Information Builders

iWay

Omni-HealthData™ API Services Reference Guide

Version 3.11

November 19, 2019

Contents

Preface

This documentation provides a reference for Omni-HealthData™ API Services. This manual is intended for developers and administrators of Omni-HealthData™.

How This Manual Is Organized

This manual includes the following chapters:

	Chapter/Appendix	Contents
1	Omni-HealthData Instance and Master Services	Describes services used for subject instance and master management.
2	Omni-HealthData Change Data Capture	Describes the Omni-HealthData change data capture (CDC) services, which allow a client to subscribe to changes of instance or master records.
3	Omni-HealthData Bulk Load	Describes the Omni-HealthData bulk loading API, which supports data loads to Omni-HealthData directly from OnRamp tables or from Omni Interface Documents (OIDs), which are converted into OnRamp tables.
4	Omni Server Status Operations	Describes the Omni Server status API calls, which allow for server state discovery, its health, and operation mode.
5	Omni Controller Deployed Bundle Artifacts	Describes API calls to download artifacts for a deployed bundle.
6	Omni-HealthData Metadata Services	Describes the Omni-HealthData metadata services, which are available as part of the Omni-HealthData Controller REST API, and allow a client to manage metadata for a domain.
7	Omni-HealthData Source Management Services	Describes the Omni-HealthData source management services that you can use to document Omni-HealthData interactions with external systems. They are available as part of the Omni-HealthData Controller REST API.

Documentation Conventions

The following table lists and describes the documentation conventions that are used in this manual.

Convention	Description

`THIS TYPEFACE`
or
`this typeface`
	Denotes syntax that you must type exactly as shown.

this typeface	Represents a placeholder (or variable), a cross-reference, or an important term. It may also indicate a button, menu item, or dialog box option that you can click or select.
underscore	Indicates a default setting.
Key + Key	Indicates keys that you must press simultaneously.
{}	Indicates two or three choices. Type one of them, not the braces.
|	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
...	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis (...).
. . .	Indicates that there are (or could be) intervening or additional commands.

Related Publications

Visit our Technical Documentation Library at *http://documentation.informationbuilders.com*. You can also contact the Publications Order Department at (800) 969-4636.

Customer Support

Do you have questions about this product?

Join the Focal Point community. Focal Point is our online developer center and more than a message board. It is an interactive network of more than 3,000 developers from almost every profession and industry, collaborating on solutions and sharing every tips and techniques. Access Focal Point at *http://forums.informationbuilders.com/eve/forums*.

You can also access support services electronically, 24 hours a day, with InfoResponse Online. InfoResponse Online is accessible through our website, *http://www.informationbuilders.com*. It connects you to the tracking system and known-problem database at the Information Builders support center. Registered users can open, update, and view the status of cases in the tracking system and read descriptions of reported software issues. New users can register immediately for this service. The technical support section of *www.informationbuilders.com* also provides usage techniques, diagnostic tips, and answers to frequently asked questions.

Call Information Builders Customer Support Services (CSS) at (800) 736-6130 or (212) 736-6130. Customer Support Consultants are available Monday through Friday between 8:00 A.M. and 8:00 P.M. EST to address all your questions. Information Builders consultants can also give you general guidance regarding product capabilities. Be prepared to provide your six-digit site code (*xxxx.xx*) when you call.

To learn about the full range of available support services, ask your Information Builders representative about InfoResponse Online, or call (800) 969-INFO.

Help Us to Serve You Better

To help our consultants answer your questions effectively, be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following table lists the environment information that our consultants require.

Platform	
Operating System	
OS Version	
JVM Vendor	
JVM Version	

The following table lists additional questions to help us serve you better.

Request/Question	Error/Problem Details or Information
Did the problem arise through a service or event?	
Provide usage scenarios or summarize the application that produces the problem.	
When did the problem start?	
Can you reproduce this problem consistently?	
Describe the problem.	
Describe the steps to reproduce the problem.	
Specify the error messages.	
Any change in the application environment: software configuration, EIS/database configuration, application, and so forth?	
Under what circumstance does the problem *not* occur?	

The following is a list of error and problem files that might be applicable.

- ❑ Input documents (XML instance, XML schema, non-XML documents)
- ❑ Transformation files
- ❑ Error screen shots
- ❑ Error output files
- ❑ Trace files
- ❑ Custom functions and agents in use

- Diagnostic Zip
- Transaction log

User Feedback

In an effort to produce effective documentation, the Technical Content Management staff welcomes your opinions regarding this document. Please use the Reader Comments form at the end of this document to communicate your feedback to us or to suggest changes that will support improvements to our documentation. You can also contact us through our website, *http://documentation.informationbuilders.com/connections.asp*.

Thank you, in advance, for your comments.

iWay Software Training and Professional Services

Interested in training? Our Education Department offers a wide variety of training courses for iWay Software and other Information Builders products.

For information on course descriptions, locations, and dates, or to register for classes, visit our website, *http://education.informationbuilders.com*, or call (800) 969-INFO to speak to an Education Representative.

Interested in technical assistance for your implementation? Our Professional Services department provides expert design, systems architecture, implementation, and project management services for all your business integration projects. For information, visit our website, *http://www.informationbuilders.com/consulting*.

Chapter 1

Omni-HealthData Instance and Master Services

The services described in this section are used for subject instance and master management. These services allow a consumer to read and write instances and read and sync with masters.

In this chapter:

Load Subject Instance

Use the following REST call to initiate the load and mastering process for a subject in Omni-HealthData. Omni-HealthData will take the subject information in the form of an OID XML and will perform requested cleansing, matching, and merging of the subject.

Omni-HealthData can process fully qualified or partial OID XML documents. The required SourceName and SourceInstanceId elements are used by Omni-HealthData to determine whether a subject is new or an update to a previous instance. Omni-HealthData will construct the best possible subject document by intelligently combining the current subject instance with previous instance data. This intelligent combination may be configured at the document, source, subject, or source/subject levels.

PUT: /server/api/v1/server/instance?mode=&timeout=

The data must be a correctly formatted OID XML, which minimally requires a SourceName and SourceInstanceId. The OID may be fully qualified or partially qualified.

- mode – Specify the loading mode of sync or async. The default is async, as sync will block the caller until the data is fully processed or a timeout occurs.
- timeout – When mode=sync, the caller will be blocked until the document is fully processed or until this timeout is exceeded.

Example:

PUT https://localhost:9514/server/api/v1/server/instance

Content-Type: application/xml

```
<customer>
       <SourceName>ReadTest</SourceName>
       <SourceInstanceId>ReadTestCustomer</SourceInstanceId>
       <cu_title>DEVELOPER</cu_title>
       <cu_salutation>MR</cu_salutation>
       <cu_suffix>PHD</cu_suffix>
       <cu_first_name>SCOTT</cu_first_name>
       <cu_middle_name>W</cu_middle_name>
       <cu_last_name>BISHOPP</cu_last_name>
       <cu_full_name>MR SCOTT W BISHOPP PHD</cu_full_name>
       <cu_gender>M</cu_gender>
       <cu_dob format="yyyy-MM-dd">1965-02-21</cu_dob>
       <cu_ssn>874-98-4546</cu_ssn>
       <cu_type>P</cu_type>
       <cu_bus_name>IBI</cu_bus_name>
       <cu_dba_name>SCOTTYB</cu_dba_name>
       <cust_addressOmniCollection>
         <customercust_address version="">
             <SourceName>ReadTest</SourceName>
             <SourceInstanceId>ReadTest:address</SourceInstanceId>
           <ad_1>4212 S LIVONIA RD</ad_1>
           <ad_2>ATTN: SCOTT BISHOPP</ad_2>
           <ad_city>LIVONIA</ad_city>
           <ad_state>NY</ad_state>
           <ad_zip>14487</ad_zip>
           <ad_zip4>0212</ad_zip4>
           <ad_country>US</ad_country>
         </customercust_address>
       </cust_addressOmniCollection>
   </customer>
```

- response -

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Content-Length:  202
Date:  Tue, 08 May 2018 15:34:06 GMT

<RestResponse>
  <status>0</status>
  <statusText></statusText>
  <responseType>java.lang.String</responseType>
  <response>
  Submitted request for customer:ReadTest:ReadTestCustomer
  </response>
</RestResponse>
```

Get an Instance

Use the following service to get an instance using its unique identifier.

GET: /server/api/v1/server/instance/{subject}/{sourceName}/{sourceInstanceId}

- {subject} – Required subject name.
- {sourceName} – Required source name of the instance.
- {sourceInstanceId} – Required instance identifier.

Example:

GET https://localhost:9514/server/api/v1/server/instance/customer/ReadTest/ReadTestCustomer

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Content-Length:  1256
Date:  Tue, 08 May 2018 15:42:07 GMT

<RestResponse>
<status>0</status>
<statusText></statusText>
<responseType>java.lang.String</responseType>  <response>
<customer>
  <SourceName>ReadTest</SourceName>
  <SourceInstanceId>ReadTestCustomer</SourceInstanceId>
  <MasterId>26</MasterId>
  <cu_title>Developer</cu_title>
  <cu_salutation>Mr</cu_salutation>
  <cu_suffix>Phd</cu_suffix>
  <cu_first_name>Scott</cu_first_name>
  <cu_middle_name>W</cu_middle_name>
  <cu_last_name>Bishopp</cu_last_name>
  <cu_full_name>Mr Scott W Bishopp Phd</cu_full_name>
  <cu_gender>M</cu_gender>
  <cu_dob format="yyyy-MM-dd">1965-02-21</cu_dob>
  <cu_ssn>874-98-4546</cu_ssn>
  <cu_type>P</cu_type>
  <cu_bus_name>IBI</cu_bus_name>
  <cu_dba_name>SCOTTYB</cu_dba_name>
```

```
 <cust_addressOmniCollection>
    <customercust_address>
      <SourceName>ReadTest</SourceName>
      <SourceInstanceId>ReadTest:address</SourceInstanceId>
      <ad_1>4212 S LIVONIA RD</ad_1>
      <ad_2>ATTN: SCOTT BISHOPP</ad_2>
      <ad_city>LIVONIA</ad_city>
      <ad_state>NY</ad_state>
      <ad_zip>14487</ad_zip>
      <ad_zip4>0212</ad_zip4>
      <ad_country>US</ad_country>
    </customercust_address>
  </cust_addressOmniCollection>
</customer>
</response>
</RestResponse>
```

Get a Master

Use the following service to get a master with a specified master ID.

GET: /server/api/v1/server/master/{masterSubject}/{masterId}

- {masterSubject} – Name of mastered subject.
- {masterId} – Master ID.

Example:

GET https://localhost:9514/server/api/v1/server/master//customerMaster/26

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Content-Length:  1146
Date:  Tue, 08    May 2018 15:51:20 GMT

<RestResponse>
<status>0</status>
<statusText></statusText>
<responseType>java.lang.String</responseType>  <response>
<customerMaster>
  <MasterId>26</MasterId>
  <cu_title>Developer</cu_title>
  <cu_salutation>Mr</cu_salutation>
  <cu_suffix>Phd</cu_suffix>
  <cu_first_name>Scott</cu_first_name>
  <cu_middle_name>W</cu_middle_name>
  <cu_last_name>Bishopp</cu_last_name>
```

```
  <cu_full_name>Mr Scott W Bishopp Phd</cu_full_name>
  <cu_gender>M</cu_gender>
  <cu_dob format="yyyy-MM-dd">1965-02-21</cu_dob>
  <cu_ssn>874-98-4546</cu_ssn>
  <cu_type>P</cu_type>
  <cu_bus_name>IBI</cu_bus_name>
  <cu_dba_name>SCOTTYB</cu_dba_name>
  <cust_addressOmniCollection>
    <customercust_addressMaster>
      <MasterChildId>customer:26:cust_address:1</MasterChildId>
      <ad_1>4212 S LIVONIA RD</ad_1>
      <ad_2>ATTN: SCOTT BISHOPP</ad_2>
      <ad_city>LIVONIA</ad_city>
      <ad_state>NY</ad_state>
      <ad_zip>14487</ad_zip>
      <ad_zip4>0212</ad_zip4>
      <ad_country>US</ad_country>
    </customercust_addressMaster>
  </cust_addressOmniCollection>
</customerMaster>
</response>
</RestResponse>
```


Query for a Master

Use the following REST call to check if subject data matches any currently mastered subject. This service expects data in the form of a subject OID with data used for the defined matching criteria. This service will NOT cause changes in any instance or master data, including the matching indexes.

The implementation of this service will cause execution of defined cleansing plans against the input data to ensure it is standardized before the match attempt. After cleansing, the READ ONLY matching service will be executed to see if a high confidence match and associated master are available.

If a match is available this service can respond with the master ID or the full master XML document based on the responseType parameter. If a match is not found, the service will respond with an HTTP status of 404 (Not Found).

POST: /api/v1/server/master?responseType=oid|masterId

The body of the post must be a properly formatted XML OID. Only the data used for the matching service is evaluated and other data will simply be ignored.

Note: This will NOT create data in Omni-HealthData.

Example:

POST https://localhost:9514/server/api/v1/server/master?responseType=oid

Content-Type: application/xml

```
<customer>
    <cu_first_name>SCOTT</cu_first_name>
    <cu_middle_name>W</cu_middle_name>
    <cu_last_name>BISHOPP</cu_last_name>
    <cu_gender>M</cu_gender>
    <cu_dob format="yyyy-MM-dd hh:mm:ss">1965-02-21 01:34:09</cu_dob>
    <cu_ssn>874-98-4546</cu_ssn>
</customer>
```

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Content-Length:  1146
Date:  Tue, 08 May 2018 16:00:47 GMT

<RestResponse>
<status>0</status>
<statusText></statusText>
<responseType>java.lang.String</responseType>
<response>
<customerMaster>
  <MasterId>26</MasterId>
  <cu_title>Developer</cu_title>
  <cu_salutation>Mr</cu_salutation>
  <cu_suffix>Phd</cu_suffix>
  <cu_first_name>Scott</cu_first_name>
  <cu_middle_name>W</cu_middle_name>
  <cu_last_name>Bishopp</cu_last_name>
  <cu_full_name>Mr Scott W Bishopp Phd</cu_full_name>
  <cu_gender>M</cu_gender>
  <cu_dob format="yyyy-MM-dd">1965-02-21</cu_dob>
  <cu_ssn>874-98-4546</cu_ssn>
  <cu_type>P</cu_type>
  <cu_bus_name>IBI</cu_bus_name>
  <cu_dba_name>SCOTTYB</cu_dba_name>
  <cust_addressOmniCollection>
    <customercust_addressMaster>
      <MasterChildId>customer:26:cust_address:1</MasterChildId>
      <ad_1>4212 S LIVONIA RD</ad_1>
      <ad_2>ATTN: SCOTT BISHOPP</ad_2>
      <ad_city>LIVONIA</ad_city>
      <ad_state>NY</ad_state>
      <ad_zip>14487</ad_zip>
      <ad_zip4>0212</ad_zip4>
      <ad_country>US</ad_country>
    </customercust_addressMaster>
  </cust_addressOmniCollection>
</customerMaster>
</response>
</RestResponse>
```

Chapter 2

Omni-HealthData Change Data Capture

The Omni-HealthData change data capture (CDC) services allow a client to subscribe to changes of instance or master records. When subscribed, changes are detected, and the instance or master OID will be published to a location that the subscriber designated.

In this chapter:

Subscribe to Changes

Use the following REST call to subscribe to changes for a specific subject instance or master. The payload sent to the destination will be the OID that has been modified.

POST: /server/api/v1/server/cdc/subscribe

The data body is the configuration information about where to publish, the subscriber and optional the source to watch.

```
<SubscriptionConfiguration>
     <subject/>
     <source/>
     <subscriber/>
     <connectionType/>
     <url/>
</SubscriptionConfiguration>
```

- subject – Required subject to subscribe to. This may be an instance or master subject.
- source – Optional source when subscribing to instance changes.
- subscriber – Required identifier of the subscriber.

- connectionType – Required connection type, which must be set to REST.
- url – Required when the connectionType is REST and is used to specify the endpoint that can accept the POST operation.

Response:

A RestResponse will be returned with the <response> containing the unique subscription ID.

Example:

POST https://localhost:9514/server/api/v1/server/cdc/subscribe

Content-Type: application/xml

```
<SubscriptionConfiguration>
  <subject>customer</subject>
  <subscriber>Test</subscriber>
  <connectionType>rest</connectionType>
  <url>
    https://localhost:9514/server/api/v1/server/cdc/subscribe/loopback
  </url>
</SubscriptionConfiguration>
```

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Thu, 03 May 2018 11:37:45 GMT

<RestResponse>
  <status>0</status>
  <statusText/>
  <responseType>java.lang.String</responseType>
  <response>271192d4-48d0-4e91-869e-dc1b477311be</response>
</RestResponse>
```

Cancel a Subscription

Use the following REST call to cancel a subscription.

DELETE: /server/api/v1/server/cdc/subscribe/{subscriptionId}

Specify the subscription ID to cancel.

Example:

DELETE https://localhost:9514/server/api/v1/server/cdc/subscribe/271192d4-48d0-4e91-869e-dc1b477311be

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Thu, 03 May 2018 11:39:19 GMT


<RestResponse>
  <status>0</status>
  <statusText/>
  <responseType>java.lang.String</responseType>
  <response>271192d4-48d0-4e91-869e-dc1b477311be</response>
</RestResponse>
```

Enable or Disable a Subscription

Use the following REST call to enable or disable a subscription. When a subscription is disabled no changes will be set to the subscriber.

PUT: /server/api/v1/server/cdc/subscribe/{subscriptionId}?status=ENABLE|DISABLE

Specify the subscription ID and the new status of the subscription.

Example:

PUT https://localhost:9514/server/api/v1/server/cdc/subscribe/5f1021c3-234b-4a05-ae88-86ff6af8aa8e?status=disable

Content-Type: application/xml

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Thu, 03 May 2018 11:41:36 GMT


<RestResponse>
  <status>0</status>
  <statusText/>
  <responseType>java.lang.String</responseType>
  <response>DISABLED</response>
</RestResponse>
```

Get List of Subscription

Use the following REST call to the list of current subscriptions

GET: /server/api/v1/server/cdc/subscribe

Retrieves the list of current subscriptions.

Example:

GET https://localhost:9514/server/api/v1/server/cdc/subscribe

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Tue, 08    May 2018 11:25:11 GMT

<RestResponse>
  <status>0</status>
  <statusText/>
  <responseType>java.util.ArrayList</responseType>
  <responses>
    <responseItem>
      <CdcSubscriptionInfo>
        <subscriber>Test</subscriber>
        <subject>customer</subject>
        <source/>
        <connectionType>rest</connectionType>
        <connectionInfo/>
        <url>
    https://localhost:9500/controller/api/v1/server/cdc/subscribe/loopback
        </url>
        <createdDate>2016-05-10</createdDate>
        <lastUpdate>2016-05-10</lastUpdate>
        <id>ab529810-e4a8-4718-bb7d-b37240ceeffe</id>
        <status>E</status>
      </CdcSubscriptionInfo>
    </responseItem>
  </responses>
</RestResponse>
```

Get Change Data Capture Process Status

Use the following REST call to get the current status of CDC process.

GET: /server/api/v1/server/cdc/status

Get the current status of the CDC poller process.

Example:

GET https://localhost:9514/server/api/v1/server/cdc/status

-- response --

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Tue, 08 May 2018 13:32:06 GMT

<RestResponse>
  <status>0</status>
  <statusText/>
<responseType>com.ibi.omnigen.controller.cdc.NotificationServiceStatus</
responseType>
  <response>
    <NotificationServiceStatus>
      <status>Stopped</status>
      <configurationValid>true</configurationValid>
      <serviceName>cdc</serviceName>
      <lastRun/>
      <pollInterval>300</pollInterval>
    </NotificationServiceStatus>
  </response>
</RestResponse>
```

Start the Change Data Capture Process

Use the following REST call to start the CDC process with an optional interval.

GET: /server/api/v1/server/cdc/start?interval=<seconds>

Start the CDC process.

Example:

GET https://localhost:9514/server/api/v1/server/cdc/start?interval=5

-- response --

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Tue, 08 May 2018 13:39:53 GMT

<RestResponse>
  <status>0</status>
  <statusText/>
```

```
<responseType>
  com.ibi.omnigen.controller.cdc.NotificationServiceStatus
  </responseType>
  <response>
    <NotificationServiceStatus>
      <status>Running</status>
      <configurationValid>true</configurationValid>
      <serviceName>cdc</serviceName>
      <lastRun/>
      <pollInterval>5</pollInterval>
    </NotificationServiceStatus>
  </response>
</RestResponse>
```

Stop the Change Data Capture Process

Use the following REST call to start the CDC process with an optional interval.

GET: /server/api/v1/server/cdc/stop

Stop the CDC process.

Example:

GET https://localhost:9514/server/api/v1/server/cdc/stop

– response –

```
200 OK
Server:  Apache-Coyote/1.1
X-Application-Context:  application:9500
Content-Type:  application/xml;charset=UTF-8
Transfer-Encoding:  chunked
Date:  Tue, 08    May 2018 13:44:53 GMT

<RestResponse>
  <status>0</status>
  <statusText/>
  <responseType>
    com.ibi.omnigen.controller.cdc.NotificationServiceStatus
  </responseType>
  <response>
    <NotificationServiceStatus>
      <status>Stopped</status>
      <configurationValid>true</configurationValid>
      <serviceName>cdc</serviceName>
      <lastRun>2018-05-08</lastRun>
      <pollInterval>5</pollInterval>
    </NotificationServiceStatus>
  </response>
</RestResponse>
```

Chapter 3

Omni-HealthData Bulk Load

The Omni-HealthData bulk loading API supports data loads to Omni-HealthData directly from OnRamp tables or from Omni Interface Documents (OIDs), which are converted into OnRamp tables.

For more detailed information on loading data into Omni-HealthData, see the *Omni-HealthData Integration Services User's Guide*.

In this chapter:

Load Bulk Subject

Use the following REST call to populate OnRamp tables with data represented by an Omni Interface Document (OID). The records written are assigned a batchId provided as a query parameter to the call.

The OnRamp batch may then be loaded into Omni through subsequent Process Bulk Subject or Process Ramp REST calls. Many OIDs may be processed together using the same batchId. However, a batch may only contain records with distinct primary keys. This is enforced by OnRamp constraints.

PUT: /server/api/v1/server/bulk/load/{subject}?batchId=

The data payload must be a correctly formatted OID XML, which minimally requires a SourceName and SourceInstanceId. The OID may be fully or partially qualified.

- subject – Specify the name of the subject that is being loaded. This must match the subject that is being loaded in the payload.
- batchId – Generated that should not collide with any other bulk loading operation. This batchId will define the set of data to process using the bulk processing command.

Example:

PUT https://localhost:9514/server/api/v1/server/bulk/load/Facility?batchId=4

```
<Facility version="1.1.7">
    <SourceName>test_system</SourceName>
    <SourceInstanceId>1499870996285_86_3</SourceInstanceId>
    <Type sourceName="test_system" codeSet="Types">type_code</Type>
    <Name>TestHospitalName</Name>
...
</Facility>
```

– response –

```
{
  "status": "OK",
  "code": 0,
  "message": "Added Facility:test_system:1499870996285_86_3 OID XML
document to Ramp.",
  "developerMessage": null,
  "responseType": "com.ibi.omni.rest.support.ServiceOperationDto",
  "response": {
    "service": "RampProcessing",
    "operation": "writeOidToRamp",
    "code": 0,
    "message": "Added Facility:test_system:1499870996285_86_3 OID XML
document to Ramp.",
    "start": 1500059394757,
    "end": 1500059396014
  },
  "exception": null
}
```

Use "code" to determine the result of the request. Any value other than zero (0) should be considered an error and the message will contain the reason.

Error Codes

- 100 – Missing required SourceName element.
- 110 – Missing required SourceInstanceId element.
- 120 – Fatal error processing the OID.
- 130 – Invalid XML document format.

Process Bulk Subject

Use the following REST call to schedule processing for a specific subject and batchId. This call will return status when the request is scheduled. The processing is carried out by the server, subsequently.

Note: If multiple subjects are loaded into a single OnRamp batch, they need to be processed with separate bulk/process calls, one for each subject. Processing follows the rules of batch_type UPSERT. Process Bulk provides a function similar to Process Ramp, with all defaults taken.

GET: /server/api/v1/server/bulk/process/{subject}?batchId=&subject=

Execute the processing of a loaded subject and batch.

- batchId – Specify the batchId used to load the data.
- subject – Specify the name of the subject to process.

Example:

GET https://localhost:9514/server/api/v1/server/bulk/process/Facility?batchId=4

-- response --

```
{
  "status": "OK",
  "code": 0,
  "message": "Scheduled ramp load of Facility with count = 4",
  "developerMessage": null,
  "responseType": "com.ibi.omni.rest.support.ServiceOperationDto",
  "response": {
    "service": "RampProcessing",
    "operation": "processRampData",
    "code": 0,
    "message": "Scheduled ramp load of Facility with count = 4",
    "start": 1500059103385,
    "end": 1500059103427
  },
  "exception": null
}
```

Use "code" to determine the result of the request. Any value other than zero (0) should be considered an error and the message will contain the reason.

Error Codes

- 90 – No records in Ramp tables for subject and batch.

- 100 – Error reading database for subject and batch.
- 110 – Other error processing error.

Process Immediate

Use the following REST call to synchronously load and process an Omni Interface Document (OID).

Note: policy = (merge/replace/delete) is respected. See the *Omni-HealthData Integration Services User's Guide* for more detailed information.

Example:

POST https://localhost:9514/server/api/v1/server/processImmediateService

```
<Facility version="1.1.7">
 <SourceName>test_system</SourceName>
 <SourceInstanceId>1499870996285_86_3</SourceInstanceId>
 <Type sourceName="test_system" codeSet="Types">type_code</Type>
 <Name>TestHospitalName</Name>
</Facility>
```

-- response --

```
{
  "status": "OK",
  "code": 0,
  "message": "Processed OID Immediately",
  "developerMessage": null,
  "responseType": "com.ibi.omni.rest.support.ServiceOperationDto",
  "response": {
    "service": "DataProcessing",
    "operation": "processImmediate",
    "code": 0,
    "message": "Processed OID Immediately",
    "start": 1574113093346,
    "end": 1574113107599
  },
  "exception": null
}
```

Error Codes

- 400 – Unable to load OID to ramp. See message for more information.

Process Ramp

Use the following REST call to schedule processing ramp data for a specific subject and batchId, governed by the various batch options. This call will return status when the request is scheduled. The processing is carried out by the server subsequently.

The HTTP PUT request accepts the parameters listed and described in the following table.

Parameter	Type	Description and Values
batchId	string	Batch to execute.
subject	string	Subject to process as part of the batch.
sourceName	string	Restrict the source system to participate in the batch.
batchType	string	Mode corresponding to the *batch_type*: ❑ UPSERT (default) ❑ INSERT_ONLY ❑ REPLACE_SELECTED ❑ REPLACE_ALL ❑ DELETE

The following table provides the batch options (sub directives) for the process ramp service and the corresponding *batch_options* column in *os_ramp_control*.

Name	Type	Description and Values
dataTransferMode	string	Overrides the default data transfer mode runtime configuration setting. ❑ JPA (default) Database neutral. ❑ Native_SQL High performance, bulk database-specific. Currently supported for SQL Server and PostgreSQL. **Note:** Native_SQL generates a runtime error for unsupported databases.

Name	Type	Description and Values
changeDetection	string	Determines whether or not to eliminate duplicates from further processing (data quality operations). ❑ ENFORCE (default) Eliminates the duplicates. ❑ IGNORE Guarantees processing of all records in a batch. Useful at certain times for error recovery. **Note:** ChangeDetection = IGNORE replaces batch option *FORCE_REPROCESS*.
upsertNullHandling (previously *addUpdateNullHandling*)	string	UPSERT only. Governs how to handle missing values on an update. ❑ PRESERVE (default) ❑ OVERRIDE

Name	Type	Description and Values
truncateBeforeInsert	boolean	Truncate data before an INSERT_ONLY operation. Used for a full replacement of the data set. Applies only to transactional subjects. ❏ FALSE (default) ❏ TRUE **Note:** Is an error unless loadType is INSERT_ONLY.

Example:

PUT https://localhost:9514/server/api/v1/server/processRamp.v4?batchId=8&batchType=UPSERT&changeDetection=ENFORCE&dataTransferMode=JPA&subject=Facility&truncateBeforeInsert=FALSE&upsertNullHandling=PRESERVE

-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": "Loaded Facility from Ramp. (batch=8, instructions={\n
\"loadType\": \"MERGE\",\n
  \"truncateTablePolicy\": \"NO_TRUNCATE\",\n  \"addUpdateNullHandling\":
\"PRESERVE\",\n
  \"duplicateRecordPolicy\": \"IGNORE\",\n  \"dataTransferMode\": \"JPA
\"\n}, sourceName=null",
  "developerMessage": null,
  "responseType": "com.ibi.omni.rest.support.ServiceOperationDto",
  "response": {
    "service": "DataProcessing",
    "operation": "processRamp",
    "code": 0,
  "message": "Loaded Facility from Ramp. (batch=8, instructions={\n
\"loadType\":  \"MERGE\",\n
  \"truncateTablePolicy\": \"NO_TRUNCATE\",\n  \"addUpdateNullHandling\":
\"PRESERVE\",\n
  \"duplicateRecordPolicy\": \"IGNORE\",\n  \"dataTransferMode\": \"JPA
\"\n}, sourceName=null",
   "start": 1574115854275,
   "end": 1574115854341
  },
  "exception": null}
```

Error Codes

- 400 – Error in Subject or Planning failure. See message for more information.

Chapter 4

Omni Server Status Operations

The Omni Server status API calls allow for server state discovery, its health, and operation mode.

In this chapter:

Status

Use the following operation to receive the server status object.

GET: /server/

Example:

GET https://localhost:9514/server/

–response –

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.lang.String",
  "response": "Server Running",
  "exception": null
}
```

Ping Server

Use the following operation to ping the server and receive string "ok".

GET: /server/api/v1/ping

Example:

GET https://localhost:9514/server/api/v1/ping

–response –

ok

Running Product

Use the following operation to determine edition of the running product.

GET: /server/api/v1/status/omniProduct

Example:

GET https://localhost:9514/server/api/v1/status/omniProduct

–response–

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.lang.String",
  "response": "MD_BASE_EDITION",
  "exception": null
}
```

Running Mode

Use the following operation to determine the product running mode.

GET: /server/api/v1/status/omniRunningMode

Example:

GET https://localhost:9514/server/api/v1/status/omniRunningMode

–response–

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.lang.String",
  "response": "DEVELOPMENT",
  "exception": null
}
```

Chapter 5

Omni Controller Deployed Bundle Artifacts

This section describes API calls to download artifacts for a deployed bundle.

In this chapter:

Download Bundle Artifact

Use the following API call to extract different types of artifacts for a deployed bundle.

It could be XSD, documentation, examples, bundle itself.

GET: /api/v1/deploy/bundle/artifact

Example

GET https://localhost:9500/api/v1/deploy/bundle/artifact

- {filename} – Artifact name.
- {artifact} – Requested artifact type from the list: documentation, xsd, example, ids, bundle

–request –

https://localhost:9500/api/v1/deploy/bundle/artifact?filename=SourceCodeStandard.xml&artifact=example

–response–

```
<?xml version="1.0" encoding="UTF-8"?>
<OmniInterface>
  <SourceCodeStandard version="3.0.0">
  <BaseCode sourceName="test_system"  codeSet="BaseCodes">base_code</
BaseCode>
  <StandardCode sourceName="test_system"
codeSet="StandardCodes">standard_code</StandardCode>
  <Description>description</Description>
  <SourceStatusCode sourceName="test_system"
codeSet="SourceStatusCodes">source_status_code</SourceStatusCode>
  <SourceCreatedDate format="yyyy-MM-dd">2017-10-18</SourceCreatedDate>
  <SourceCreatedBy>source_created_by</SourceCreatedBy>
  <SourceModifiedDate format="yyyy-MM-dd">2017-10-1</SourceModifiedDate>
  <SourceModifiedBy>source_modified_by</SourceModifiedBy>
 </SourceCodeStandard>
</OmniInterface>
```

Chapter 6

Omni-HealthData Metadata Services

The Omni-HealthData metadata services, which are available as part of the Omni-HealthData Controller REST API, allow a client to manage metadata for a domain.

In this chapter:

List All Subjects

Use the following REST call to return all subjects defined in the project. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/subjects

The list of subjects is returned in JSON format.

Example:

GET https://localhost:9500/api/v1/metadata/subjects

– response –

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.Vector",
  "response": [
    "Customer",
    "Item",
    "SalesOrder",
    "SalesOrg"
 ],
  "exception": null
}
```

List All Domains

Use the following REST call to return all domains defined in the project. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/domains

The list of domains is returned in JSON format.

Example:

GET https://localhost:9500/api/v1/metadata/domains

– response –

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.Vector",
  "response": [
    "Customer",
    "Item",
    "SalesOrder",
    "SalesOrg"
  ],
  "exception": null
}
```

List Subjects in a Domain

Use the following REST call to list all the subjects defined in a specific domain. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/domains/{domain}

The list of subjects is returned in JSON format. {domain} is one of the domains in the list returned by the method above.

Example:

GET https://localhost:9500/api/v1/metadata/domains/SalesOrg

-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "com.ibi.omni.controller.services.metadata.DomainDto",
  "response": {
    "domainName": "SalesOrg",
    "subjects": [
      "SalesOrg"
    ]
},
  "exception": null
}
```


Get Metadata for a Subject Instance

Use the following REST call to return detailed metadata for a subject instance. See the data dictionary for the metadata repository for an explanation of the attributes returned. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/subject/instance/{name}

The response is returned in JSON format. {name} is the name of a subject instance in the project.

Example:

GET https://localhost:9500/api/v1/metadata/subject/instance/SalesOrg

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "com.ibi.omni.model.ids.IdsDocumentModel",
  "response": {
    "name": "SalesOrg",
    "version": null,
    "bundleCreatedBy": "IBI Deployment Bundler",
    "projectName": "CustomerAutoiwqaomnia",
    "bundleReleaseNumber": "3.5.0.Alpha.10",
    "bundleVersion": "CustomerAutoiwqaomnia_3_5_0_Alpha_10_190506131145",
    "idsVersion": "2.0",
    "udlName": "SalesOrg",
    "type": "instance",
    "master": "SalesOrgMaster",
    "instance": null,
    "access": "standard",
    "tableName": "og_sales_org",
    "sourceTableName": "og_sales_org_s",
    "rampTableName": "og_sales_org_r",
    "tableSpace": null,
    "domain": "SalesOrg",
    "elementOrder":
"OmniStatus,OmniStatusReason,SourceName,SourceInstanceId,SourceInstanceIdNam
e,MasterId,sales_org_name,sales_org_desc,sales_org_start_dt,sales_org_end_dt
,primary_geo_area,SourceStatusCode,SourceCreatedDate,SourceCreatedBy,SourceM
odifiedDate,SourceModifiedBy",
    "captureHistory": false,
    "excludeFromRamp": false,
    "backingClass": "com.ibi.omni.model.DIBMasteredInstance",
    "implementationClass": null,
    "allowInitialUpdates": false,
    "persistenceUnit": null,
    "cleanse": true,
    "match": true,
    "merge": false,
    "remediate": false,
    "system": false,
    "documentation": "The enterprise's sales organization information such
as name, identifier and description.",
    "longDocumentation": "",
    "changeLogs": [],
"elements": [
      {
        "name": "sales_org_desc",
        "udlName": "sales_org_desc",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "sales_org_desc",
        "dbRampColumns": null,
```

```
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": false,
        "merge": true,
        "remediate": false,
        "system": false,
        "documentation": "The textual description for a sales
organization.",
        "shortDescription": null
      },
{
        "name": "SourceStatusCode",
        "udlName": "sourceStatusCode",
        "typeName": "code",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_status_code_code",
        "dbRampColumns":
"source_status_code_src,source_status_code_set,source_status_code_val",
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": false,
```

```
        "documentation": "Use this element to specify a customer specific
status for the instance. This may be used during the match and merge
process.",
"shortDescription": null
      },
      {
        "name": "SourceName",
        "udlName": "sourceName",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_name",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": true,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "This element must identify the source system that
was responsible for generating this data. This value is determined by the
integration team and used in combination with SourceInstanceId will allow
this record to be traced back to its originating source.",
        "shortDescription": null
      },
{
        "name": "SourceCreatedDate",
        "udlName": "sourceCreatedDate",
        "typeName": "datetime",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_created_date",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
```

```
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "Use this element to define when this element was
modified outside of Omni.",
        "shortDescription": null
      },
{
        "name": "primary_geo_area",
        "udlName": "primary_geo_area",
        "typeName": "code",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "primary_geo_area_code",
        "dbRampColumns":
"primary_geo_area_src,primary_geo_area_set,primary_geo_area_val",
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
```

```
        "system": false,
        "documentation": "The primary geographical area assigned to a sales
organization.",
        "shortDescription": null
      },
{
        "name": "SourceModifiedDate",
        "udlName": "sourceModifiedDate",
        "typeName": "datetime",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_modified_date",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "Use this element to define when this element was
modified outside of Omni.",
        "shortDescription": null
      },
{
        "name": "SourceInstanceId",
        "udlName": "sourceInstanceId",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_instance_id",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": true,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
```

```
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "This element should contain an integration
manufactured value that uniquely identifies the collection element and can
be used to map back to the source system. Whatever value is manufactured
must not change over time as it will be used to make updates when data
changes in the source system.",
        "shortDescription": null
      },
{
        "name": "SourceCreatedBy",
        "udlName": "sourceCreatedBy",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_created_by",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "Use this element to define the id of the person
who was responsible for creating data for this record.",
        "shortDescription": null
      },
{
        "name": "SourceInstanceIdName",
        "udlName": "sourceInstanceIdName",
        "typeName": "string",
        "dbTable": "og_sales_org",
```

```
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_instance_id_name",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "This element gives an indication of where the
data came from in the source system. Will be used in the 360 viewer for
stewards to trace data from Omni back into the source system.",
        "shortDescription": null
      },
{
        "name": "sales_org_start_dt",
        "udlName": "sales_org_start_dt",
        "typeName": "date",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "sales_org_start_dt",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": false,
```

```
        "merge": true,
        "remediate": false,
        "system": false,
        "documentation": "The start date for a sales organization.",
        "shortDescription": null
      },
{
        "name": "SourceModifiedBy",
        "udlName": "sourceModifiedBy",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "source_modified_by",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "Use this element to define the id of the person
who was responsible for modifying data for this record.",
        "shortDescription": null
      },
{
        "name": "OmniStatus",
        "udlName": "omniStatus",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "status",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": true,
        "key": false,
        "excludeInDoc": false,
```

```
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": true,
        "documentation": "Use this element to set the status of this
record. This field will default to 'ACTIVE' if not supplied. Possible
values are: 'ACTIVE', 'INACTIVE', 'MERGED', 'LINKED', 'UNLINK'",
        "shortDescription": null
      },
{
        "name": "sales_org_name",
        "udlName": "sales_org_name",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
        "rampTable": "og_sales_org_r",
        "dbColumn": "sales_org_name",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": true,
        "match": true,
        "merge": true,
        "remediate": false,
        "system": false,
        "documentation": "The textual name for a sales organization.",
        "shortDescription": null
      },
{
        "name": "OmniStatusReason",
        "udlName": "omniStatusReason",
        "typeName": "string",
        "dbTable": "og_sales_org",
        "sourceTable": "og_sales_org_s",
```

```
            "rampTable": "og_sales_org_r",
            "dbColumn": "status_reason",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
            "hidden": true,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": true,
            "match": true,
            "merge": true,
            "remediate": false,
            "system": true,
            "documentation": "Use to specify the reason for the status.",
            "shortDescription": null
        },
    {
            "name": "sales_org_end_dt",
            "udlName": "sales_org_end_dt",
            "typeName": "date",
            "dbTable": "og_sales_org",
            "sourceTable": "og_sales_org_s",
            "rampTable": "og_sales_org_r",
            "dbColumn": "sales_org_end_dt",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
            "hidden": false,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": true,
            "match": false,
```

```
            "merge": true,
            "remediate": false,
            "system": false,
            "documentation": "The end date for a sales organization.",
            "shortDescription": null
          },
{
            "name": "MasterId",
            "udlName": "masterId",
            "typeName": "string",
            "dbTable": "og_sales_org",
            "sourceTable": "og_sales_org_s",
            "rampTable": "og_sales_org_r",
            "dbColumn": "master_id",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
            "hidden": true,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": false,
            "match": true,
            "merge": true,
            "remediate": false,
            "system": true,
            "documentation": "Master id for master instance relationship.",
            "shortDescription": null
          }
        ],
        "lists": [],
        "groups": []
      },
      "exception": null
}
```


Get Metadata for a Subject Master

Use the following REST call to return detailed metadata for a subject master in the project. See the data dictionary for the metadata repository for an explanation of the attributes returned. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/subject/instance/{name}

The response is returned in JSON format. {name} is the name of a subject master in the project.

Example:

GET https://localhost:9500/api/v1/metadata/subject/master/SalesOrg

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "com.ibi.omni.model.ids.IdsDocumentModel",
  "response": {
    "name": "SalesOrgMaster",
    "version": null,
    "bundleCreatedBy": "IBI Deployment Bundler",
    "projectName": "CustomerAutoiwqaomnia",
    "bundleReleaseNumber": "3.5.0.Alpha.10",
    "bundleVersion": "CustomerAutoiwqaomnia_3_5_0_Alpha_10_190506131145",
    "idsVersion": "2.0",
    "udlName": "SalesOrgMaster",
    "type": "master",
    "master": null,
    "instance": "SalesOrg",
    "access": "standard",
    "tableName": "og_sales_org_m",
    "sourceTableName": null,
    "rampTableName": null,
    "tableSpace": null,
    "domain": "SalesOrg",
    "elementOrder":
"MasterStatus,MasterStatusReason,MasterId,MasterStatusCode,sales_org_name,sa
les_org_desc,sales_org_start_dt,sales_org_end_dt,primary_geo_area",
    "captureHistory": false,
    "excludeFromRamp": false,
    "backingClass": "com.ibi.omni.model.DIBMaster",
    "implementationClass": null,
    "allowInitialUpdates": false,
    "persistenceUnit": null,
    "cleanse": true,
    "match": true,
    "merge": false,
    "remediate": false,
    "system": false,
    "documentation": "The enterprise's sales organization information such
as name, identifier and description.",
    "longDocumentation": "",
    "changeLogs": [],
```

```
"elements": [
      {
        "name": "sales_org_start_dt",
        "udlName": "sales_org_start_dt",
        "typeName": "date",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "sales_org_start_dt",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": false,
        "match": false,
        "merge": false,
        "remediate": false,
        "system": false,
        "documentation": "The start date for a sales organization.",
        "shortDescription": null
      },
{
        "name": "sales_org_desc",
        "udlName": "sales_org_desc",
        "typeName": "string",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "sales_org_desc",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
```

```
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": false,
        "match": false,
        "merge": false,
        "remediate": false,
        "system": false,
        "documentation": "The textual description for a sales
organization.",
        "shortDescription": null
      },
{
        "name": "sales_org_name",
        "udlName": "sales_org_name",
        "typeName": "string",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "sales_org_name",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": false,
        "match": false,
        "merge": false,
        "remediate": false,
        "system": false,
        "documentation": "The textual name for a sales organization.",
        "shortDescription": null
      },
      {
        "name": "MasterStatus",
        "udlName": "masterStatus",
        "typeName": "string",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "status",
        "dbRampColumns": null,
        "contains": null,
```

```
            "keys": null,
            "immediate": false,
            "hidden": true,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": true,
            "match": true,
            "merge": true,
            "remediate": false,
            "system": true,
            "documentation": "",
            "shortDescription": null
        },
    {
            "name": "primary_geo_area",
            "udlName": "primary_geo_area",
            "typeName": "code",
            "dbTable": "og_sales_org_m",
            "sourceTable": null,
            "rampTable": null,
            "dbColumn": "primary_geo_area_code",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
            "hidden": false,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": false,
            "match": false,
            "merge": false,
            "remediate": false,
            "system": false,
            "documentation": "The primary geographical area assigned to a sales
organization.",
            "shortDescription": null
        },
```

```
{
        "name": "MasterStatusCode",
        "udlName": "masterStatus",
        "typeName": "masteromnicode",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "master_status_code",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": false,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
        "codeSourceName": null,
        "codeSet": null,
        "cleanse": false,
        "match": false,
        "merge": false,
        "remediate": false,
        "system": true,
        "documentation": "",
        "shortDescription": null
      },
{
        "name": "MasterId",
        "udlName": "masterId",
        "typeName": "string",
        "dbTable": "og_sales_org_m",
        "sourceTable": null,
        "rampTable": null,
        "dbColumn": "master_id",
        "dbRampColumns": null,
        "contains": null,
        "keys": null,
        "immediate": false,
        "hidden": false,
        "key": true,
        "excludeInDoc": false,
        "required": false,
        "index": false,
        "allowCreate": true,
        "maxLength": null,
        "precision": null,
        "scale": null,
        "excludeFromRamp": false,
```

```
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": false,
            "match": false,
            "merge": false,
            "remediate": false,
            "system": true,
            "documentation": "Master id for master instance relationship.",
            "shortDescription": null
        },
    {
            "name": "MasterStatusReason",
            "udlName": "masterStatusReason",
            "typeName": "string",
            "dbTable": "og_sales_org_m",
            "sourceTable": null,
            "rampTable": null,
            "dbColumn": "status_reason",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
            "hidden": true,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": true,
            "match": true,
            "merge": true,
            "remediate": false,
            "system": true,
            "documentation": "Use to specify the reason for the status.",
            "shortDescription": null
        },
    {
            "name": "sales_org_end_dt",
            "udlName": "sales_org_end_dt",
            "typeName": "date",
            "dbTable": "og_sales_org_m",
            "sourceTable": null,
            "rampTable": null,
            "dbColumn": "sales_org_end_dt",
            "dbRampColumns": null,
            "contains": null,
            "keys": null,
            "immediate": false,
```

```
            "hidden": false,
            "key": false,
            "excludeInDoc": false,
            "required": false,
            "index": false,
            "allowCreate": true,
            "maxLength": null,
            "precision": null,
            "scale": null,
            "excludeFromRamp": false,
            "codeSourceName": null,
            "codeSet": null,
            "cleanse": false,
            "match": false,
            "merge": false,
            "remediate": false,
            "system": false,
            "documentation": "The end date for a sales organization.",
            "shortDescription": null
          }
        ],
        "lists": [],
        "groups": []
      },
      "exception": null
    }
```


Get All Metadata for a Subject

Use the following REST call to return both instance and master metadata for the specified subject. This REST call is available as of Version 3.10.

GET: /api/v1/metadata/{subject}

Returns detailed metadata for both the subject instance and master. {subject} is the name of a subject defined in the project.

Example:

GET https://localhost:9500/api/v1/metadata/SalesOrg

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "com.ibi.omni.controller.services.metadata.SubjectDto",
  "response": {
    "name": "SalesOrg",
    "projectName": "CustomerAutoiwqaomnia",
    "bundleVersion": "CustomerAutoiwqaomnia_3_5_0_Alpha_10_190506131145",
    "bundleRelease": "3.5.0.Alpha.10",
    "instanceTable": "og_sales_org",
    "masterTable": "og_sales_org_m",
    "sourceTable": "og_sales_org_s",
    "rampTable": "og_sales_org_r",
    "instanceModel": {
...
    },
    "masterModel": {
...
  },
  "exception": null
}
```

Chapter 7

Omni-HealthData Source Management Services

You can use the Omni-HealthData source management services to document Omni-HealthData interactions with external systems. They are available as part of the Omni-HealthData Controller REST API.

In this chapter:

Source Management Services Terminology

A *source system* can be a system that provides Omni-HealthData with data, a system that consumes data that has been processed by Omni-HealthData, or an Omni-HealthData system.

Source Metadata allows the user to enrich the definition of a source system by attaching additional facts about that system.

An *Integration Point* is a defined interaction between two source systems, such as an ETL process that loads data from an external system into one or more Omni-HealthData ramp tables.

A *Source Mapping* associates an integration point with a specific subject in the Omni-HealthData model. You can get a full description of a subject using the metadata services described above.

A *Source Mapping Item* is the atomic mapping between an item in the source mapping and an element of the subject in Omni-HealthData, as defined in the IDS document. You can get a full description of an element, using the metadata services described above.

Connect a Database and Issue a Query

The following procedure shows a process that connects to the Acme Corporation customer database and issues a query every night at midnight.

1. The following query is issued:

   ```
   SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;
   ```

   The result set is shown in the following image.

2. This data is loaded into the Customer ramp table in the production Omni-HealthData environment, as follows:

In this example, we have two source systems, Acme and Omni. We have one integration point, the scheduled ETL job. We have one source mapping, from the ETL job to the Customer subject, and five mapping items, one for each of the Customer elements affected. Many of the examples below will relate to this scenario.

Model

Source Entity

```
{
  "description": "string",
  "id": "string",
  "internalIndicator": true,
  "name": "string",
  "owningBusinessUnitId": "string",
  "owningBusinessUnitName": "string",
  "sensitivity": "string",
  "thirdPartyIntegratorId": "string",
  "thirdPartyIntegratorName": "string",
  "virtualIndicator": true
}
```

Source Metadata Entity

```
{
  "category": "string",
  "codeReference": "string",
  "endDate": {
    "day": 0,
    "month": 0,
    "year": 0
  },
  "id": "string",
  "omniSourceId": "string",
  "specVersion": "string",
  "startDate": {
    "day": 0,
    "month": 0,
    "year": 0
  },
  "subCategory": "string"
}
```

Integration Point Entity

```
{
  "earliestStartTime": {
    "hour": 0,
    "minute": 0
  },
  "expectedDuration": "string",
  "expectedSize": "string",
  "id": "string",
  "integrationDaysOfMonth": "string",
  "integrationDaysOfWeek": "string",
  "integrationDirection": "string",
  "integrationFrequency": "string",
  "integrationScope": "string",
  "latestEndTime": {
    "hour": 0,
    "minute": 0
  },
  "omniProcessingPolicy": "string",
  "sourceId": "string",
  "sourceLocation": {
    "domain": "string",
    "hostName": "string",
    "password": "string",
    "port": "string",
    "protocol": "string",
    "resourceLocation": "string",
    "resourceName": "string",
    "userName": "string"
  },
```

```
"sourceLocationType": "string",
  "sourceResourceType": "string",
  "targetId": "string",
  "targetLocation": {
    "domain": "string",
    "hostName": "string",
    "password": "string",
    "port": "string",
    "protocol": "string",
    "resourceLocation": "string",
    "resourceName": "string",
    "userName": "string"
  },
  "targetLocationType": "string",
  "targetResourceType": "string"
}
```

Mapping Entity

```
{
  "description": "string",
  "id": "string",
  "integrationPointId": "string",
  "name": "string",
  "targetIdsId": "string",
  "version": "string"
}
```

Mapping Item Entity

```
{
  "codeSet": "string",
  "codeSourceName": "string",
  "codeValue": "string",
  "id": "string",
  "idsDocElementId": "string",
  "linkSourceId": "string",
  "linkSourceName": "string",
  "mappedItem": "string",
  "mappingId": "string",
  "serviceLevelAgreement": "string"
}
```

Create a Source System

Use the following REST call to create a new source system. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/create

Parameters (body): Source Entity

Content type: application/json

Do not send include the ID in the source entity when creating a new source system. A unique ID will be generated.

Example:

POST https://localhost:9500/api/v1/sourcesystem/create

Content-Type: application/json

```
{
  "description": "Sellers of fine products since 1974",
  "internalIndicator": false,
  "name": "Acme",
  "owningBusinessUnitName": "Amalgamated Industries",
  "sensitivity": "insensitive",
  "thirdPartyIntegratorName": "Super Integration Services",
  "virtualIndicator": false
}
```

– response –

```
{
  "status": "CREATED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.OmniSourceDto",
  "response": {
    "id": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "name": "Acme",
    "description": "Sellers of fine products since 1974",
    "owningBusinessUnitId": null,
    "owningBusinessUnitName": "Amalgamated Industries",
    "thirdPartyIntegratorId": null,
    "thirdPartyIntegratorName": "Super Integration Services",
    "internalIndicator": false,
    "virtualIndicator": false,
    "sensitivity": "insensitive"
  },
  "exception": null
}
```

Update a Source System

Use the following REST call to update a source system. This REST call is available as of Version 3.11.

PUT: /api/v1/sourcesystem/update

Parameters (body): Source Entity

Content type: application/json

Example:

PUT https://localhost:9500/api/v1/sourcesystem/update

Content-Type: application/json

```
{
  “id”:” 4e5b765f-b807-483b-8c56-1fcc683b0b98”,
  "description": "Sellers of fine products since 1975",
  "internalIndicator": false,
  "name": "Acme",
  "owningBusinessUnitName": "Amalgamated Industries",
  "sensitivity": "hyper sensitive",
  "thirdPartyIntegratorName": "Super Dooper Integration Services",
  "virtualIndicator": false
}
```

– response –

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.OmniSourceDto",
  "response": {
    "id": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "name": "Acme",
    "description": "Sellers of fine products since 1975",
    "owningBusinessUnitId": null,
    "owningBusinessUnitName": "Amalgamated Industries",
    "thirdPartyIntegratorId": null,
    "thirdPartyIntegratorName": "Super Dooper Integration Services",
    "internalIndicator": false,
    "virtualIndicator": false,
    "sensitivity": "hyper sensitive"
  },
  "exception": null
}
```

Get a Source System By Name

Use the following REST call to get a source system by name. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem/{name}

Parameters: Source Name

Example:

GET https://localhost:9500/api/v1/sourcesystem/Acme

-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.OmniSourceDto",
  "response": {
    "id": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "name": "Acme",
    "description": "Sellers of fine products since 1975",
    "owningBusinessUnitId": null,
    "owningBusinessUnitName": "Amalgamated Industries",
    "thirdPartyIntegratorId": null,
    "thirdPartyIntegratorName": "Super Dooper Integration Services",
    "internalIndicator": false,
    "virtualIndicator": false,
    "sensitivity": "hyper sensitive"
  },
  "exception": null
}
```


List All Defined Source Systems

Use the following REST call to list all source systems. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem

Example:

GET https://localhost:9500/api/v1/sourcesystem

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "a8be97a2-1688-4855-8d76-92f86095e292",
      "name": "Omni Prod",
      "description": "Production Omni System"
    },
    {
      "id": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
      "name": "Acme",
      "description": "Sellers of fine products since 1975"
    }
  ],
  "exception": null}
```


List Source Systems That Contribute to a Subject

Use the following REST call to return all source systems that contribute to a subject. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem?subject={subject}

Parameters (body): Subject Name

Returns all source systems with integration points that have at least one mapping for the requested subject.

Example:

GET https://server/api/v1/sourcesystem?subject=Customer

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
      "name": "Acme",
      "description": "Sellers of fine products since 1975"
    }
  ],
  "exception": null
}
 "exception": null
}
```


Delete a Source System

Use the following REST call to delete a source system. This REST call is available as of Version 3.11.

DELETE: /api/v1/sourcesystem/delete/{id}

Parameters: Source System ID

Example:

DELETE https::://localhost:9500/api/v1/sourcesystem/delete/0276981d-1f09-401f-b097-a9f228faae86

– response –


```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```


Add New Metadata to a Source System

Use the following REST call to add metadata to a source system. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/metadata/create

Parameters (body): Source Metadata Entity

Content type: application/json

Do not send include the ID in the source metadata entity when creating a new source system. A unique ID will be generated. For the start and end date months, enter 0 for January and 11 for December. Do not use leading zeroes for the day, month, or year.

Example:

POST https://localhost:9500/api/v1/sourcesystem/metadata/create

Content-Type: application/json

```
{
  "omniSourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
  "specVersion": "1.0.0",
  "category": "Vendor",
  "subCategory": "Anvils",
  "startDate": {
    "day": 1,
    "month": 0,
    "year": 2019
  },
  "endDate": {
    "day": 2,
    "month": 0,
    "year": 2019
  }
}
```

– response –


```
{
  "status": "CREATED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.SourceMetadataDto",
  "response": {
    "id": "f9c9af9d-a6f9-4822-b9bf-9d0245e5ba4c",
    "omniSourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "codeReference": null,
    "category": "Vendor",
    "subCategory": "Anvils",
    "specVersion": "1.0.0",
    "startDate": {
      "year": 2019,
      "month": 0,
      "day": 1
    },
    "endDate": {
      "year": 2019,
      "month": 0,
      "day": 2
    }
  },
  "exception": null
}
```


Update Metadata for a Source System

Use the following REST call to update metadata for a source system. This REST call is available as of Version 3.11.

PUT: /api/v1/sourcesystem/metadata/update

Parameters (body): Source Metadata Entity

Content type: application/json

For the start and end date months, enter 0 for January and 11 for December. Do not use leading zeroes for the day, month, or year.

Example:

PUT https://localhost:9500/api/v1/sourcesystem/metadata/update

Content-Type: application/json


```
{
   "id": "f9c9af9d-a6f9-4822-b9bf-9d0245e5ba4c",
  "omniSourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
  "specVersion": "1.0.0",
  "category": "Vendor",
  "subCategory": "Iron Anvils",
  "startDate": {
    "day": 1,
    "month": 0,
    "year": 2020
  },
  "endDate": {
    "day": 2,
    "month": 0,
    "year": 2020
  }
}
```


-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.SourceMetadataDto",
  "response": {
    "id": "f9c9af9d-a6f9-4822-b9bf-9d0245e5ba4c",
    "omniSourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "codeReference": null,
    "category": "Vendor",
    "subCategory": "Iron Anvils",
    "specVersion": "1.0.0",
    "startDate": {
      "year": 2020,
      "month": 0,
      "day": 1
    },
    "endDate": {
      "year": 2020,
      "month": 0,
      "day": 2
    }
  },
  "exception": null
}
```

List All Metadata for a Source System

Use the following REST call to list metadata for a source system. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem/metadatadata/{sourceId}

Example:

GET https://localhost:9500/api/v1/sourcesystem/metadata/4e5b765f-b807-483b-8c56-1fcc683b0b98

-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "f9c9af9d-a6f9-4822-b9bf-9d0245e5ba4c",
      "omniSourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
      "codeReference": null,
      "category": "Vendor",
      "subCategory": "Iron Anvils",
      "specVersion": "1.0.0",
      "startDate": {
        "year": 2020,
        "month": 0,
        "day": 1
      },
      "endDate": {
        "year": 2020,
        "month": 0,
        "day": 2
      }
    }
  ],
  "exception": null
}
```


Delete a Metadata Item From a Source System

Use the following REST call to delete a metadata item from a source system. This REST call is available as of Version 3.11.

DELETE: /api/v1/sourcesystem/metadata/delete/{Id}

Parameters: Source Metadata ID

Example:

DELETE https://localhost:9500/api/v1/sourcesystem/metadata/delete/f9c9af9d-a6f9-4822-b9bf9d0245e5ba4c

– response –

```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```

Create an Integration Point

Use the following REST call to create an integration point. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/integration/create

Parameters (body): Integration Point Entity

Content type: application/json

Do not include the ID in the integration point entity when creating a new integration point. A unique ID will be generated.

Example:

POST https://localhost:9500/api/v1/sourcesystem/integration/create

Content-Type: application/json


```
{
  "earliestStartTime": {
    "hour": 11,
    "minute": 0
  },
  "expectedDuration": "1 hour",
  "expectedSize": "1000 rows",
  "integrationDirection": "TargetRequest",
  "integrationFrequency": "daily",
  "integrationScope": "PROD",
  "latestEndTime": {
    "hour": 1,
    "minute": 0
  },
  "sourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
  "sourceLocation": {
    "domain": "acme.com",
    "hostName": "ahost",
    "password": "secret",
    "port": "8080",
    "protocol": "jdbc",
    "resourceLocation": "/customers",
    "userName": "user"
  },
  "sourceLocationType": "database",
  "targetId": "a8be97a2-1688-4855-8d76-92f86095e292",
  "targetLocation": {
    "domain": "amalgamated.com",
    "hostName": "omniprod",
    "password": "secret",
    "port": "9500",
    "protocol": "http",
    "userName": "user"
  },
  "targetLocationType": "omni",
}
```

– response –


```
{
  "status": "CREATED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.IntegrationPointDto",
  "response": {
    "id": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
    "sourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "sourceLocationType": "database",
    "sourceLocation": {
      "protocol": "jdbc",
      "hostName": "ahost",
      "domain": "acme.com",
      "port": null,
      "userName": "user",
      "password": "secret",
      "resourceLocation": "/customers",
      "resourceName": null
    },
    "sourceResourceType": null,
    "targetId": "a8be97a2-1688-4855-8d76-92f86095e292",
    "targetLocationType": "omni",
    "targetLocation": {
      "protocol": "http",
      "hostName": "omniprod",
      "domain": "amalgamated.com",
      "port": null,
      "userName": "user",
      "password": "secret",
      "resourceLocation": null,
      "resourceName": null
    },
"targetResourceType": null,
    "integrationScope": "PROD",
    "integrationDirection": "TargetRequest",
    "integrationFrequency": "daily",
    "integrationDaysOfWeek": null,
    "integrationDaysOfMonth": null,
    "earliestStartTime": {
      "hour": 11,
      "minute": 0
    },
    "latestEndTime": {
      "hour": 1,
      "minute": 0
    },
    "expectedDuration": "1 hour",
    "expectedSize": "1000 rows",
    "omniProcessingPolicy": null
  },
  "exception": null
}
```

Update an Integration Point

Use the following REST call to update an integration point. This REST call is available as of Version 3.11.

PUT: /api/v1/sourcesystem/integration/create

Parameters (body): Integration Point Entity

Content type: application/json

Example:

PUT https://localhost:9500/api/v1/sourcesystem/integration/update

Content-Type: application/json

```
{
    "id": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
    "earliestStartTime": {
        "hour": 11,
        "minute": 30
    },
    "expectedDuration": "1 hour",
    "expectedSize": "5000 rows",
    "integrationDirection": "TargetRequest",
    "integrationFrequency": "daily",
    "integrationScope": "PROD",
    "latestEndTime": {
        "hour": 1,
        "minute": 30
    },
    "sourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "sourceLocation": {
        "domain": "acme.com",
        "hostName": "ahost",
        "password": "secret",
        "port": "8080",
        "protocol": "jdbc",
        "resourceLocation": "/customers",
        "userName": "user"
    },
    "sourceLocationType": "database",
    "targetId": "a8be97a2-1688-4855-8d76-92f86095e292",
    "targetLocation": {
        "domain": "amalgamated.com",
        "hostName": "omniprod",
        "password": "secret",
        "port": "9500",
        "protocol": "http",
        "userName": "user"
    },
    "targetLocationType": "omni"
}
```

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.IntegrationPointDto",
  "response": {
    "id": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
    "sourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
    "sourceLocationType": "database",
    "sourceLocation": {
      "protocol": "jdbc",
      "hostName": "ahost",
      "domain": "acme.com",
      "port": null,
      "userName": "user",
      "password": "secret",
      "resourceLocation": "/customers",
      "resourceName": null
    },
    "sourceResourceType": null,
    "targetId": "a8be97a2-1688-4855-8d76-92f86095e292",
    "targetLocationType": "omni",
    "targetLocation": {
      "protocol": "http",
      "hostName": "omniprod",
      "domain": "amalgamated.com",
      "port": null,
      "userName": "user",
      "password": "secret",
      "resourceLocation": null,
      "resourceName": null
    },"targetResourceType": "omni",
    "integrationScope": "PROD",
    "integrationDirection": "TargetRequest",
    "integrationFrequency": "daily",
    "integrationDaysOfWeek": null,
    "integrationDaysOfMonth": null,
    "earliestStartTime": {
      "hour": 11,
      "minute": 30
    },
    "latestEndTime": {
      "hour": 1,
      "minute": 30
    },
    "expectedDuration": "1 hour",
    "expectedSize": "5000 rows",
    "omniProcessingPolicy": null
  },
  "exception": null
}
```

List All Integration Points for a Source System

Use the following REST call to list integration points for a source system. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem/integration/{sourceId}

Example:

GET https://localhost:9500/api/v1/sourcesystem/integration/4e5b765f-b807-483b-8c56-1fcc683b0b98

-- response --


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
      "sourceId": "4e5b765f-b807-483b-8c56-1fcc683b0b98",
      "sourceLocationType": "database",
      "sourceLocation": {
        "protocol": "jdbc",
        "hostName": "ahost",
        "domain": "acme.com",
        "port": null,
        "userName": "user",
        "password": "secret",
        "resourceLocation": "/customers",
        "resourceName": null
      },
      "sourceResourceType": null,
      "targetId": "a8be97a2-1688-4855-8d76-92f86095e292",
    …
```


Delete an Integration Point

Use the following REST call to delete an integration point. This REST call is available as of Version 3.11.

DELETE: /api/v1/sourcesystem/integration/create

Parameters: Integration Point ID

Example:

DELETE https://localhost:9500/api/v1/sourcesystem/integration/delete/f9c9af9d-a6f9-4822-b9bf9d0245ba4c

– response –

```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```

Create a Mapping

Use the following REST call to create a new mapping. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/integration/mapping/create

Parameters (body): Mapping Entity

Content type: application/json

Do not include the ID in the mapping entity when creating a new mapping. A unique ID will be generated.

Example:

POST https://localhost:9500/api/v1/sourcesystem/integration/mapping/create

Content-Type: application/json

```
{
  "description": "SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;",
  "integrationPointId": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
  "name": "Acme Customer",
  "targetIdsId": "36de029d-5baa-4f4a-8252-8d1c5c8c2b78",
  "version": "1.0.0"
}
```

– response –


```
{
  "status": "CREATED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.MappingDto",
  "response": {
    "id": "4ca397e9-7c39-432f-abe7-c150f193e11b”,
    "name": "Acme Customer",
    "version": "1.0.0",
    "targetIdsId": "36de029d-5baa-4f4a-8252-8d1c5c8c2b78",
    "integrationPointId": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
    "description": "SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;"
  },
  "exception": null
}
```


Update a Mapping

Use the following REST call to update a mapping. This REST call is available as of Version 3.11.

PUT: /api/v1/sourcesystem/integration/mapping/update

Parameters (body): Mapping Entity

Content type: application/json

Example:

PUT https://localhost:9500/api/v1/sourcesystem/integration/mapping/update

Content-Type: application/json


```
{
  "id": "4ca397e9-7c39-432f-abe7-c150f193e11b",
  "description": "SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;",
  "integrationPointId": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
  "name": "Acme Customer Database",
  "targetIdsId": "36de029d-5baa-4f4a-8252-8d1c5c8c2b78",
  "version": "1.0.0"
}
```

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.MappingDto",
  "response": {
    "id": "ac167bd5-c58b-4159-9019-7f2d4d57fa22",
    "name": "Acme Customer Database",
    "version": "1.0.0",
    "targetIdsId": "36de029d-5baa-4f4a-8252-8d1c5c8c2b78",
    "integrationPointId": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
    "description": "SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;"
  },
  "exception": null
}
```


List All Mappings for an Integration Point

Use the following REST call to list all mappings for an integration point. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem/integration/{integrationPointId}/mapping

Parameters: Integration Point ID

Example:

GET https://localhost:9500/api/v1/sourcesystem/integration/6cb6d742-3f23-4095-a9d1-deaeb88edeaf/mapping

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "4ca397e9-7c39-432f-abe7-c150f193e11b",
      "name": "Acme Customer",
      "version": "1.0.0",
      "targetIdsId": "36de029d-5baa-4f4a-8252-8d1c5c8c2b78",
      "integrationPointId": "6cb6d742-3f23-4095-a9d1-deaeb88edeaf",
      "description": "SELECT CUSTID, FNAME, MNAME, LNAME FROM CUSTOMER;"
    }
  ],
  "exception": null
}
```


Delete a Mapping

Use the following REST call to delete a mapping. This REST call is available as of Version 3.11.

DELETE: /api/v1/sourcesystem/integration/mapping/update

Parameters Mapping ID

Example:

DELETE https://localhost:9500/api/v1/sourcesystem/integration/mapping/delete/f9c9af9822-b9bf9d0245ba4c

– response –


```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```

Create a Mapping Item

Use the following REST call to create a new mapping item. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/integration/mapping/item/create

Parameters (body): Mapping Item Entity

Content type: application/json

Do not include the ID in the mapping item entity when creating a new mapping item. A unique ID will be generated.

Example:

POST https://localhost:9500/api/v1/sourcesystem/integration/mapping/item/create

Content-Type: application/json

```
{
  "idsDocElementId": "62745a35-1790-4994-a39a-efb0dc63d142",
  "mappedItem": "FNAME",
  "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b"
}
```

– response –

```
{
  "status": "CREATED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.MappingItemDto",
  "response": {
    "id": "cb1af21e-7335-4dbd-85fa-82eaf27d2634",
    "mappedItem": "FNAME",
    "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
    "idsDocElementId": "62745a35-1790-4994-a39a-efb0dc63d142",
    "linkSourceName": null,
    "linkSourceId": null,
    "codeSourceName": null,
    "codeSet": null,
    "codeValue": null,
    "serviceLevelAgreement": null
  },
  "exception": null
}
```

Update a Mapping Item

Use the following REST call to update a mapping item. This REST call is available as of Version 3.11.

PUT: /api/v1/sourcesystem/integration/mapping/item/update

Parameters (body): Mapping Item Entity

Content type: application/json

Example:

PUT https://localhost:9500/api/v1/sourcesystem/integration/mapping/item/update

Content-Type: application/json

```
{
    "id": "cb1af21e-7335-4dbd-85fa-82eaf27d2634",
    "mappedItem": "FNAME",
    "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
    "idsDocElementId": "62745a35-1790-4994-a39a-efb0dc63d142",
    "serviceLevelAgreement": "super SLA"
  }
```

-- response --

```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType":
"com.ibi.omni.controller.services.sourcesystem.MappingItemDto",
  "response": {
    "id": "cb1af21e-7335-4dbd-85fa-82eaf27d2634",
    "mappedItem": "FNAME",
    "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
    "idsDocElementId": "62745a35-1790-4994-a39a-efb0dc63d142",
    "linkSourceName": null,
    "linkSourceId": null,
    "codeSourceName": null,
    "codeSet": null,
    "codeValue": null,
    "serviceLevelAgreement": "super SLA"
  },
  "exception": null
}
```

List All Items for a Mapping

Use the following REST call to list all items for a mapping. This REST call is available as of Version 3.11.

GET: /api/v1/sourcesystem/integration/mapping/{mappingId}

Parameters: Integration Point ID

Example:

GET https://localhost:9500/api/v1/sourcesystem/integration/mapping/4ca397e9-7c39-432f-abe7-c150f193e11b

– response –


```
{
  "status": "OK",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": "java.util.ArrayList",
  "response": [
    {
      "id": "efc256fc-989a-49e7-85c0-316f7519db9a",
      "mappedItem": "MNAME",
      "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
      "idsDocElementId": "ddf07c78-b8f8-4fde-9415-d114a9cfc1bc",
      "linkSourceName": null,
      "linkSourceId": null,
      "codeSourceName": null,
      "codeSet": null,
      "codeValue": null,
      "serviceLevelAgreement": null
    },
    {
      "id": "18440f2a-22c1-4c09-891a-bb82eedf5d6c",
      "mappedItem": "LNAME",
      "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
      "idsDocElementId": "3f412216-f7ad-4f5a-a726-9ef54711b9a7",
      "linkSourceName": null,
      "linkSourceId": null,
      "codeSourceName": null,
      "codeSet": null,
      "codeValue": null,
      "serviceLevelAgreement": null
    },
```

```
{
      "id": "cb1af21e-7335-4dbd-85fa-82eaf27d2634",
      "mappedItem": "FNAME",
      "mappingId": "4ca397e9-7c39-432f-abe7-c150f193e11b",
      "idsDocElementId": "62745a35-1790-4994-a39a-efb0dc63d142",
      "linkSourceName": null,
      "linkSourceId": null,
      "codeSourceName": null,
      "codeSet": null,
      "codeValue": null,
      "serviceLevelAgreement": "super SLA"
    }
  ],
  "exception": null
}
```


Delete a Mapping Item

Use the following REST call to delete a mapping item. This REST call is available as of Version 3.11.

DELETE: /api/v1/sourcesystem/integration/mapping/item/delete/{itemId}

Parameters: Mapping ID

Example:

DELETE https://localhost:9500/api/v1/sourcesystem/integration/mapping/item/delete/f9c9af9822-b9bf95ba4c

-- response --


```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```


Load Mapping From Documentation

Use the following REST call to load mappings from Microsoft Excel into the metadata repository. This REST call is available as of Version 3.11.

POST: /api/v1/sourcesystem/integration/mapping/load?int_pt_id={id}&subjects={s1,s2}

Parameters:

int_pt_id: the ID of the integration point to which this mapping belongs.

subjects: optionally, a comma delimited list of specific subjects to load from the spreadsheet.

excel_file: an Omni generated documentation file containing mappings from a source system to one or more subjects. These will correspond to sheet names in the spreadsheet.

When a model is deployed into Omni server, Excel spreadsheets are generated to allow users to document the mappings from external systems into Omni subjects. This service allows information captured in those spreadsheets to be loaded into the metadata repository.

Example:

POST /api/v1/sourcesystem/integration/mapping/load?int_pt_id=f9c9af9822-b9bf95ba4c&subjects=Customer

-- response --

```
{
  "status": "ACCEPTED",
  "code": 0,
  "message": null,
  "developerMessage": null,
  "responseType": null,
  "response": null,
  "exception": null
}
```

Feedback

Customer success is our top priority. Connect with us today!

Information Builders Technical Content Management team is comprised of many talented individuals who work together to design and deliver quality technical documentation products. Your feedback supports our ongoing efforts!

You can also preview new innovations to get an early look at new content products and services. Your participation helps us create great experiences for every customer.

To send us feedback or make a connection, contact Sarah Buccellato, Technical Editor, Technical Content Management at *Sarah_Buccellato@ibi.com.*

To request permission to repurpose copyrighted material, please contact Frances Gambino, Vice President, Technical Content Management at *Frances_Gambino@ibi.com.*

Information Builders

iWay

Omni-HealthData™ API Services Reference Guide

Version 3.11

DN3502335.1119

Information Builders, Inc.
Two Penn Plaza
New York, NY 10121-2898