Before You Begin

Did you sign up for a special OmniStudio Developer Edition org already? You'll need one to do the steps in this guide. Here's how to request one if this is your first time completing an OmniStudio module:

1. Sign up for a free Developer Edition org with OmniStudio.
2. Fill out the form.
    a. For Email, enter an active email address.
    b. For Username, enter a username that looks like an email address and is unique, but it doesn't need to be a valid email account (for example, yourname@omnistudiotrails.com).
    c. After you fill out the form, click **Sign me up**. A confirmation message appears.
3. When you receive the activation email (this might take about 10 minutes), open it and click **Verify Account**.
4. Complete your registration by setting your password and security challenge question. **Tip:** Write down your username, password, and login URL for easy access later.

You are logged in to your Developer Edition and you can begin practicing.

Extract External Data with an Integration Procedure

Requirements

"As a developer, I'd like the data in the Weather FlexCard of my Master Account FlexCard to come from an external source."

External data includes all data from outside a Salesforce org. When you need data that is not in Salesforce, you use an external data source. Displaying external data on a FlexCard or an OmniScript is a common need for implementations. Using a weather API is a great way to practice getting data via a web API. There are lots of use cases for weather since severe weather is an important factor for many industries. In this exercise, you explore how to bring weather data into your org.

Prerequisites

- None

Tasks

1. Extract Account Data
2. Gather External URLs
3. Add External Data to an Integration Procedure

Time

- 40 mins

Task 1: Extract Account Data

1. Add a DataRaptor Extract to pull Account Data into the Weather FlexCard.

   When your FlexCard doesn't use a data source in the Setup that pulls Account data into the layout, you need to have a DataRaptor Extract that will pull the Account data your FlexCard will need into the Integration Procedure.

   a. Open the **App Launcher**, and select the **OmniStudio** app.

   b. From the dropdown, select **OmniStudio Integration Procedures**.

   c. Locate and access **team/getWeatherForecast > Team Stub Get Weather and Forecast Data (Version 1)**.

   d. Click **Create Version**.

   e. Edit the **Integration Procedure Name** to remove **Stub**. The new name is **Team Get Weather and Forecast Data**.

   f. Delete the Set Values element **SetValues** which holds the stub data.

   g. From **Actions**, drag a **DataRaptor Extract Action** component into the STRUCTURE panel below Procedure Configuration and above Response Action.

   h. Fill in the fields as follows:

Property	Value
Element Name	`DRGetAccountInfo`
DataRaptor Interface	sampleGetAccountDetailsDRExtract

   i. In the **INPUT PARAMETERS** table, click **+ Add Input Parameter** and fill in the fields as follows:

Property	Value	Notes

Data Source	`AccountId`	You may see suggestions for this value, but you must enter it yourself.
Filter Value	`AccountId`	

j. Click **PREVIEW**.

NOTE:
If you don’t already have it, locate Acme's Record Id as follows:

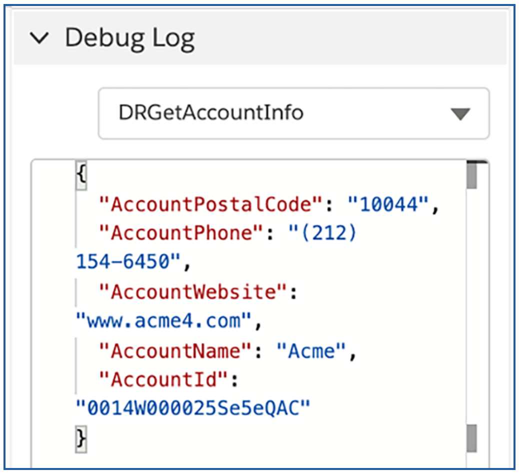
1. Duplicate the browser tab, then use the App Launcher to open the **OmniStudio Console**.
2. Select **Accounts** from the Object dropdown.
3. If needed, switch the view to **All Accounts**.
4. Click **Acme** to view Acme's detail page.
5. Copy the RecordId from the URL (Account Ids always begin with 001, Contact Ids with a 003) and paste it somewhere to use it again.

k. Click **+ Add New Key/Value Pair** and enter the following:

Property	Value
Key	`AccountId`
Value	[Acme’s AccountId]

l. Click **Execute**.

m. Under Debug Log, select **DRGetAccountInfo** from the dropdown and confirm you see Acme's information. The DataRaptor is now connected to the Integration Procedure.



```
{
  "AccountPostalCode": "10044",
  "AccountPhone": "(212)
154-6450",
  "AccountWebsite":
"www.acme4.com",
  "AccountName": "Acme",
  "AccountId":
"0014W000025Se5eQAC"
}
```

Task 2: Gather External URLs

1. Sign up for a weatherbit.io account.

   a. Open a new browser tab and access the following URL: https://www.weatherbit.io/account/create.

   b. Enter a **Username**, **Password**, **Password Verification**, and **Email**.

   c. Accept the **Terms and Conditions** and prove you're not a robot.

   d. Click **Submit**.

   e. Click **Sign in** to sign in to your account.

   f. Enter more details such as your name, the name of your company, and your country.

   g. For **API Tier**, select **Free**.

   h. For **API Usage Purpose**, select **Other**.

   i. Click **Proceed to Dashboard**.

   j. Read the Welcome message and click **Okay!**

   k. Locate the API Key from the API Keys/Admin dashboard. Note that it can take up to 30 minutes to provision the key.

2. Obtain a URL for a static image. (You will use this later when you build a DataRaptor Transform in the next unit).

   a. Open a new browser tab and access the following URL: **https://www.weatherbit.io/api**.

   b. Scroll to the bottom of the page to right-click and open the **Weather API icon / weather condition codes** link under Additional Documentation, Metadata & API's in a new tab.

   c. Select that tab, and then right-click on any icon and select **Open image in a new tab**.

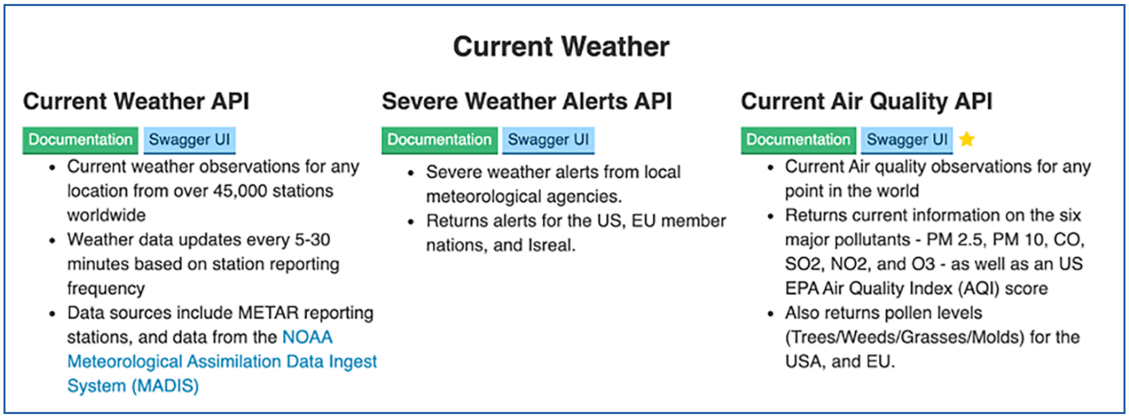
   d. Select that tab and copy the URL of the image and save it to use in the DataRaptor Transform you create in the next unit.

3. Use the weatherbit.io website to build the URL you need to use with your API key.

   a. Return to the Weather API Documentation tab.

   b. Locate the section on the page for **Current Weather API**.

   c. Click **Swagger UI**.



Current Weather

Current Weather API

- Current weather observations for any location from over 45,000 stations worldwide
- Weather data updates every 5-30 minutes based on station reporting frequency
- Data sources include METAR reporting stations, and data from the NOAA Meteorological Assimilation Data Ingest System (MADIS)

Severe Weather Alerts API

- Severe weather alerts from local meteorological agencies.
- Returns alerts for the US, EU member nations, and Isreal.

Current Air Quality API

- Current Air quality observations for any point in the world
- Returns current information on the six major pollutants - PM 2.5, PM 10, CO, SO2, NO2, and O3 - as well as an US EPA Air Quality Index (AQI) score
- Also returns pollen levels (Trees/Weeds/Grasses/Molds) for the USA, and EU.

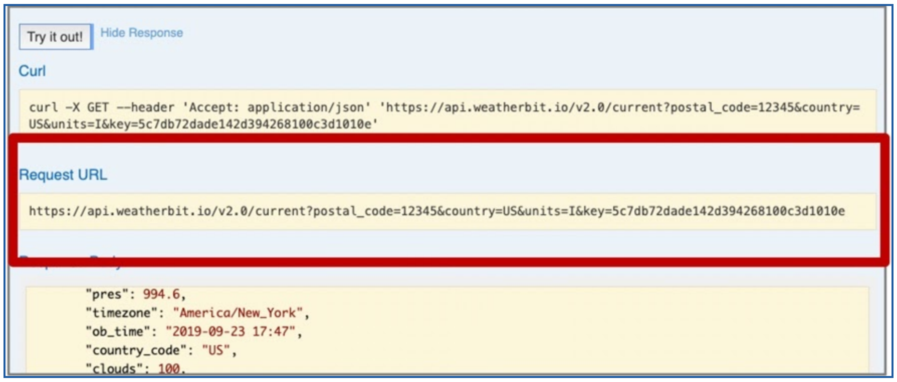
   d. Click the link for **Returns a current observation by postal code**.

   e. Enter the following parameters:

Property	Value
postal_code	`12345`
country	US
units (for metric) or units (for Fahrenheit)	[leave blank] I
key	[Your API key]

   f. Click **Try it out!** (You may have to scroll down to see the button).

g. Copy the **Request URL** and save it for later.



Curl

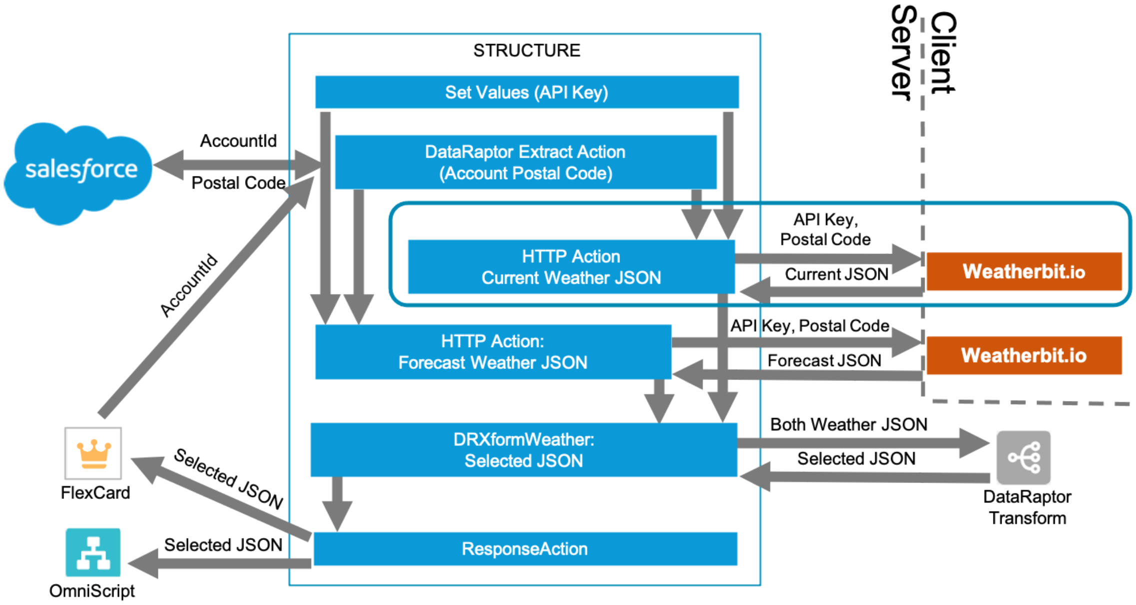
```
curl -X GET --header 'Accept: application/json' 'https://api.weatherbit.io/v2.0/current?postal_code=12345&country=
US&units=I&key=5c7db72dade142d394268100c3d1010e'
```

Request URL

```
https://api.weatherbit.io/v2.0/current?postal_code=12345&country=US&units=I&key=5c7db72dade142d394268100c3d1010e
```

```
        "pres": 994.6,
        "timezone": "America/New_York",
        "ob_time": "2019-09-23 17:47",
        "country_code": "US",
        "clouds": 100,
```

Task 3: Add External Data to an Integration Procedure



1. Transfer the request URL to your Integration Procedure.

   a. Return to the browser tab that has the **Team Get Weather and Forecast Data (Version 2)** Integration Procedure open.

   b. If needed, click **PROPERTIES**.

   c. From **Actions**, drag an **HTTP Action** component to the STRUCTURE panel below **DRGetAccountInfo**.

   d. Enter the following information:

Property	Value	Notes
Element Name	`HTTPGetCurrentWeather`	
HTTP Path	[your Request URL from Task 2 Step 3]	
HTTP Method	`GET`	The value is case sensitive

    e. In the **HTTP Path** field, locate the postal code you used (**12345**) and replace it with the merge code for the postal code of the record: `%DRGetAccountInfo:AccountPostalCode%`.

2. Create a Set Values component to more efficiently use your API key.

   In most cases, the API is worth money, so it is better to place it in a SetValues component where it is more visible and easier to maintain.

    a. From **Actions**, add a **Set Values** component to the STRUCTURE panel, above **HTTPGetCurrentWeather**.

    b. In the **Element Name** field enter `SVWeather`.

    c. Under **Element Value Map**, click **+ Add New Value** and fill in the information as follows:

Element Name	Value
`APIKey`	[Your API key]

    d. Return to the **HTTPGetCurrentWeather** element.

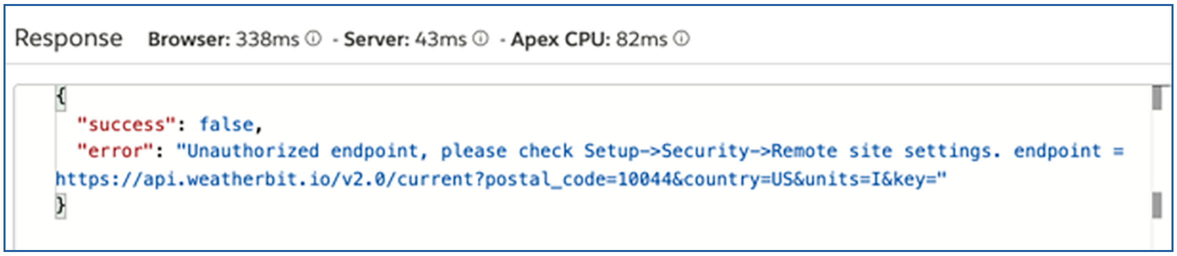
    e. In the **HTTP Path** field, locate the API key in the response URL (**key=API KEY**) and replace it with the merged field for the API key: `%SVWeather:APIKey%`.

TIPS:
You must do a merge code with the full path because you are referencing a field in the SetValues element.

    f. Click **PREVIEW**.

    g. Click **Execute**. (You will receive an error. See the note explaining why below.)

    h. Copy the base URL from the error message: **https://api.weatherbit.io/** to use when allowing the URL in Task 3 Step 3b below.



Response Browser: 338ms ⓘ - Server: 43ms ⓘ - Apex CPU: 82ms ⓘ

```
{
  "success": false,
  "error": "Unauthorized endpoint, please check Setup->Security->Remote site settings. endpoint =
https://api.weatherbit.io/v2.0/current?postal_code=10044&country=US&units=I&key="
}
```

NOTE:
You received the error message because weatherbit.io has not been added to the allowed list in your Salesforce org. You must add the API URL to this list.

3. Allow the weatherbit API's URL in your Salesforce org and test the Integration Procedure a second time.

    a. Access **Setup** > **Remote Site Settings** > **New Remote Site**.

    b. Fill in the form as follows:

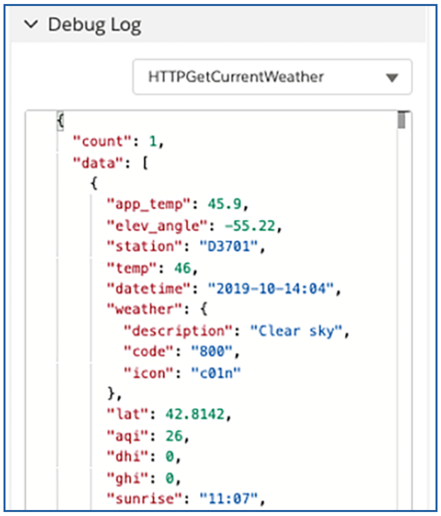
Property	Value
Remote Site Name	`Weatherbit`
Remote Site URL	`https://api.weatherbit.io`

    c. Click **Save**.

    d. Close the Setup browser tab and return to your **Integration Procedure** browser tab.

    e. Click **Execute**.

    f. From the **Debug Log** dropdown, select **HTTPGetCurrentWeather** and confirm the weather is returned.



```
{
  "count": 1,
  "data": [
    {
      "app_temp": 45.9,
      "elev_angle": -55.22,
      "station": "D3701",
      "temp": 46,
      "datetime": "2019-10-14:04",
      "weather": {
        "description": "Clear sky",
        "code": "800",
        "icon": "c01n"
      },
      "lat": 42.8142,
      "aqi": 26,
      "dhi": 0,
      "ghi": 0,
      "sunrise": "11:07",
```

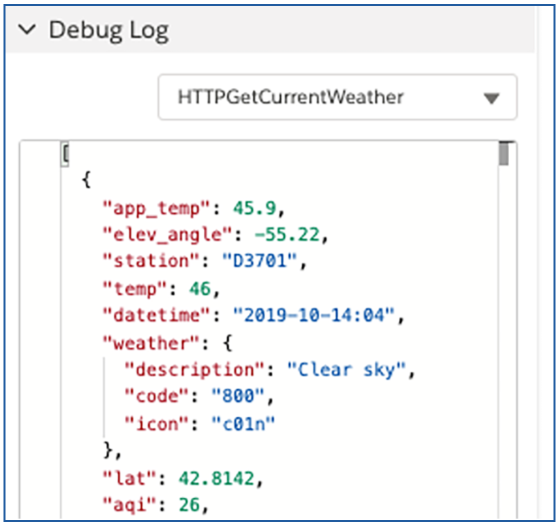
4. Focus the Response JSON to only return the weather data.

    a. Click **PROPERTIES** and select the **HTTPGetCurrentWeather** element.

    b. Expand **SEND/RESPONSE TRANSFORMATIONS**.

    c. In the **Response JSON Path**, enter `data`. This trims the data node off the JSON.

    d. Click **PREVIEW > Execute > Debug Log > HTTPGetCurrentWeather** and confirm the only data visible is the data node.



```
[
  {
    "app_temp": 45.9,
    "elev_angle": -55.22,
    "station": "D3701",
    "temp": 46,
    "datetime": "2019-10-14:04",
    "weather": {
      "description": "Clear sky",
      "code": "800",
      "icon": "c01n"
    },
    "lat": 42.8142,
    "aqi": 26,
```

Review

Confirm your understanding by answering these questions.

1. What are ways that you could use weather data in your own project?
2. How do you build your API Return URL?
3. What components or steps do you use in your Integration Procedure to incorporate the external data?

Edit the Integration Procedure to Include Forecast Weather Data

To solidify what you learned, update the Integration Procedure to bring in Forecast weather data for 5 days in the future. We've summarized the requirements; refer to the step-by-step instructions above if you get stuck.

1. Generate the Return URL for Forecast Data. Follow the same process as in Task 2 above to generate the Return URL for current weather conditions to generate 5 days of forecast weather data for the same test postal code.
   - Use the Forecast API (16 day / daily) for a Postal Code. *(Hint: don't forget your units setting).*
2. Add Forecast Data to the Integration Procedure. Follow the same process as in Task 3 above to add a new HTTP Action element that pulls in the forecast data that you just generated a Return URL for.
   - You will need a second input; call this input `days` and use the merge code of `%Days%`. This is what the OmniScript and FlexCard send to this Integration Procedure. *(Hint: don't forget your Response JSON Path)*