

blueprism[®]

Blue Prism 6.2

IBM Watson Intelligent Services User Guide

Document Revision: 1.0



Trademarks and Copyright

The information contained in this document is the proprietary and confidential information of Blue Prism Limited and should not be disclosed to a third-party without the written consent of an authorized Blue Prism representative. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying without the written permission of Blue Prism Limited.

© **Blue Prism Limited, 2001 – 2021**

© “Blue Prism”, the “Blue Prism” logo and Prism device are either trademarks or registered trademarks of Blue Prism Limited and its affiliates. All Rights Reserved.

All trademarks are hereby acknowledged and are used to the benefit of their respective owners. Blue Prism is not responsible for the content of external websites referenced by this document.

Blue Prism Limited, 2 Cinnamon Park, Crab Lane, Warrington, WA2 0XP, United Kingdom.
Registered in England: Reg. No. 4260035. Tel: +44 370 879 3000. Web: www.blueprism.com

Contents

Introduction	4
Solution Overview	5
Limitations	5
Pre-Requisites and Environment Configuration	6
Cloud Services Prerequisites	6
Blue Prism Configuration	6
Natural Language Understanding VBO	19
Configure	19
Get Categories	20
Get Concepts	21
Get Emotion	22
Get Entities	23
Get Keywords	24
Get Metadata	25
Get Relations	26
Get Semantic Roles	27
Get Sentiment	28
List Models	29
Delete Model	29
Language Translator VBO	30
Configure	30
Translate	30
Get Identifiable Languages	30
Identify Language	31
List Models	32
Get Model Details	33
Delete Model	34
Visual Recognition VBO	35
Configure	35
Classify Image	35
Detect Faces	36

Introduction


Robotic Process Automation (RPA) is an ever-expanding market which allows for back end office automation via intelligent robots. With the addition of a Blue Prism® Artificial Intelligence offering, the list of use cases for RPA will increase dramatically.

This document focuses on the design of the integration between IBM's Watson Services implemented through REST calls which covers following APIs:

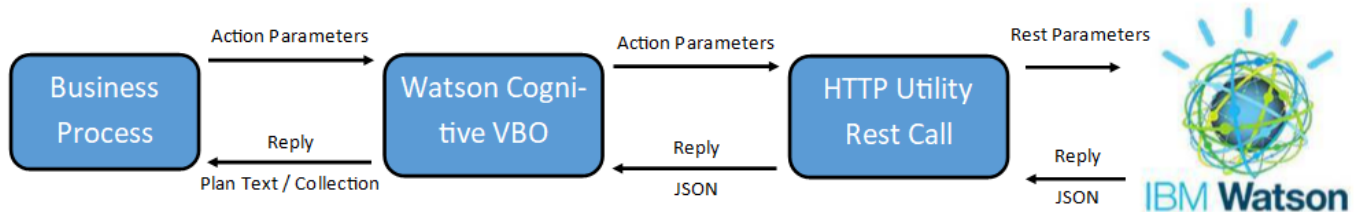
- Text Analytics (Natural Language Understanding)
- Visual Recognition
- Language Translation

Solution Overview

The basic design is to provide one or more VBOs that encapsulate the different IBM Watson services. These VBOs can then be offered as an easy bridge to connect a client's processes to the different services provided by IBM Watson.

 The provided VBOs are only supported with Blue Prism 6.2 and above.

All of IBM Watson Services require a username and password which is provided when you define a new IBM Watson Service (i.e. Natural Language Understanding) in the configuration stage.



Limitations

The following limitations should be understood before attempting to use these integrations:

- It is the customer / partner responsibility to configure and maintain the relevant cloud subscriptions and services. Blue Prism can not provide any support on the configuration of the cloud environment itself.
- Use of the APIs may incur additional costs, depending on usage
- There is always a possibility with external services that the APIs will change. Blue Prism will attempt to maintain compatibility with the latest versions, however no formal SLAs are provided around this.

Pre-Requisites and Environment Configuration

This section outlines the pre-requisites that are required in order to use the integrations. Note that Blue Prism is not able to provide any support in configuring the Cloud Services themselves.

Cloud Services Prerequisites

The following is required, before configuring the Blue Prism environment and attempting to use the VBOs

- An IBM Cloud account
- Create an IBM Watson service
- Obtain service username and password or API key for the Service

All of IBM Watson Services require a username and password which is provided when you define a new IBM Watson Service (i.e. Natural Language Understanding) in the configuration stage.


- IBM Watson Intelligent Services VBOs (packaged with this document on the Blue Prism portal)

Blue Prism Configuration

First, import the bprelease file. Before using the VBOs, it is necessary that the following environment variables be defined, as well as their appropriate values set. These values link up to Credentials in BluePrism which hold the subscription keys.

Environmental Variables

VBO	Variable Name	Description
Natural Language Processing	IBM Watson Natural Language Processing	Links to the Credential Provider where the username, password or API Key is kept in a secure manner.
Translation	IBM Watson Translation	Links to the Credential Provider where the username, password or API Key is kept in a secure manner.
Visual Recognition	IBM Watson Visual Recognition	Links to the Credential Provider where the username, password or API Key is kept in a secure manner.

 All are of the type Text.

Credentials

The credential manager holds the API keys / Username and Password in relation to each IBM Watson API. Each credential name will be the same as the environmental name. This section shall explain how to add a new credential for the Computer Vision API. If you require a different API, follow the exact same steps but with the according name.

Creating a Credential for the Visual Recognition VBO

This example shows how to set up the IBM Watson Visual Recognition API.

1. In Blue Prism, click the **System** tab and select **Security > Credentials** from the navigation tree.
2. Click **New** from the Credentials menu. The Credential Details dialog displays.

Credential Details

Name:

Description:

Application Credentials | **Access Rights**

Username Expires

Password Marked as invalid

Additional Properties

Name	Value

3. Enter a name for the credential - in this example, *IBM Watson Visual Recognition*. This exact name will be used for the Environmental Variable.
4. Leave the Username and Password fields blank.
5. In the Additional Properties, enter *API_key* in the Name column and set a value the same as the subscription key generated in the IBM Watson Cloud panel.

Credential Details

Name: IBM Watson Visual Recognition

Description:

Application Credentials | Access Rights

Username:

Password:

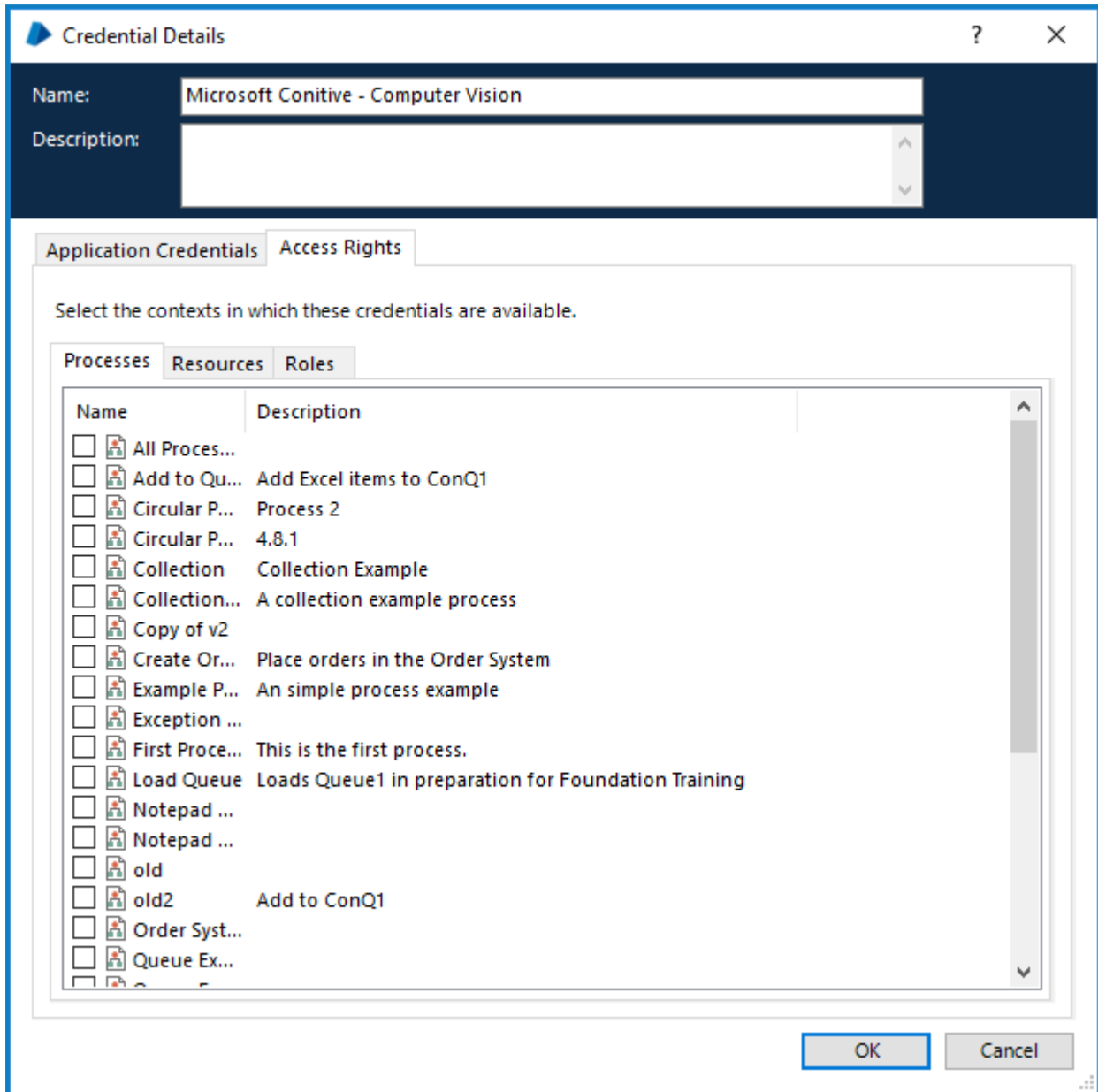
Expires:

Marked as invalid

Additional Properties

	Name	Value
	API Key

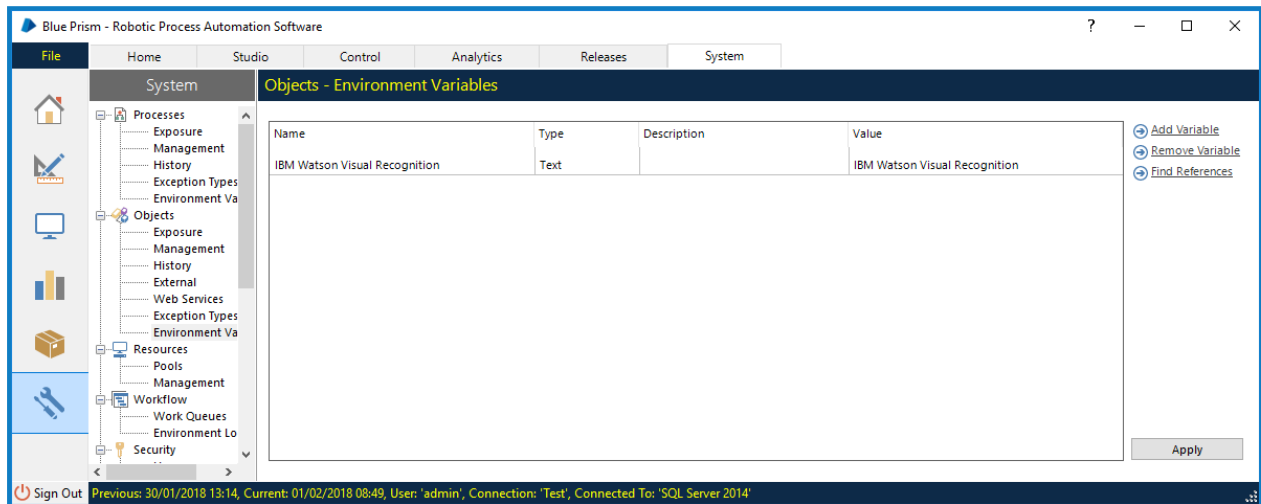
6. Select the **Access Rights** tab. Access rights must be configured before these new credentials can be used in a process to determine which processes, resources, and roles can use these credentials.



7. Apply the required access rights in the **Processes**, **Resources**, and **Roles** tabs.
8. Click **OK**.

9. Select **Objects > Environment Variables** from the Navigation tree.
10. Click **Add Variable** to create a new line in the table.
11. Give the variable the same name as the credential - *IBM Watson Visual Recognition*.

By linking the environment variable to the credential in this way, the IBM Watson Visual Recognition VBO can access the API Key securely.



The IBM Watson Visual Recognition VBO has now been correctly configured.

Creating a Credential for the Natural Language Processing

This example shows how to set up the IBM Natural Language Processing API.

1. In Blue Prism, click the **System** tab and select **Security > Credentials** from the navigation tree.
2. Click **New** from the Credentials menu. The Credential Details dialog displays.

The screenshot shows the 'Credential Details' dialog box. It has a title bar with a question mark and a close button. The dialog is divided into several sections:

- Name:** A text input field.
- Description:** A text area with a scroll bar.
- Application Credentials:** A tabbed section with two tabs: 'Application Credentials' (selected) and 'Access Rights'.
- Username:** A text input field.
- Password:** A text input field.
- Expires:** A date picker field.
- Marked as invalid:** A checkbox.
- Additional Properties:** A table with two columns: 'Name' and 'Value'. The table is currently empty.

3. Enter a name for the credential - in this example, *IBM Watson Natural Language Processing*. This exact name will be used for the Environmental Variable.
4. In the Username and Password fields, enter the credentials gathered from your IBM Watson Portal.

Credential Details

Name: IBM Watson Natural Language Processing

Description:

Application Credentials | Access Rights

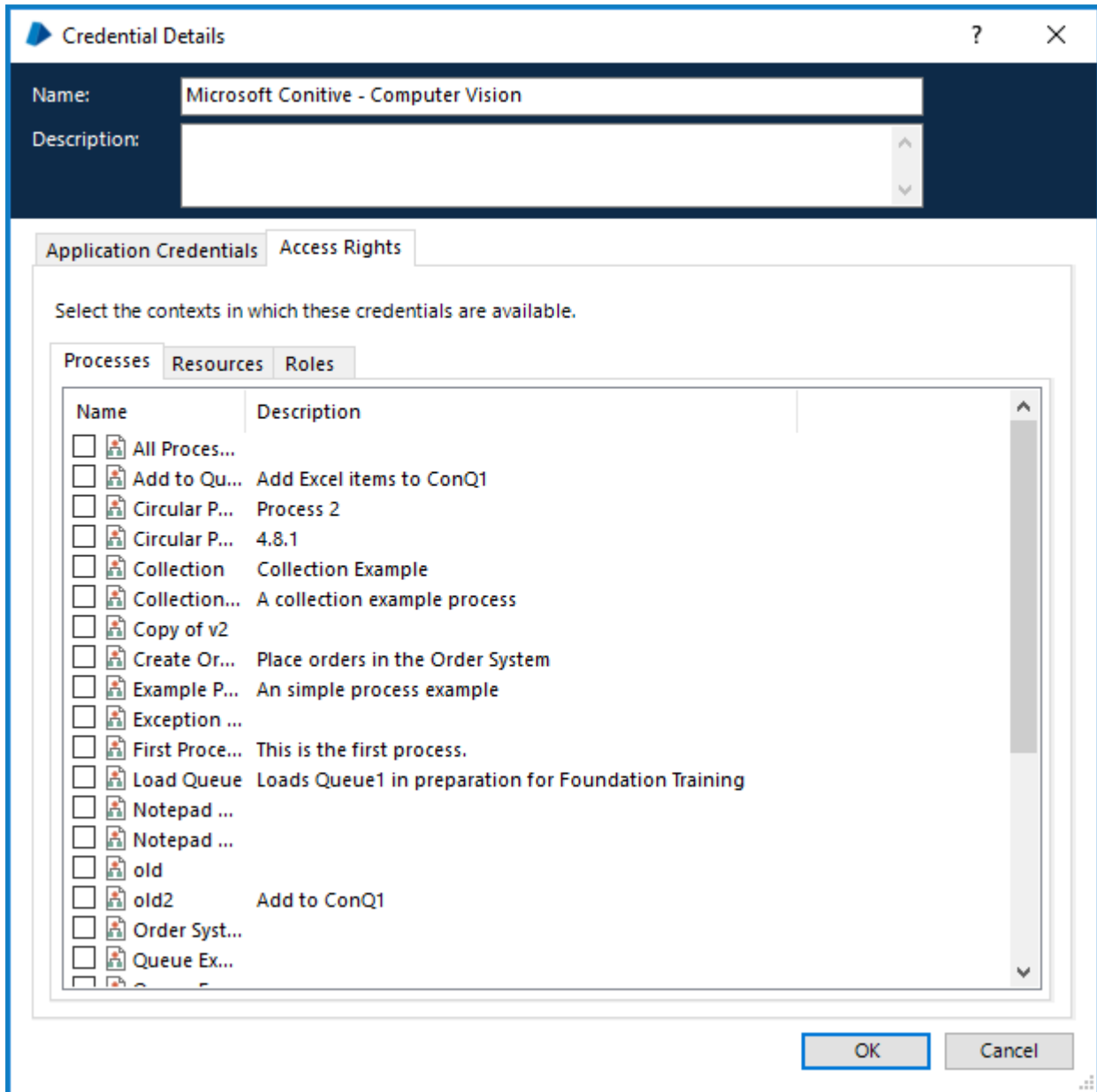
Username: bf16768d-fd40-4ce0-beee-daa55cbc1474 Expires:

Password: ●●●●●●●● Marked as invalid

Additional Properties

Name	Value

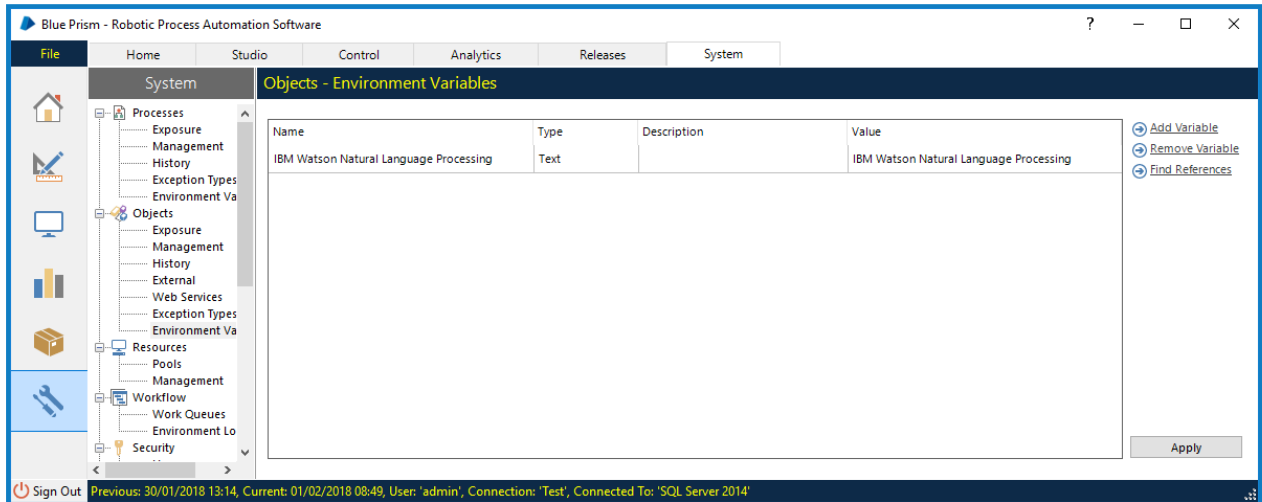
5. Select the **Access Rights** tab. Access rights must be configured before these new credentials can be used in a process to determine which processes, resources, and roles can use these credentials.



6. Apply the required access rights in the **Processes**, **Resources**, and **Roles** tabs.
7. Click **OK**.

8. Select **Objects > Environment Variables** from the Navigation tree.
9. Click **Add Variable** to create a new line in the table.
10. Give the variable the same name as the credential - *IBM Watson Natural Language Processing*.

By linking the environment variable to the credential in this way, the IBM Watson Visual Recognition VBO can access the username and password securely.



The IBM Watson Visual Recognition VBO has now been configured.

Creating a Credential for the Translation VBO

This example will be for setting up the IBM Watson Translation API.

1. In Blue Prism, click the **System** tab and select **Security > Credentials** from the navigation tree.
2. Click **New** from the Credentials menu. The Credential Details dialog displays.

Credential Details ? X

Name:

Description:

Application Credentials | **Access Rights**

Username Expires

Password Marked as invalid

Additional Properties

Name	Value

3. Enter a name for the credential - in this example, *IBM Watson Translation*. This exact name will be used for the Environmental Variable.
4. In the Username and Password fields, enter the credentials gathered from your IBM Watson Portal.

Credential Details

Name: IBM Watson Translation

Description:

Application Credentials | **Access Rights**

Username: a37bdd4d-49e8-401a-8cd4-0ac55f32ff41

Expires:

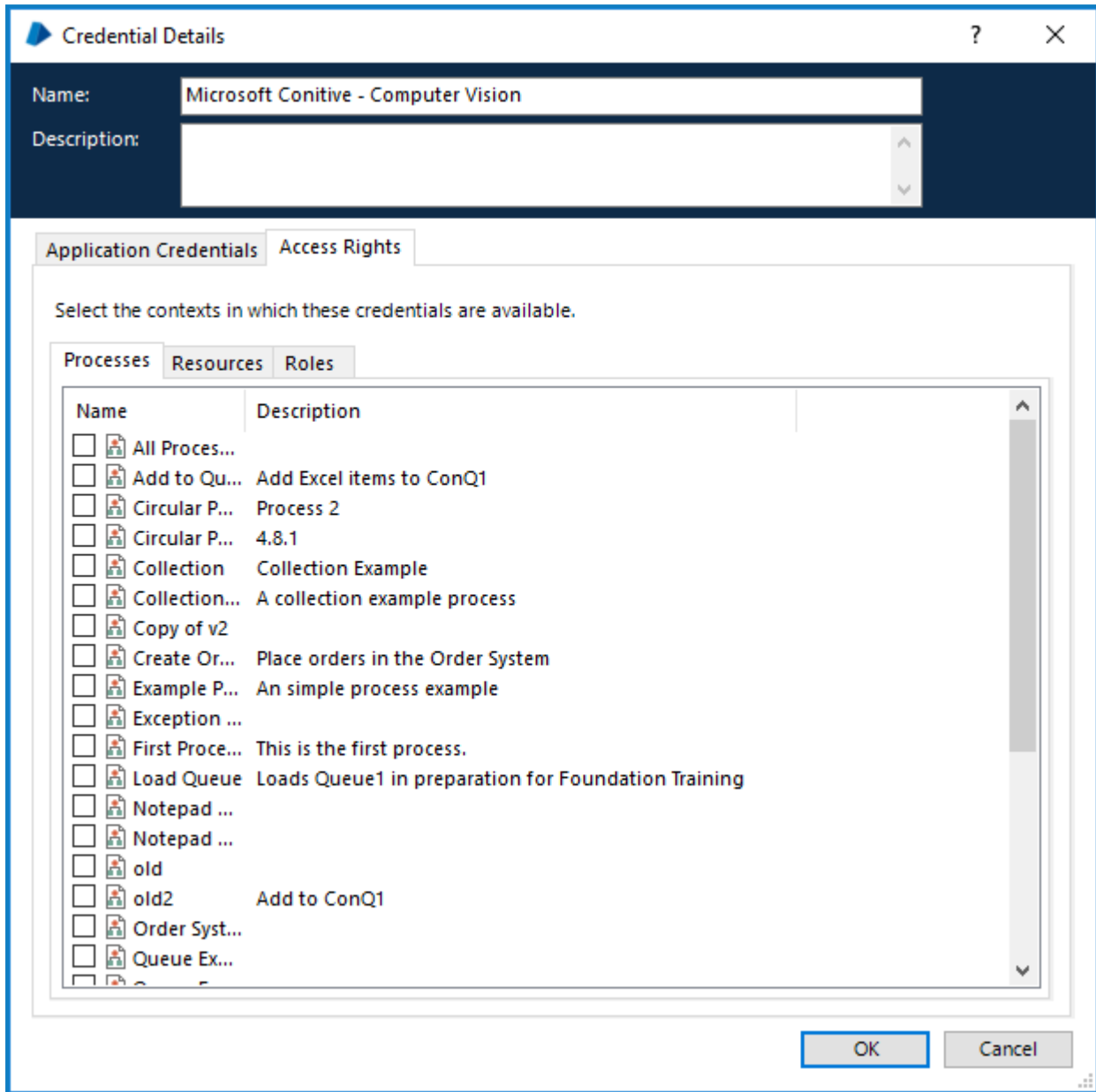
Password: ●●●●●●●●

Marked as invalid

Additional Properties

Name	Value

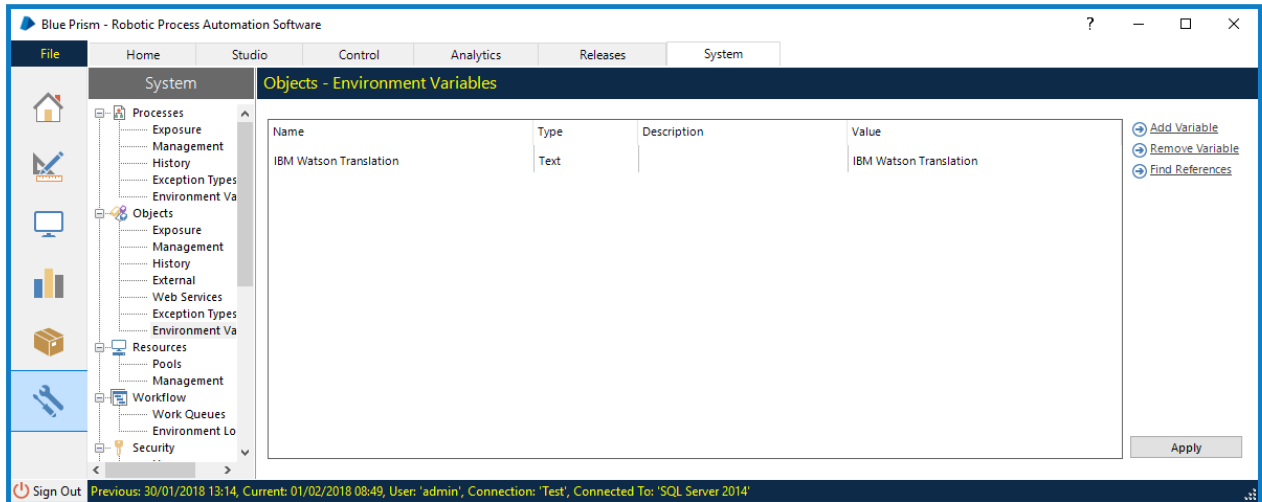
5. Select the **Access Rights** tab. Access rights must be configured before these new credentials can be used in a process to determine which processes, resources, and roles can use these credentials.



6. Apply the required access rights in the **Processes**, **Resources**, and **Roles** tabs.
7. Click **OK**.

8. Select **Objects > Environment Variables** from the Navigation tree.
9. Click **Add Variable** to create a new line in the table.
10. Give the variable the same name as the credential - *IBM Watson Translation*.

By linking the environment variable to the credential in this way, the IBM Watson Visual Recognition VBO can access the username and password securely.



The IBM Watson Translation VBO has now been correctly configured.

Natural Language Understanding VBO

This Business object contains several actions covering text analytics of the IBM Watson NLU Services:

- [Configure below](#)
- [Get Categories on the next page](#)
- [Get Concepts on page 21](#)
- [Get Emotion on page 22](#)
- [Get Entities on page 23](#)
- [Get Keywords on page 24](#)
- [Get Metadata on page 25](#)
- [Get Relations on page 26](#)
- [Get Semantic Roles on page 27](#)
- [Get Sentiment on page 28](#)

The Business object contains two additional actions covering management of deployed custom models created in Watson Knowledge Studio:

- [List Models on page 29](#)
- [Delete Model on page 29](#)

Configure

This action must be called before any additional VBO action as it sets up all required parameters for Natural Language Understanding Service.

Parameter	Direction	Data Type	Description
Watson URL	In	Text	Specify Watson URL (i.e. https://gateway.watsonplatform.net/natural-language-understanding/api/v1/)
Method	In	Text	Specify method (at this moment only "Analyse" is supported)
Version	In	Text	Specify version (i.e. "2017-02-27")
Username	In	Text	Provide username under which you are going to access the Watson Service
Password	In	Password	Provide password under which you are going to access the Watson Service
Data Collection	In	Flag	Set whether you prefer to let IBM collect usage statistic data. By default it is turned off (false).

Get Categories

Supported languages: Arabic, English, French, Italian, Korean, Portuguese, Spanish.

The request response is formatted into the following easy to use output:

- **Categories** - This collection contains two fields:
 - **Score** - A ranking of labels with confidence score ranging from 0 to 1.0; 0 means it is not confident in the categorization and a 1 means it is highly confident.
 - **Label** - Category label based on a 5-level taxonomy. Complete list of categories can be found here <https://console.bluemix.net/docs/services/natural-language-understanding/categories.html#categories-hierarchy>. The top three categories will be returned.
- **Language** – The language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed

Output

Categories (Collection)	Language (Text)
Field 1: Score	Language code
Field 2: Label	

Get Concepts

Get Concepts action recognizes high-level concepts that are related to your text.

Supported languages: English, Korean, Spanish

The request response is formatted into the following easy to use output:

- **Concepts** - This collection contains three fields:
 - **Text** - Name of the concept
 - **Relevance** - Relevance score for the concept ranging from 0 to 1. A 0 means it's not relevant, and a 1 means it's highly relevant
 - **Dbpedia_Resource** - Link to the concept's associated DBpedia resource.
- **Language** - The language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Concepts Limits	In	Number	The maximum number of concepts to return, default is 8, maximum is 50

Output

Concepts (Collection)	Language (Text)
Field 1: Text	Language code
Field 2: Relevance	
Field 3: Dbpedia_Resource	

Get Emotion

Get Emotion action detects emotion conveyed by the entire body of text, or by the context around target phrases specified in the targets parameter.

Supported languages: English

The request response is formatted into the following easy to use outputs:

- **Emotion** - This collection contains six fields:
 - **Target** - Target object for which the emotion is Analysed (E.G. a word or sentence)
 - **Sadness** - Scores ranging from 0 to 1 for sadness emotion. A 0 means the text does not convey any sadness, whereas a 1 means the text carries sadness.
 - **Joy** - Scores ranging from 0 to 1 for joy emotion. A 0 means the text does not convey any joy, whereas a 1 means the text carries joy.
 - **fear** - Scores ranging from 0 to 1 for fear emotion. A 0 means the text does not convey any fear, whereas a 1 means the text carries fear.
 - **Disgust** - Scores ranging from 0 to 1 for disgust emotion. A 0 means the text does not convey any disgust, whereas a 1 means the text carries disgust.
 - **Anger** - Scores ranging from 0 to 1 for anger emotion. A 0 means the text does not convey any anger, whereas a 1 means the text carries anger.
- **Document** - Collection containing emotion analysis results for the entire document and contains five fields:
 - **Sadness** - Scores ranging from 0 to 1 for sadness emotion. A 0 means the text does not convey any sadness, whereas a 1 means the text carries sadness.
 - **Joy** - Scores ranging from 0 to 1 for joy emotion. A 0 means the text does not convey any joy, whereas a 1 means the text carries joy.
 - **Fear** - Scores ranging from 0 to 1 for fear emotion. A 0 means the text does not convey any fear, whereas a 1 means the text carries fear.
 - **Disgust** - Scores ranging from 0 to 1 for disgust emotion. A 0 means the text does not convey any disgust, whereas a 1 means the text carries disgust.
 - **Anger** - Scores ranging from 0 to 1 for anger emotion. A 0 means the text does not convey any anger, whereas a 1 means the text carries anger.
- **Language** - The language used for analysis in the [ISO language format](#)

Inputs

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public page which will be Analysed
Targets	In	Collection	Set each target string to Analyse emotion for
Document	In	Flag	Set this false to hide document-level results

Output

Emotion (Collection)	Document (Collection)	Language (Text)
Field 1: Target	Field 1: sadness	Language code
Field 2: sadness	Field 2: joy	
Field 3: joy	Field 3: fear	
Field 4: fear	Field 4: disgust	
Field 5: disgust	Field 5: anger	
Field 6: anger		

Get Entities

Get Entities action identifies people, cities, organizations, and many other types of entities in your text. The complete list of entity types and subtypes can be found here <https://console.bluemix.net/docs/services/natural-language-understanding/entity-types.html#entity-types-and-subtypes>.

Supported languages: English, French, German, Italian, Korean, Portuguese, Russian, Spanish, Swedish

The request response is formatted into the following easy to use output:

- **Language** - The language used for analysis in the [ISO language format](#)
- **Entities Document** - Is a collection containing entity information related to the request, formatted into the following five fields:
 - **Entity Name** - The entity which corresponds to the rest of the data row
 - **Entity Type** - Text
 - **Entity Sentiment** - Collection holding two fields, Score and Label
 - **Entity Relevance** - Indicates how much relevance the entity name has to the whole adornment
 - **Entity Emotion** - A collection holding emotion content as seen in section V - Get Emotion.

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Modal ID	In	Text	A custom model ID to override the standard entity detection model.
Limit	In	Number	Maximum number of entities to return. Default 50, maximum 250

Output

Entities Document (Collection)	Language (Text)
Field 1: Entity Name	Language code
Field 2: Entity Type	
Field 3: Entity Sentiment (Collection)	
Field 4: Entity Relevance	
Field 5: Entity Emotion (Collection)	

Get Keywords

Get Keywords action identifies the important keywords in your content.

Supported languages: English, French, German, Italian, Korean, Portuguese, Russian, Spanish, Swedish

The request response is formatted into the following easy to use output:

- **Language** - The language used for analysis in the [ISO language format](#)
- **Keyword Entities** - A collection containing the four following fields:
 - **Keyword Name** - Indicates what keyword has been analysed.
 - **Keyword Sentiment** - A collection containing sentiment information about the keyword (Score and Label)
 - **Keyword Relevance** - Indicates the overall relevance to the document
 - **Keyword Emotion** - Contains emotion content corresponding to the Keyword Name (As seen in Section V – Get Emotions)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Limit	In	Number	Maximum number of keywords to return. Default 50, maximum 250

Output

Keyword Entities	Language (Text)
Field 1: Keyword Name	Language code
Field 2: Keyword Sentiment (Collection)	
Field 3: Keyword Relevance	
Field 4: Keyword Emotion (Collection)	

Get Metadata

Get Metadata action gets document metadata, including author name, title, RSS/ATOM feeds, prominent page image, and publication date. (HTML or web page input only)

Supported languages: English, Korean, Spanish

The request response is formatted into the following easy to use output:

- **Meta Data** - This collection is formatted into the five following fields:
 - **Name**
 - **Publication Date**
 - **Image**
 - **News Feed** (Collection)
 - **Authors** (Collection)
- **Language** - The language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed

Output

Meta Data (Collection)	Language (Text)
Field 1: Name	Language code
Field 2: Publication Date	
Field 3: Image	
Field 4: News Feed (Collection)	
Field 5: Authors (Collection)	

Get Relations

Get Relations action recognizes when two entities are related, and identify the type of relation. For example, an "awardedTo" relation might connect the entities "Oscar" and "Milos Forman". The complete list of relation types can be found here <https://console.bluemix.net/docs/services/natural-language-understanding/relations.html#relation-types>.

Supported languages: Arabic, English, Korean, Spanish

The request response is formatted into the following easy to use output:

- **Relation** - This output contains four fields:
 - **Sentence**
 - **Score**
 - **Entity** (Collection holding Type and Text fields)
 - **Additional Entities** (Collection holding Type and Text fields)
- **Language** - The language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Model ID	In	Text	Specify the ID of a deployed Watson Knowledge Studio custom model to override the default model.

Output

Relation (Collection)	Language (Text)
Field 1: Sentence	Language code
Field 2: Score	
Field 3: Entity (Collection)	
Field 4: Additional Entities (Collection)	

Get Semantic Roles

Get Semantic Roles action parses sentences into subject, action, and object form.

Supported languages: English, Korean, Spanish

The request response is formatted into the following easy to use output:

- **Semantic Roles** - This collection is split up into four fields:
 - **Subject**
 - **Sentence**
 - **Object**
 - **Verbs** (Collection holding two fields; Text and Tense)
- **Language** - The language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Limit	In	Number	Maximum number of semantic role results to return. Default: 50

Output

Semantic Roles (Collection)	Language (Text)
Field 1: Subject	Language code
Field 2: Sentence	
Field 3: Object	
Field 4: Verbs (Collection)	

Get Sentiment

Get Sentiment action analyses the general sentiment of your content, or analyses the sentiment toward specific target phrases found in the text which has a relation to the original information. For example, if you were analysing the sentiment of a stock website, then a target variable could be “Stocks”.

Supported languages: Arabic, English, French, German, Italian, Korean, Portuguese, Russian, Spanish

The request response is formatted into the following easy to use output:

- **Sentiment** - Collection containing three fields:
 - **Target** - Target object for which the sentiment is Analysed
 - **Score** - Sentiment score ranging from -1 (negative sentiment) to 1 (positive sentiment)
 - **Label** - Indicates whether the sentiment is positive, neutral or negative
- **Document Sentiment** - Collection containing sentiment analysis result for the entire document:
 - **Score** - Sentiment score ranging from -1 (negative sentiment) to 1 (positive sentiment) at the document level
 - **Label** - indicates whether the sentiment is positive, neutral or negative at the document level
- **Language** - the language used for analysis in the [ISO language format](#)

Input

Parameter	Direction	Data Type	Description
Text To Analyse	In	Text	If plain text analysis is required enter plain text here
HTML To Analyse	In	Text	if HTML analysis is required formatted source code is accepted
URL To Analyse	In	Text	URL to public web page which will be Analysed
Targets	In	Collection	The service Analyses sentiment for each target string found in the text up to 20 targets.
Document	In	Flag	Set this false to hide document-level sentiment results

Output

Sentiment (Collection)	Document (Collection)	Language (Text)
Field 1: Target	Field 1: score	Language code
Field 2: Score	Field 2: label	
Field 3: Label		

List Models

This action lists available custom models.

The request response is formatted into the following easy to use output:

- **Models** - Collection containing four fields:
 - **Status** - Status of the custom model (available, training, etc.)
 - **Model_ID** - Unique model_id of the custom model
 - **Language** - Language of the model
 - **Description** - Custom description of the model

Output

Models (Collection)
Field 1: Status
Field 2: Model_ID
Field 3: Language
Field 4: Description

Delete Model

This action deletes a custom model.

Output

Status (Text)
Status

Language Translator VBO

The Language Translator VBO consists of eight different actions:

- [Configure below](#)
- [Translate below](#)
- [Get Identifiable Languages below](#)
- [Identify Language on the next page](#)
- [List Models on page 32](#)
- [Get Model Details on page 33](#)
- [Delete Model on page 34](#)

Configure

This action must be called before any additional VBO action as it sets up all required parameters for Language Translator Service.

Translate

Translate action provides translation of provided text from a source language to a target language. You need to specify a text to translate, you can define which translation model to use or set source and target languages. Once request is sent the response is formatted in the following output:

- **Paragraphs** - This collection contains all paragraphs translated into the target language.
- **Word Count** - Number of words of the complete input text
- **Character Count** - Number of characters of the complete input text

Outputs

Paragraphs (Collection)	Word Count (Text)	Character Count (Text)
Field 1: Translation	Word Count	Character Count

Get Identifiable Languages

This action provides the list of languages which can be detected by IBM Watson. The output is a collection with two columns.

- **Languages** - This collection contains two columns:
 - **Language** - Two or five letter language code
 - **Name** - Full language name.

Outputs

Languages (Collection)
Field 1: Language
Field 2: Name

Identify Language

This action identifies the language in which a text is written. The output is a collection with two columns.

- **Identified Languages** - This collection contains two fields; Language and Confidence (a ranking of languages with confidence scores, 1.0 is the highest value)
 - **Language** - Two letter language code (ISO language format)
 - **Confidence** - A ranking of languages with confidence score ranging from 0 to 1.0

Outputs

Identified Languages (Collection)
Field 1: Language
Field 2: Confidence

List Models

This action lists available models for the Language translator service with option to filter by source or by target language. The output is provided in a form of a collection.

- **Models** - This collection contains columns describing the model.
 - **Model_ID** - A unique string that identifies the underlying model used for translation. This string contains all the information about source language, target language, and domain.
 - **Source** - A translation source language in 2 or 5 letter language code. When model_id is used directly, it will override the source-target language combination, when a 2 letter language code is used, and no suitable default is found, it returns an error.
 - **Target** - A translation target language in 2 or 5 letter language code. When model_id is used directly, it will override the source-target language combination, when a 2 letter language code is used, and no suitable default is found, it returns an error.
 - **Base_Model_ID** - A base model it was trained on. For a base model, the response value is empty.
 - **Domain** - A domain of the translation model.
 - **Customizable parameter** - Defines whether this model can be used as a base for customization. Customized models are not further customizable, and IBM don't allow the customization of certain base models.
 - **Default_Model** - Describes whether this model is considered default, and whether it is used when the source and target languages are specified without the model_id.
 - **Owner** - Can be an empty string, indicating it's a model trained by IBM, or the ID of the service instance where the model was created.
 - **Status** - Current availability of the model. Possible values are
 - **Name** - An optional parameter attached during training to help the user identify his model.
 - **Train_Log** - Undocumented feature.

Outputs

Models (Collection)
Field 1: Model_ID
Field 2: Source
Field 3: Target
Field 4: Base_Model_ID
Field 5: Domain
Field 6: Customizable
Field 7: Default_Model
Field 8: Owner
Field 9: Status
Field 10: Name
Field 11: Train_Log

Get Model Details

This action provides information, including training status, about a specified translation model. The output is provided in a form of a collection.

- **Model Details** - This collection contains columns describing the model.
 - **Model_id** - A unique string that identifies the underlying model used for translation. This string contains all the information about source language, target language, and domain.
 - **Source** - A translation source language in 2 or 5 letter language code. When model_id is used directly, it will override the source-target language combination, when a 2 letter language code is used, and no suitable default is found, it returns an error.
 - **Target** - A translation target language in 2 or 5 letter language code. When model_id is used directly, it will override the source-target language combination, when a 2 letter language code is used, and no suitable default is found, it returns an error.
 - **Base_model_id** - A base model it was trained on. For a base model, the response value is empty.
 - **Domain** - Domain of the translation model.
 - **Customizable parameter** - Defines whether this model can be used as a base for customization. Customized models are not further customizable, and IBM don't allow the customization of certain base models.
 - **Default_model** - Describes whether this model is considered default, and whether it is used when the source and target languages are specified without the model_id.
 - **Owner** - Can be an empty string, indicating it's a model trained by IBM, or the ID of the service instance where the model was created.
 - **Status** - Current availability of the model. Possible values are
 - **Name** - An optional parameter attached during training to help the user identify his model.
 - **Train_Log** - Undocumented feature.

Outputs

Model Details (Collection)
Field 1: model_id
Field 2: source
Field 3: target
Field 4: base_model_id
Field 5: domain
Field 6: customizable
Field 7: default_model
Field 8: owner
Field 9: status
Field 10: name
Field 11: train_log

Delete Model

This action deletes trained translation model identified by model_id. Returns Status OK in case the model was successfully deleted.

Visual Recognition VBO

The Visual Recognition VBO consists of three different actions:

- [Configure below](#)
- [Classify Image below](#)
- [Detect Faces on the next page](#)

Configure

This action must be called before any additional VBO action as it sets up all required parameters for Visual Recognition Service.

Classify Image

This action provides image classification. Once the request is sent the response is formatted in the following output:

- **Classes** - This collection contains three columns:
 - **Class** - The name of the class identified in the image
 - **Score** - The confidence score for the property in the range of 0 to 1. A higher score indicates greater likelihood that the class is depicted in the image. The default threshold for returning scores from a classifier is 0.5
 - **Type_Hierarchy** - A parameter included only if identified

Outputs

Class (Collection)
Field 1: Class
Field 2: Score
Field 3: Type_Hierarchy

Detect Faces

This action Analyses and get data about faces in images. This feature uses a built-in classifier, so you do not train it on custom classifiers. The detect faces methods do not support general biometric facial recognition. Once the request is sent the response is formatted in the following output:

- **Age** - This collection contains three columns:
 - **Max** - Estimated maximum age
 - **Min** - Estimated minimum age.
 - **Score** - The confidence score for the ages in range of 0 to 1.
- **Gender** - This collection contains two columns:
 - **Gender** - Can be Male or Female.
 - **Score** - The confidence score for the gender in range of 0 to 1.
- **Face Location** - This collection contains four columns:
 - **Height** - Height in pixels of the face region.
 - **Width** - Width in pixels of the face region.
 - **Left** - X-position of the top-left pixel of the face region.
 - **Top** - Y-position of the top-left pixel of the face region.
- **Identity** - The collection contains three columns:
 - **Name** - The name of the person.
 - **Score** - The confidence score for the ages in range of 0 to 1.
 - **Type_hierarchy** - A knowledge graph of the celebrity, included only if identified.

Outputs

Age (Collection)	Gender (Collection)	Face Location (Collection)	Identity (Collection)
Field 1: max	Field 1: gender	Field 1: height	Field 1: name
Field 2: min	Field 2: score	Field 2: width	Field 2: score
Field 3: score		Field 3: left	Field 3: type_hierarchy
		Field 4: top	