

# Informatica Cloud Application Integration

## ServiceNow Connector Package

### September 2018

#### Contents

|  |   |
|--|---|
| Overview: .....  | 1 |
| Prerequisites: .....   | 1 |
| Use Case #1: Create a new ServiceNow Account .....                   | 2 |
| Use Case #2: Retrieve details of an existing ServiceNow Account..... | 2 |
| Quick Setup:.....  | 3 |
| Detailed Setup: .....  | 5 |

#### Overview:

ServiceNow is one of the leading cloud-based, customer-service solutions available today.

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

This ServiceNow Connector and associated assets demonstrate how to authenticate against ServiceNow, how to create an account in ServiceNow, and how to retrieve account information from ServiceNow. Additional functionality is available for the reader to explore on their own.

#### Prerequisites:

- A ServiceNow account with API access and the following login credentials
  - Client ID
  - Client Secret
  - User Name
  - User Password
  - Service URL
  - Grant Type
- Appropriate role/permissions on Informatica Cloud Application Integration to perform an IMPORT operation

## Use Case #1: Create a new ServiceNow Account

1. With credentials embedded in the Connection, create an account
  - a. Inputs:
    - i. inputName
    - ii. inputCity
    - iii. inputZIP
  - b. Outputs
    - i. o\_AccessToken
    - ii. o\_RefreshToken
    - iii. a\_sys\_id
    - iv. o\_city
    - v. o\_name

## Use Case #2: Retrieve details of an existing ServiceNow Account

1. With credentials embedded in the Connection, invoke the “Retrieve account in ServiceNow” process.
  - a. Input
    - i. input\_sys\_id
  - b. Output
    - i. o\_AccessToken
    - ii. o\_RefreshToken
    - iii. a\_sys\_id
    - iv. o\_city
    - v. o\_name

## Quick Setup:

1. Download the [ServiceNowConnectorPackage0918.zip](#) from the [Informatica CAI Git Repository](#)
2. Import the package "ServiceNowConnectorPackage.zip" into your CAI workspace
3. This creates a project named "ServiceNow", with 4 assets in it.
  - a. "ServiceNow Table API" Service Connector
  - b. "ServiceNow-Table-API-actions" Connection
  - c. "Create an account in ServiceNow" Process
  - d. "Retrieve account data from ServiceNow" Process
4. Go to the "ServiceNow" project
5. Publish the "ServiceNow Table API" Service Connector
6. Edit the "ServiceNow-Table-API-actions" Connection
  - a. Under Properties, enter your organization's ServiceNow credentials for:
    - i. SNOW\_client\_id
    - ii. SNOW\_client\_secret
    - iii. SNOW\_username
    - iv. SNOW\_password
    - v. SNOW\_ServiceUrl
    - vi. SNOW\_grant\_type
  - b. Save the Connection
  - c. Publish the Connection
7. Publish the other components in this order
  - a. "Create an account in ServiceNow" Process
  - b. "Retrieve account data from ServiceNow" Process
8. Test the "Create an account in ServiceNow" 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 "Create an account in ServiceNow" process
    - i. Actions-> Properties Detail...-> Service URL-> Copy
  - c. The request body is of type "application/json"
  - d. Specify your input account variables:

```
{  
  "inputName": "<account name to be created>",  
  "inputCity": "<city of new account>",
```

```
"inputZIP": <ZIPCode>
```

```
}
```

- 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_RefreshToken": "<refresh token>",  
  "o_AccessToken": "<access token>",  
  "a_sys_id": "<unique ServiceNow ID for the newly created account>",  
  "o_city": "<city of the new account>",  
  "o_name": "<name of the new account>"  
}
```

- 9. Test the "Retrieve account data from ServiceNow" process in your REST tool of choice (Postman, RESTed, etc.)

- a. Define a POST operation
- b. Get the Request URL - the Service Address from the published "Retrieve account data from ServiceNow" 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:

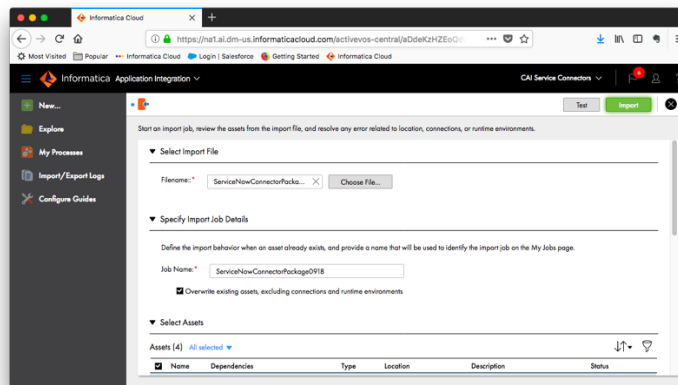
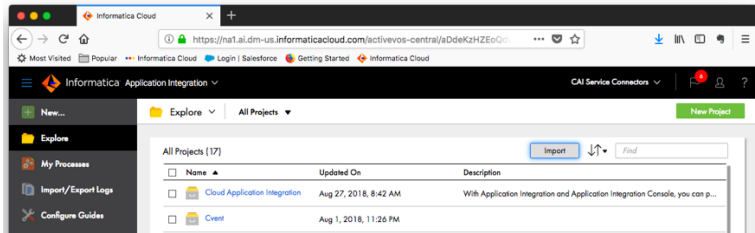
```
{  
  "input_sys_id": "<sys_id of an existing ServiceNow account>"  
}
```

- 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:

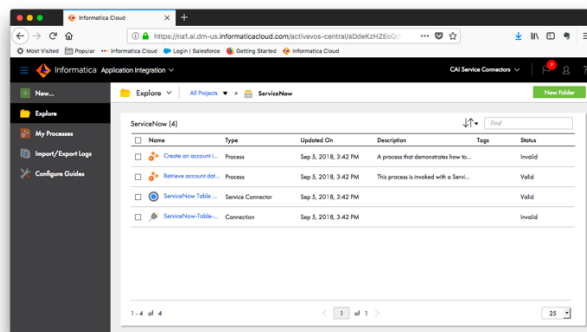
```
{  
  "o_RefreshToken": "<refresh token>",  
  "o_AccessToken": "<access token>",  
  "a_sys_id": "<unique ServiceNow ID for the newly created account>",  
  "o_city": "<city of the new account>",  
  "o_name": "<name of the new account>"  
}
```

## Detailed Setup:

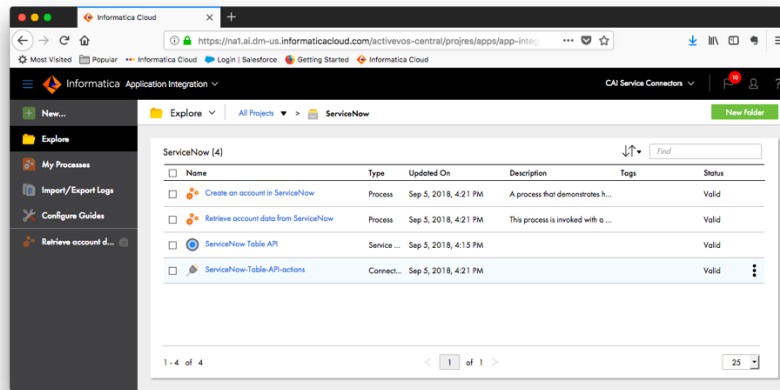
1. Download the [ServiceNowConnectorPackage0918.zip](#) from the [Informatica CAI Git Repository](#)
2. Import the package “ServiceNowConnectorPackage.zip” into your CAI workspace



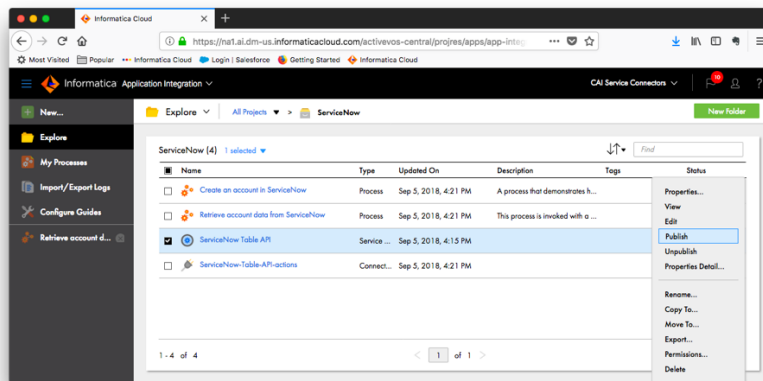
3. This creates a project named “ServiceNow”, with 4 assets in it.
  - a. "ServiceNow Table API" Service Connector
  - b. “ServiceNow-API-actions” Connection
  - c. "Create an account in ServiceNow" Process
  - d. “Retrieve account data from ServiceNow” Process



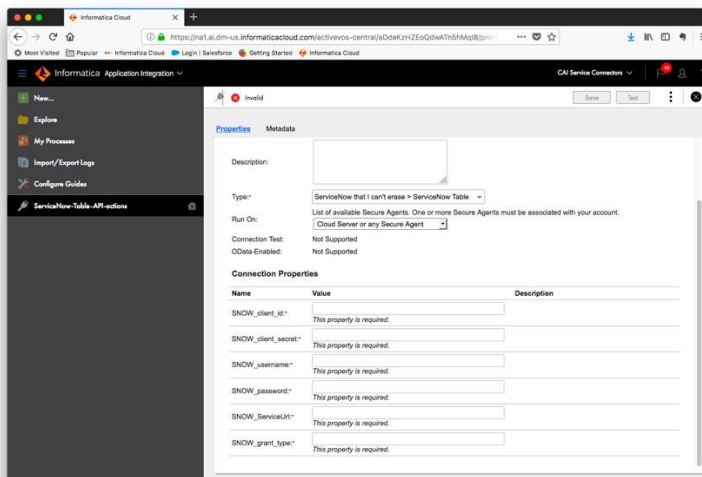
4. Go to the “ServiceNow” project



5. Publish the “ServiceNow Table API” Service Connector



6. Edit the “ServiceNow-Table-API-actions” Connection



- a. Under Properties, enter your organization’s ServiceNow credentials for:
  - i. SNOW\_client\_id
  - ii. SNOW\_client\_secret

- iii. SNOW\_username
  - iv. SNOW\_password
  - v. SNOW\_ServiceUrl
  - vi. SNOW\_grant\_type
- b. Save the Connection

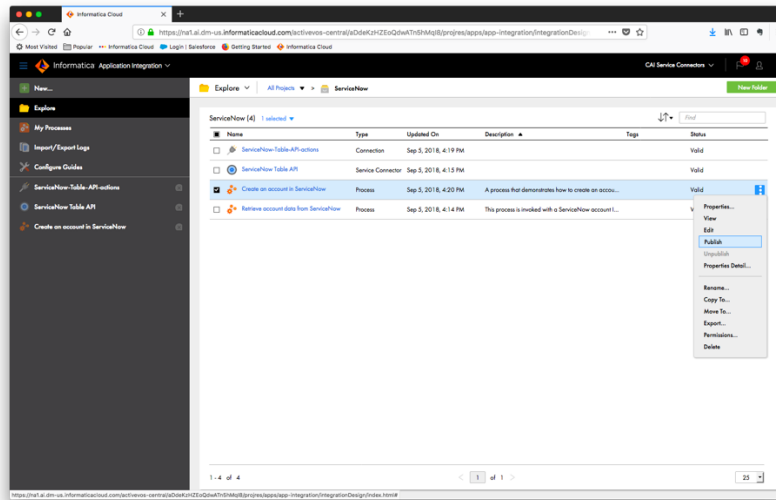
The screenshot shows the Informatica Cloud web interface. On the left is a navigation pane with options like 'New...', 'Explore', 'My Processes', 'Import/Export Logs', 'Configure Guides', and 'ServiceNow-Table-API-actions'. The main area is titled 'Valid' and contains a 'Properties' tab. The 'Location' is set to 'ServiceNow'. The 'Type' is 'ServiceNow that I can't erase > ServiceNow Table'. The 'Run On' is 'Cloud Server or any Secure Agent'. Below this is a 'Connection Properties' table with fields for SNOW\_client\_id, SNOW\_client\_secret, SNOW\_username, SNOW\_password, SNOW\_ServiceUrl, and SNOW\_grant\_type, each with a masked input field.

| Name                | Value    | Description |
|---------------------|----------|-------------|
| SNOW_client_id*     | [Masked] |             |
| SNOW_client_secret* | [Masked] |             |
| SNOW_username*      | [Masked] |             |
| SNOW_password*      | [Masked] |             |
| SNOW_ServiceUrl*    | [Masked] |             |
| SNOW_grant_type*    | [Masked] |             |

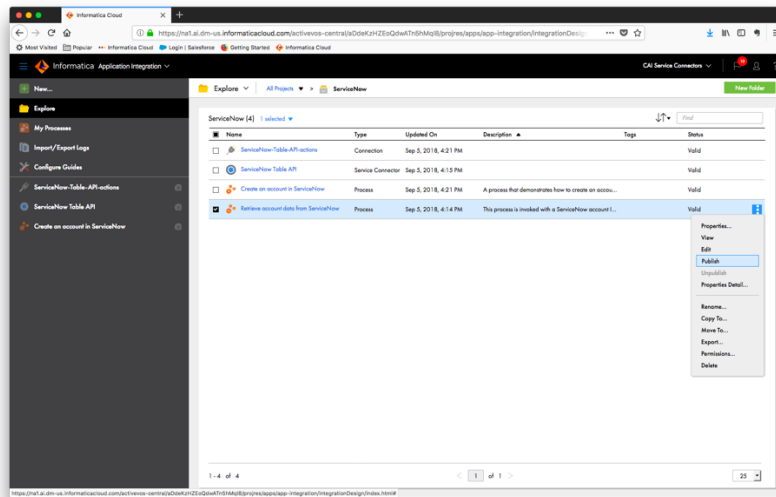
- c. Publish the Connection

This screenshot is identical to the previous one, but with an additional 'Publish' button visible in the top right corner of the main configuration area. The button is blue and labeled 'Publish'. Below it are links for 'Unpublish' and 'Properties Detail...'. The rest of the interface remains the same.

7. Publish the other components in this order
  - a. “Create an account in ServiceNow” Process



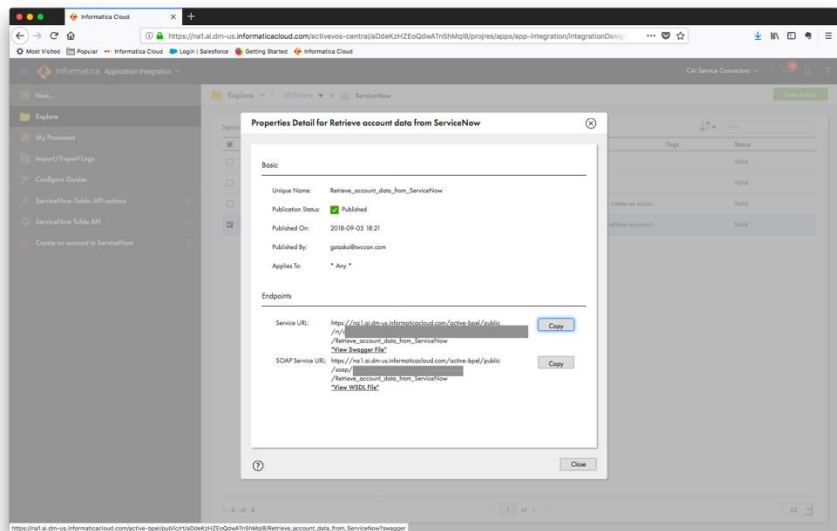
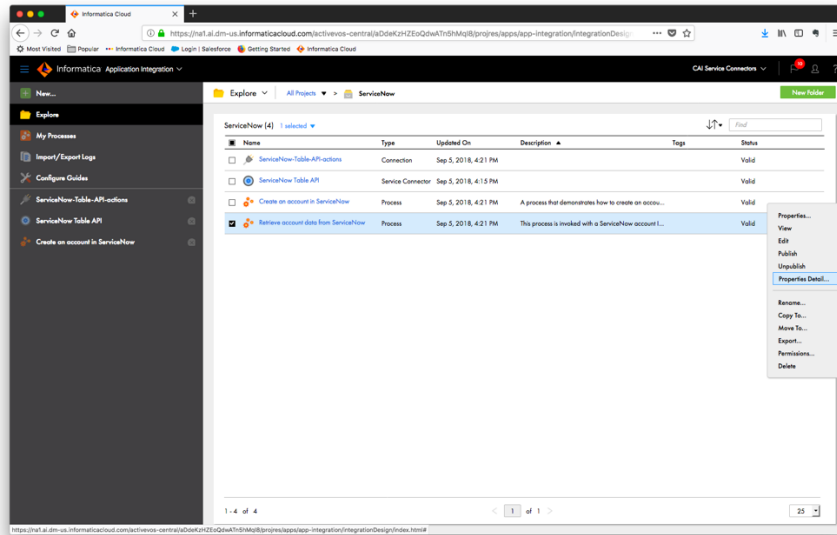
- b. “Retrieve account data from ServiceNow” Process



8. Test the “Create an account in ServiceNow” 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 “Create an account in ServiceNow” process



i. Actions-> Properties Detail...-> Service URL-> Copy



- c. The request body is of type “application/json”
- d. Specify your input account variables:
 

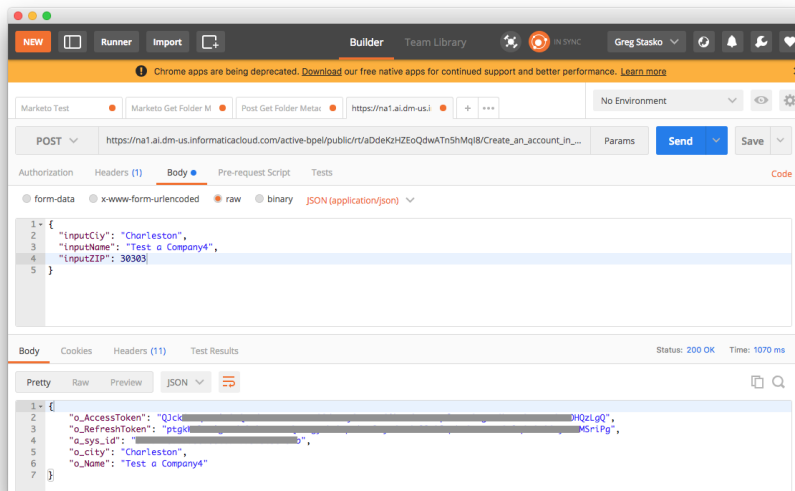
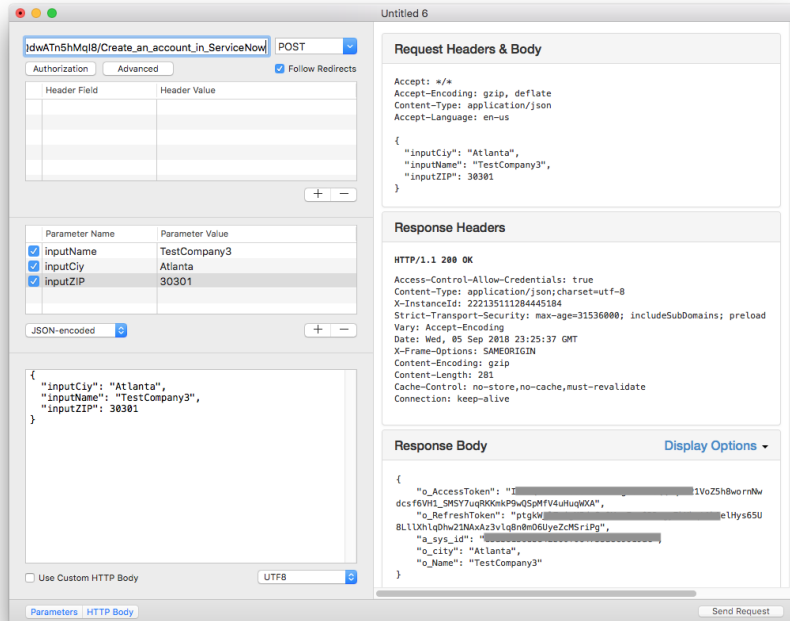
```
{
        "inputName": "<account name to be created>",
        "inputCity": "<city of new account>",
        "inputZIP": "<ZIPCode>"
      }
```
- e. Invoke the operation (“Send”, in the case of Postman, “Send Request” with Rested)
- f. Assuming valid credentials and inputs, you should receive a response that looks similar to this:

```
{
  "o_RefreshToken": "<refresh token>",
```

```

"o_AccessToken": "<access token>",
"a_sys_id": "<unique ServiceNow ID for the newly created account>",
"o_city": "<city of the new account>",
"o_name": "<name of the new account>"
}

```



9. Test the “Retrieve account data from ServiceNow” process in your REST tool of choice (Postman, RESTed, etc.)
  - a. Define a POST operation
  - b. Get the Request URL - the Service Address from the published “Retrieve account data from ServiceNow” 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:

```
{  
  "input_sys_id": "<sys_id of an existing ServiceNow account>"  
}
```
- e. Invoke the operation ("Send", in the case of Postman, "Send Request" with Rested)



