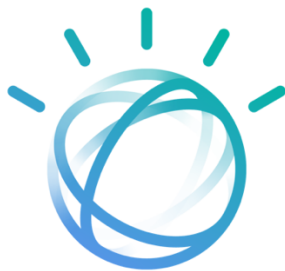


Collegamento al back-end di un assistente virtuale costruito con IBM Watson Assistant tramite IBM Cloud Functions

Francesca Gigante: francesca_gigante@it.ibm.com





Indice

Introduzione	3
Obiettivi.....	3
Prerequisiti	3
STEP 1: Creazione di una action con IBM Cloud Function	4
STEP 2: Info necessarie ad invocare la action da WA	8
Endpoint	8
API Key	8
Parametri	9
STEP 2: Come invocare la action da WA.....	10



Introduzione

Un assistente virtuale costruito con IBM Watson Assistant può essere collegato a qualsiasi web-services o API attraverso una 'action' realizzata tramite IBM Cloud Functions che agisce da wrapper.

La documentazione relativa a questa capability è disponibile qui

<https://console.bluemix.net/docs/services/conversation/dialog-actions.html#dialog-actions>

E' importante indicare che la location geografica delle Action realizzate in Cloud Function deve coincidere con quella del servizio WA (unica eccezione UK che fa riferimento a US)

Quindi abbiamo:

- WA US -> Cloud Function US
- WA UK -> Cloud Function US
- WA DE -> Cloud Function DE

Obiettivi

- Creare un aggancio tra un nodo di Dialogo e una Cloud Function

Prerequisiti

Prerequisito per questo laboratorio è avere realizzato almeno un nodo di dialogo con il Watson Assistant.

Il laboratorio è disponibile a questo link:

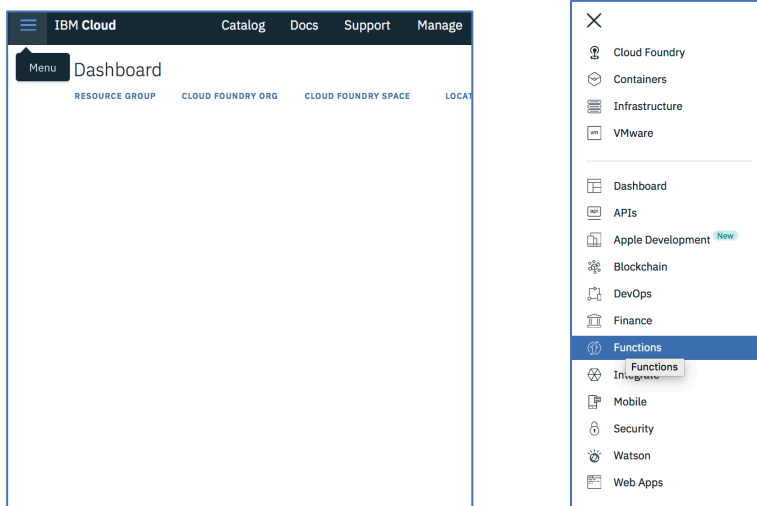
<https://github.com/francesca-gigante/watson-assistant-hands-on>



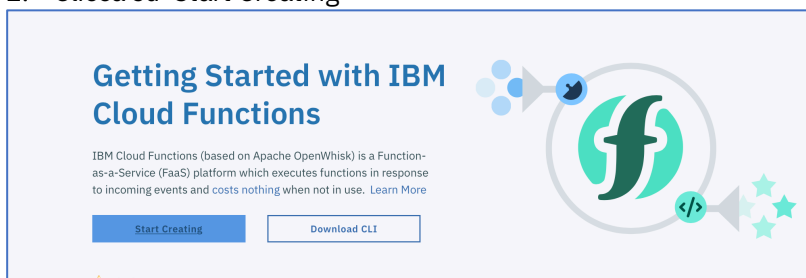
STEP 1: Creazione di una action con IBM Cloud Function

Entrare nel tool di gestione delle Cloud Functions

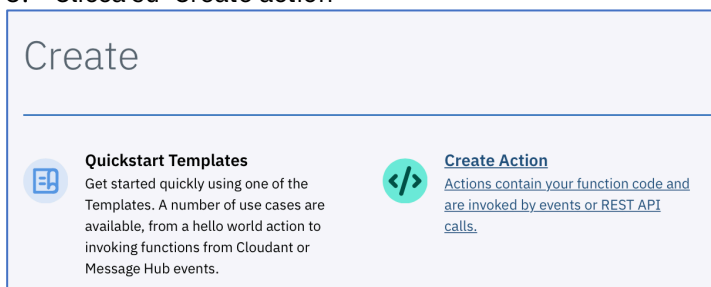
1. Clicca su 'Menu' in alto a sinistra e poi seleziona la voce 'Functions':



2. Clicca su 'Start Creating'



3. Clicca su 'Create action'





4. Inserisci il nome della tua Action, crea o associa un package (i packages servono ad organizzare/raggruppare le functions), definisci il linguaggio di programmazione associato alla tua Action (per questo laboratorio selezioniamo 'Node.js 8'). Alla fine clicca su 'Create'.

Create Action

Actions contain your function code and are invoked by events or REST API calls.

[Learn more about Actions](#)
[Learn more about Packages](#)

Action Name

Enclosing Package ¹
(Default Package)

Runtime ¹

☒ Node.js 8
☐ Node.js 6
☐ Python 3
☐ Swift 4
☐ Swift 3.1.1
☐ PHP 7.1

La tua action è pronta

Functions / Actions / prova1 Region: Germany Org: FANIA GIGANTE ORG Space: dev_de_personal

prova1 Web Action

Code Parameters Runtime Endpoints Connected Triggers Enclosing Sequences Logs

Code ¹ Node.js 8 Edit mode - press ESC to exit Reset Save

```
1 ~ /**
2 ~ *
3 ~ * main() will be run when you invoke this action
4 ~ *
5 ~ * @param Cloud Functions actions accept a single parameter, which must be a JSON object.
6 ~ *
7 ~ * @return The output of this action, which must be a JSON object.
8 ~ *
9 ~ */
10 ~ function main(params) {
11 ~   return { message: 'Hello World '};
12 ~ }
13 ~
```

5. Modifichiamo il codice per fare in modo che ritorni non un messaggio fisso ma un messaggio variabile in base ad un parametro di input fornito alla funzione.

Ad esempio, cambiamo:

```
return { message: 'Hello World ' };
```

con:

```
return { message: 'Hello World ' + params.date };
```

6. Dobbiamo quindi definire un parametro di nome 'date' fornendo anche un valore di default che ci permette di testare la funzione anche dallo stesso tool:

Functions / Actions / prova1 Region: Germany Org: FANIA GIGANTE ORG Space: dev_de_personal

prova1 Web Action

Code Parameters Runtime Endpoints Connected Triggers

Parameters ¹ Add

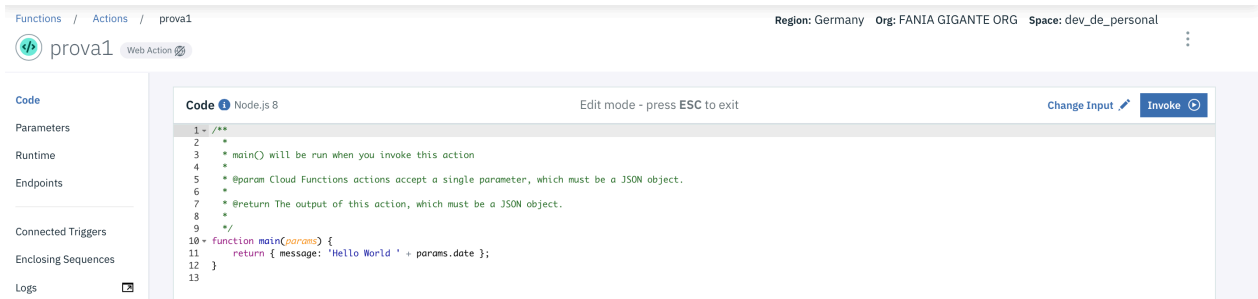
Parameter Name	Parameter Value
date	"2018-01-12"



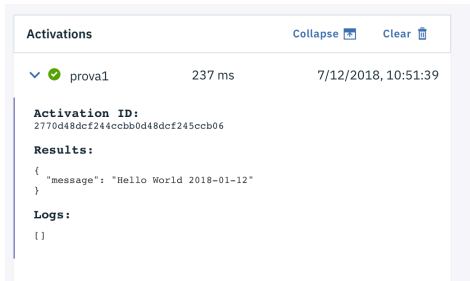
Nota 1: Alla voce 'Runtime' del menu è possibile impostare Timeout e memoria associata all'esecuzione della action.

Nota 2: Le actions possono essere anche concatenate in sequence. Quindi è possibile sviluppare action elementari e poi concatenarle tra loro.

7. Per testare la funzione clicca sul tasto 'Invoke'

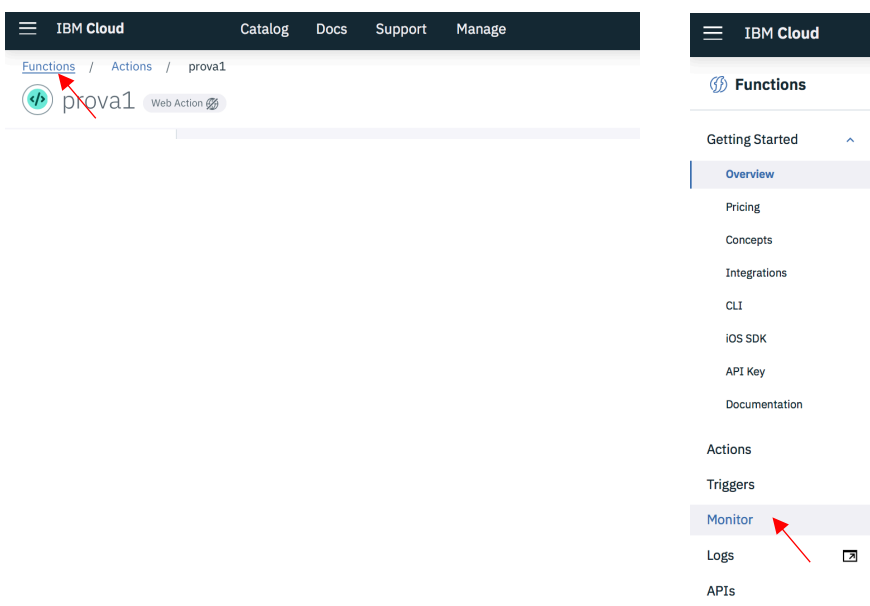


Avrai tutte le informazioni sulla invocazione ed il messaggio di ritorno:



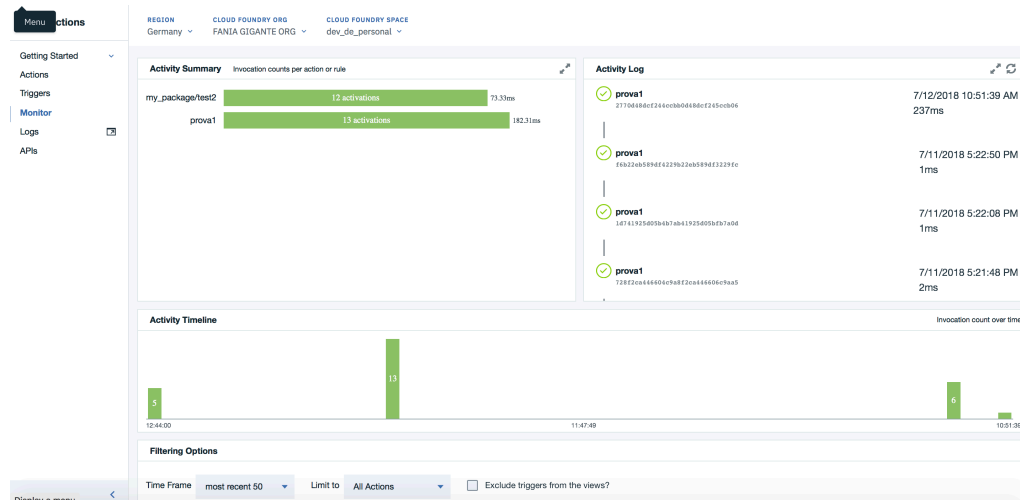
Il 'baffo verde' accanto al nome della funzione significa che l'esecuzione è andata a buon fine.
Il messaggio di ritorno è riportato sotto 'Results'

8. Puoi recuperare le informazioni delle invocazioni della tua funzione in ogni momento attraverso il menu monitor.
Vai su 'Functions' per tornare alla schermata principale del tool e poi seleziona Monitor





Vedrai tutte le attivazioni delle tue funzioni, il tempo di esecuzione e i dettagli cliccando su ciascuna di esse.



Come già evidenziato in introduzione, la action può essere collegata a qualsiasi web-services o API accessibile al Cloud IBM perché esposta su Internet o in maniera sicura tramite il Gateway (<https://console.bluemix.net/catalog/services/secure-gateway>) .

Ora è necessario recuperare le informazioni necessarie per permettere alla action di essere invocata dal WA.



STEP 2: Info necessarie ad invocare la action da WA

Per invocare la action da WA abbiamo bisogno delle seguenti informazioni:

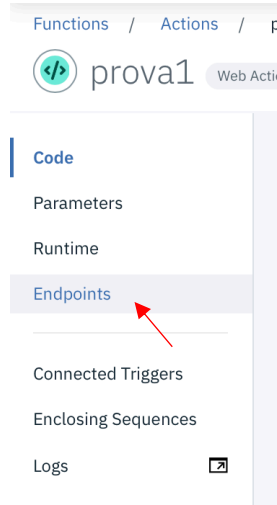
Endpoint

API Key (user e password)

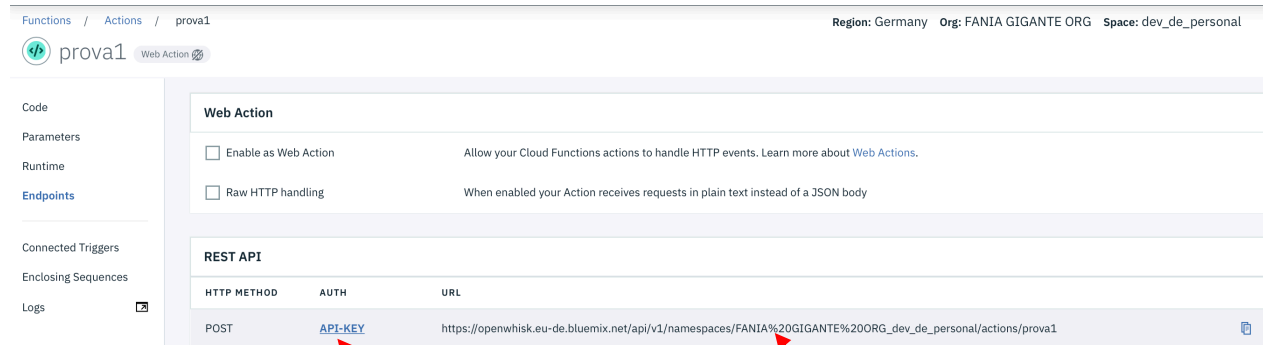
Parametri

Endpoint

Clicca su



Seleziona URL



L'endpoint che ci serve è composto da: `"/org+space/nome_action"` (rimuovendo opportunamente i %xx)
Quindi è l'URL privato della parte iniziale <https://openwhisk.eu-de.bluemix.net/api/v1/namespaces> e privato anche della parte interna `"/actions"`.

API Key

Ora clicca su API-KEY. Si apre la seguente schermata:



IBM Cloud

Functions

Getting Started

Overview

Pricing

Concepts

Integrations

CLI

IOS SDK

API Key

Documentation

Actions

REGION: Germany

CLOUD FOUNDRY ORG: FANIA GIGANTE ORG


CLOUD FOUNDRY SPACE: dev_de_personal

API Key

The IBM Cloud Functions [REST API](#) is protected with an API Key. Each Namespace has a unique API Key assigned, which needs to be provided via HTTP Basic authentication when calling any API in that Namespace, except for Web Actions invocations.

Start Creating

CURRENT NAMESPACE	HOST	KEY
FANIA GIGANTE ORG_dev_de_personal	openwhisk.eu-de.bluemix.net

Clicca su  per vedere la API Key e poi clicca su  per copiarla nella tua clipboard.

Usa un editor di testo, incolla la API Key copiata e separa *username* e *password* che sono concatenati tramite “:”. Tienili da parte perché serviranno come credenziali nel WA.

Parametri

Per quanto riguarda i parametri della action sono quelli definiti al passo #6 dello step 1.

Bene, procediamo ad agganciare la action dal WA.



STEP 2: Come invocare la action da WA

Per prima cosa dobbiamo creare una variabile di contesto con le credenziali (API key) per l'accesso alla Action. E' buona norma creare questa variabile nel nodo di benvenuto per fare in modo che sia sempre accessibile a tutti i nodi di dialogo sin dall'inizio della conversazione

Andare nel nodo di Benvenuto e aprire il 'Context editor'

The screenshot shows the Watson Assistant interface. On the left, a flowchart shows the 'Benvenuto' node (welcome) leading to a node labeled '#prova1'. The 'Benvenuto' node has a status of '1 Response / 1 Context set / Does not return'. The '#prova1' node has a status of '1 Response / 1 Context set / Jump to'. Below it, a link says 'Jump to answer prova1 (Evaluate responses)'. On the right, the 'Benvenuto' node is selected, showing its configuration. Under 'If bot recognizes:', the text 'welcome' is entered. Under 'Then respond with:', the text '1. Salve. Come posso aiutarti?' is entered. A red arrow points to the 'Open context editor' button in the bottom right corner.

Definisci la seguente variabile \$private così composta

```
{"my_credentials":{"user":"username","password":"password"}}
```

Sostituisci *username* e *password* con quelle recuperate dalla API Key nello step precedente

The screenshot shows the 'Then set context:' section in the Watson Assistant interface. It contains a table with two columns: 'Variable' and 'Value'. The first row shows the variable '\$ private' and the value '{"my_credentials":{"user":"fefa39d7'.

Variable	Value
\$ private	{"my_credentials":{"user":"fefa39d7

Below the table, there is a button labeled '+ Add variable'.

Ora dobbiamo gestire il nodo della conversazione che riceve una informazione ed invoca la action e poi un nodo che gestisce la risposta.



Supponiamo che il nodo che invoca la Action sia il nodo **#prova1**. Possiamo indicare una risposta che il WA fornisce per indicare che sta chiamando la action (es. “attendere prego...”), e successivamente dobbiamo impostare il ‘Jump to’ al nodo di risposta come indicato nella figura precedente.

Richiamiamo il Json editor (non è stato ancora predisposta una GUI per il collegamento con le action)

Aggiungiamo le info necessarie ad invocare la action.
Il JSON finale sarà:

```
{
  "context": {
    "date": "@sys-date"
  },
  "output": {
    "text": {
      "values": [
        "chiamo prova 1 con $date"
      ],
      "selection_policy": "sequential"
    }
  },
  "actions": [
    {
```



```
"name": "/FANIA GIGANTE ORG_dev_de_personal/prova1",  
"type": "server",  
"parameters": {  
  "date": "$date"  
},  
"credentials": "$private.my_credentials",  
"result_variable": "$risposta"  
}  
]  
}
```

Come indicato nel JSON la risposta della action (result_variable) viene inserita in una variabile di contesto che abbiamo chiamato \$risposta. Questa variabile potrà essere usata nel nodo di dialogo che fornisce la risposta all'utente come indicato nella figura seguente

The screenshot shows the 'Then respond with' section of the Watson Assistant interface. At the top, there is a search bar with 'answer prova1' and buttons for 'Customize' and a close icon. Below this, the 'If bot recognizes' section has a placeholder 'Enter an intent, entity or context variable...'. The 'Then respond with' section contains a list of responses. The first response is '1. questa è la risposta da prova 1 \$risposta.message'. Below the list is a button 'Add a variation to this response'.

Ora non ti resta che provare:

The screenshot shows the 'Try it out' interface of the Watson Assistant. At the top, there are buttons for 'Clear', 'Manage Context', and a close icon. The main area shows a conversation. The user input is 'Salve. Come posso aiutarti?'. The bot response is 'prova 1 domani'. Below this, there is a dropdown menu showing '#prova1' and a button '@sys-date:2018-07-13'. The conversation history shows 'chiamo prova 1 con 2018-07-13' and 'questa è la risposta da prova 1 Hello World 2018-07-13'.



Acknowledgements and Disclaimers

Copyright © 2017 by International Business Machines Corporation (IBM). No part of this document may be reproduced or transmitted in any form without written permission from IBM.

U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.

Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed “as is” without any warranty, either express or implied. In no event shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted according to the terms and conditions of the agreements under which they are provided.

IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply.”

Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.

Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.

References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.

Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.

It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer is in compliance with any law.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a particular, purpose.**

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right. IBM, the IBM logo, ibm.com, Aspera®, Bluemix, Blueworks Live, CICS, Clearcase, Cognos®, DOORS®, Emptoris®, Enterprise Document Management System™, FASP®, FileNet®, Global Business Services®, Global Technology Services®, IBM ExperienceOne™, IBM SmartCloud®, IBM Social Business®, Information on Demand, ILOG, Maximo®, MQIntegrator®, MQSeries®, Netcool®, OMEGAMON, OpenPower, PureAnalytics™, PureApplication®, pureCluster™, PureCoverage®, PureData®, PureExperience®, PureFlex®, pureQuery®, pureScale®, PureSystems®, QRadar®, Rational®, Rhapsody®, Smarter Commerce®, SoDA, SPSS, Sterling Commerce®, StoredIQ, Tealeaf®, Tivoli® Trusteer®, Unica®, urban{code}®, Watson, WebSphere®, Worklight®, X-Force® and System z® Z/OS, are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.