

Quick2Cloud for Cloud Systems

Built on IBM Cloud



Quick2Cloud College - Course 101

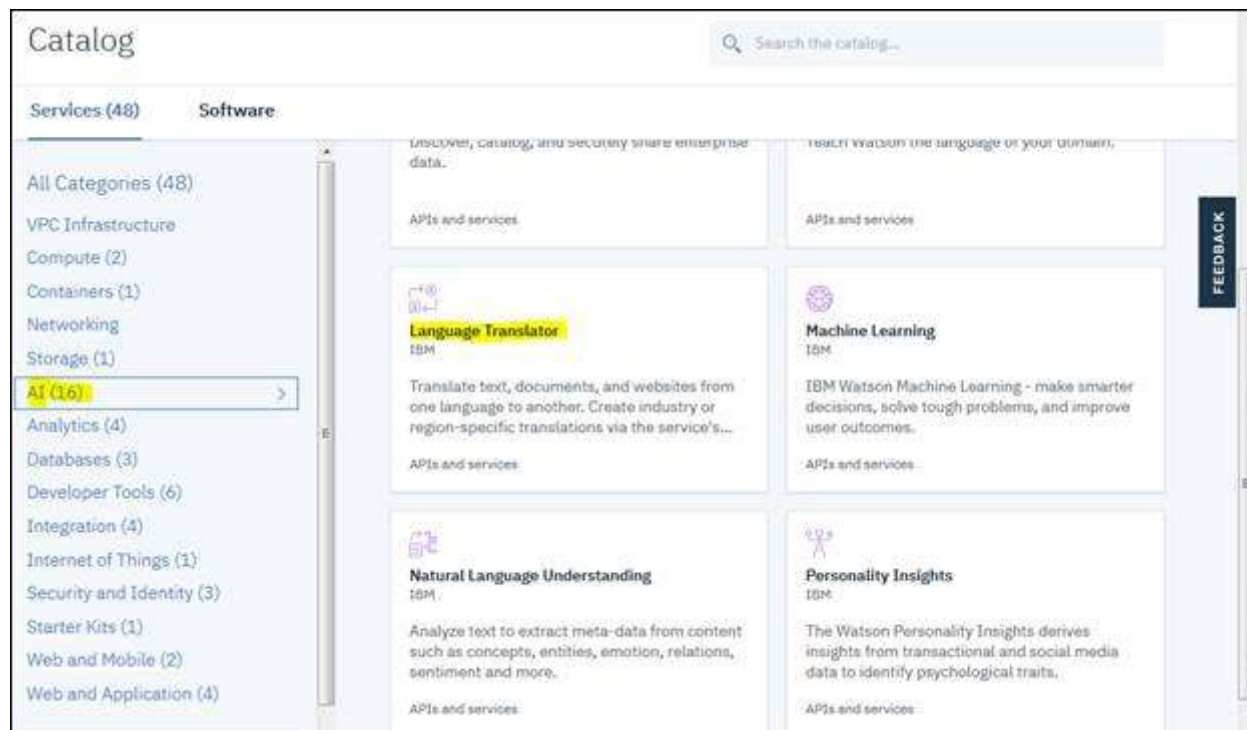
Provisioning Services for your IBM Cloud Application

Soon you will see how the VCAP_SERVICES Cloud Environment Variable is accessed out of your *Quick2Cloud* sample web application allowing it to connect and begin using this newly provisioned Service.

But, before we get involved with VCAP concepts you will provision your second style of IBM Service so you get a feeling of what it takes to connect to a slightly more transactional Service. This Service will be the IBM Watson Language Translator which will be set to translate English to Spanish.

Step 4

Again, just as you did for the Cloudant Service you will locate the Watson Language Translation Service from the IBM Service Catalog. Click on the AI and Language Translator links in yellow below:



It is important that you remember the **Service Name** because you will want to reference this name later when your sample application wants to find the credentials from the passed VCAP environment variable. You can use the default name if you like or pick one more easily remembered. Of course, if you forget the name you can always come back to the Services display and find it.

IBM Cloud

My Resources Catalog Docs Support Manage Jeff Hannan's Account

Language Translator

Life IBM Service

Need Help? [Contact Support](#) [View docs](#) [API docs](#)

Author: IBM Watson • Date of last update: 10/30/2019

[Create](#) [About](#)

Select a region

Dallas

Select a pricing plan Monthly prices shown are for country or region: [United States](#)

PLAN	FEATURES	PRICING
✓ Lite	1,000,000 Characters per Month	Free
Standard	Standard Translations (First 250,000 characters are free)	\$0.02 USD/THOUSAND CHAR
Advanced	Standard Translations Custom Translations Custom Model Maintenance (Pro-Rated Daily)	\$0.02 USD/THOUSAND CHAR \$0.10 USD/THOUSAND CHAR \$15.00 USD/INSTANCE MONTH

[Create](#) [Add to estimate](#) [View terms](#)

FEEDBACK

Step 5

Click on [Create](#) link and notice this Service provisions much faster than the Cloudant Service. That is because this Watson Service is already up and running at a Watson Cloud Center and all that was needed was for a URL and API Key as a credential. In the case of this Service, you don't have to manually generate a credential - it has already been provided. You can see the credential by clicking on the [Service Credentials](#) link on the far left menu. Remember to click on the [View Credentials](#) link to unhide it.

Resource list /

Language Translator-w3 0% Used | 999974 Char available

Resource group: Default Location: Dallas [Add Tags](#)

Service credentials

Credentials are provided in JSON format. The JSON snippet lists credentials, such as the API key and secret, as well as connection information for the service. [Learn more](#)

Service credentials [New credential](#)

Items per page 10 | 1-2 of 2 items 1 of 1 pages

<input type="checkbox"/> KEY NAME	DATE CREATED	ACTIONS
<input type="checkbox"/> Auto-generated service credentials	NOV 7, 2019 - 03:13:15 PM	View credentials

```
{
  "apikey": "w19c...m16jeXfUw38AbhNnJ1vz4Cbc8iLKdQAZip",
  "iam_apikey_description": "Auto-generated for key 687e67a8-dc4c-4372-a88e-e36557399c5c",
  "iam_apikey_name": "Auto-generated service credentials",
  "iam_role_crn": "crn:v1:bluemix:public:iam:::serviceRole:Manager",
  "iam_serviceid_crn": "crn:v1:bluemix:public:iam-identity::a/a48df2bdcbcd48b58af1b491627840a::serviceid:ServiceId-b4450d78-5c26-42d2-96fe-cc4ad8852ea7",
  "url": "https://gateway.watsonplatform.net/language-translator/api"
}
```

Step 6

Since this Service is just a REST API secured with an API key you can test it out by issuing a REST transaction and see what comes back as translation. The best way to accomplish this is to use the [CURL](#) facility that is part of the BASH Command Line Interface. Typed below is a simple CURL formatted command that will call the Watson Language Translator with a few salutations. This will translate English to Spanish.

```
curl -X POST -u "apikey:GNIKPoXGKqNVlh1Q6PvWT2ZCgSr62Q384Ney1E3mSkNJ" --header "Content-Type: application/json" --data '{"text":["Hello, world! ", "How are you?"], "model_id":"en-es"}' "https://gateway.watsonplatform.net/language-translator/api/v3/translate?version=2018-05-01"
```

Replace the API key in green above with the API key in your Watson Language credential.

Cut and Paste this substituted command string into the BASH console as shown below:

```

MINGW64:/c/Users/Paul
Paul@Paul-PC MINGW64 ~ (master)
$ curl -X POST -u "apikey:GNIKPoXGKqNVlh1Q6AvWT2ZCgSr62V384Ney1E3mSkNJ" --header
"Content-Type: application/json" --data '{"text": ["Hello, world! ", "How
are you?"], "model_id": "en-es"}' "https://gateway.watsonplatform.net/lanqua
ge-translator/api/v3/translate?version=2018-05-01"
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100  226  100  162  100    64    100     39  0:00:01  0:00:01 --:--:-- 140{
  "translations" : [ {
    "translation" : "¡Hola, mundo! "
  }, {
    "translation" : "¿Cómo estás?"
  } ],
  "word_count" : 5,
  "character_count" : 26
}

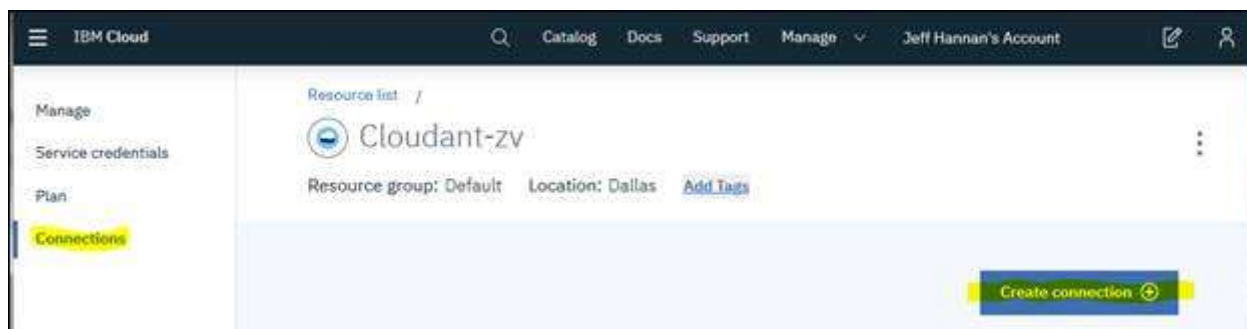
Paul@Paul-PC MINGW64 ~ (master)
$ |

```

As you can see, this Service does not require a set of Service API calls that get called out of code like will be required in Cloudant (described a bit later in your next college course). Any application can make use of this service just by issuing a REST API with the API key attached to make the request secure.

Step 7

Let us now bind this Watson Service to our application via the [Connections](#) link on the far left menu. Binding this will give your sample application knowledge of the URL and API key such that the application can make REST calls programmatically. Again, just like with Cloudant, those bits of credential json are passed inside the VCAP variable. With the addition of the Watson Service, that VCAP variable now has two Services defined in it and you will use a facility that will parse each Service entry to get their information.

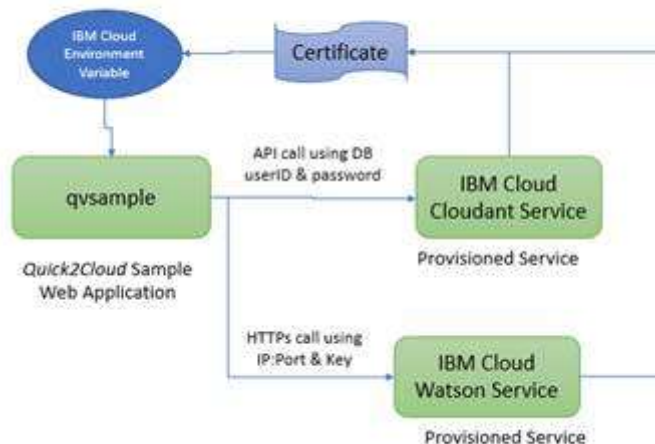


This will bring you to the following display:



Click on the application name you want to connect this Service too. When this is complete, restart your application from the [IBM Dashboard](#) for this to all be recognized.

Provisioning and using different IBM Services



Fantastic! You now have two IBM Cloud Services running that are available to your sample application. They both should be visible on the IBM Cloud Dashboard. In the next section you will see how to access the Cloud environment variables (VCAP_APPLICATION and VCAP_SERVICES) using the CFENV facility. These variables will contain information that will allow your sample application to programmatically interface with the two Services.

[Continue](#)

Quick2Cloud Consulting - Moorpark, CA. 93021 - Developed in Node.js - Stage & Production in the IBM Cloud

[Feedback](#)