target transformation:

Property	Description
Connection	Name of the target connection.
Operation	REST API operation to perform on the FHIR resource. Select a POST, PUT, DELETE, or PATCH operation.

The following table describes the advanced properties that you can configure in a Target transformation:

Property	Description
Path	Path to the FHIR resource, such as /Patient. This path is appended to the base path in the FHIR connection. For example, if the base URL of the FHIR server is `https://hapi.fhir.org/baseR4`, this property appends `/Patient` to access a patient resource.
Request Body Type	Type of input to send in the request body. Select buffer or file.
Content Type	Media type of the request. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json Overrides the media type of the request specified in the connection.
Accept	Media type of the response. Select one of the following options: - application/fhir+xml - application/fhir+json - application/xml - application/json Overrides the media type of the response specified in the connection.
Additional Headers	Additional headers that the connection requires. Define headers in JSON format. For example: `[{"Name":"Content-Type","Value":"application/fhir+json"},{"Name":"accept","Value":"text/xml"}]` Overrides the additional headers specified in the connection.
Convert request/ response data format	Converts the data format in the REST API request body between XML and JSON. If the request body is in XML format, this option converts it to JSON format. If the request body is in JSON format, this option converts it to XML format.
Response Directory	Directory to store the response.
Response Headers Directory	Directory to store response headers.
Tracing Level	Amount of detail that appears in the log for the source. Use the following tracing levels: - Terse - Normal - Verbose Initialization - Verbose Default is Normal.
Cache Size for Web Service Response (KB)	Memory available for the web service response. If the web service response contains many rows or columns, you might want to increase the cache size. Default is 1024 KB. Maximum is 99999 KB.
Transaction Commit Control	Control to commit or roll back transactions based on the set of rows that pass through the transformation. Use the transaction commit control if you have a large amount of data and you want to control how it is processed.

Property	Description
Transaction Scope	Scope of the transaction. Select all input or only the transaction.
Optimize Transformation	Select this option to improve performance for processing records from hierarchical to relational format or from relational to hierarchical format.

FHIR mapping example

In this example, you use a FHIR connection to access a patient's records on the HAPI FHIR test server using a GET REST API operation.

1. In Data Integration, click **New** > **Mappings** > **Mapping** and then click **Create**.
2. In the Source transformation, select the FHIR connection.
3. Select GET as the operation.
4. In the request options, configure the request message.
5. Copy and paste the following request message:

```
<!--1 or more repetitions:-->
<proc:GET_INPUT xmlns:proc="http://xml.schemas/infa/procedure/">
  <!--Optional:-->
  <GET>
    <!--Optional:-->
    <requestHeaders/>
<!--Optional:-->
    <pathParams>11111,_history,<pathParams/>
    <!--Optional:-->
    <queryParams>given=Benjamin;id=2123;identifier=bcook74;gender=male<queryParams/>
  </GET>
</proc:GET_INPUT>
```

6. In the advanced properties, configure the following properties:
   a. Enter `/Patient` as the path.
   b. Select buffer as the request body type.
   c. Enable the option to convert the request data format.
   d. Enter `/root/FHIR/` as the response directory.
7. On the **Field Mapping** tab, map the output of the GET request to the output field.
8. In the Target transformation, configure a flat file connection and use a POST request to write the output to a flat file.

INDEX

C

connections
  FHIR connection properties 6

F

FHIR
  connection properties 6