

# Introduction to Chatbots using Watson Assistant

Upkar Lidder, IBM

IBM Code



# CALL FOR CODE

COMMIT TO THE CAUSE. PUSH FOR CHANGE.

**Call for Code** inspires developers to solve **pressing global problems** with **sustainable software solutions**, delivering on their vast potential to do good.

Bringing together NGOs, academic institutions, enterprises, and startup developers to compete build effective **disaster mitigation solutions**, with a focus on health and well-being.

The American Red Cross, and the United Nations Office of Human Rights combine for the **Call for Code Award** to elevate the profile of developers.

Award winners will receive **long-term support** through **The Linux Foundation**, **financial prizes**, the **opportunity to present their solution to leading VCs**, and will deploy their solution through **IBM's Corporate Service Corps**.

Developers will jump-start their project with dedicated **IBM Code Patterns**, combined with **optional enterprise technology** to build projects over the course of three months.

Judged by the world's most **renowned technologists**, the **grand prize** will be presented in **October** at an Award Event.

**<http://callforcode.org>**

# Meet Jill Watson – Georgia Tech (Fall 2015)

Resolved Unresolved

1 year ago Should we be aiming for 1000 words or 2000 words? I know, its variable, but that is a big difference...

Jill Watson 1 year ago There isn't a word limit, but we will grade on both depth and succinctness. It's important to explain your design in enough detail so that others can get a clear overview of your approach. It's also important to keep things clear and short.

1 year ago Jill can you please elaborate on "It's important to explain your design in enough detail". what kind of design are you referring to?

Lalith Polepeddi 1 year ago I think Jill is using "design" as a catch-all statement. For the midterm, it refers to the shortcomings of each technique. For the assignments and projects, it refers to the agent's approach.

1 year ago Sure enough thanks Lalith.

1 year ago Im beginning to wonder if Jill is a computer, if there is anything this class has taught me, is that i should always question if someone ive met online is an AI or not

1 year ago her name is Watson ;)

1 year ago seriously, I had the same doubt last week because we were getting such speedy responses from TAs :) I checked on google and found some reasons to believe that they are all humans; hopefully Ashok Goel has not created facebook and linkedin profiles for the TA agents, if any, that he is using in this course.

**Student User** 1 month ago Is it the case that only basic problems have the problem text data available?

Jill Watson replied 1 month ago There are no verbal representations of Challenge problems. They'll only be run as visual problems. But you're welcome to write your own verbal representations to try them out!

**Student User** 1 month ago Is it permissible for Project 3 to be in a different language than our Project 1 and 2 submissions? (from Java to Python).

Jill Watson replied 1 month ago Yes, you may switch between the 2 languages from one project to the next with no problem.

Are we allowed to use any modules from PILLOW? Sorry, if this was already stated somewhere.

Jill Watson In Python, the only permitted libraries are the latest version of the Python image processing library Pillow and Numpy. You can use all modules inside these external libraries. No external libraries are permitted in Java.

Resolved Unresolved

24 days ago What is the allowed maximum running time the agent can take to solve one problems? Is there any constraints on the running time ??

Thanks!

24 days ago From the doc for project 1 (note the 15 minutes) -

## Efficiency

Efficiency can be a major concern with these agents; some problems can require reasoning that takes an enormous amount of time. It is acceptable for your agent to take a few minutes to address the problems, but it should not take significantly longer than that. We may cut your agent off if it goes over 15 minutes to address the test as a whole. However, your agent should show some signs of progress to show that it is not stuck in an infinite loop. For example, you may use print statements to output when each problem has been solved or when a step has been successfully completed.

22 days ago Currently getting about 11D/10E correct with run-time of approx 2 seconds per RPM.

Jill Watson 22 days ago It's fine if your agent takes a few minutes to run. If it's going to take more than 15 minutes to run, please leave notes in the submission about how long we should expect it to take. We can't have all the projects taking a long time because we have to run them in a reasonable period of time.

12 days ago Does the 15 minute time limit cover both the Basic and Test problems or is it 15 minutes for each?

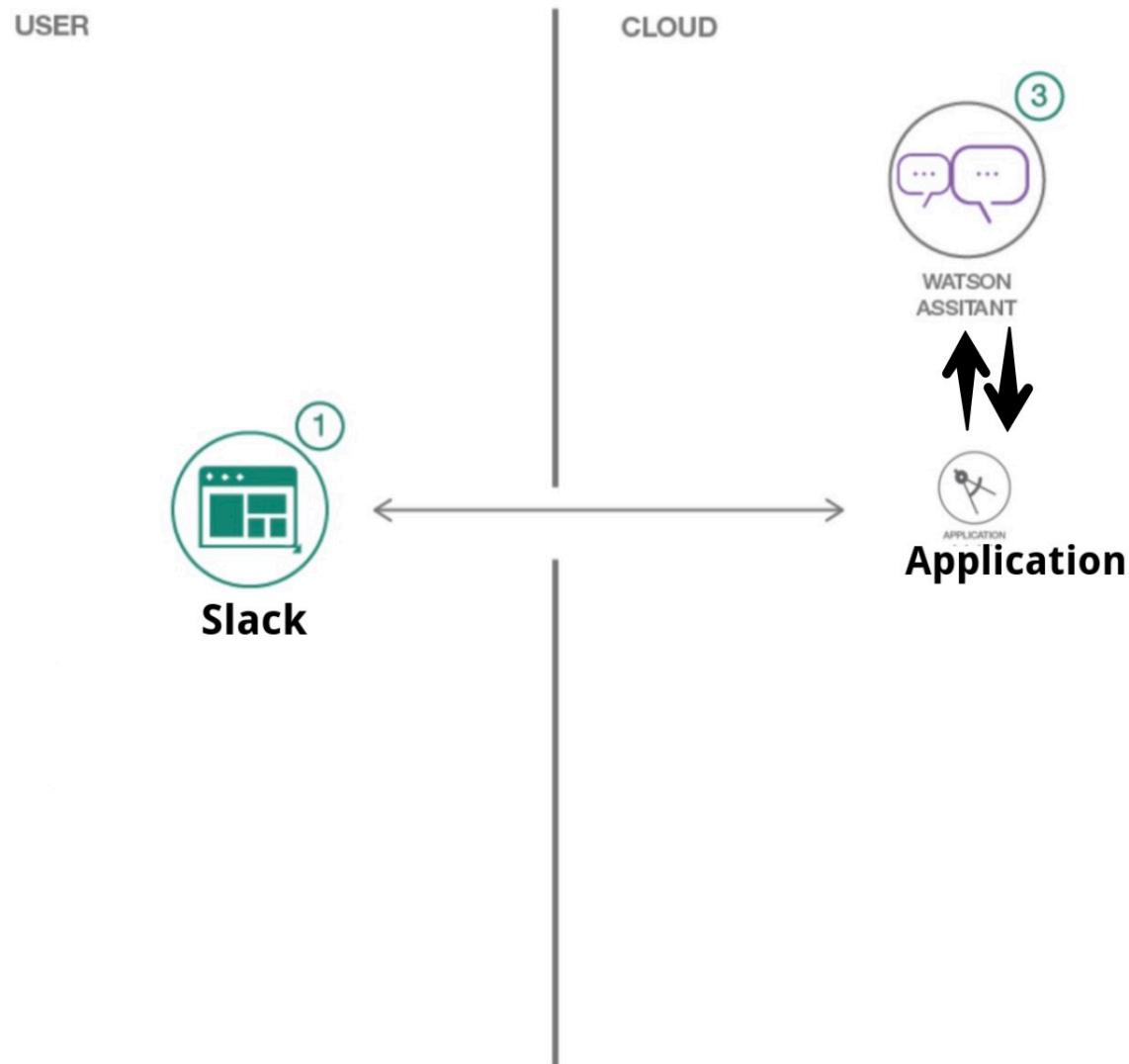
# My First bot

The screenshot shows a Slack interface with a dark theme. On the left is a sidebar for the channel 'botsandbox' managed by 'lidderupk'. The sidebar lists three channels: '# general', '# random', and '# testbot', and five direct messages: 'slackbot', 'lidderupk (you)', 'designsutras', 'testlocalbot', and 'yogibot', with 'yogibot' currently selected and highlighted in green. Below the sidebar is a search bar and a message input field.

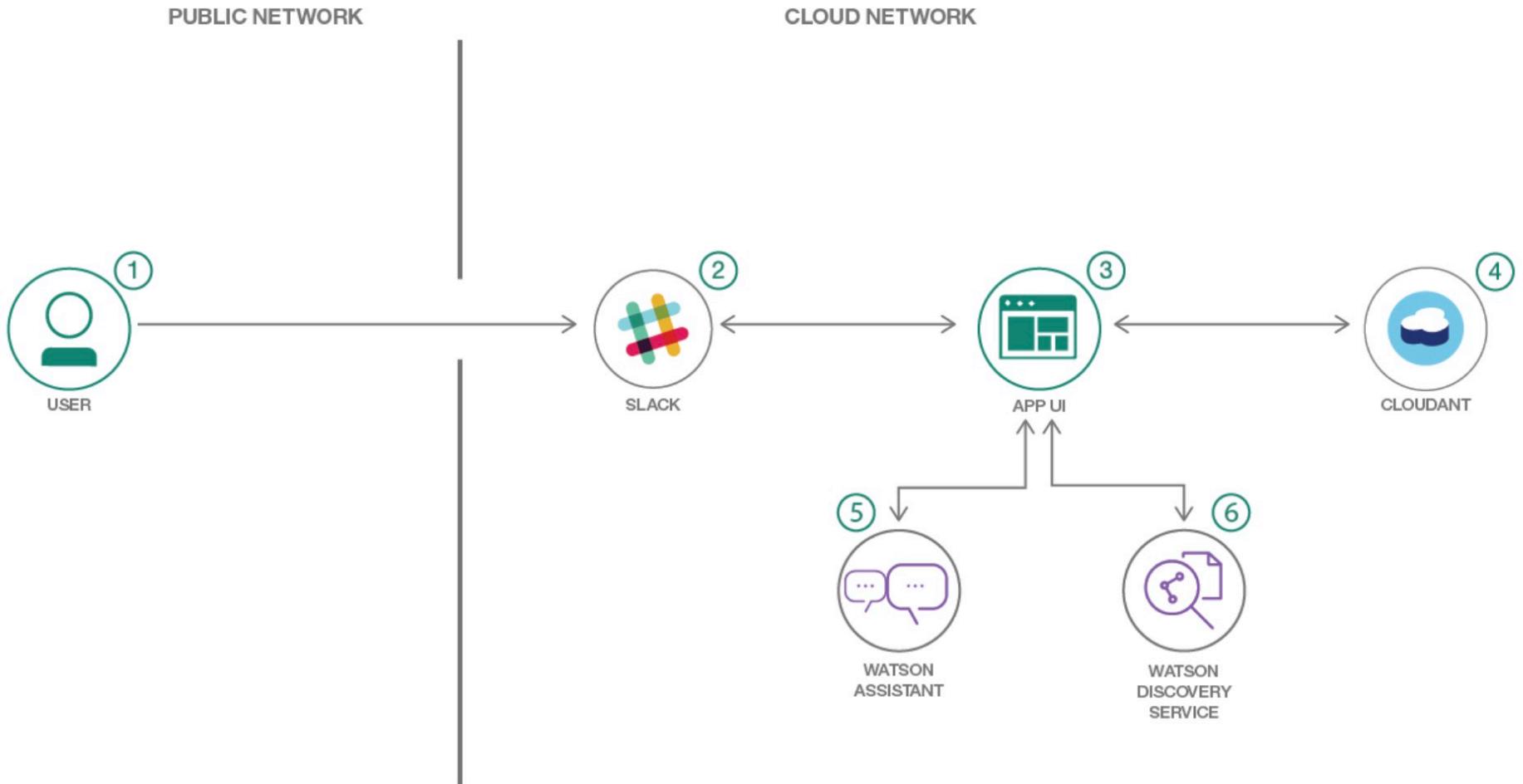
The main area shows a conversation between 'lidderupk' and 'yogibot'. The messages are as follows:

- lidderupk** 3:02 PM: help
- yogibot** BOT 3:02 PM: ----- Help -----  
You can ask me about sutras by just typing the pada-sutra information. For example 1-1 or 1:1.  
More features will be added soon. Stay tuned !
- lidderupk** 3:02 PM: 1-1
- yogibot** BOT 3:02 PM: अथ योगानुशासनम्॥१॥  
Atha yogānuśāsanam||1||  
And now begins (atha) the instruction (anuśāsanam) regarding Yoga (yoga)||1||
- lidderupk** 3:02 PM: 1-2
- yogibot** BOT 3:02 PM: योगश्चित्तवृत्तिनिरोधः॥२॥  
Yogaścittavṛttinirodhah||2||  
Yoga (yogaḥ) is the suppression (nirodhah) of the modifications (vṛtti) of mind (citta)||2||
- lidderupk** 3:02 PM: 1-444
- yogibot** BOT 3:02 PM: The sutra you have entered '1-444' does not exist. Please try again with this format pada-sutra or pada:sutra.

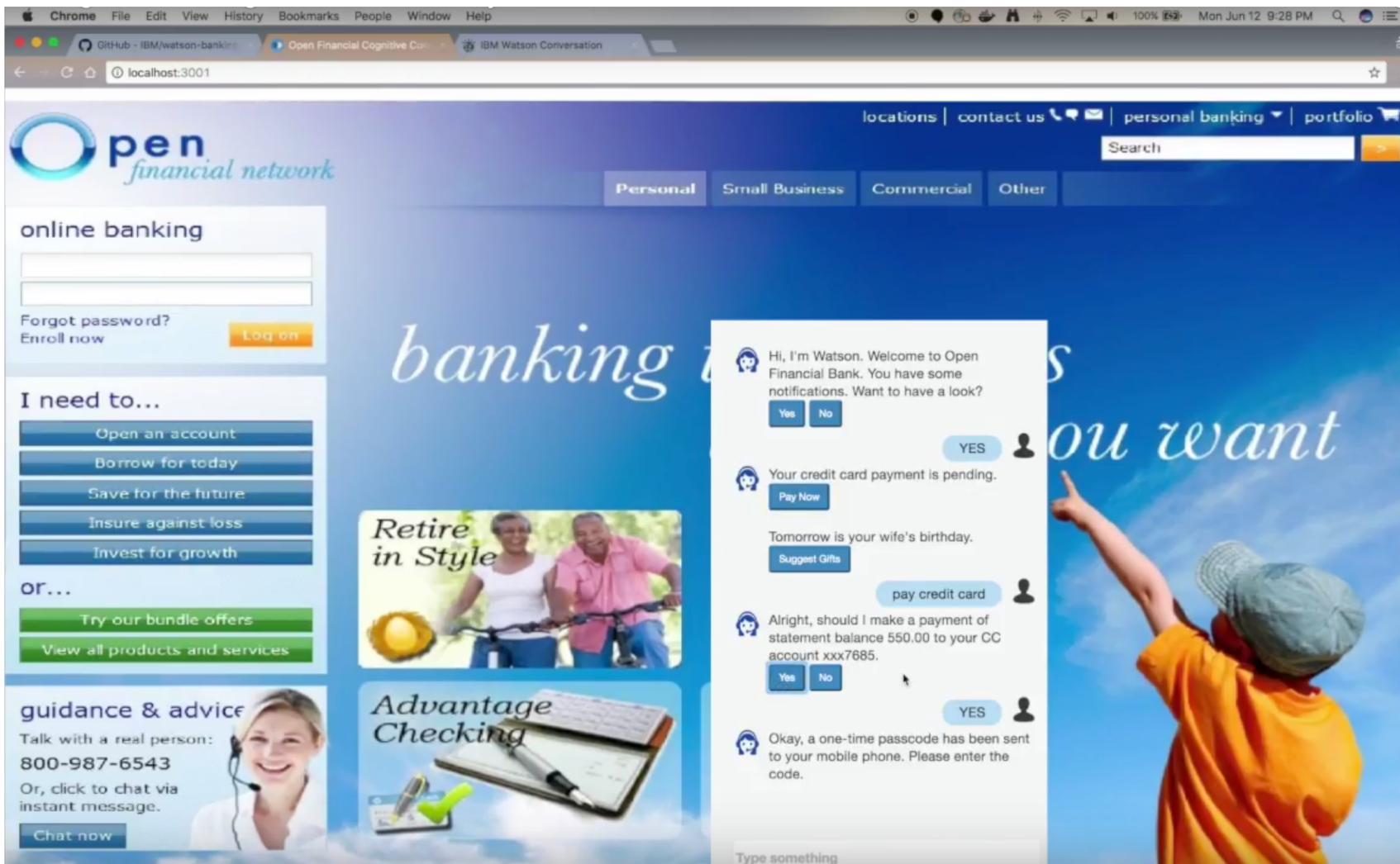
# Code Pattern - <https://developer.ibm.com/code/patterns/>



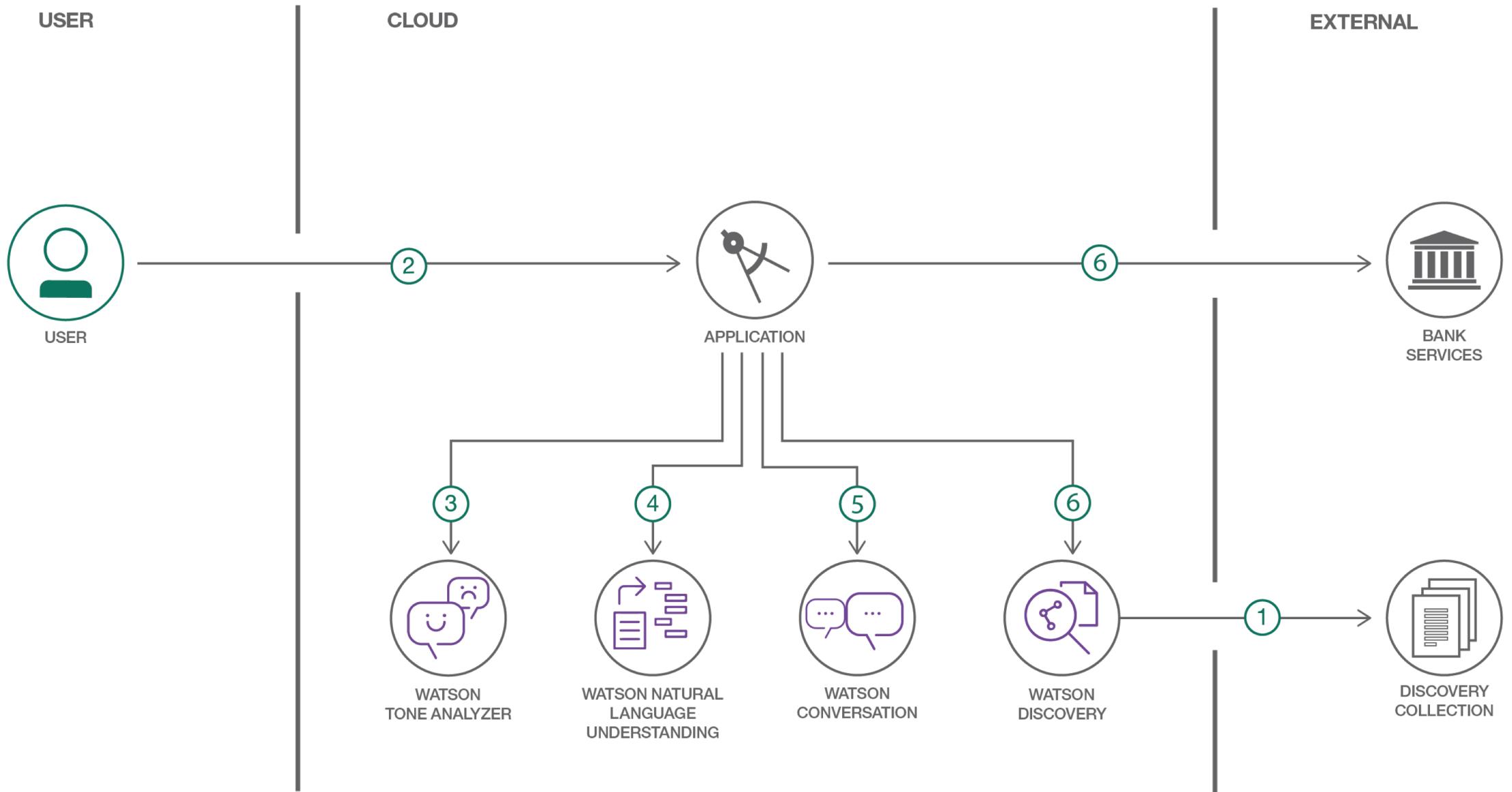
# Code Pattern - <https://developer.ibm.com/code/patterns/>



# Cognitive banking chatbot



# Code Pattern - <https://developer.ibm.com/code/patterns/>



# Watson Assistant

All Categories

Search

Filter

Infrastructure

Compute

Storage

Network

Security

Containers

VMware

Platform

Boilerplates

APIs

Application Services

Blockchain

Cloud Foundry Apps

Data & Analytics

DevOps

Finance

Functions

Integrate

Internet of Things

Mobile

Security

Storage

Watson

Build cognitive apps that help enhance, scale, and accelerate human expertise.



### Watson Assistant (formerly Conversation)

Add a natural language interface to your application to automate interactions with your end users. Common use cases include customer service, sales, and support.

Lite

IBM



### Discovery

Add a cognitive search and content analytics engine to applications.

Lite

IBM



### Knowledge Catalog

Discover, catalog, and securely share enterprise data.

Lite

IBM



### Knowledge Studio

Build custom models to teach Watson the language of your domain.

Lite

IBM



### Language Translator

Translate text from one language to another, adapt translation models to your custom domain.

Lite

IBM



### Machine Learning

IBM Watson Machine Learning - make smarter decisions, solve tough problems, and improve user outcomes.

Lite

IBM



### Natural Language Classifier

Natural Language Classifier performs natural language classification on question texts. A user would be able to ask a question and receive a classification response.

IBM



### Natural Language Understanding

Analyze text to extract meta-data from content such as concepts, entities, emotion, relations, sentiment and more.

Lite

IBM



### Speech to Text

Low-latency, streaming transcription

Lite

IBM



### Text to Speech

Synthesizes natural-sounding speech from text.

Lite

IBM



### Visual Recognition

Find meaning in visual content! Analyze images for scenes, objects, faces, and other content. Choose a default model.

Lite

IBM



### Watson Studio

Embed AI and machine learning into your business. Create custom models using your own data.

Lite

IBM

FEEDBACK

# Workshop Prep

[Github - http://ibm.biz/weworkibm081518](http://ibm.biz/weworkibm081518)

[IBM Cloud - http://ibm.biz/slackchatbot070918](http://ibm.biz/slackchatbot070918)

Create a Slack Workspace

# Workspace

Collection of artifacts – Intents,  
Entities and Dialogs

Build

Deploy

Improve / Metrics

IBM Watson Assistant

Workspaces / SampleWorkspace / Build

Intents Entities Dialog Content Catalog



No intents yet.

An intent is the goal or purpose of the user's input. Adding examples to intents helps your virtual assistant understand different ways in which people would say them.

Browse available intents to get started using the Content Catalog

Add intent

Browse content catalog

Import intents

The screenshot shows the IBM Watson Assistant interface for a workspace named "SampleWorkspace". The "Intents" tab is selected. A central area displays a placeholder icon for intents and the message "No intents yet.". Below this, a descriptive text explains what an intent is and how it helps the virtual assistant understand user input. At the bottom, three buttons provide options to "Add intent", "Browse content catalog", and "Import intents". On the left, a vertical sidebar lists navigation items: "Build", "Deploy", and "Improve / Metrics".

# Intents

An *intent* is the goal or purpose of the user's input. Adding examples to intents helps your virtual assistant understand different ways in which people would say them.

Provide at least five examples for each intent.

← #turn\_on

**Intent name**  
#turn\_on

**Description**  
Add a description to this intent

**Add user examples**  
Add user examples to this intent

**Add example**

User examples (693) ▾

80's music

activate the jams

activate the music

air condition on

Air conditioner on

air on

Air on please.

also turn on the lights

# Content Catalog

Intents Entities Dialog Content Catalog

Get started faster by adding existing intents from the content catalog. These intents are trained on common questions that users may ask.

Get started faster by adding existing intents from the content catalog. These intents are trained on common questions that users may ask.

| Category      | Description  | Intents |                                    |
|---------------|--|---------|------------------------------------|
| Banking       | Basic transactions for a banking use case.   | 13      | <a href="#">+ Add to workspace</a> |
| Bot Control   | Functions that allow navigation within a conversation.                               | 9       | <a href="#">+ Add to workspace</a> |
| Customer Care | Understand and assist customers with information about themselves and your business. | 18      | <a href="#">+ Add to workspace</a> |
| eCommerce     | Payment, billing, and basic management tasks for orders.                             | 14      | <a href="#">+ Add to workspace</a> |
| General       | General conversation topics most users ask.  | 10      | <a href="#">+ Add to workspace</a> |
| Insurance     | Issues related to insurance policies and claims.                                     | 12      | <a href="#">+ Add to workspace</a> |
| Telco         | Questions and issues related to a user's telephony service, device, and plan.        | 21      | <a href="#">+ Add to workspace</a> |
| Utilities     | Help a user with utility emergencies and their utility service.                      | 10      | <a href="#">+ Add to workspace</a> |

# Entities

An *entity* is a portion of the user's input that you can use to provide a different response to a particular intent. Adding values and synonyms to entities helps your virtual assistant learn and understand important details that your users mention.

Entity name  
@appliance

Value name  
Enter value

Synonyms ▾

Add synonym... +

Add value

|                          | Entity values (7) ▾ | Type     |  |
|--------------------------|---------------------|----------|--|
| <input type="checkbox"/> | AC                  | Synonyms | ac, acs, air conditioner, air conditioners, air con  |
| <input type="checkbox"/> | fan                 | Synonyms | cooler, fans   |
| <input type="checkbox"/> | heater              | Synonyms | heat, heating, heat vent, heat vents, hotter         |
| <input type="checkbox"/> | lights              | Synonyms | bulb, bulbs, foglight, fog light, foglights, fog lig |
| <input type="checkbox"/> | music               | Synonyms | itune, itunes, music station, music stations, ra     |
| <input type="checkbox"/> | volume              | Synonyms | sound, speaker, speakers, vol                        |
| <input type="checkbox"/> | wipers              | Synonyms | windscreen wiper, windscreen wipers, windsh          |

# System Entities

System entities are common entities created by IBM that could be used across any use case. They are ready to use as soon as you add them.

Intents    **Entities**    Dialog    Content Catalog

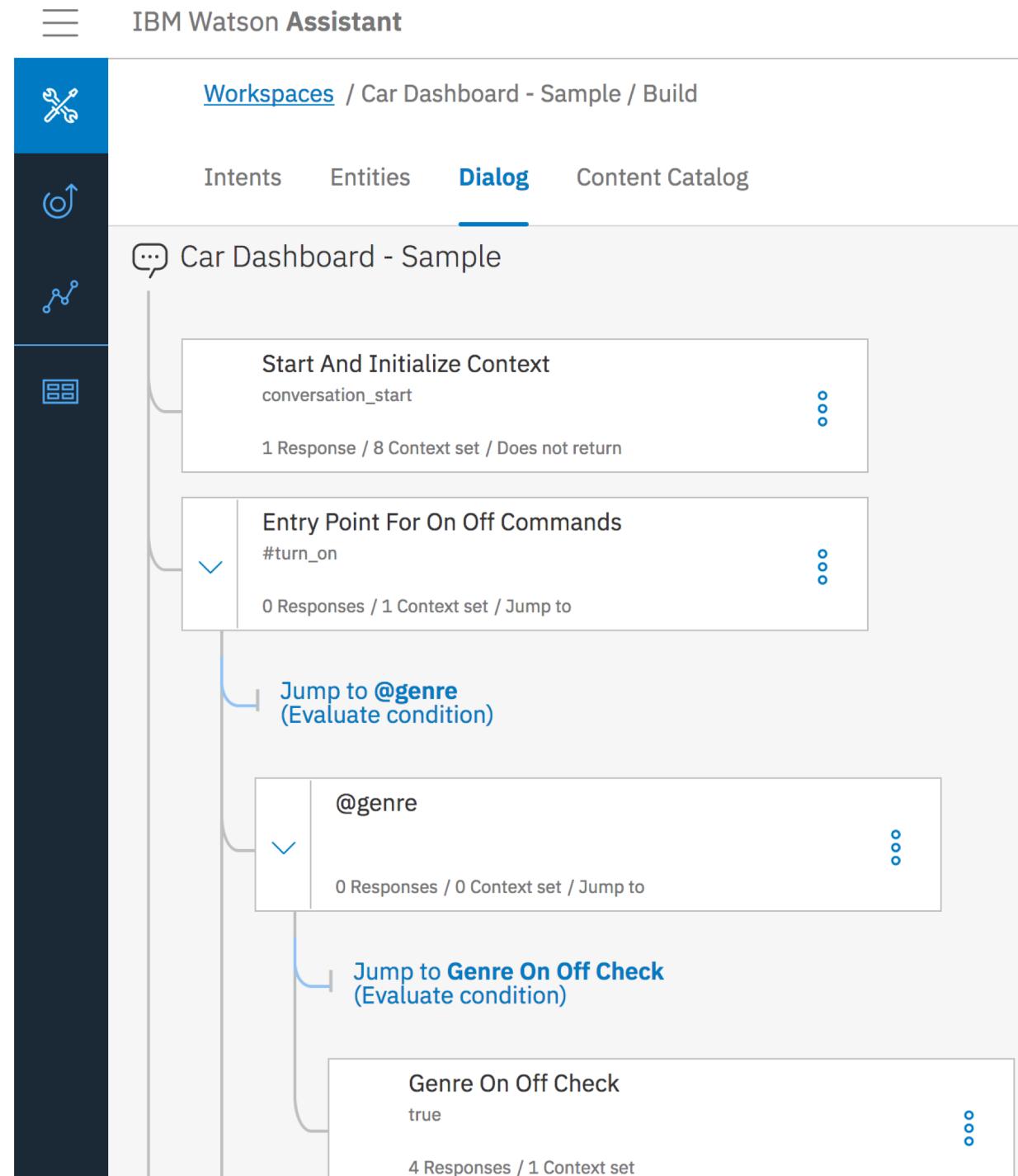
My entities    **System entities**

- > **@sys-currency** Extracts currency values from user examples including the amount and the unit. (20 cents)  Off
- > **@sys-date** Extracts date mentions (Friday)  Off
- > **@sys-location BETA** The @sys-location system entity extracts place names (country, state/province, city, town, etc.) from the user's input. (Boston)  Off
- > **@sys-number** Extracts numbers mentioned from user examples as digits or written as numbers. (21)  Off
- > **@sys-percentage** Extracts amounts from user examples including the number and the % sign. (15%)  Off
- > **@sys-person BETA** The @sys-person system entity extracts names from the user's input. (Anna)  Off
- > **@sys-time** Extracts time mentions (at 10)  Off

# Dialog

A dialog uses intents, entities, and context from your application to define a response to each user's input.

Creating a dialog defines how your virtual assistant will respond to what your users are saying.

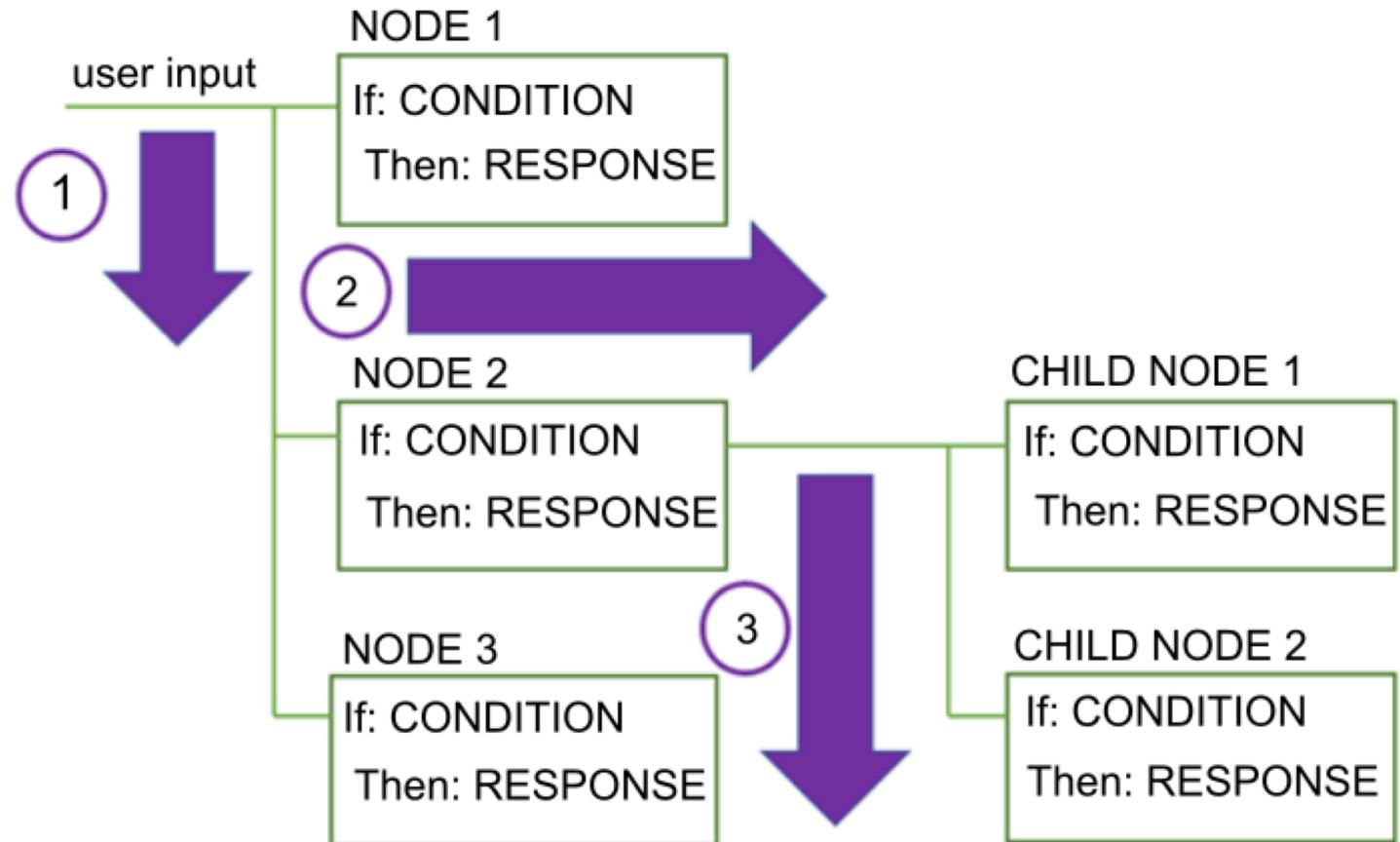


# Dialog

Dialog is processed by the service from the first node in the tree to the last.

If the service finds a condition that is met, it triggers that node.

It then moves along the triggered node to check the user input against any child node conditions.



# Slots

Add slots to a dialog node to gather multiple pieces of information from a user within that node.

Slots collect information at the users' pace. Details the user provides upfront are saved, and the service asks only for the details they do not.

## Book Reservation

Then check for:

| Check for     | Save it as | If not present, ask                  |
|---------------|------------|--------------------------------------|
| 1 @cuisine    | \$cuisine  | What type of cuisine would you like? |
| 2 @sys-date   | \$date     | What day would you like to eat?      |
| 3 @sys-time   | \$time     | What time would you like to eat?     |
| 4 @sys-number | \$number   | How many people will be there?       |

 Add slot

# Digressions

Use digressions to handle interruptions and off topic short conversations.

Digressions allow the user to change topics or ask for more information and allows them to come back to the original intent.

Example:

Make an appointment.

What times are you open?

IBM Watson Assistant

[Workspaces](#) / Customer Service - Sample / Build

Intents Entities Dialog Content Catalog

Customer Service - Sample

- Opening welcome  
1 Response / 1 Context set / Does not return
- Hours of Operation  
#Customer\_Care\_Store\_Hours  
5 Responses / 0 Context set / Returns
- > Directions and location  
#Customer\_Care\_Store\_Location  
3 Responses / 0 Context set / Skip user input
- Make an appointment  
#Customer\_Care\_Appointments  
3 Responses / 7 Context set / 5 Slots / Does not return
- > Transfer to agent  
#General\_Connect\_to\_Agent  
1 Response / 0 Context set / Does not return
- Small Talk  
3 Dialog nodes / No digressions

Customize "Make an appointment"

[Customize node](#) **Digressions**

This node has **edited** digressions settings ⓘ

▼ Digressions can go away from this node ⓘ

Allow digressions away while slot filling

Users can divert the conversation away from this node in the middle of processing slots.

Only digress from slots to nodes that allow returns

If a user goes off topic, only nodes with digressions that allow returns will be considered.

▼ Digressions can come into this node ⓘ

Allow digressions into this node

Users can digress to this node from other dialog flows.

Return after digression

After this dialog flow is processed, return to the dialog flow that was previously in progress.

Cancel Apply

# Handlers

Use handlers to

- provide responses to questions users might ask during the interaction that are tangential to the purpose of the node
- provide exit route

Example: I need to talk to an agent.

IBM Watson Assistant  
Workspaces / Customer Service - Sample / Dialog

Handlers are how your virtual assistant will respond when the user's answer to a prompt is not found. These handlers will be checked before trying the "Not found" responses in a slot.

If answer to any prompt is not found and:

| If bot recognizes | Respond with  |
|-------------------|---|
| 1 #Help           | I see you need help making an appoi   |
| 2 #Cancel         | OK. Canceling your request...         |
| 3 #Thanks         | You are welcome. Let's continue.      |

+ Add handler

Skip user input. The first child node

Provide location  
true  
1 Response / 0 Context set

Make an appointment  
#Customer\_Care\_Appointments  
3 Responses / 7 Context set / 5 Slots / Does not return

Transfer to agent  
#General\_Connect\_to\_Agent  
1 Response / 0 Context set / Does not return

Cancel Save

Manage handlers for "Make an appointment"

# Some tips before you start

- Start on paper and have a conversation with a friend (***discover intents, entities and create dialog***).
- Don't repeat yourself too many times - unless the information is absolutely necessary to meet the goal or help the end user (***slots***).
- Keep it natural - let the users digress and come back to finish the current conversation (***digressions***).
- Give the users a way out ! (***handlers***)
- Remember details about the users (***context***)
- Forget details when the user starts again (***empty out context***)
- Confirm details in a natural way “I have booked a table for two at 7pm tonight”.
- Enable Fuzzy logic – beta feature

# Deployment

1. Deploy to Slack from within Watson Assistant using the **Watson Assistant Connector**.
2. Publish to Slack with Botkit and botkit-middleware.
3. Use API with existing App.
4. Starter apps (Nodejs, Python)  
[<https://console.bluemix.net/developer/appservice/starter-kits>]

## Deploy to a Slack app

Deploy your workspace to Slack using your own Slack app. This option gives you full control over your app, but it requires you to create a Slack app, or to have access to your Slack app settings.

 Takes about 15 minutes

[Deploy to Slack app](#)



### Already deployed?

Great! That means you've allowed us to use your Cloud Functions space, and we've placed some valuable things there that you can use to make powerful customizations.

#### Latest documentation

Find the most up-to-date documentation on how to customize or extend your virtual assistant in our GitHub repository. If you do not find a solution to your problem, you can submit an issue for us to review.

[GitHub README](#)

#### Customize and manage app elements

Find out which actions were placed into your Cloud Functions space and how to customize them.

[Cloud Functions](#)

### Manage in Slack

Slack provides ways to manage your current applications and authorizations. Here are some quick links to get you there.

[Manage Slack apps](#)

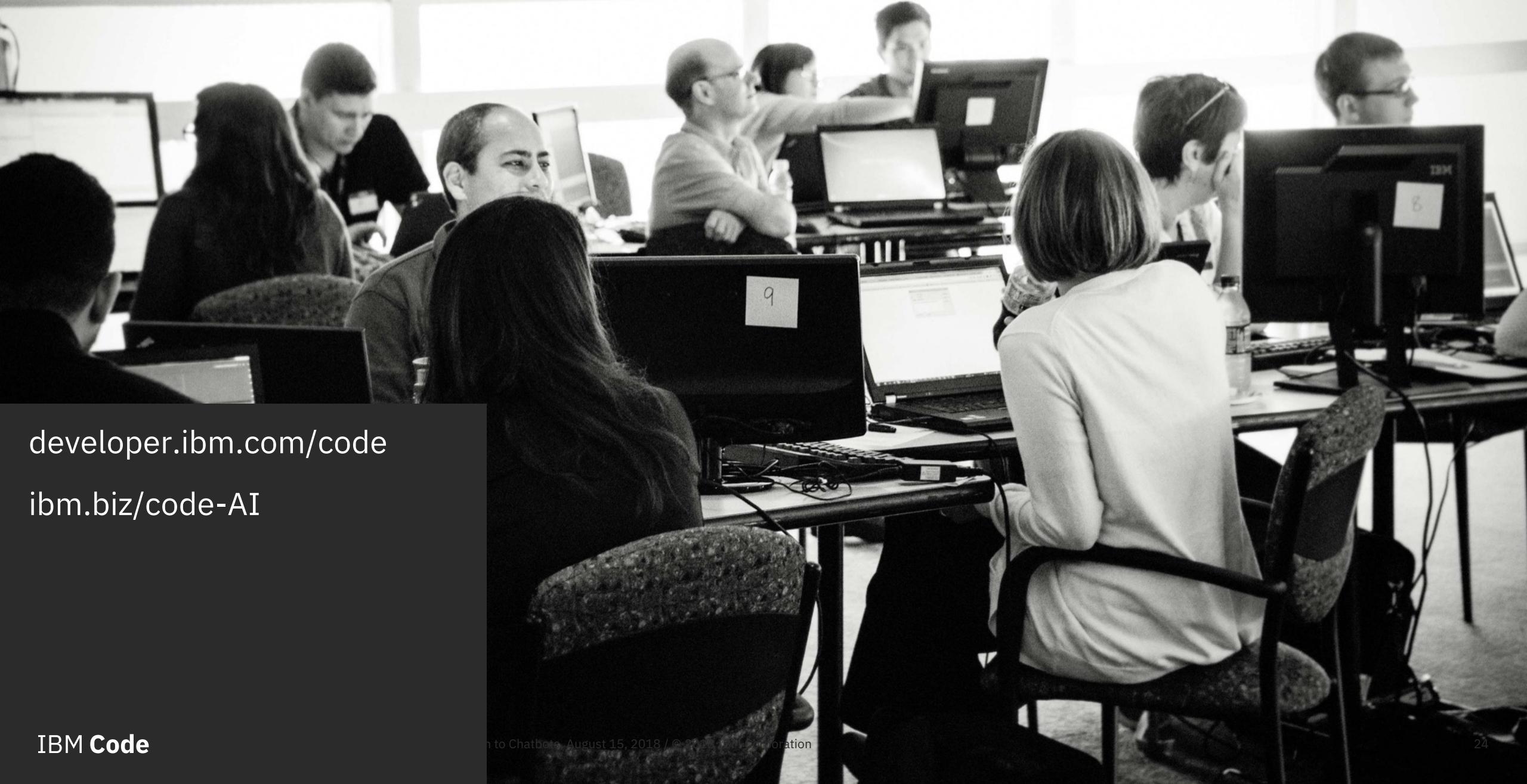
[Manage Slack authorizations](#)

# Workshop Prep

[Github - http://ibm.biz/weworkibm081518](http://ibm.biz/weworkibm081518)

[IBM Cloud - http://ibm.biz/slackchatbot070918](http://ibm.biz/slackchatbot070918)

## Create a Slack Workspace



[developer.ibm.com/code](http://developer.ibm.com/code)

[ibm.biz/code-AI](http://ibm.biz/code-AI)

Thank you

∞ ulidder@us.ibm.com

twitter.com/lidderupk

github.com/lidderupk

developer.ibm.com/code





# Watson SDK

This organization Search Pull requests Issues Marketplace Explore

@watson-dev IBM Watson APIs

A collection of REST APIs and SDKs that use cognitive computing to solve complex problems.

USA https://www.ibm.com/watson/developer/

Repositories 84 People 30

### Pinned repositories

**node-sdk**  
Node.js library to access IBM Watson services.  
TypeScript ★ 1k 516

**python-sdk**  
Client library to use the IBM Watson services in Python and available in pip as watson-developer-cloud  
Python ★ 849 509

**swift-sdk**  
The Watson Swift SDK enables developers to quickly add Watson Cognitive Computing services to their Swift applications.  
Swift ★ 724 186

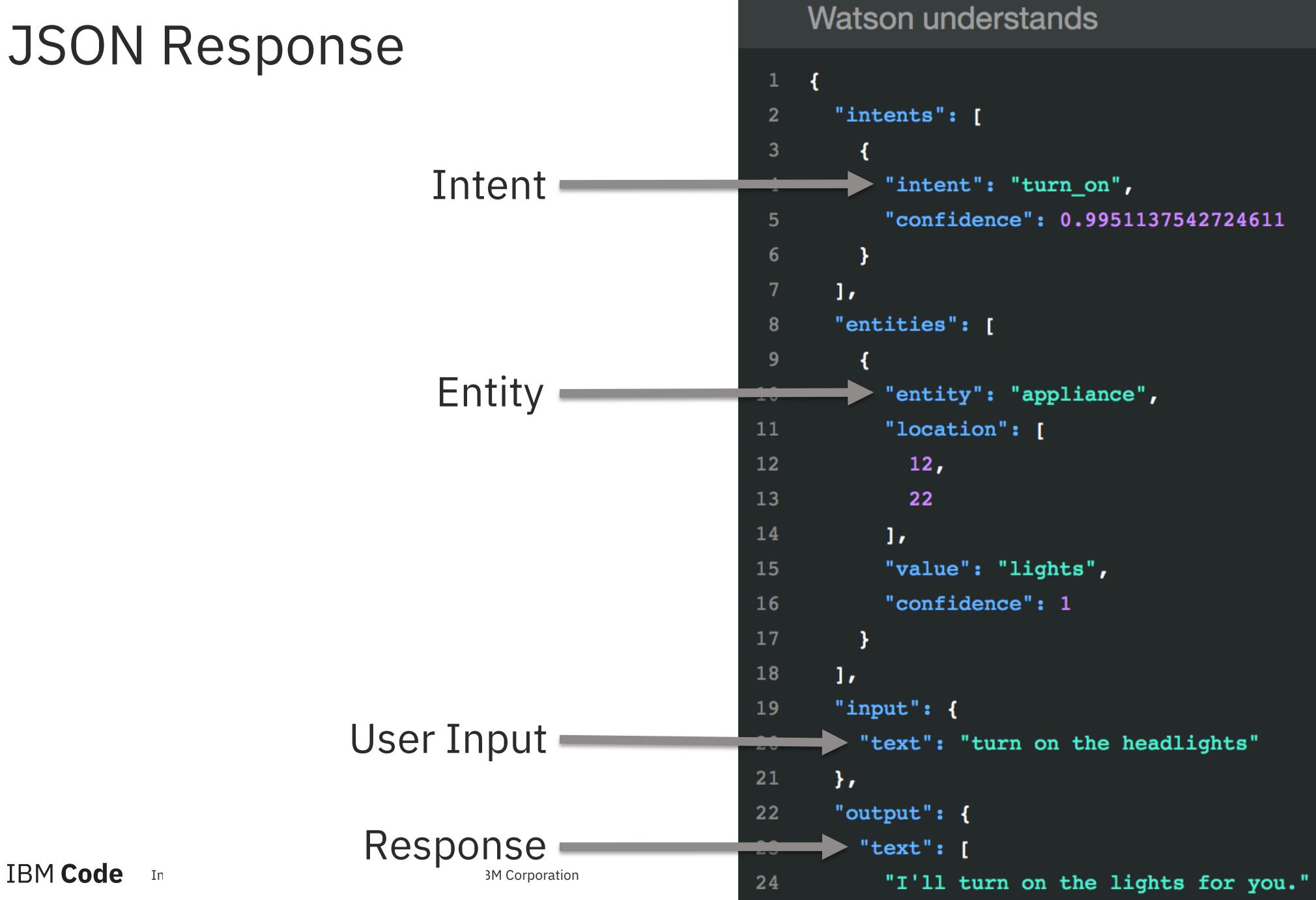
**java-sdk**  
Java SDK to use the IBM Watson services.  
Java ★ 474 456

**unity-sdk**  
Unity SDK to use the IBM Watson services.  
C# ★ 339 135

**dotnet-standard-sdk**  
.NET Standard library to access Watson Services.  
C# ★ 79 67

{...}

# JSON Response



# BotKit Middleware

The screenshot shows the GitHub repository page for 'watson-developer-cloud / botkit-middleware'. The repository has 106 commits, 2 branches, 6 releases, and 9 contributors. The latest commit was made 14 days ago. The repository description states: "A middleware to connect Watson Conversation Service to different chat channels using Botkit".

A middlewarer to connect Watson Conversation Service to different chat channels using Botkit  
<https://www.npmjs.com/package/botkit-...>

| Branch: master   | New pull request                                 | Create new file | Upload files | Find file | Clone or download                 |
|--|--|-----------------|--------------|-----------|-----------------------------------|
| germanattanasio committed on GitHub Merge pull request #82 from Naktibalda/patch-2 ... |  |                 |              |           | Latest commit cd79ce4 14 days ago |
| examples   | Merge pull request #77 from Naktibalda/patch-1   |                 |              |           | 28 days ago                       |
| lib/middleware   | updateContext ignores error of storage.users.get |                 |              |           | a month ago                       |
| test   | updateContext ignores error of storage.users.get |                 |              |           | a month ago                       |
| .gitignore   | Initial commit 🚀                                 |                 |              |           | 11 months ago                     |
| .npmignore   | ignore the test folder                           |                 |              |           | 10 months ago                     |
| .travis.yml  | add travis file                                  |                 |              |           | 10 months ago                     |
| CHANGELOG.md   | updateContext ignores error of storage.users.get |                 |              |           | a month ago                       |
| LICENSE  | rename license file                              |                 |              |           | 10 months ago                     |
| README.md  | [Readme] Fixed typo in updateContext example     |                 |              |           | 14 days ago                       |
| package-lock.json  | 1.4.2  |                 |              |           | 28 days ago                       |
| package.json   | 1.4.2  |                 |              |           | 28 days ago                       |

# IBM Bot Asset Exchange

IBM Code

Code

Content

Community

Open Source

## Bot Asset Exchange

Your community-driven chatbot development hub.

Discover, configure, deploy, and be rewarded for bots built with Watson Assistant. Join the revolution--build a chatbot today.

[Get a Bot](#)[Upload a Bot](#)[Learn More](#)

### Featured bots



Shrink - The Emotion Bot

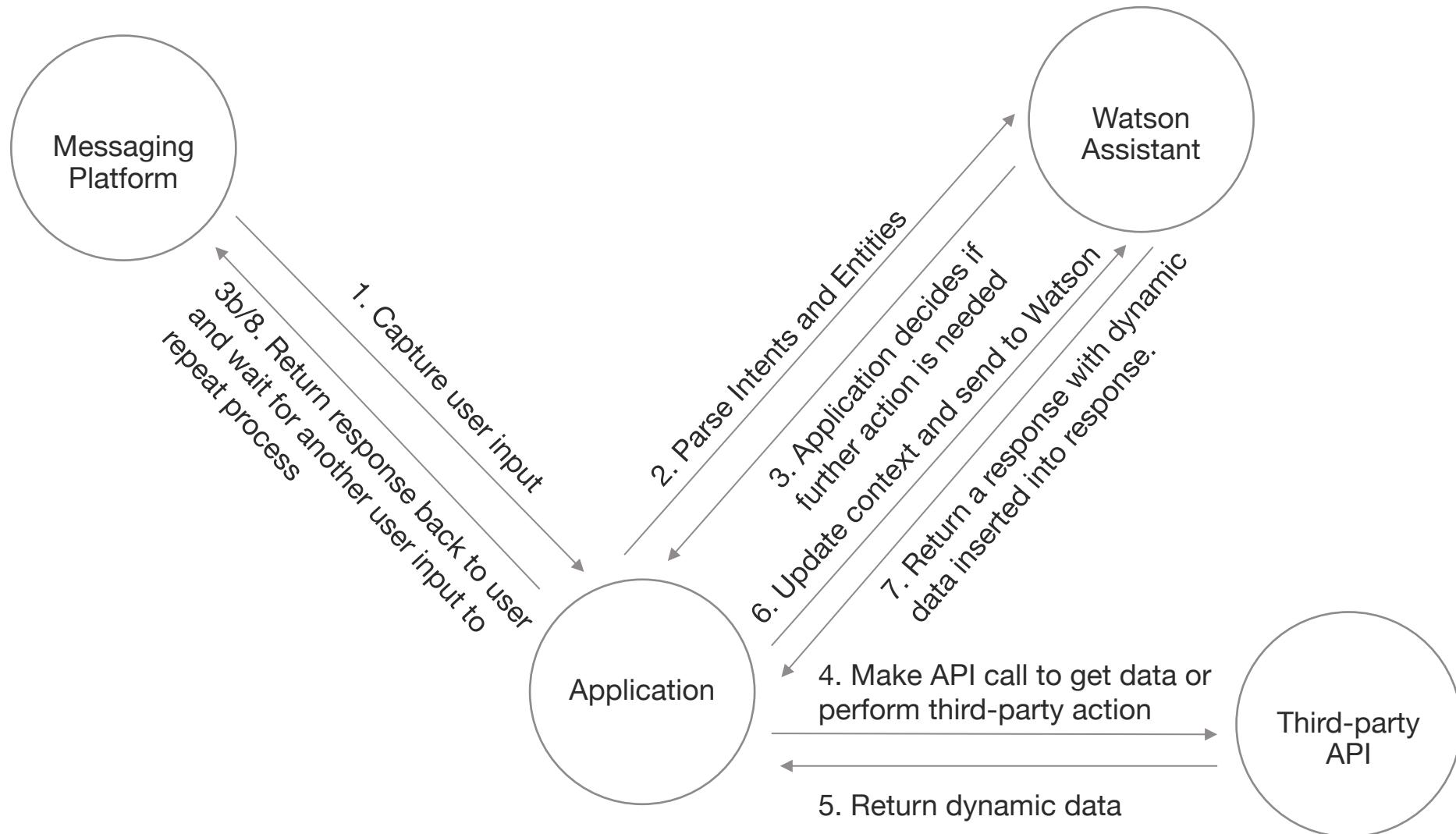


Online Store Chatbot



Simple Ranking Chatbot

# Connecting Third-Party APIs via Client



# Calling an IBM Cloud Function

Define actions that can make programmatic calls to external applications or services and get back a result as part of the processing that occurs within a dialog turn.

```
{  
  "context": {  
    "variable_name" : "variable_value"  
  },  
  "actions": [  
    {  
      "name": "<actionName>",  
      "type": "client | server",  
      "parameters": {  
        "<parameter_name>": "<parameter_value>",  
        "<parameter_name>": "<parameter_value>"  
      },  
      "result_variable": "<result_variable_name>",  
      "credentials": "<reference_to_credentials>"  
    }  
  ],  
  "output": {  
    "text": "response text"  
  }  
}
```