

DialogFlow 1

CS571: Building User Interfaces

Cole Nelson & Yuhang Zhao

Before Lecture

1. Log into Dialogflow [here](#)!
 - a. You shouldn't need to enter a credit card - we are using DialogFlow ES (free)
2. Submit your favorite joke [here](#)! Please keep it PG :)
Your email will be associated with your submission.
 - a. Selectees get their choice of candy bar!

Announcements

- No lecture on Thursday, have a happy Thanksgiving!
- Start thinking about the Final Exam...
 - Monday, December 18th at 10:05 am

Final Exam

- Monday, December 18th at 10:05 am
 - Fill out [the conflict form](#) by 12/11.
- 90 minutes for 40 MC questions.
 - 20 design and 20 implementation
- **Double-sided** notesheet (unless you brought a double-sided notesheet to the midterm...)
- Cumulative, with a heavier emphasis on content from the second-half of the semester.

HW10 Walkthrough

[Link to Canvas assignment.](#)

What will we learn today?

- What are VUIs?
- What are their potential applications?
- How can we use DialogFlow to create VUIs?
- An Introduction to DialogFlow Webhook Fulfillment

Voice User Interfaces

VUIs are a common form of **agent-based design** as opposed to **direct manipulation**.

Conversational interfaces can be used to...

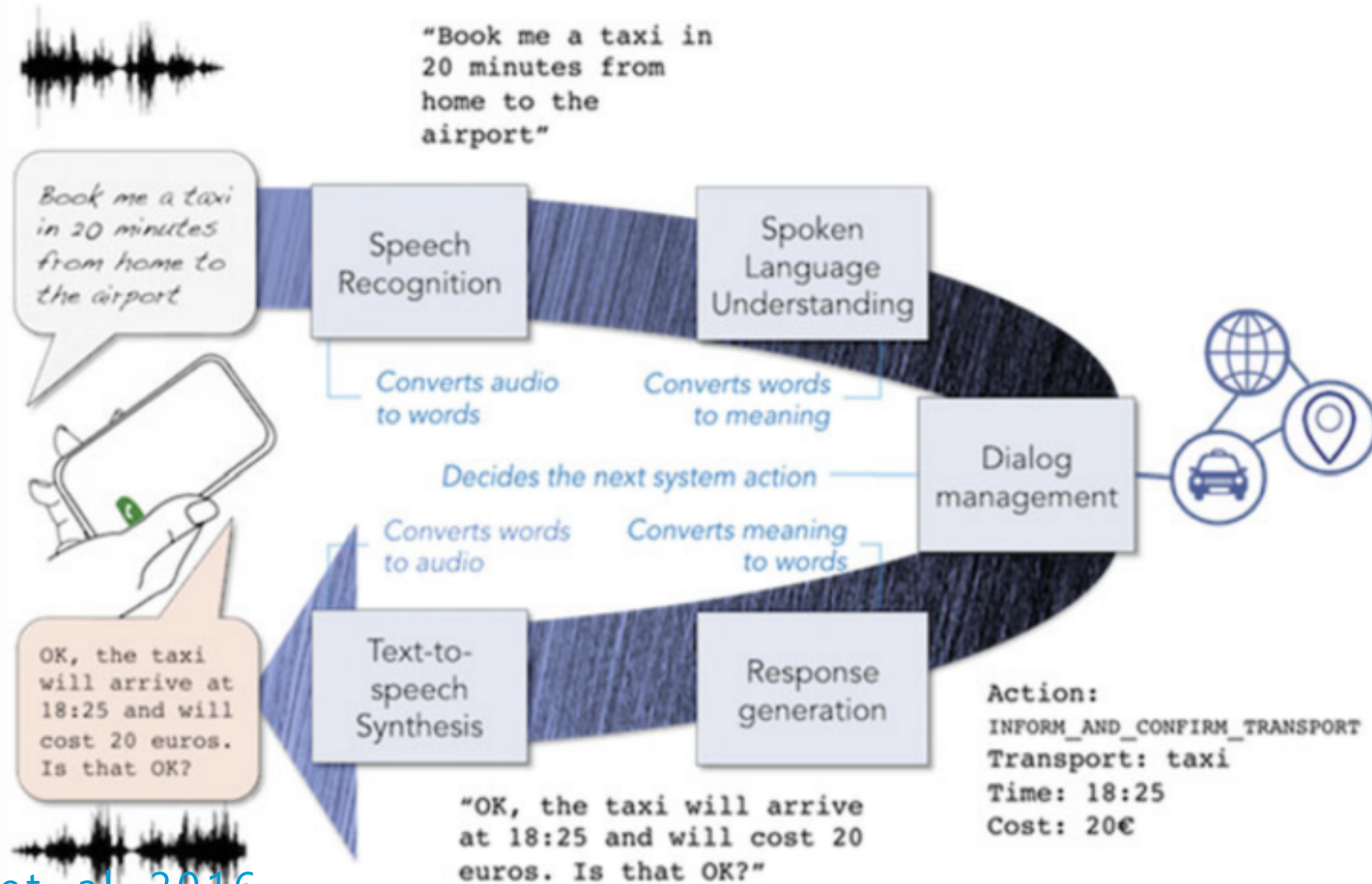
- Address accessibility needs
- Address context-specific problems (e.g. driving)
- Augment the user experience

Voice User Interfaces

VUIs integrate a number of technologies and ideas...

1. Speech recognition
2. Spoken language understanding
3. Dialog management
4. Response generation
5. Text-to-speech synthesis

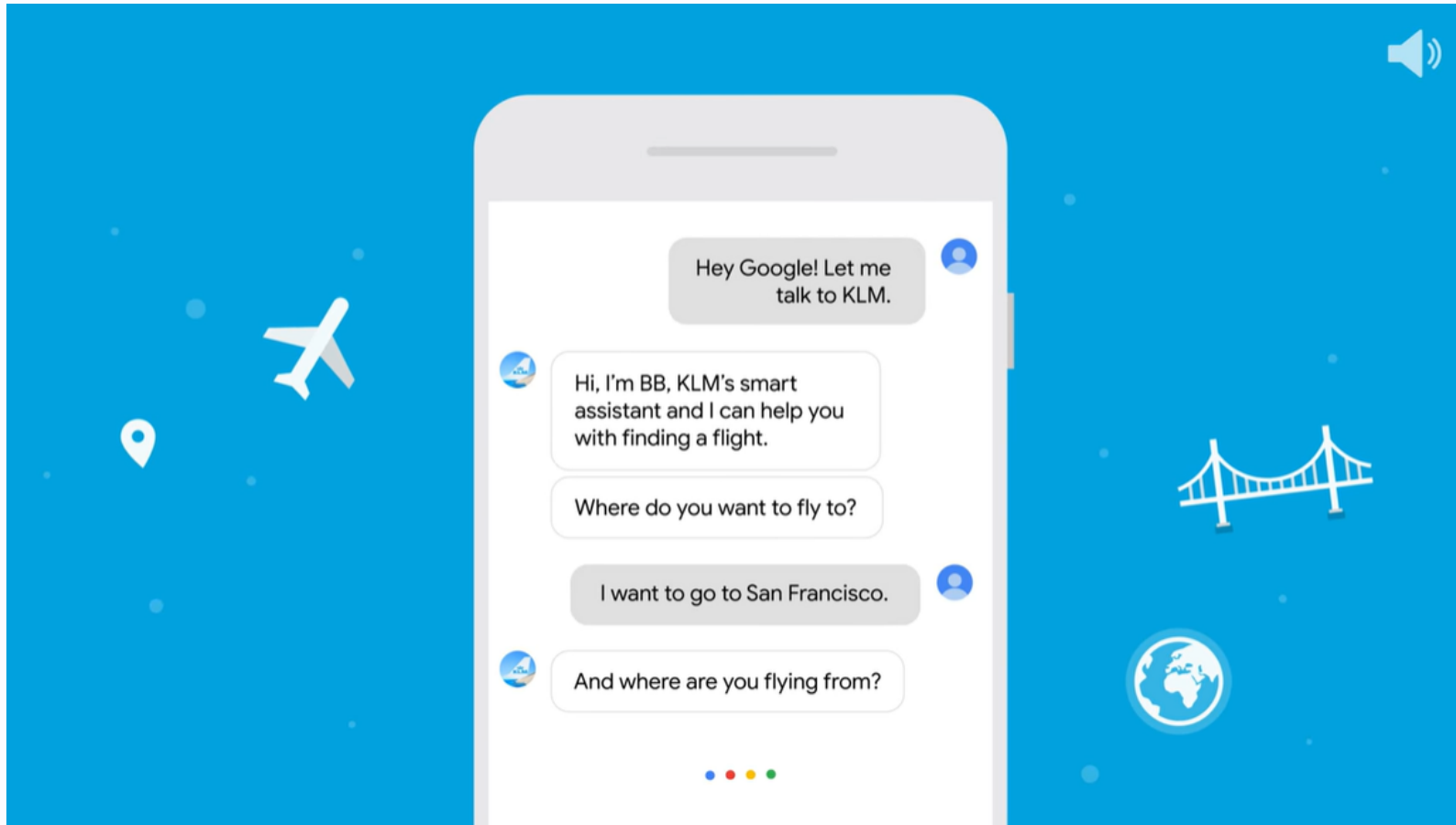
[McTear et. al. 2016](#)



McTear et. al. 2016

Potential Applications

Where can I use my Voice Agent?



Find KLM flights on the Google Assistant



Order Domino's with Alexa!

Implementation Options

We focus on just one avenue of implementation!

- [DialogFlow](#) by Google
- [Wit.ai](#) by Facebook
- [Watson Assistant](#) by IBM
- [Lex](#) by Amazon
- [Azure Bot Service](#) by Microsoft

A Consideration: [Killed by Google](#)

Key Concepts in DialogFlow

- Agents
 - Intents
 - Training Phrases
 - Parameters
 - Responses
 - Entities

Agents

Agents are high-level containers for a number of building blocks...

- Agent settings
- Intents
- Entities
- Integrations
- Fulfillment

Think of it like a project!

Intents

Consider the following user requests...

- What is the weather like today?
- Will it rain sometime today?

What is the intent of these requests?

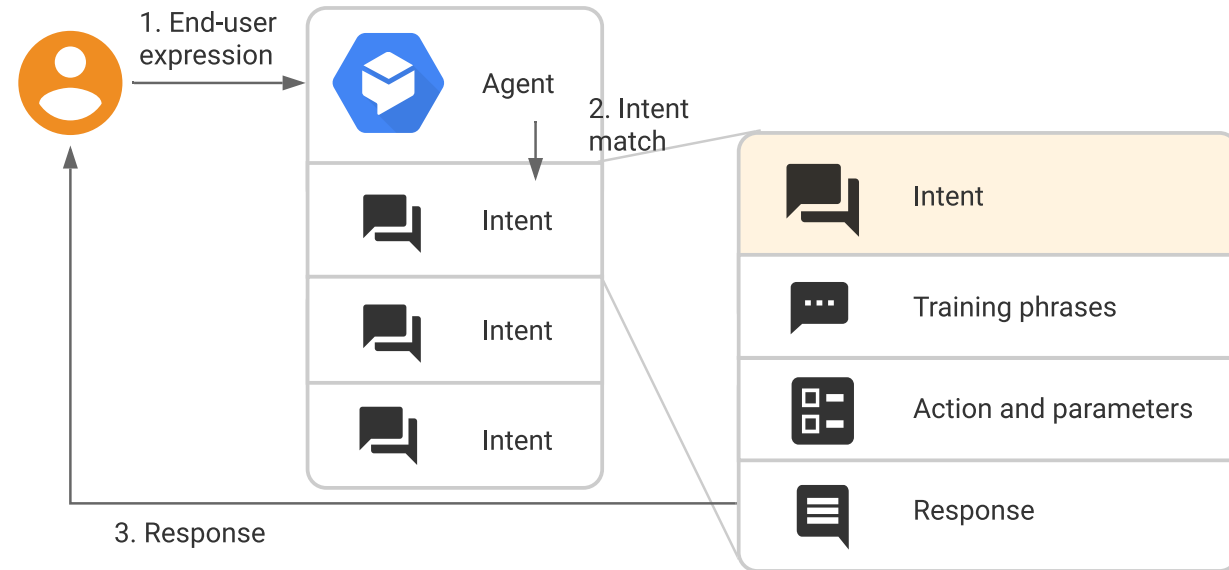
Intents

The goals of the user that are expressed to the agent.

Consists of training phrases, parameters (optional), and responses.

Image Source

CS571 Building User Interfaces | Cole Nelson & Yuhang Zhao | Lecture 21: DialogFlow 1



Intents

Training Phrases: Things the user may say to express an intent. DialogFlow recommends having many!

Parameters: Things that may vary in an expression, e.g. time, quantity, location.

Responses: How the system responds to the expression. Can include text, buttons, links, etc!

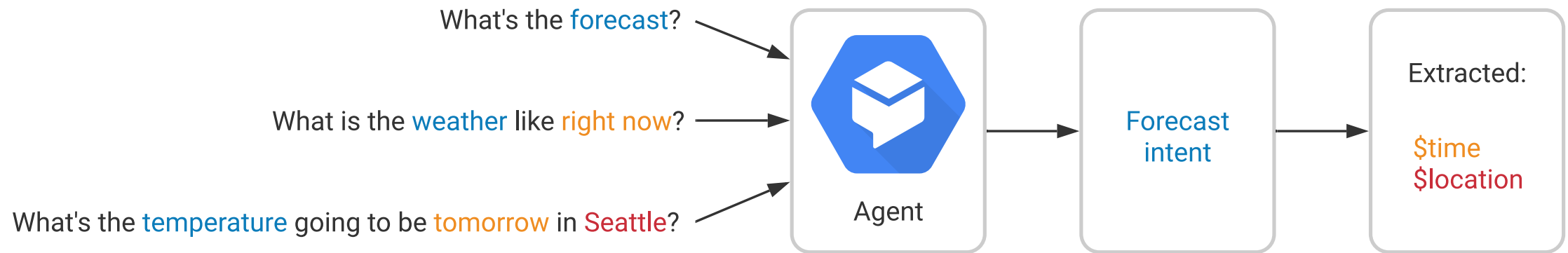


Image Source

Fallback Intent

A special type of intent that dictates how the agent responds if an expression cannot match an intent.

e.g. "Sorry, I didn't get that."

Parameters, Entity Types, and Entities

These allow for more specificity of requests without exploding the intent space.

- **Parameter:** A variable to be expressed by the user
- **Entity Type:** The type of variable expressed
- **Entity:** A specific value that can be expressed

Example

Consider the following expression...

What is the weather like today in Seattle?

- **Intent:** weather inquiry
- **Parameter** city name
- **Entity Type:** geolocation
- **Entity:** Seattle

Activity

Identify intents, parameters, and entity types of parameters in the following training phrases. Then, identify possible entities and write potential responses.

- How are you?
- How do I get to Middleton?
- What is the price of a basketball?
- Buy one box of tissues from Amazon.

Entities

Entities can have *synonyms*, e.g. green onions are also known as spring onions or scallions.

Entities can be *fuzzy-matched*, e.g. gween onion is probably misspelled green onion.

Entities can be *automatically expanded*, e.g. if saw and hammer are tools, drill likely is too.

Training phrases

Search training phrases  

” Add user expression

” Take that.

” Grab this from me.

” Take this wrench

” Pick up the hammer over there.

” How about getting that box of screws for me?

” Can you get the screwdriver for me?

” Please get the green ball

” Get the green ball

” Grab the toy

” Pick up that red cube

Let's Build an Agent!

Using DialogFlow ES for a Joke VUI

DialogFlow Advanced Concepts

- WebHook Fulfillment < Brief Intro!
- Contexts
 - Followup Intents
- Integrations
- Events

Webhook Fulfillment

Doing something "smart" with our agent.

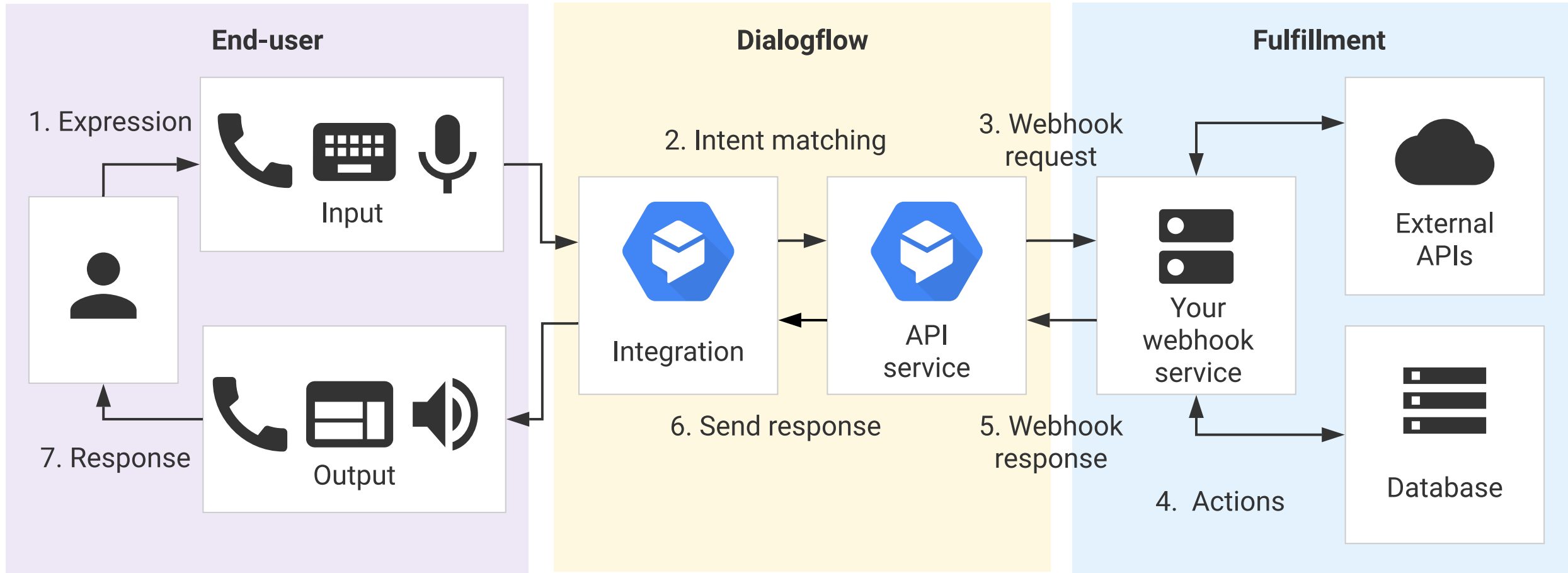


Image Source

Webhook Fulfillment

Many reasons to use webhook fulfillment...

- Used instead of giving static responses.
- Interact with backend APIs and databases.
- Provides the business logic for the agent.

Types of Webhook Fulfillment

Inline Editor (Powered by Cloud Functions for Firebase)

Build and manage fulfillment directly in Dialogflow via Cloud Functions for Firebase. [Docs](#)

index.js

package.json

Download icon

```
10
11 exports.dialogflowFirebaseFulfillment = functions.https.onRequest((request, response) => {
12   const agent = new WebhookClient({ request, response });
13   console.log('Dialogflow Request headers: ' + JSON.stringify(request.headers));
14   console.log('Dialogflow Request body: ' + JSON.stringify(request.body));
15
16   function welcome(agent) {
17     agent.add('Welcome to my agent!');
18   }
19
20   function fallback(agent) {
21     agent.add('I didn\'t understand');
22     agent.add('I\'m sorry, can you try again?');
23   }
24
```

DEPLOY

Fulfillment ?

Enable webhook call for this intent

Enable webhook call for slot filling

Webhook

ENABLED

Your web service will receive a POST request from Dialogflow in the form of the response to a user query matched by intents with webhook enabled. Be sure that your web service meets all the [webhook requirements](#) specific to the API version enabled in this agent.

| | | |
|------------|-------------------------------|----------------|
| URL* | Enter URL | |
| BASIC AUTH | Enter username | Enter password |
| HEADERS | Enter key | Enter value |
| | + Add header | |
| SMALL TALK | Disable webhook for Smalltalk | |

Creating a Webhook

Many, many, many options!

- Google Cloud Functions
- AWS Lambdas
- C# & .NET
- Java & Spring
- Python & Flask
- **JavaScript & Express**

Creating a Server/Webhook

Quick Intro to Backend Development

- Example Express Server
- S23 HW10 API
- F23 HW9 API

DialogFlow Webhook

DialogFlow makes a
POST *request* to `/`.

[Explore Docs](#)

```
{
  "responseId": "response-id ✎",
  "session": "projects/project-id ✎/agent/sessions/s",
  "queryResult": {
    "queryText": "End-user expression",
    "parameters": {
      "param-name": "param-value"
    },
    "allRequiredParamsPresent": true,
    "fulfillmentText": "Response configured for match",
    "fulfillmentMessages": [
      {
        "text": {
          "text": [
            "Response configured for matched intent"
          ]
        }
      }
    ],
    "outputContexts": [
      {
        "name": "projects/project-id ✎/agent/session",
        "lifespanCount": 5,
        "parameters": {
          "param-name": "param-value"
        }
      }
    ]
  }
}
```

Next Class

We'll make a "webhook" on our machine using `expressjs` and `ngrok`.

What did we learn today?

- What are VUIs?
- What are their potential applications?
- How can we use DialogFlow to create VUIs?
- An Introduction to DialogFlow Webhook Fulfillment

Questions?