TGH

Making Integrations Simpler

Routing Headers in API Service Component

Author

Sowjanya Gurrala

Contents

API Service Component:

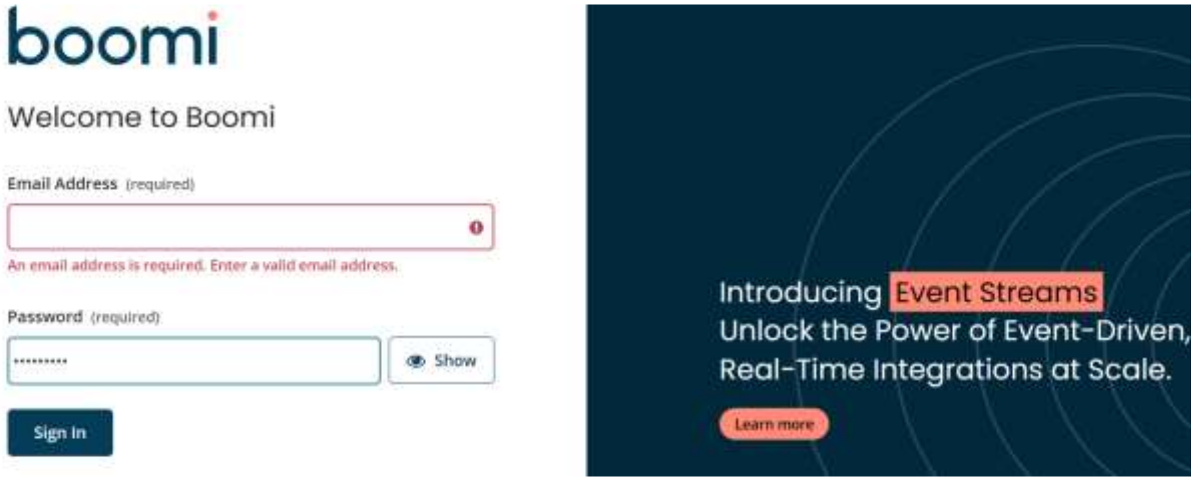
- API Service components are deployable and versioned components used to expose sets of REST, SOAP, or OData API endpoints
- You can build API Service components only you are using an account for which the Service Enablement feature is enabled.
- you can deploy API Service components only to Atoms for which **API Type** is set to **Advanced** in the Shared Web Server panel.
- By using API Service components, you can expose different sets of endpoints for use by different customers
- Each defined endpoint has a linked Web Services Server listener process configured to listen for and process requests for a particular operation
- The default settings for an operation specified for an endpoint are derived from the linked process. The defaults can optionally be overridden.
- For a REST endpoint we have routing headers option.

Routing Headers:

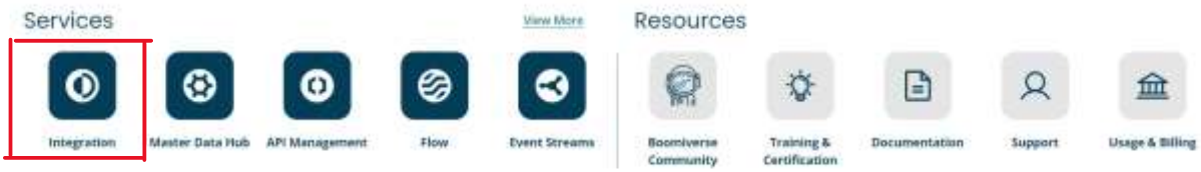
When we have two or more REST APIs with same endpoint, the Routing header will help us to route to specific API or webservice (i.e., linked subprocess) based on specified header value.

Steps to expose an API using web service server connector

Step 1.1: Log onto the Boomi platform (https://platform.boomi.com/) with the required credentials (Email Address and Password).

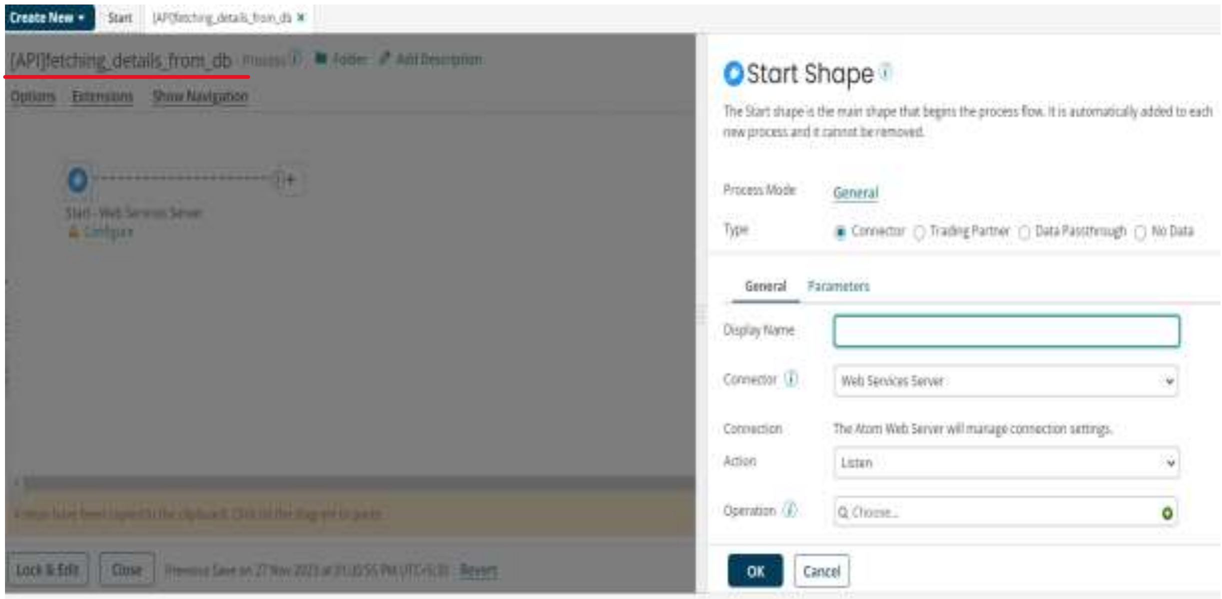

boomi
Welcome to Boomi
Email Address (required)
An email address is required. Enter a valid email address.
Password (required)
•••••••••
Show
Sign In
Introducing Event Streams
Unlock the Power of Event-Driven, Real-Time Integrations at Scale.
Learn more

Step 1.2: Under Services choose Integration



Step 1.3: Create a new process and configure the start shape with connector and choose web services server connector and give a proper name for process

(**ex:** [API]fetching_details_from_db)



Start Shape

The Start shape is the main shape that begins the process flow. It is automatically added to each new process and it cannot be removed.

Process Mode: General

Type: ● Connector ○ Trading Partner ○ Data Passthrough ○ No Data

General | Parameters

Display Name:

Connector: Web Services Server

Connection: The Atom Web Server will manage connection settings.

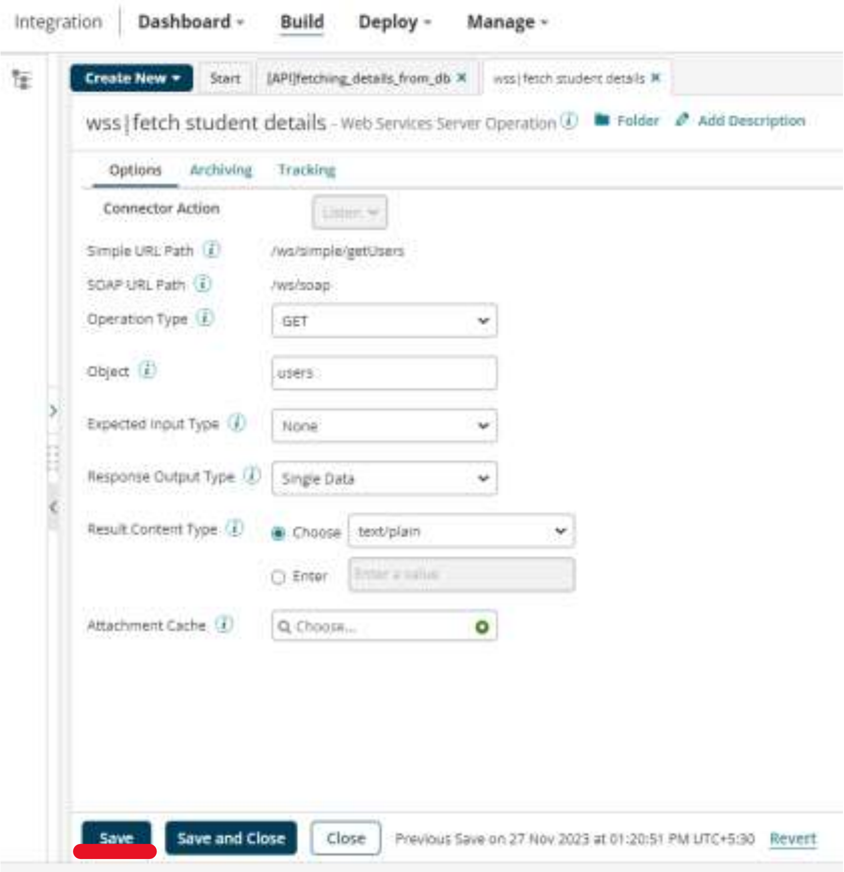
Action: Listen

Operation: Choose...

OK | Cancel

WKT, connection settings of web services server connector are managed by the runtime engine’s shared web server.

Step 1.4: configure the operation component as shown in below image and click on save



wss|fetch student details - Web Services Server Operation

Connector Action: Listen

Simple URL Path: /ws/simple/getUsers

SOAP URL Path: /ws/soap

Operation Type: GET

Object: users

Expected Input Type: None

Response Output Type: Single Data

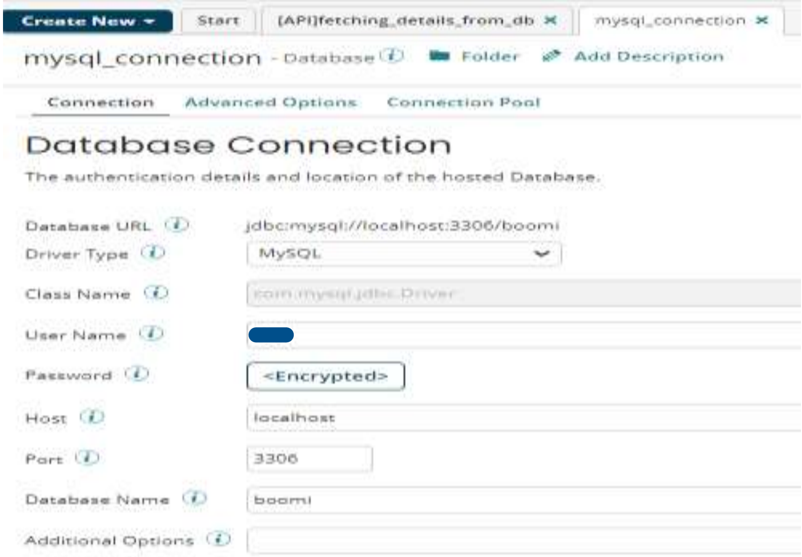
Result Content Type: ● Choose text/plain ○ Enter

Attachment Cache: Choose...

Previous Save on 27 Nov 2023 at 01:20:51 PM UTC+5:30

Step 1.5: Now drag and drop a database connector into process canvas

Step 1.6: Configure Connection Component with Proper Details



mysql_connection - Database

Database Connection

The authentication details and location of the hosted Database.

Database URL: jdbc:mysql://localhost:3306/boomi

Driver Type: MySQL

Class Name: com.mysql.jdbc.Driver

User Name: [redacted]

Password: <Encrypted>

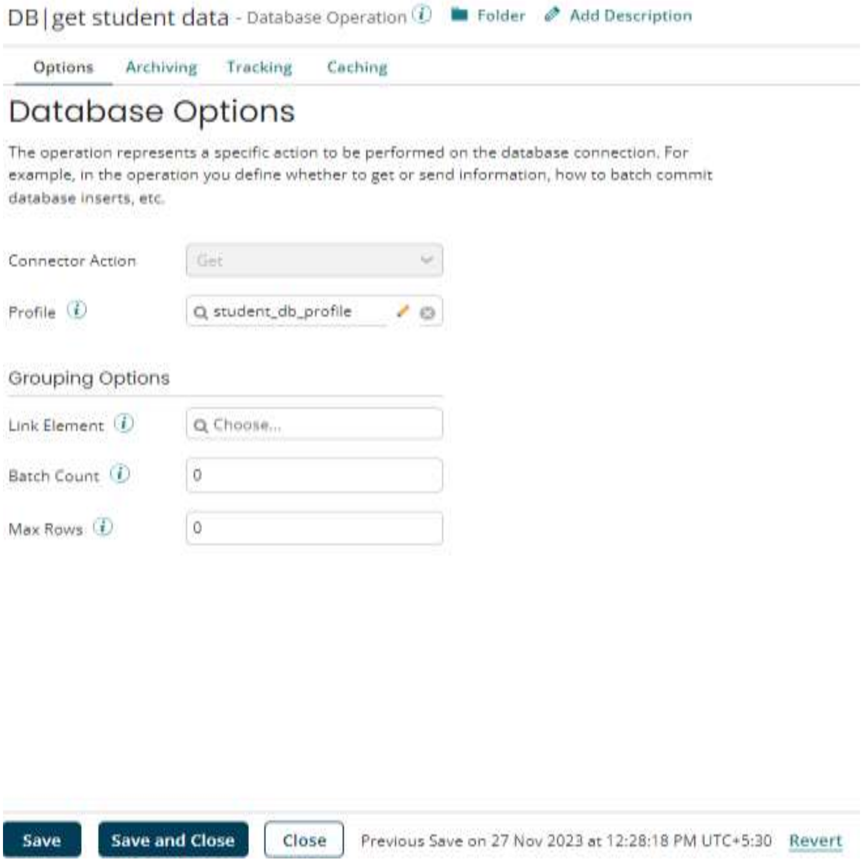
Host: localhost

Port: 3306

Database Name: boomi

Additional Options:

Step 1.7: Choose the action as get and configure the operation component and import the database profile into Boomi.



DB|get student data - Database Operation ⓘ Folder Add Description

Database Options

The operation represents a specific action to be performed on the database connection. For example, in the operation you define whether to get or send information, how to batch commit database inserts, etc.

Connector Action: Get

Profile ⓘ: student_db_profile

Grouping Options

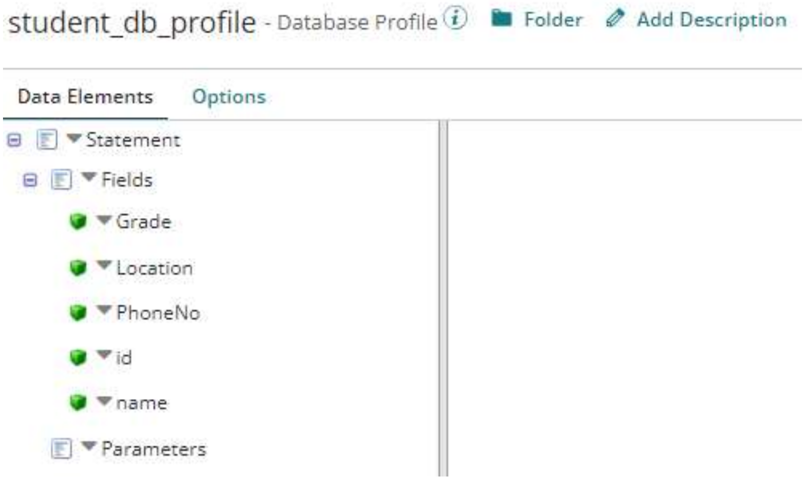
Link Element ⓘ: Choose...

Batch Count ⓘ: 0

Max Rows ⓘ: 0

Previous Save on 27 Nov 2023 at 12:28:18 PM UTC+5:30 Revert

Below image represents a sample database profile which I imported from database



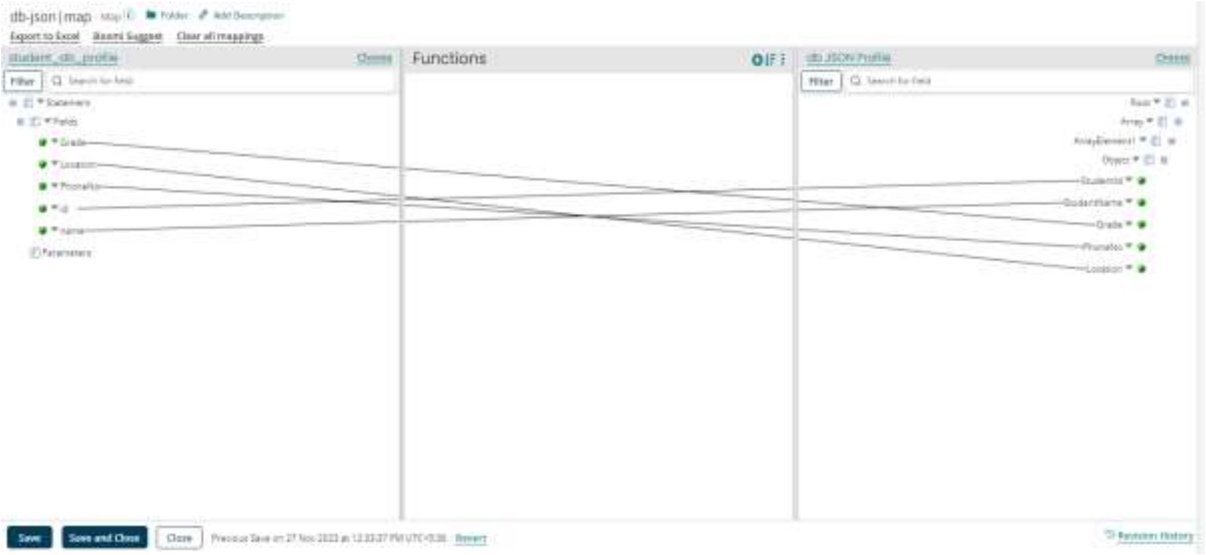
student_db_profile - Database Profile ⓘ Folder Add Description

- Statement
  - Fields
    - Grade
    - Location
    - PhoneNo
    - id
    - name
  - Parameters

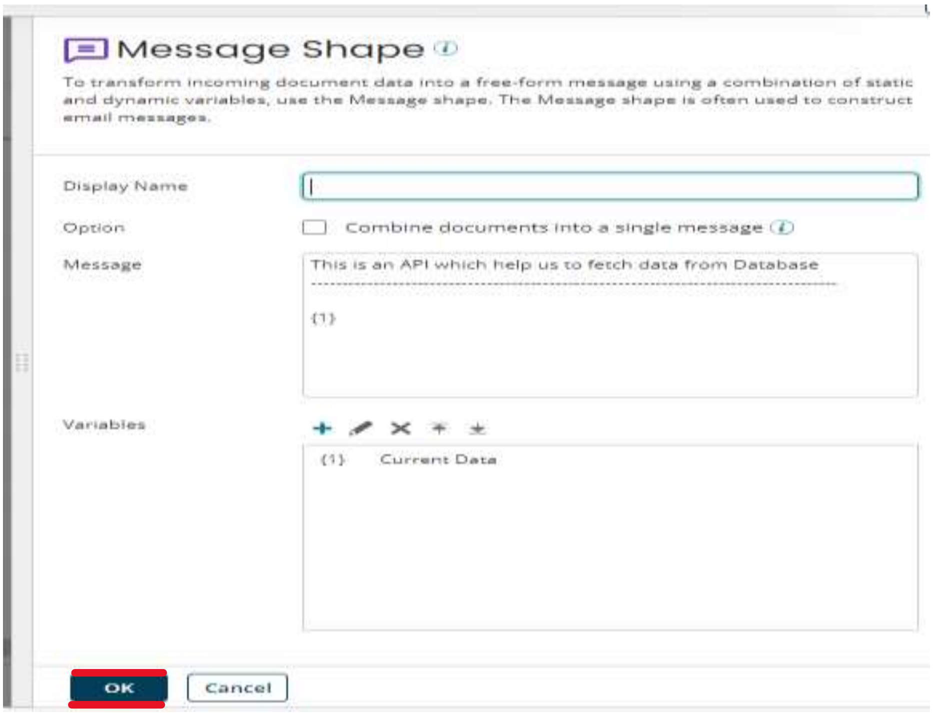
Step 1.8: Drag and drop a map shape into the process canvas. For source profile choose database profile which we configured in the database connector operation component and configure a JSON profile as target profile.

Mapping Details:

Source Element	Target Element
id	StudentId
name	StudentName
PhoneNo	PhoneNo
Location	Location
Grade	Grade



Step 1.9: Drag and drop a message shape into the process canvas and configure as shown in below image.



Message Shape

To transform incoming document data into a free-form message using a combination of static and dynamic variables, use the Message shape. The Message shape is often used to construct email messages.

Display Name

Option □ Combine documents into a single message

Message

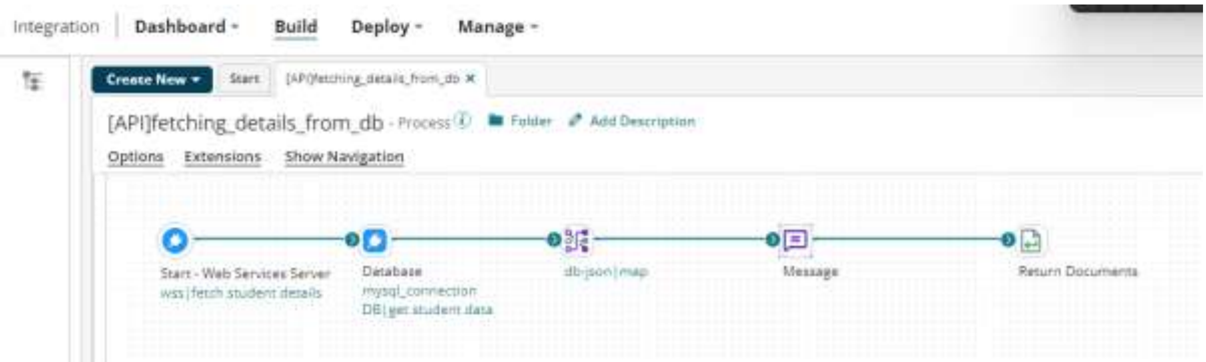
```
This is an API which help us to fetch data from Database
------------------------------------------------------------
{1}
```

Variables

{1} Current Data

Step 1.10: Drag and drop a return documents shape and attach it to the message shape.

The complete flow will look as shown below.



[API]fetching_details_from_db - Process

Start - Web Services Server
wss | fetch student details

Database
mysql_connection
DB | get student data

db-json | map

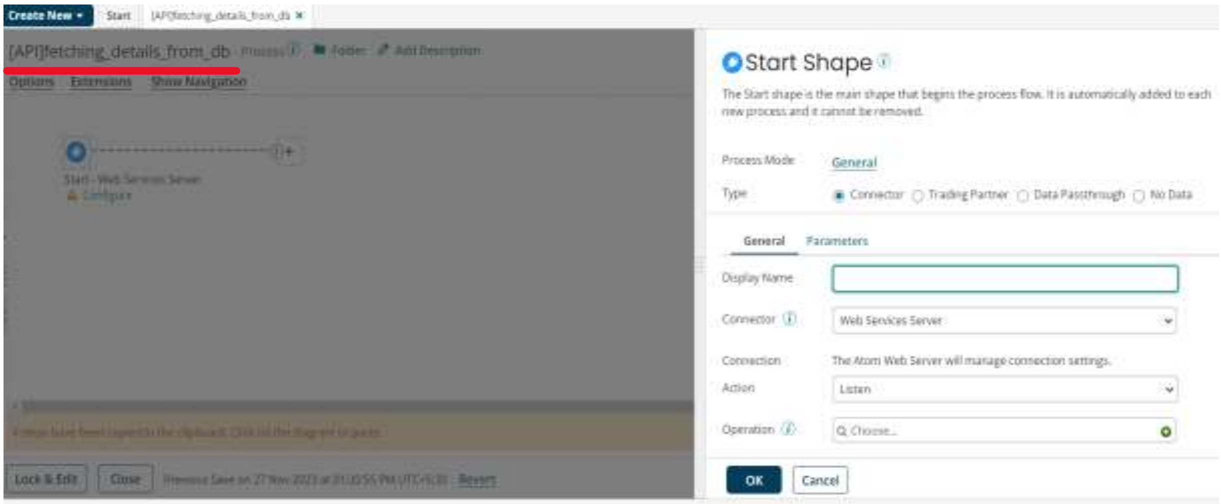
Message

Return Documents

Now expose one more API with same endpoint using web services server connector by following the below steps.

Step 2.1: Create a new process and configure the start shape with connector and choose web services server connector and give a proper name for process

(**ex:** [API]fetching_details_from_sfdc)



Start Shape

The Start shape is the main shape that begins the process flow. It is automatically added to each new process and it cannot be removed.

Process Mode: General

Type: ● Connector ○ Trading Partner ○ Data Passthrough ○ No Data

Display Name:

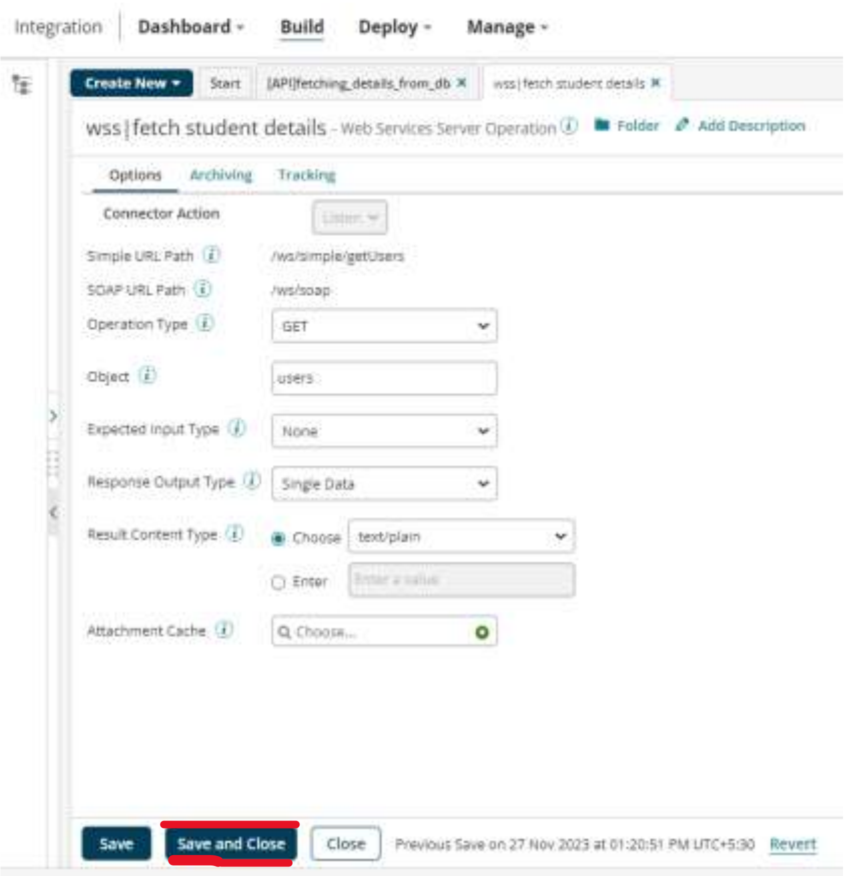
Connector: Web Services Server

Connection: The Atom Web Server will manage connection settings.

Action: Listen

Operation: Choose...

Step 2.2: configure the operation component as shown in below image and click on save



wss | fetch student details - Web Services Server Operation

Connector Action: Listen

Simple URL Path: /ws/simple/getUsers

SOAP URL Path: /ws/soap

Operation Type: GET

Object: users

Expected Input Type: None

Response Output Type: Single Data

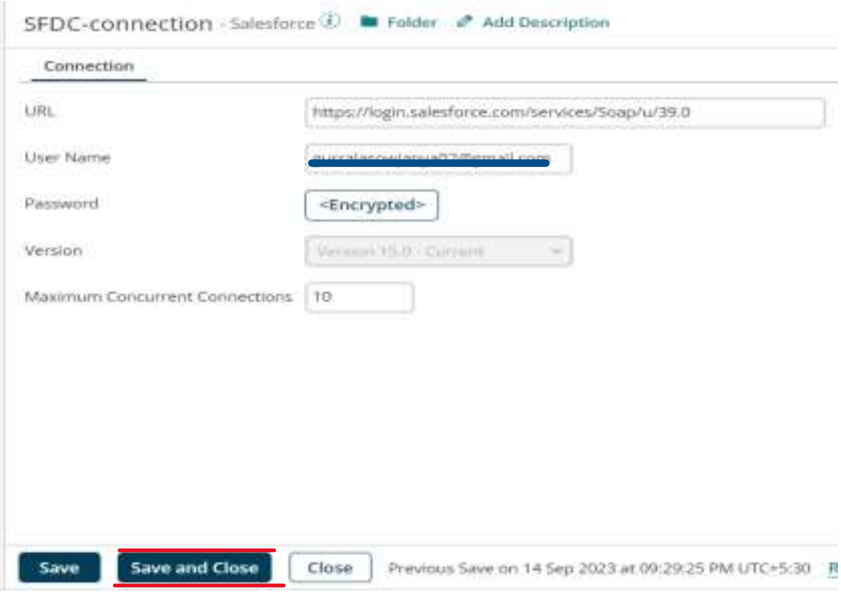
Result Content Type: ● Choose text/plain ○ Enter

Attachment Cache: Choose...

Previous Save on 27 Nov 2023 at 01:20:51 PM UTC+5:30

Step 2.3: Now drag and drop a salesforce connector into process canvas

Step 2.4: Configure Connection Component with valid credentials



SFDC-connection · Salesforce

URL: https://login.salesforce.com/services/Soap/u/39.0

User Name: [redacted]

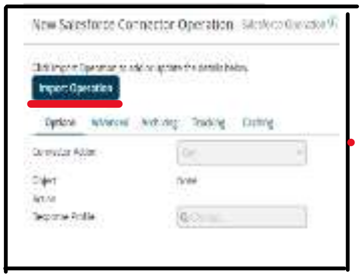
Password: <Encrypted>

Version: Version 15.0 - Current

Maximum Concurrent Connections: 10

Previous Save on 14 Sep 2023 at 09:29:25 PM UTC+5:30

Step 2.5: Choose the action as get and configure the operation component and import the profile into Boomi.



New Salesforce Connector Operation · Salesforce Operation

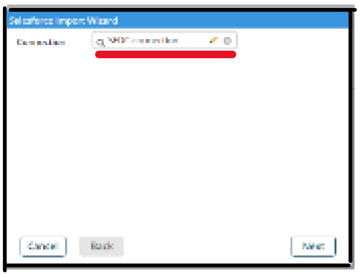
Click Import Operation to add or update the details below.

Connector Action: Get

Object: None

Response Profile: Choose...

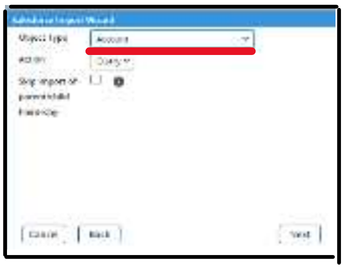
(1) click on import operation



Salesforce Import Wizard

Connection: SFDC-connection

(2) choose connection



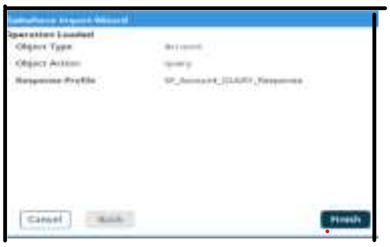
Salesforce Import Wizard

Object Type: Account

Action: Query

Skip import of parent/child Relationship: □

(3) choose Account



Salesforce Import Wizard

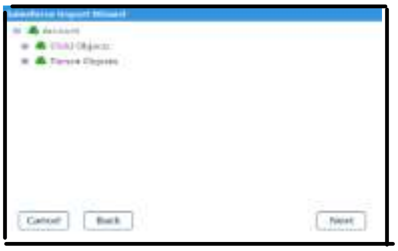
Operation Loaded

Object Type: Account

Object Action: query

Response Profile: SF_Account_QUERY_Response

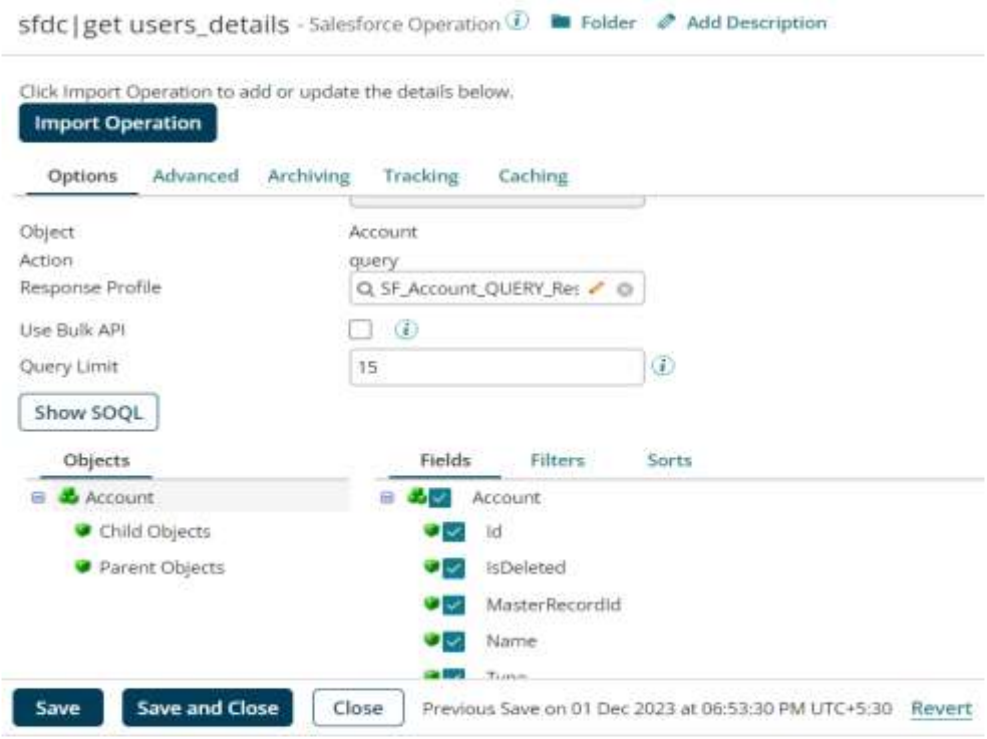
(5) Click on Finish



Salesforce Import Wizard

(4) click on Next

Now the operation component will look as shown below



sfdc|get users_details - Salesforce Operation

Click Import Operation to add or update the details below.

Object: Account
Action: query
Response Profile: SF_Account_QUERY_Res
Use Bulk API: □
Query Limit: 15

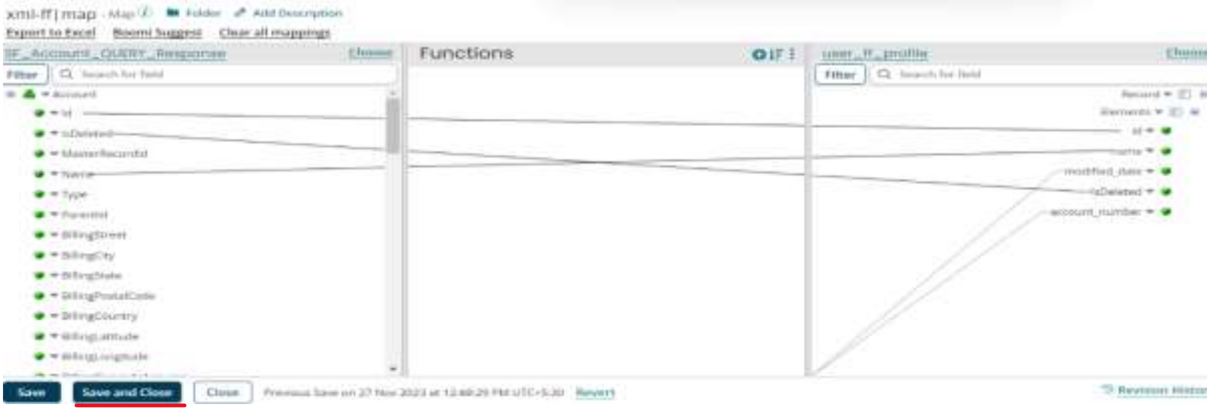
Fields: ☑ Account, ☑ Id, ☑ IsDeleted, ☑ MasterRecordId, ☑ Name, ☑ Type

Previous Save on 01 Dec 2023 at 06:53:30 PM UTC+5:30

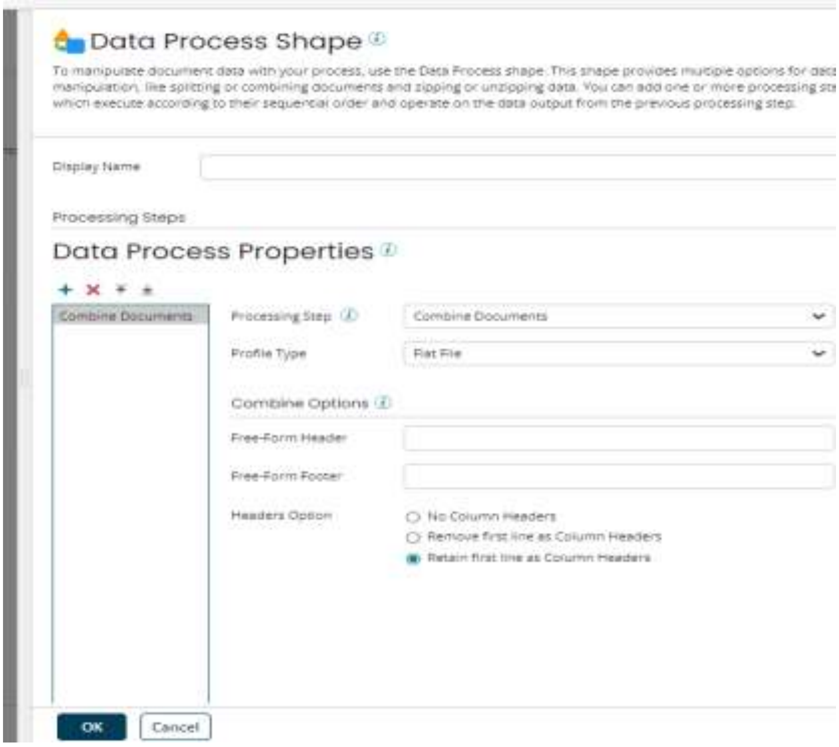
Step 2.6: Drag and drop a map shape into the process canvas. For source profile choose the response profile which we imported and configure a flat file profile as target profile

Mapping Details:

Source Element	Target Element
Id	id
IsDeleted	IsDeleted
Name	name
LastModifiedDate	modified_date
AccountNumber	account_number



Step 2.7**:** Drag and drop a data process shape into the process canvas and configure as shown in below image and click on OK.



Data Process Shape

To manipulate document data with your process, use the Data Process shape. This shape provides multiple options for dat manipulation, like splitting or combining documents and zipping or unzipping data. You can add one or more processing st which execute according to their sequential order and operate on the data output from the previous processing step.

Display Name

Processing Steps

Data Process Properties

Combine Documents

Processing Step: Combine Documents

Profile Type: Flat File

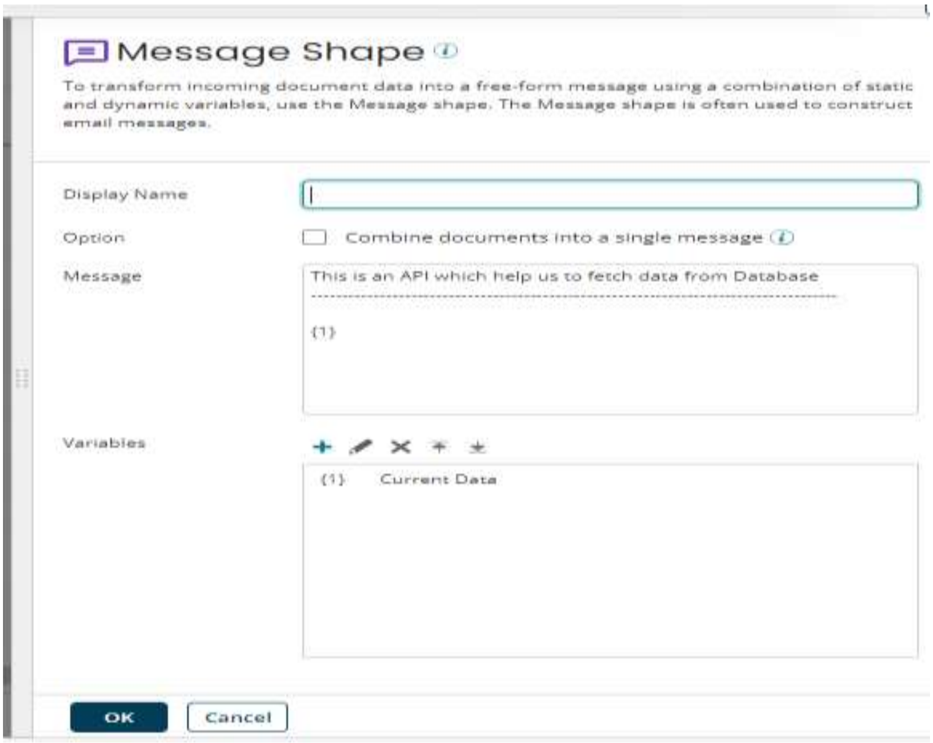
Combine Options

Free-Form Header

Free-Form Footer

Headers Option:
○ No Column Headers
○ Remove first line as Column Headers
● Retain first line as Column Headers

Step 2.8**:** Drag and drop a message shape into the process canvas and configure as shown in below image.



Message Shape

To transform incoming document data into a free-form message using a combination of static and dynamic variables, use the Message shape. The Message shape is often used to construct email messages.

Display Name

Option: □ Combine documents into a single message

Message:

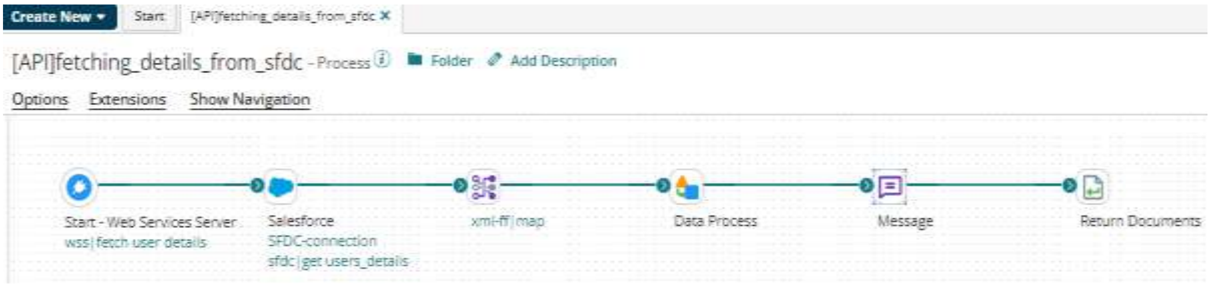
```
This is an API which help us to fetch data from Database
-----------------------------------------------------------
{1}
```

Variables:

{1} Current Data

Step 2.9: Drag and drop a return documents shape and attach it to the message shape.

The complete flow will look as shown below.

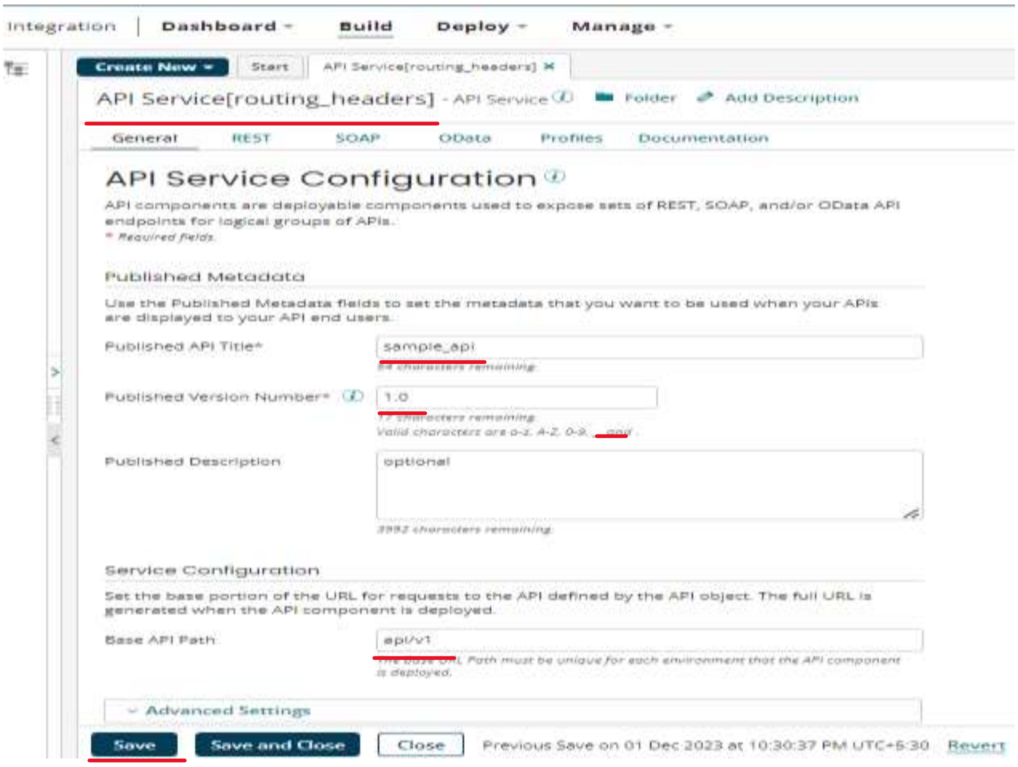


Till now we have created two APIs with same simple URL path (refer step2.2 and step1.4 and see simple URL path)

Configuring API Service Component:

Now expose the above APIs using API Service Component by following the below steps.

Step 3.1: Create an API Service Component and configure the details under General tab as shown below.



API Service Configuration

API components are deployable components used to expose sets of REST, SOAP, and/or OData API endpoints for logical groups of APIs.

* Required fields.

Published Metadata

Use the Published Metadata fields to set the metadata that you want to be used when your APIs are displayed to your API end users.

Published API Title*: sample_api
54 characters remaining.

Published Version Number*: 1.0
17 characters remaining.
Valid characters are a-z, A-Z, 0-9, _ and .

Published Description: optional
3992 characters remaining.

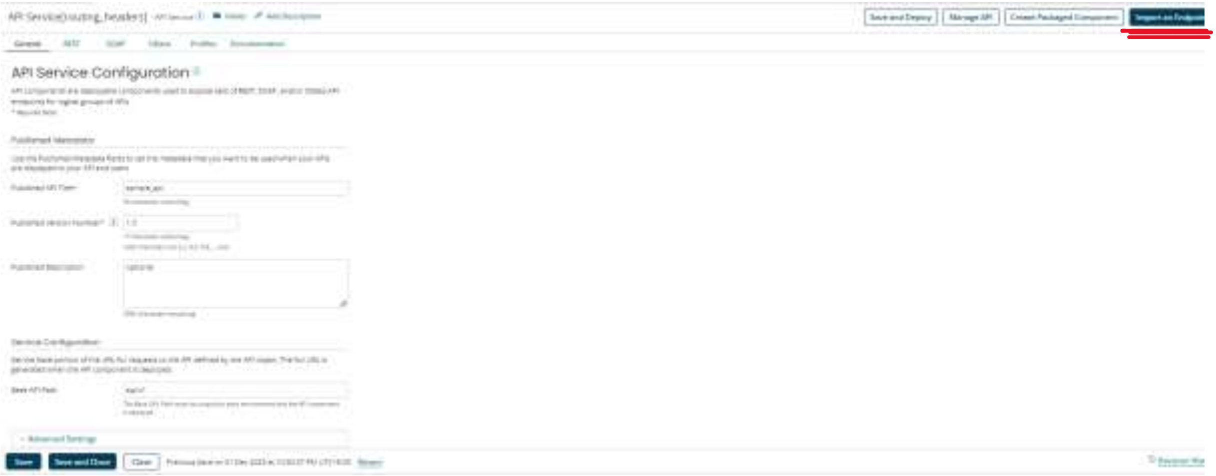
Service Configuration

Set the base portion of the URL for requests to the API defined by the API object. The full URL is generated when the API component is deployed.

Base API Path: api/v1
The base URL Path must be unique for each environment that the API component is deployed.

Previous Save on 01 Dec 2023 at 10:30:37 PM UTC+5:30

Step 3.2: Now click on import an endpoint



API Service(routing_headers) API Service

API Service Configuration

Published Metadata

Published API Name* sample_api

Published Version Number* 1.0

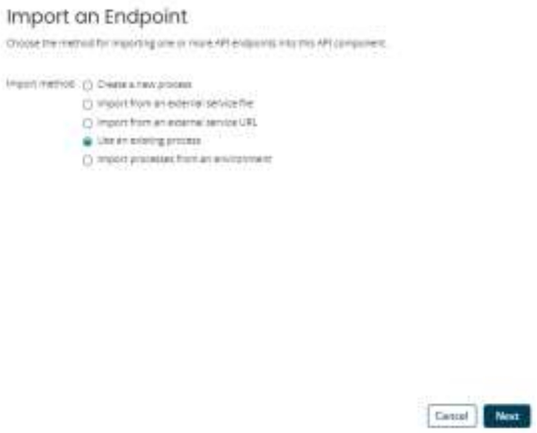
Published Description optional

Service Configuration

Base API Path ws/rest

Advanced Settings

Step3.3: select Use an existing process and click on Next

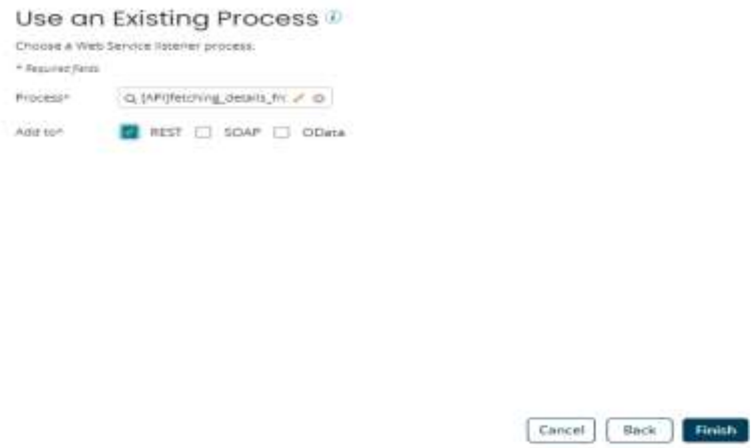


Import an Endpoint

Choose the method for importing one or more API endpoints into this API component.

Import method: ○ Create a new process
○ Import from an external service file
○ Import from an external service URL
● Use an existing process
○ Import processes from an environment

Step3.4: Choose the process ([**API]fetching_details_from_db**) which we have configured earlier and check REST checkbox and click on Finish.



Use an Existing Process

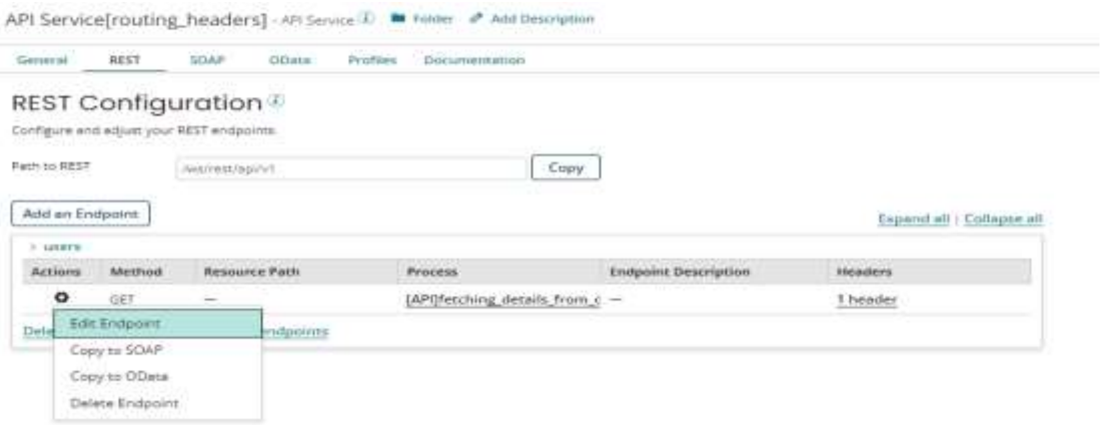
Choose a Web Service listener process.

* Required fields

Process* [API]fetching_details_fr

Add to* ☑ REST ☐ SOAP ☐ OData

Step3.5: Go to REST tab. We can see the endpoint which got imported. Click on the gear icon and click on Edit Endpoint.



API Service[routing_headers] - API Service

REST Configuration

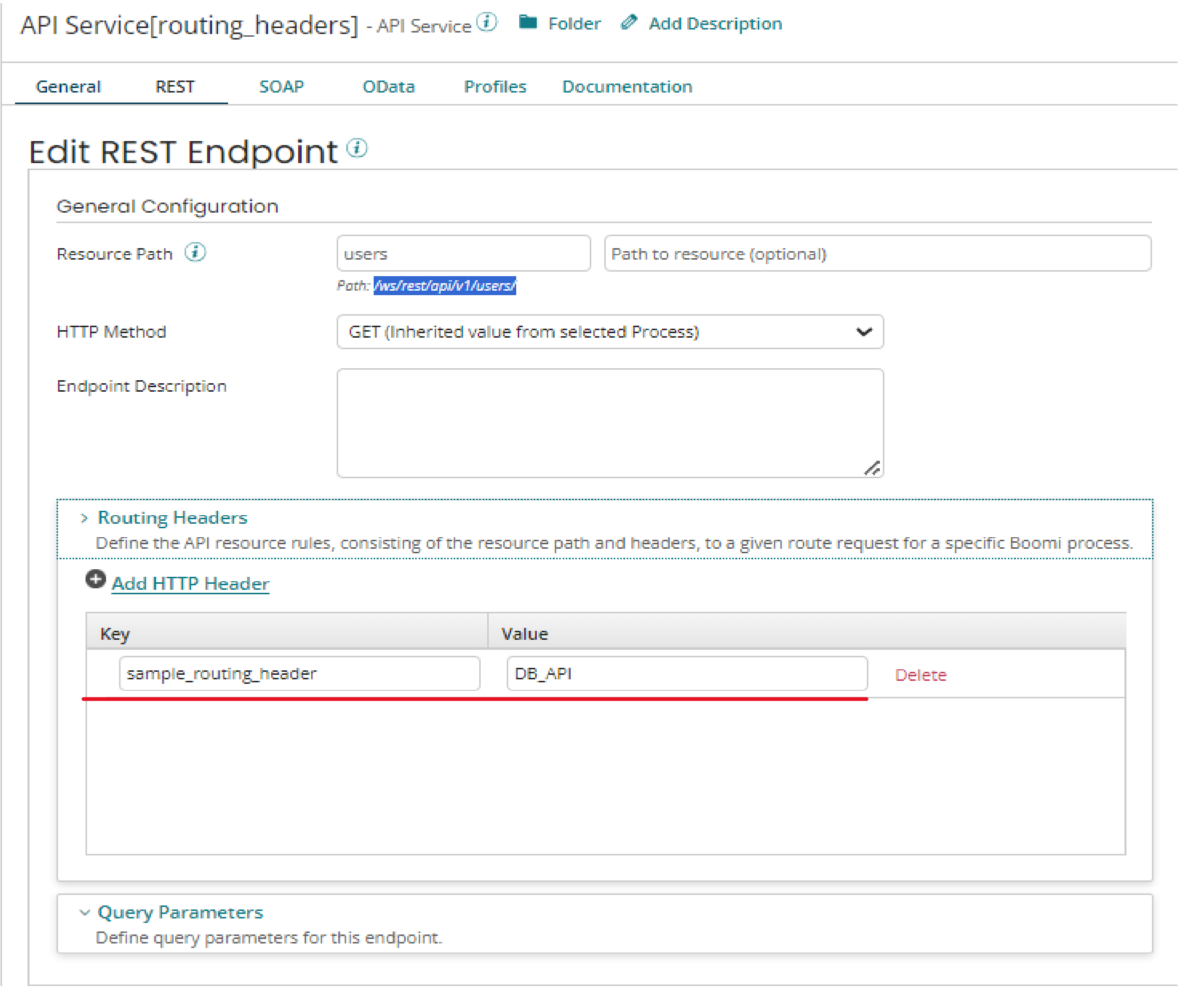
Configure and adjust your REST endpoints

Path to REST: /ws/rest/api/v1

users

Actions	Method	Resource Path	Process	Endpoint Description	Headers
⚙	GET	—	[API]fetching_details_from_c	—	1 header

Step3.6: Copy the resource path and paste it in a notepad. Click on Add HTTP Header and configure as shown below and click on ok.



API Service[routing_headers] - API Service

Edit REST Endpoint

General Configuration

Resource Path: users | Path to resource (optional)

Path: /ws/rest/api/v1/users/

HTTP Method: GET (Inherited value from selected Process)

Endpoint Description:

Routing Headers

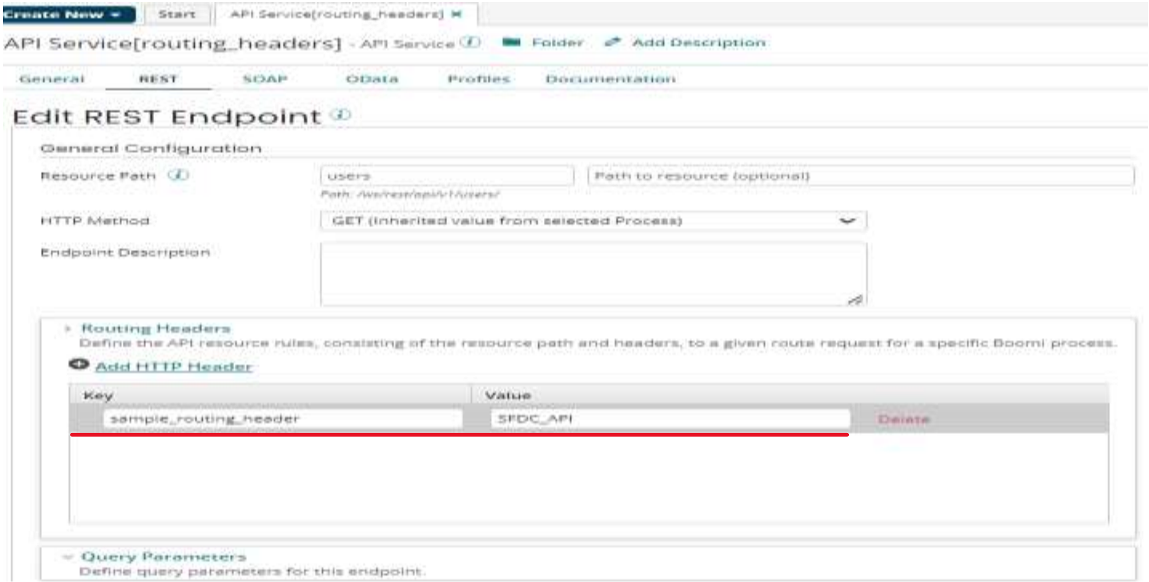
Define the API resource rules, consisting of the resource path and headers, to a given route request for a specific Boomi process.

Key	Value
sample_routing_header	DB_API

Query Parameters

Define query parameters for this endpoint.

Step3.7: Repeat the steps **3.2-3.6** to import another endpoint. [Note: choose process **[API]fetching_details_from_sfdc** in step3.4 and configure a different value i.e., SFDC_API for sample_routing_header in step3.6.



Edit REST Endpoint

General Configuration

Resource Path: users | Path to resource (optional)

Path: /ws/rest/api/v1/users/

HTTP Method: GET (Inherited value from selected Process)

Endpoint Description:

Routing Headers

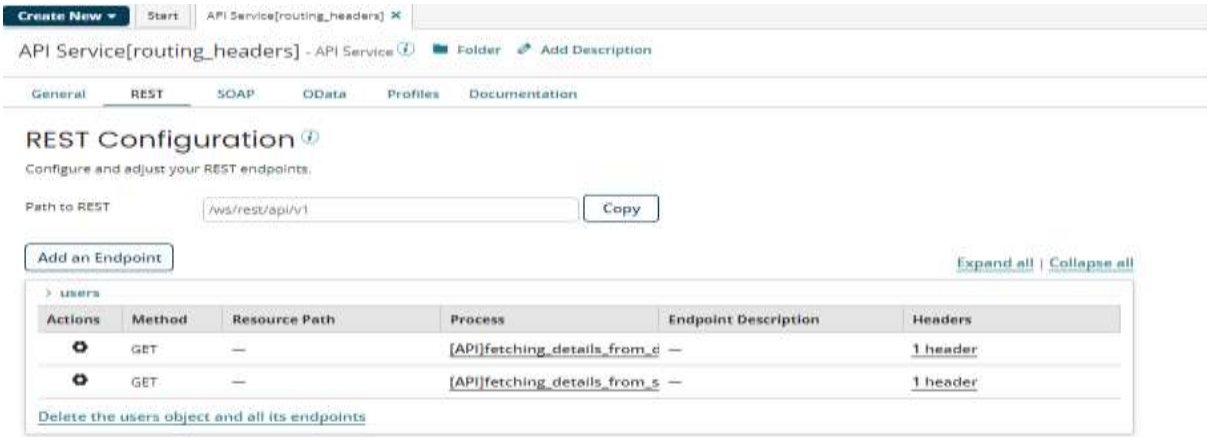
Define the API resource rules, consisting of the resource path and headers, to a given route request for a specific Boomi process.

Key	Value
sample_routing_header	SFDC_API

Query Parameters

Define query parameters for this endpoint.

Step3.8: Now the API Service Component will look as shown below.



REST Configuration

Configure and adjust your REST endpoints.

Path to REST: /ws/rest/api/v1

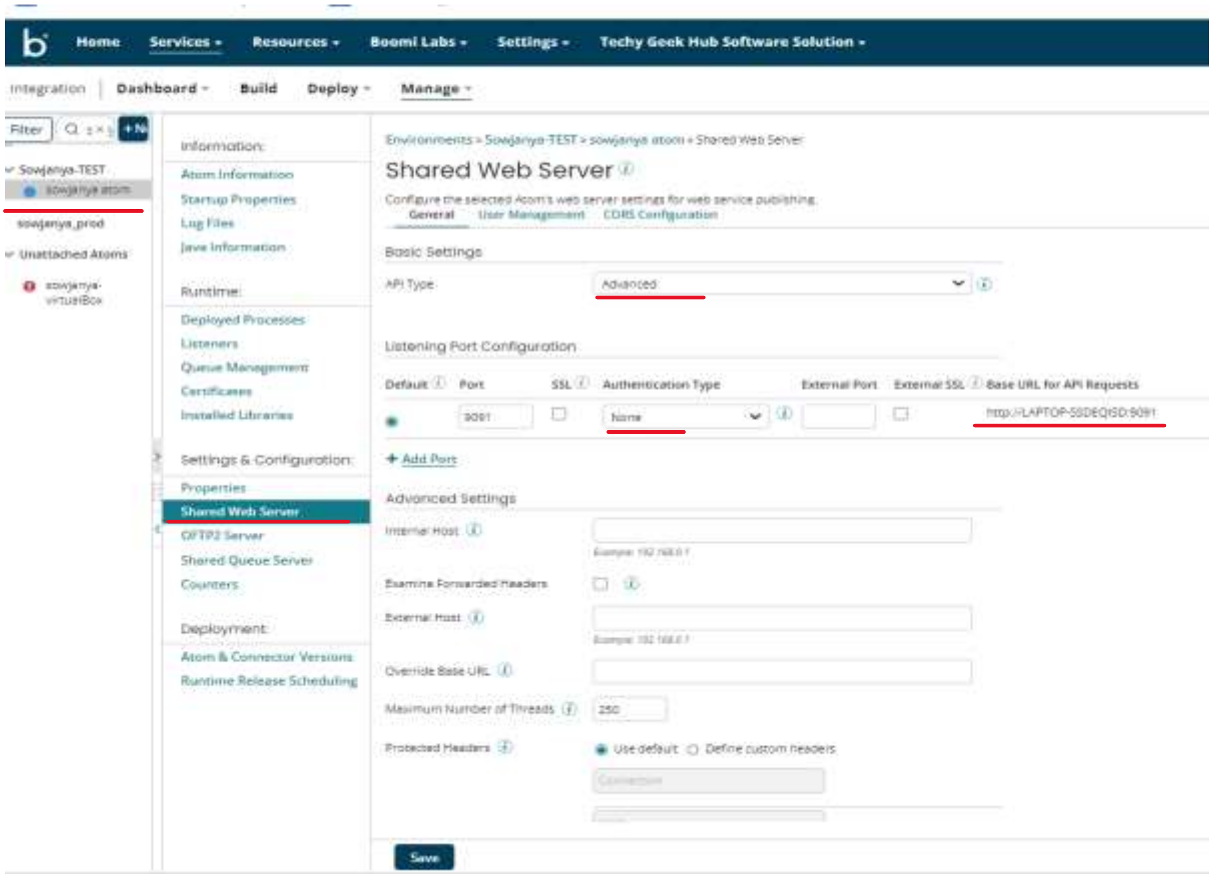
users

Actions	Method	Resource Path	Process	Endpoint Description	Headers
⚙	GET	—	[API]fetching_details_from_d	—	1 header
⚙	GET	—	[API]fetching_details_from_s	—	1 header

Delete the users object and all its endpoints

- Now create packaged component of API Service Component (API Service[routing_headers]) and the processes (**[API]fetching_details_from_sfdc, [API]fetching_details_from_db**) and deploy to your environment.

- Go to Manage→Atom Management. Choose the atom which is attached to your environment. Go to Shared Web Server panel and configure the API Type as **Advanced** and Authentication Type as **None** and copy the **Base URL**. Click on save to make the changes come into effect.



Environments > Sowjanya-TEST > sowjanya atom > Shared Web Server

Shared Web Server

Configure the selected Atom's web server settings for web service publishing.

Basic Settings

API Type: Advanced

Listening Port Configuration

Default	Port	SSL	Authentication Type	External Port	External SSL	Base URL for API Requests
●	9091	□	None		□	http://LAPTOP-SSDEQISD:9091

Advanced Settings

Internal Host: Example: 192.168.0.1

Examine Forwarded Headers: □

External Host: Example: 192.168.0.1

Override Base URL:

Maximum Number of Threads: 250

Protected Headers: ● Use default ○ Define custom headers

Connection

- Prepend the Base URL to resource path which we copied in notepad in step3.6.
- Now endpoint of two resources is same with different values for routing header as shown below

http://LAPTOP-SSDEQISD:9091/ws/rest/api/v1/users/

Routing Header	Value	Process Attached
sample_routing_header	DB_API	[API]fetching_details_from_db
sample_routing_header	SFDC_API	[API]fetching_details_from_sfdc

(Note: Base URL might change depending on server where the APIs are hosted)

- Now test the resources using testing tool like Postman.

Steps to Test in Postman:

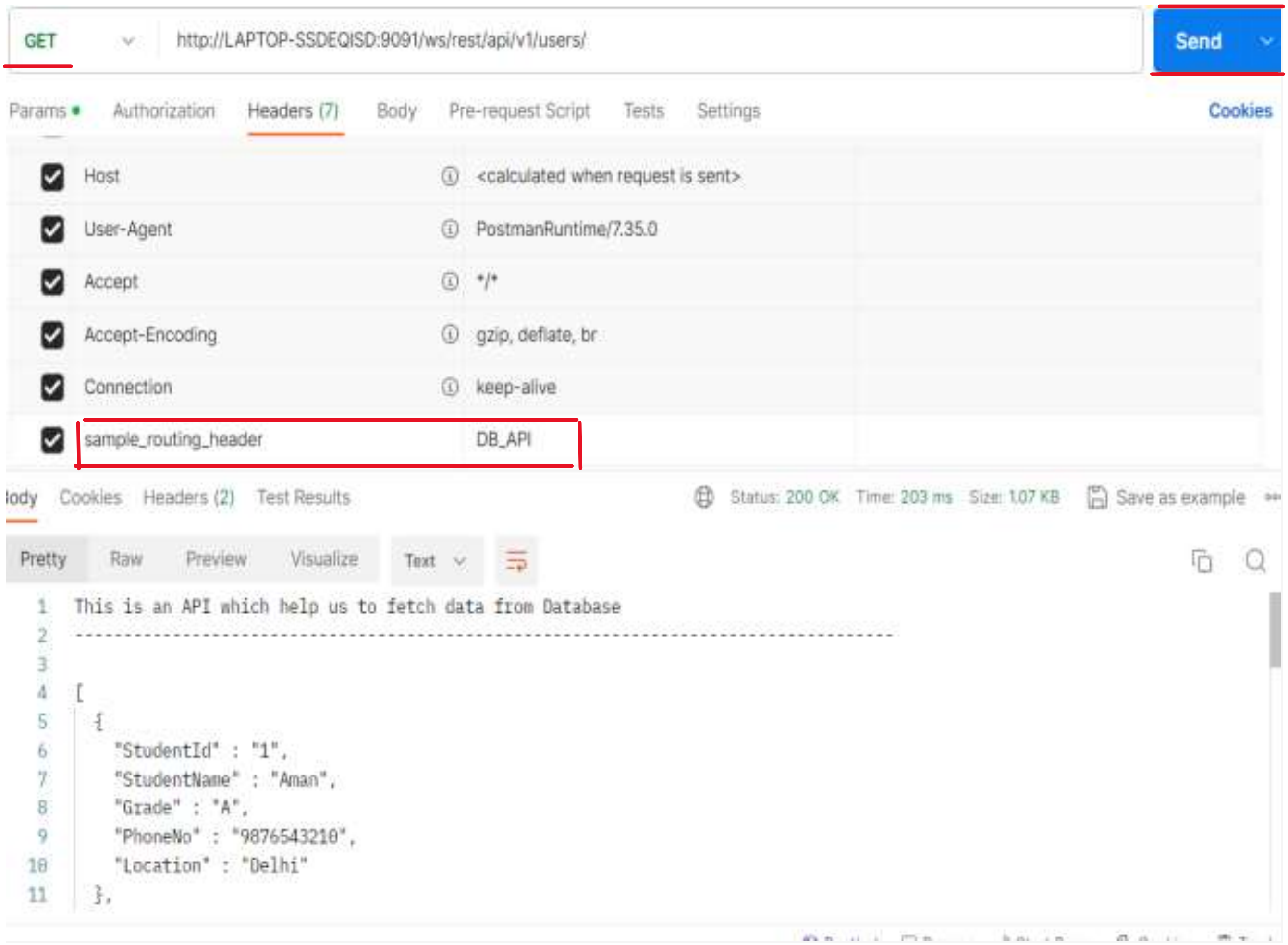
(1) Create a new collection. Within that create new request

(2) paste the endpoint (http://LAPTOP-SSDEQISD:9091/ws/rest/api/v1/users/)

(3) Go to Headers tab and add the header and configure the name as sample_routing_header

Scenario1: For the sample_routing_header assign the value as DB_API

(4) click on send



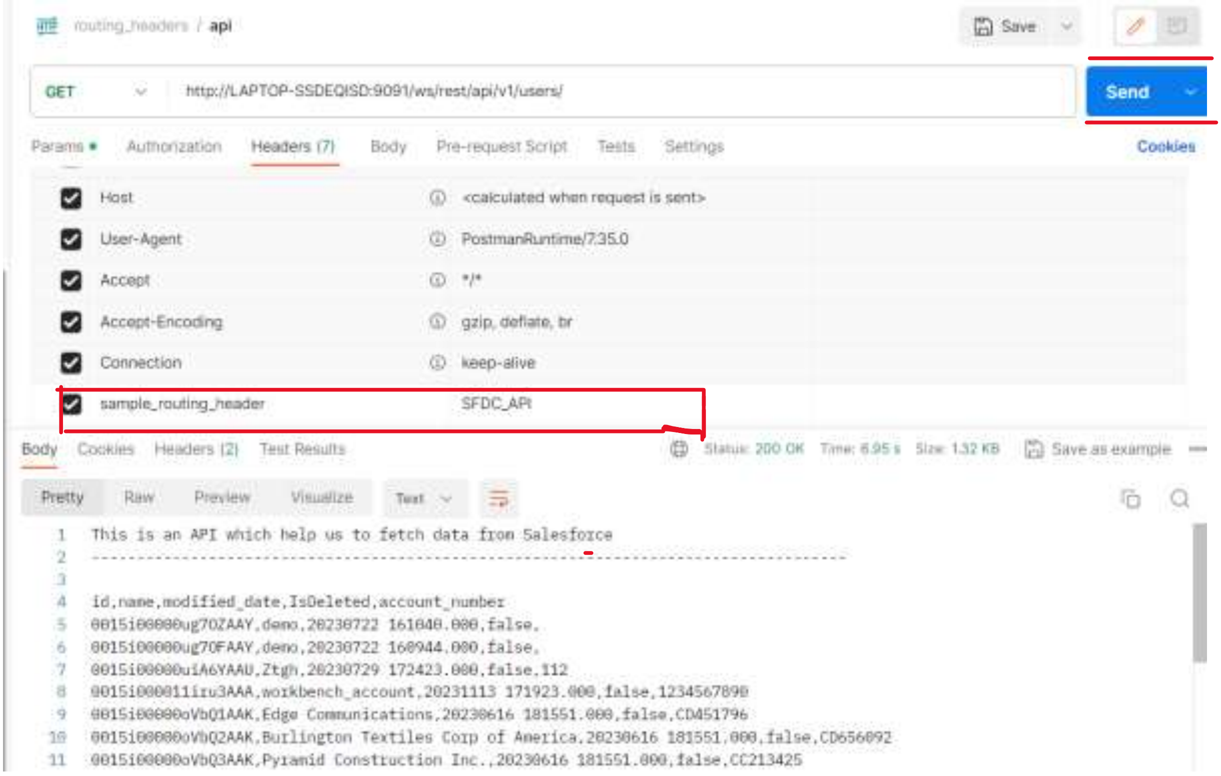
☑	Host	<calculated when request is sent>
☑	User-Agent	PostmanRuntime/7.35.0
☑	Accept	*/*
☑	Accept-Encoding	gzip, deflate, br
☑	Connection	keep-alive
☑	sample_routing_header	DB_API

```
This is an API which help us to fetch data from Database
---------------------------------------------------------------------------------


[
  {
    "StudentId" : "1",
    "StudentName" : "Aman",
    "Grade" : "A",
    "PhoneNo" : "9876543210",
    "Location" : "Delhi"
  },
```

From the response it's concluded that the request is automatically routed to the webservice **[API]fetching_details_from_db**

Now modify the value of **sample_routing_header** to SFDC_API and click on send



GET http://LAPTOP-SSDEQISD:9091/ws/rest/api/v1/users/

☑	Host	<calculated when request is sent>
☑	User-Agent	PostmanRuntime/7.35.0
☑	Accept	*/*
☑	Accept-Encoding	gzip, deflate, br
☑	Connection	keep-alive
☑	sample_routing_header	SFDC_API

```
This is an API which help us to fetch data from Salesforce
----------------------------------------------------------------------------------------------------

id,name,modified_date,IsDeleted,account_number
0015i00000ug7OZAAY,demo,20230722 161040.000,false,
0015i00000ug7OFAAY,demo,20230722 160944.000,false,
0015i00000uiA6YAAU,Ztgh,20230729 172423.000,false,112
0015i000011iru3AAA,workbench_account,20231113 171923.000,false,1234567890
0015i00000oVbQ1AAK,Edge Communications,20230616 181551.000,false,CD451796
0015i00000oVbQ2AAK,Burlington Textiles Corp of America,20230616 181551.000,false,CD656092
0015i00000oVbQ3AAK,Pyramid Construction Inc.,20230616 181551.000,false,CC213425
```

From the response we can see that the request is automatically routed to the webservice **[API]fetching_details_from_sfdc**

Conclusion:

Whenever we have multiple resources with same endpoint routing headers are used to route to specific resource based on specified header value.

References:

- https://platform.boomi.com/
- https://help.boomi.com/
- https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/int-API_Service_components_8f868bda-5099-4e1f-ad16-5648f98b68b2
- https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/t-atm-Configuring_a_REST_route_in_an_API_component_c61c0d7e-ecc9-47c3-8fc9-452c8ffc62aa
- https://help.boomi.com/docs/Atomsphere/API%20Management/Topics/r-atm-API_REST_tab_53469a70-6574-486b-a6cf-58fc147546fc

TGH
Making Integrations Simpler

boomi Partner

TGH Software Solutions Pvt. Ltd.

www.techygeekhub.com

At TGH, we specialize in driving digital transformation through seamless Integration Technologies.

Operating as an INTEGRATION FACTORY, we serve as a one-stop shop for all your integration needs. Our expert team is well-versed in enterprise software and legacy system integration, along with leading iPaaS technologies like Boomi, MuleSoft, Workato, OIC, and more.

We're committed to enhancing business processes and solving problems through our integration expertise.

Email address

connect@techygeekhub.com

Phone number

+ 011-40071137
+ 91-8810610395

Our offices

Noida Office
iThum
Plot No -40, Tower A,
Office No: 712,
Sector-62, Noida,
Uttar Pradesh, 201301

Hyderabad Office
Plot no: 6/3, 5th Floor,
Techno Pearl Building,
HUDA Techno Enclave,
HITEC City, Hyderabad,
Telangana 500081