

Informatica Cloud Application Integration

NetSuite Connector

September 2018

Contents

| | |
|--|---|
| Overview: | 1 |
| Prerequisites: | 1 |
| Use Case #1: Retrieving Customer information by ID | 2 |
| Use Case #2: Update the telephone number in an existing NetSuite Customer record | 2 |
| Quick Setup | 3 |
| Detailed Setup: | 5 |

Overview:

NetSuite by Oracle is one of the leading cloud ERP solutions available today.

This NetSuite Service Connector allows Informatica Cloud Application Integration (CAI) developers to incorporate access to NetSuite in their real-time integration scenarios.

This Service Connector and associated assets demonstrate how to authenticate against NetSuite, how to retrieve Customer data from NetSuite, and how to update Customer data that is already in NetSuite. Additional functionality is available for the reader to explore on their own.

Prerequisites:

- A NetSuite account with API access and credentials
 - Username
 - Password
 - Account
 - URL
- Appropriate role/permissions on Informatica Cloud Application Integration to perform an IMPORT operation

Use Case #1: Retrieving Customer information by ID

1. With credentials embedded in the Connection, query NetSuite with an ID
 - a. A valid ID will return:
 - i. Customer Name
 - ii. ID
 - iii. Phone
 - iv. Balance
 - v. Email
 - b. A bad ID will simply echo the ID.

Use Case #2: Update the telephone number in an existing NetSuite Customer record

1. With credentials embedded in the Connection, update NetSuite with a NetSuite ID, Customer Name, and the new telephone number
2. If the update is successful, the process will query NetSuite again with the same ID to echo the updated information.
3. If the update is unsuccessful, retrieve and display the error(s).

Quick Setup

1. Download [NetSuiteConnectorPackage-0918.zip](#) from the [Informatica CAI Git Repository](#)
2. Import the package "NetSuiteConnectorPackage.zip" into your CAI workspace
3. This creates a project named "NetSuite", with 4 assets in it.
 - a. "NetSuite API" Service Connector
 - b. "NetSuite-API-actions" Connection
 - c. "NetSuite Customer Lookup by ID" Process
 - d. "NetSuite Customer Update" Process
4. Go to the "NetSuite" project
5. Publish the "NetSuite API" Service Connector
6. Edit the "NetSuite-API-actions" Connection
 - a. Under Properties, enter your organization's NetSuite credentials for:
 - i. username
 - ii. password
 - iii. account
 - iv. URL
 - b. Save the Connection
 - c. Publish the Connection
7. Publish the other components in this order
 - a. "NetSuite Customer Lookup by ID" Process
 - b. "Netsuite Customer Update" Process
8. Test the "NetSuite Lookup by ID" process in your REST tool of choice (Postman, RESTed, etc.)
 - a. Define a POST operation
 - b. The Request URL is the Service Address from the published "NetSuite Lead Retrieval by ID" process
 - i. Actions-> Properties Detail...-> Service URL-> Copy
 - c. The request body is of type "application/json"
 - d. Set the body of the request to this:

```
{
  "InputCustomerID": "<your ID>"
}
```
 - e. Invoke the operation ("Send", in the case of Postman, "Send Request" with Rested)
 - f. Assuming valid credentials and Lead ID, you should receive a response that looks similar to this:

```
{
  "o_CustomerName": "<your customer name>",
  "o_CustomerPhone": "<your customer phone>",
  "o_CustomerBalance": "<your customer balance>",
  "o_CustomerEmail": "<your customer email>",
  "o_InputID": "<your ID>"
}
```

}

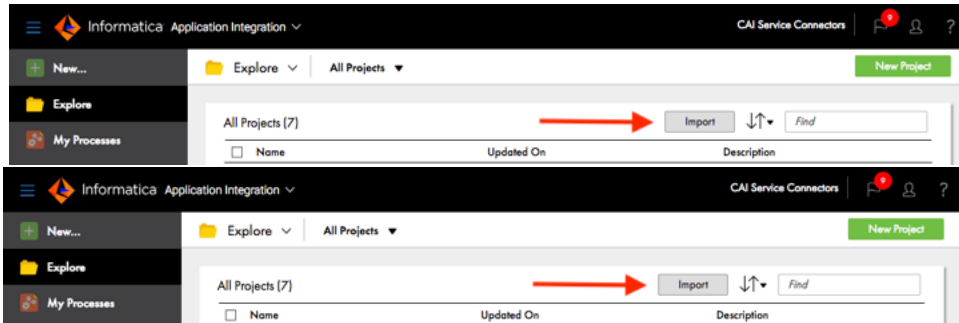
9. Test the “NetSuite Customer Update” process in your REST tool of choice (Postman, RESTed, etc.)
 - a. Define a POST operation
 - b. The Request URL is the Service Address from the published “NetSuite Customer Update” process
 - i. Actions-> Properties Detail...-> Service URL-> Copy
 - c. The request is of type application/json
 - d. Set the body of the request to this:

```
{
  "InputCoName": "<your customer name as in NetSuite>"
  "InputCustomerID": <your customer ID>,
  "InputPhone": "<your new customer phone number>"
}
```
 - e. Invoke the operation (“Send”, in the case of Postman, “Send Request” with Rested)
 - f. Assuming valid credentials and ID, you should receive a response that looks similar to this:

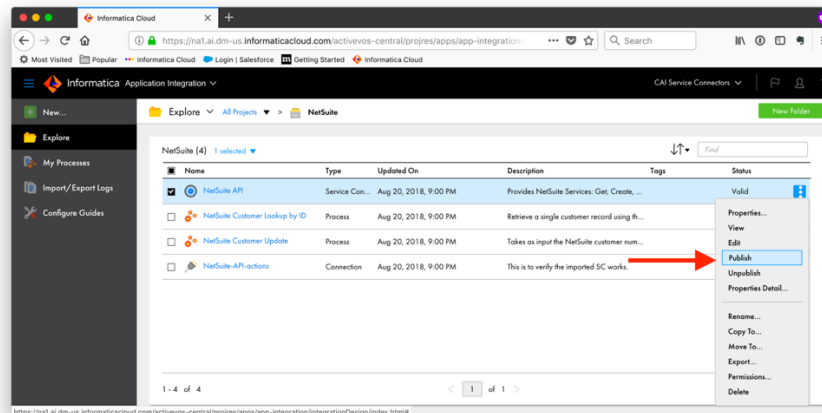
```
{
  "o_CustomerName": "<your customer name",
  "o_CustomerPhone": "<your updated customer phone number>",
  "o_CustomerBalance": "<your customer balance>",
  "o_CustomerEmail": "<your customer email address>",
  "o_InputID": "your customer id"
}
```

Detailed Setup:

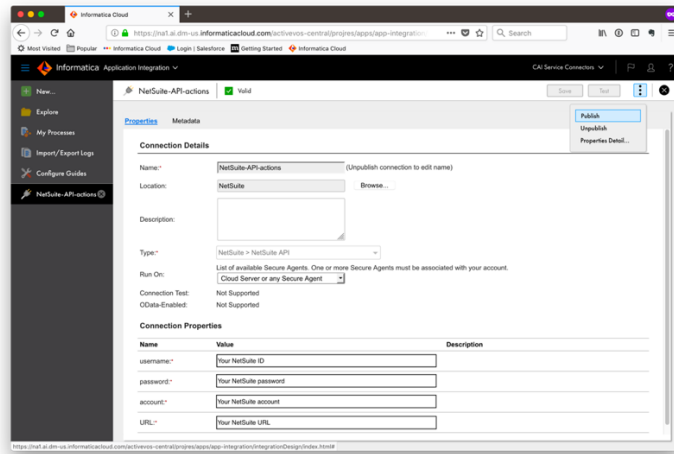
1. Download [NetSuiteConnectorPackage-0918.zip](#) from the [Informatica CAI Git Repository](#)
2. Import the package "NetSuiteConnectorPackage-0918.zip" into your CAI workspace



3. Upon completion, the import process will have created a new project named "NetSuite" at the root level of your CAI workspace, containing 4 assets.
 - i. "NetSuite API" Service Connector
 - ii. "NetSuite-API-actions" Connection
 - iii. "NetSuite Customer Lookup by ID" Process
 - iv. "NetSuite Customer Update" Process
4. Go to the newly created "NetSuite" project
5. Publish the "NetSuite API" Service Connector



6. Edit the “NetSuite-API-actions” Connection.
 - a. Insert your NetSuite credentials into the Connection Details on the Properties area.



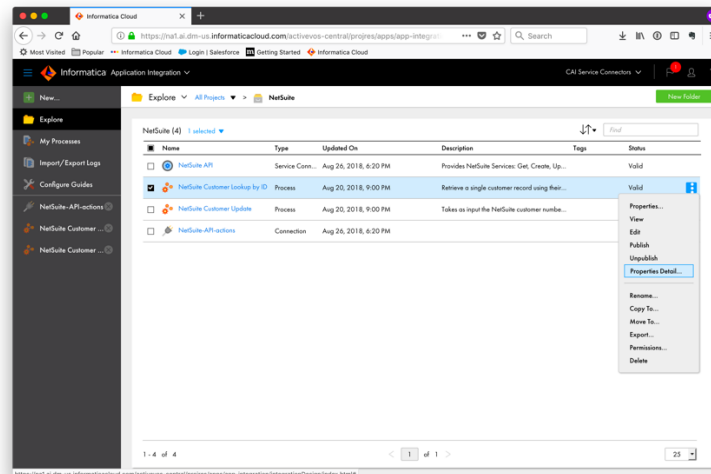
- b. Save the “NetSuite-API-actions” Connection.
 - c. Publish the “NetSuite-API-actions” Connection.

7. Publish the other assets

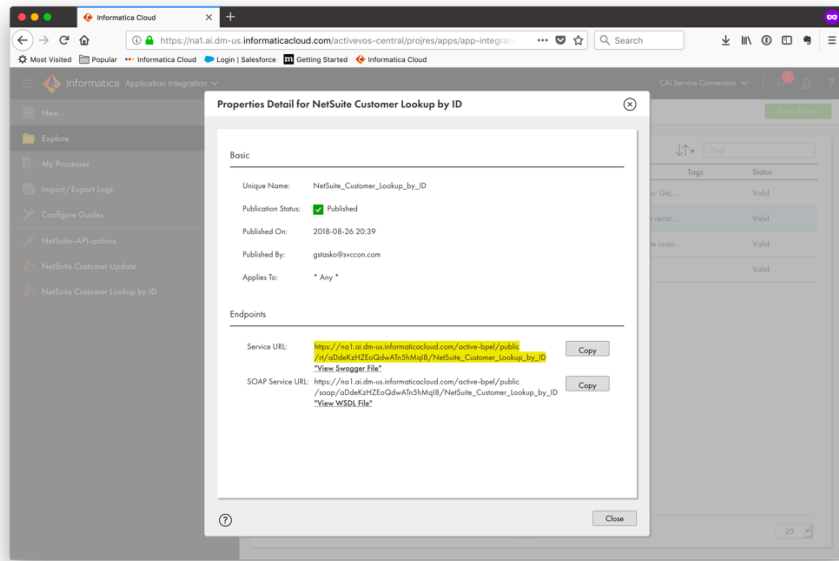
- a. Publish the "NetSuite Customer Lookup by ID" Process.
 - b. Publish the “NetSuite Customer Update” Process

8. Test the “NetSuite Customer Lookup by ID” process:

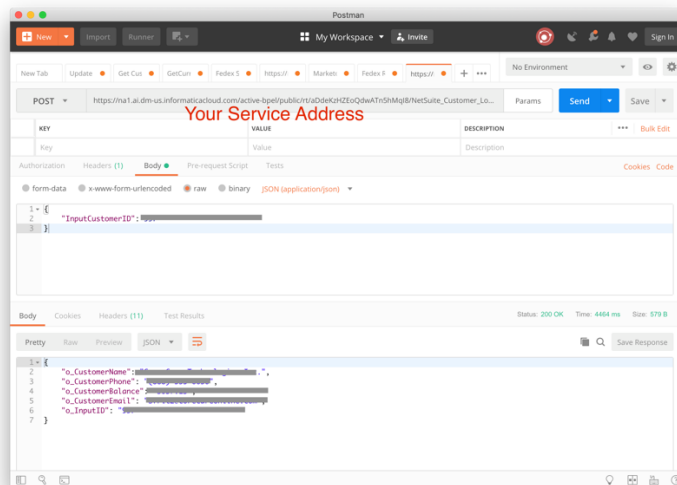
- a. Open the “Properties Detail” of the " NetSuite Customer Lookup by ID" process.



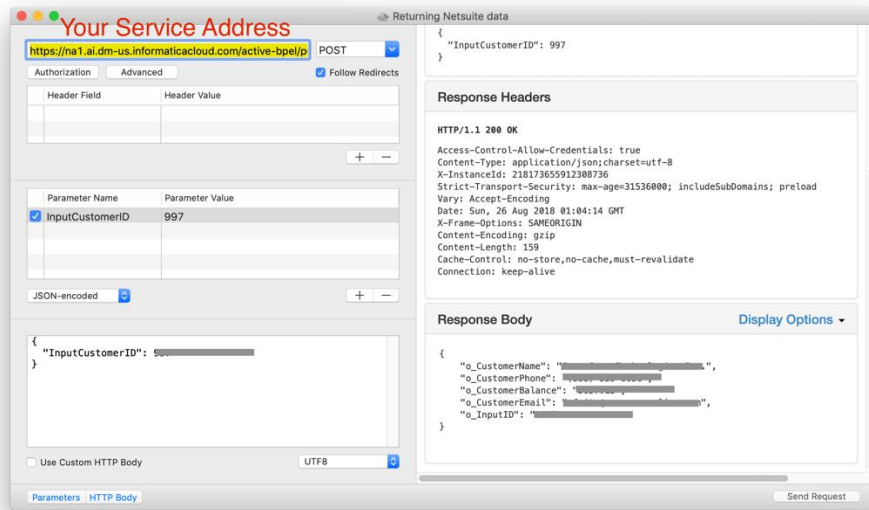
- b. Retrieve the “Service URL” of the "NetSuite Customer Lookup by ID" process.



- c. Define a POST operation in your REST tool
- d. The Request URL is the Service Address from the published “Marketo Lead Retrieval by ID” process
- e. The request is of type “application/json”
- f. Set the body of the request to this:
- ```
{
 "InputCustomerID": <your ID>
}
```
- g. Postman

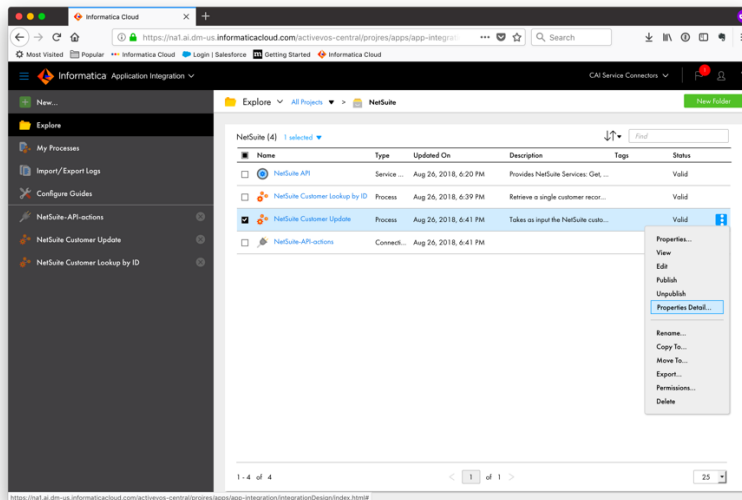


RESTed

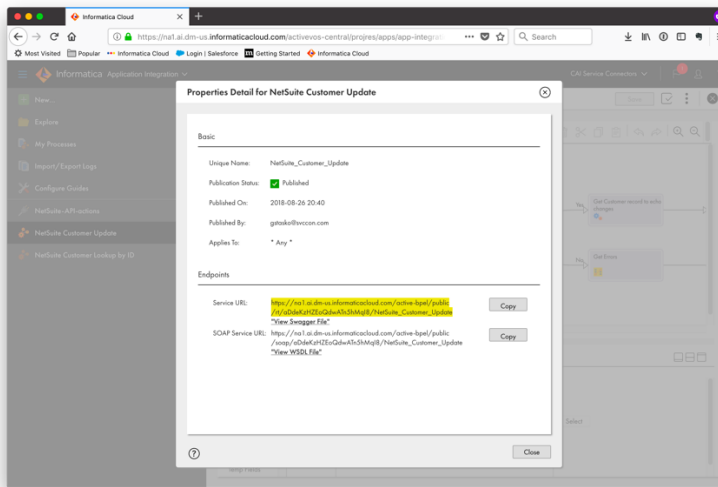




9. Test the “NetSuite Customer Update” Process in your REST tool of choice \*Postman, RESTed, etc.):
  - a. Define a POST operation
  - b. Get the Request URL of the “Properties Detail”
    - i. Open the “Properties Detail” of the "NetSuite Customer Update" process.



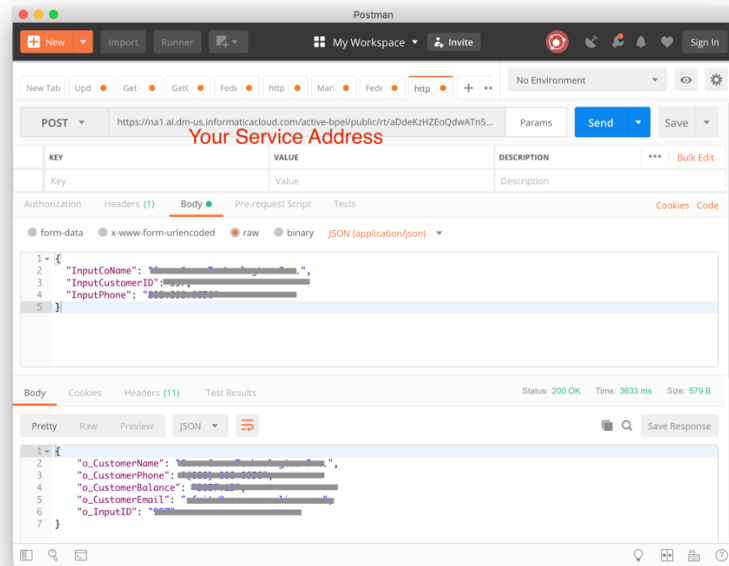
- ii. Copy the “Service URL” of the " NetSuite Customer Update" process.



- c. The request is of type “application/json”.
    - d. Set the body of the request to this:
 

```
{
 "InputCoName": "<your customer name as in NetSuite>"
 "InputCustomerID": <your customer ID>,
 "InputPhone": "<your new customer phone number>"
 }
```
    - g. Invoke the operation (“Send”, in the case of Postman, “Send Request” with Rusted)

## i. Postman



## RESTed

