



Microsoft Bot Framework

Stefan Volkmer

orange networks

IT-CONSULTING - IT-DEVELOPMENT



Stefan Volkmer

Software Architekt und Engineer

Schwerpunkt Azure und AI

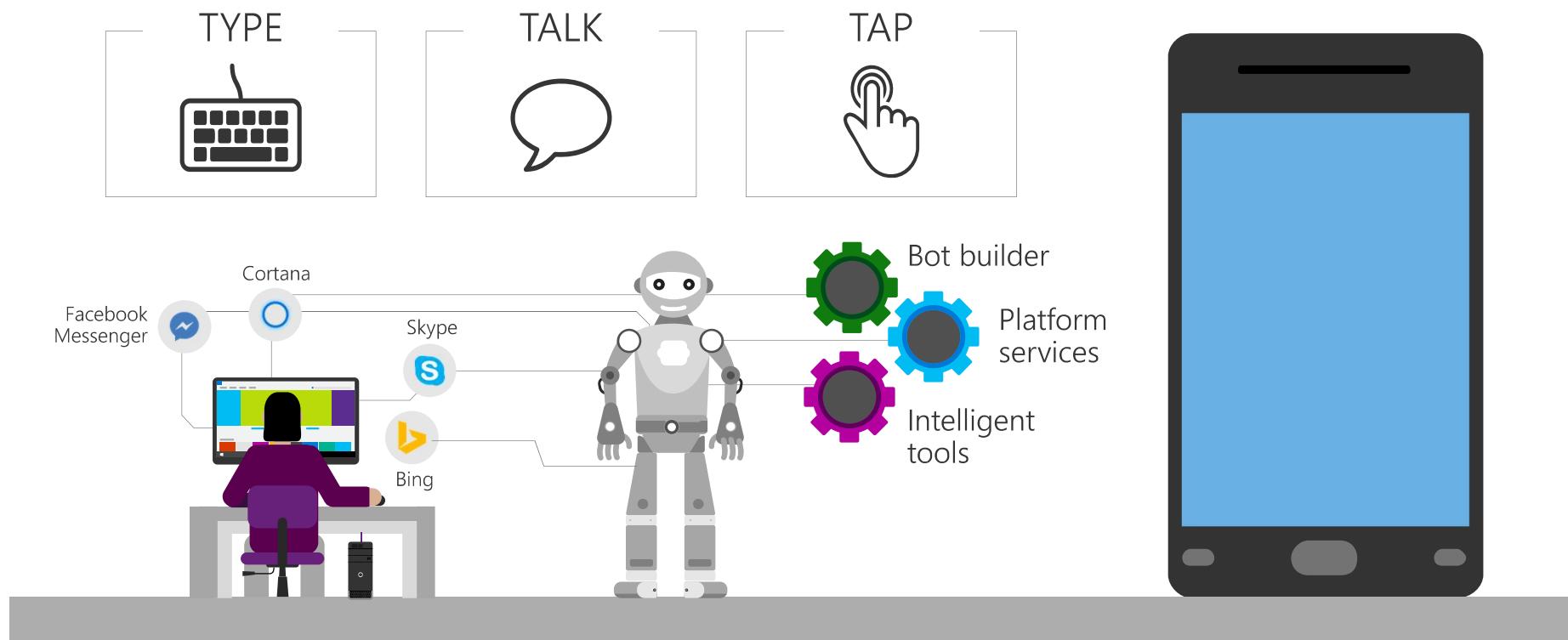
s.volkmmer@orangenet.de



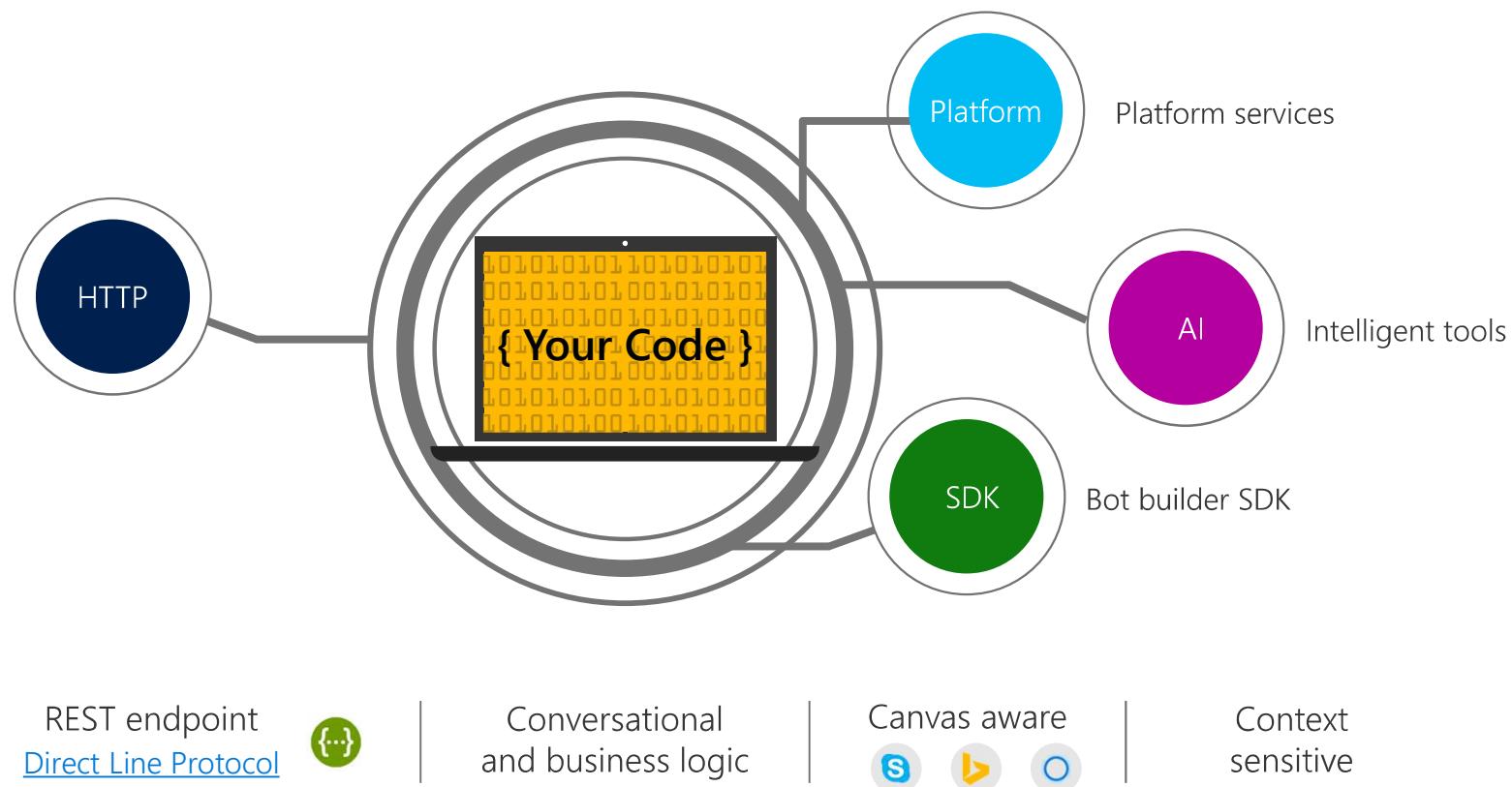
Agenda

- Organizational
- Bot Overview
- Installing the Tools
- Creating your first Bot
- Cognitive Services Overview
- Language Understanding Intelligent Service (LUIS.ai)
- QnA Maker
- Dialog Stack
- Adaptive Cards
- Authentication

Why a bot?



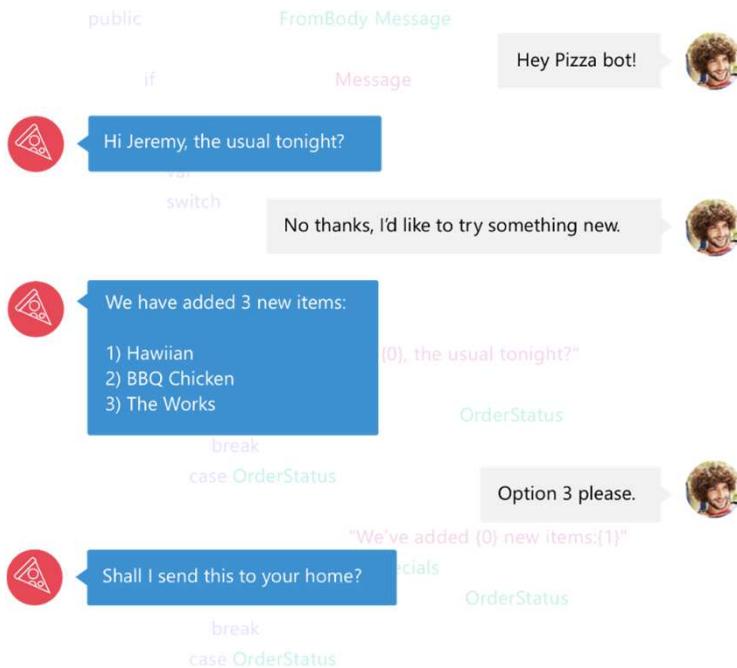
What is a bot?



- What Is a Bot?
 - A bot is an app that users interact with the user in a conversational way
 - Bots can communicate conversationally with text, cards, or speech.
 - A bot may be as simple as basic pattern matching with a response, or it may be a sophisticated weaving of artificial intelligence techniques with complex conversational state tracking and integration to existing business services.
- What is a Bot NOT?
 - AI
 - Just a chat (voice or text)

Bot sample

- Bot Framework is a Microsoft-operated service and an SDK.
- Bot Framework is one of many tools Microsoft offers for building a complete bot.
- Others include: LUIS, Speech APIs, Azure, more



Kinds of bots

1,000+ companies engaging us

Scenario	Retail	Finance	Insurance	Telecoms	Government	Automotive	Manufacturing	Healthcare	Media	Events
Customer service	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Customer retail	✓	✓	✓	✓				✓		
Audio/speech analysis	✓	✓	✓	✓	✓				✓	
Translation		✓	✓							
Surveillance		✓			✓					
Knowledge extraction		✓	✓	✓				✓		
Video/photo analysis		✓			✓				✓	
Product identification	✓						✓	✓	✓	
Digital assistant						✓				
Footfall analysis	✓						✓			✓
HD maps and object detection							✓			

What is the bot framework?

What?

- Tools for building REST websites
- Services to enrich
- Mechanisms for receive events
- Data to debug and tools to analyze

Why?

- Implements standard protocols
- Modeling conversations is hard; tools help!
- UI across multiple canvases is hard; cards rock!
- Language understanding is hard
- Common and well understood patterns

Goals

- Start simple; add complexity; no dead-ends
- Bot adapts to the user, based on context
- Composable and intelligent controls to manage complexity

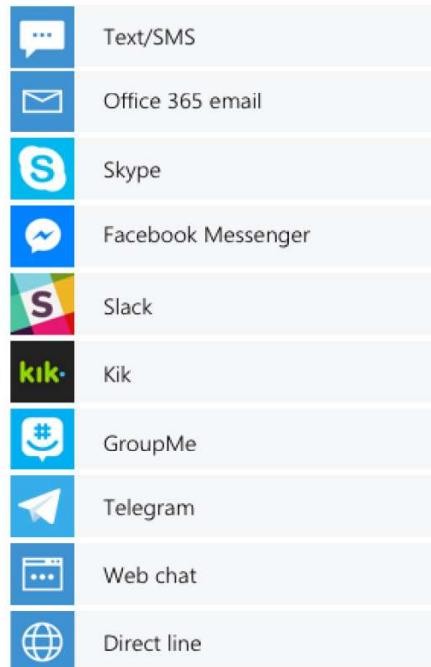


New channels

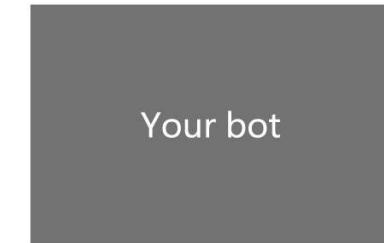
I M P R O V E M E N T S E V E R Y W H E R E !



Bot Framework Value Proposition



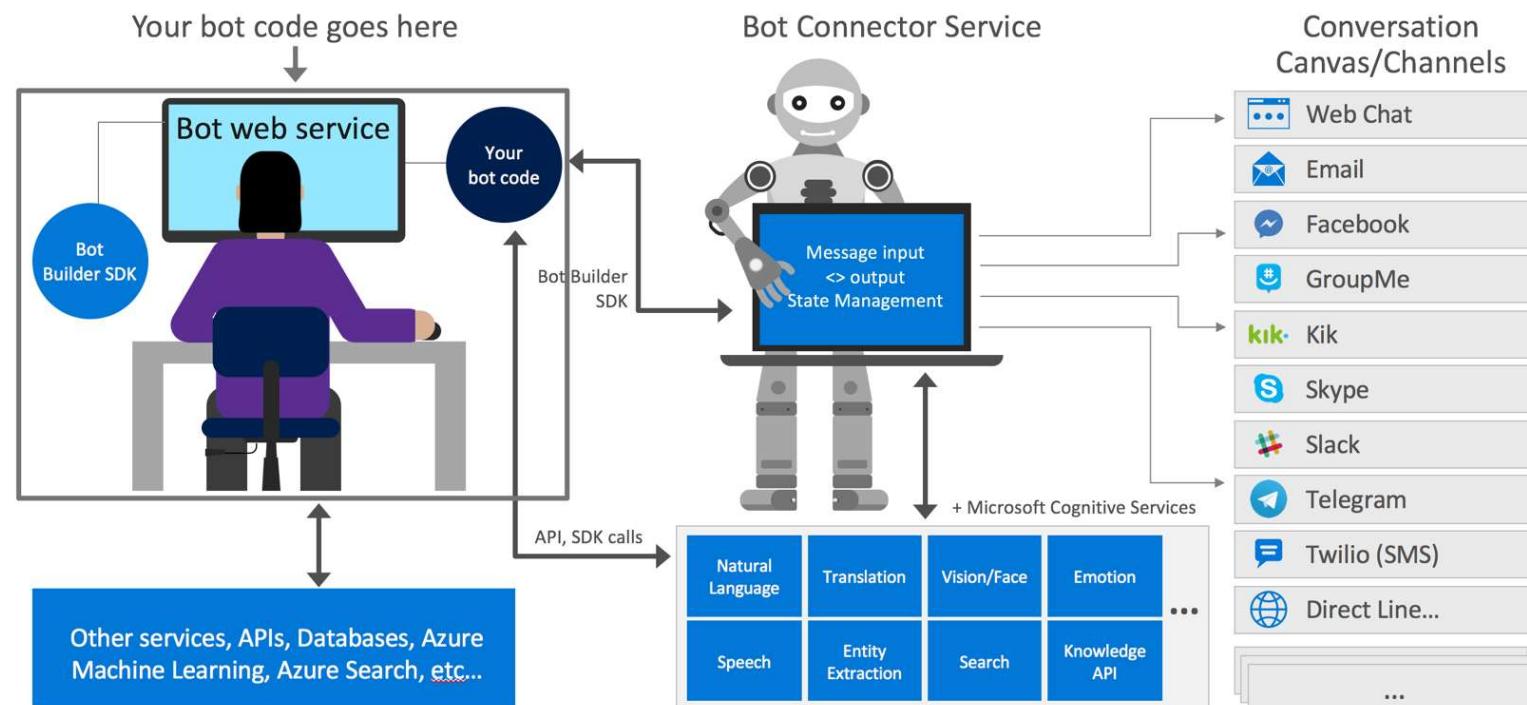
- Natural language understanding
- Routes messages
- Manages user context
- Conversation tracking
- Services (translation to 30+ languages...)



Bot Framework

Channels

Building Bots



Bot Builder

- Tools and services to build great bots that converse wherever the users are
- Open source SDK on Github for Node.js, .NET and REST
- Support for rich attachments (image, card, video, doc, etc.)
- Support for calling (Skype)
- Online/offline chat Emulator

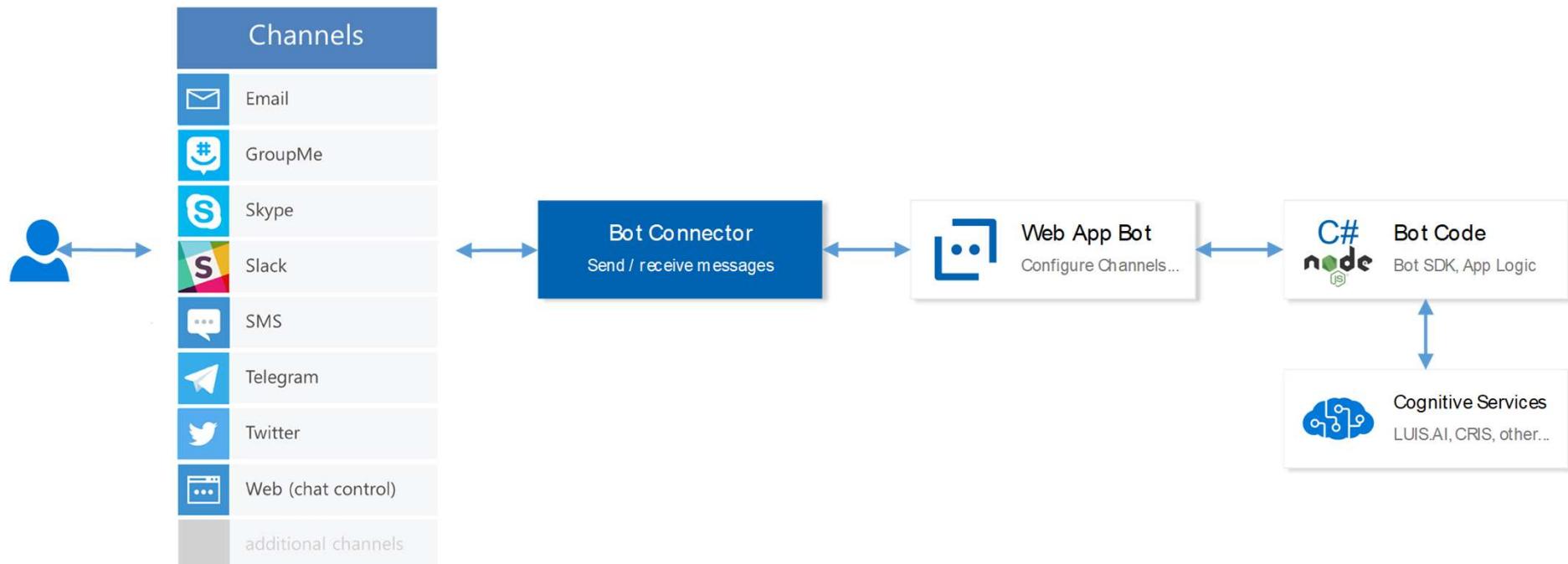
Overview

- Connector Service
 - The Bot Connector service enables the bot to exchange messages with channels that are configured in the Bot Framework Portal, by using industry-standard REST and JSON over HTTPS
- Activity
 - A specific event that occurs between a Bot and Users, such as an actual message, or conversation notification
- Messages
 - A Message is a communication between a Bot and Users, such as a User asking a question, or a Bot responding with a reply

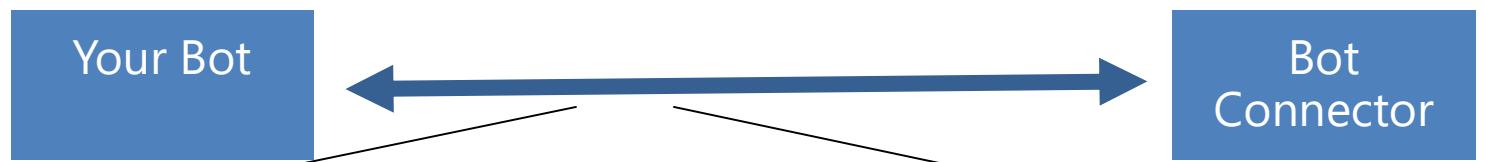
Types of Activities

Activity Type	Description
Message	Sent when general content is passed to or from a user and a bot
Conversation Update	Sent when the conversation's properties change, for example the topic name, or when user joins or leaves the group
Contact Relation Update	Sent when bot added or removed to contact list
Delete User Data	Send when user is removed from a conversation
Typing	Sent when a user is typing
Ping	Send when a keep-alive is needed

Bot Connector



Bot Connector messages



```
{  
  "type": "Message",  
  "id": "68YrxgtB53Y",  
  "conversationId": "DphPaFQrDuZDKyCez4AFGcT4vy5aQDje1lLGIjB8v18MFtb",  
  "language": "en",  
  "text": "You can say \"/order\" to order!",  
  "attachments": [ ],  
  "from": {  
    "name": "+12065551212",  
    "channelId": "sms",  
    "address": "+12065551212",  
    "id": "Ro52hKN287",  
    "isBot": false  
  },  
  "channelData": { SMS data here },  
  "botUserData": { your data here },  
  ...  
}
```



Azure Bot Service



Accelerate development cycles

Enrich your bots

Boost operational efficiencies



Azure Bot Service

Accelerate development cycles

nodeJS

C#

Develop
your way



Quick start
templates



Built in
code editor

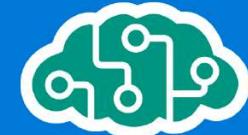


Integrated
chat window

Enrich your bots



Channel
support



Cognitive
Services



Direct
Line support



Embedded
web chat

Boost operational
efficiencies



Powered by
Azure Functions



Continuous
deployment



Scale
on demand



Reduced
dev ops

Installing the Tools

- Bot Builder SDK (for .NET)
- Bot Emulator
- Github
- Bot Service – Azure Web App Bot

RM1 Attendees could install all of the tools (i.e. setup their environments). Start the project and communicate with the bot using the Emulator.

--

This would have the benefit of making the explanation in the next section clear. i.e. what is a Activity / Message.

Rinor Maloku; 12.03.2018

RM25

Demo

Creating a Bot
with Azure Bot
Service



RM25

During Bot Emulator demo we can tell them that there are these hotkeys:

- F5 - Quick conversation refresh ("you will need it" ;P)
- F6 focus the addressbar.
- F10 focus the menu.

and then adding "the only reason we know about these is that one of our devs contributed them". - framing it as a "Fun Fact". Also should have the effect of taking higher Authority upon the talked subject.

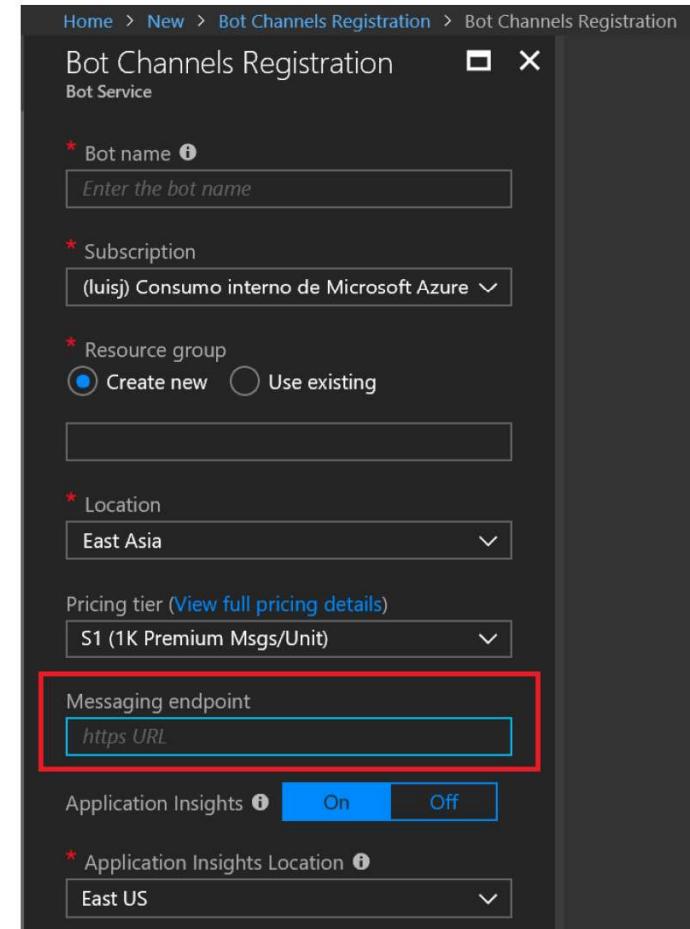
Why so? [When I am listening to someone explaining something I am constantly trying to discredit him [mentally looking for flaws], prolly all devs do the same, this way we get them from the frame of mind of evaluating us to the frame that "These guys are authority on the subject" and don't have anything to prove. If the Demo is to simplistic they miscalculated us, if it is too complex then we are too stupid]

Rinor Maloku; 13.03.2018

Register a Bot

Register a Bot – Portal

- Enable alternative hosting
- Simple process
- Basic Information
- Endpoint
 - Can be changed later
- Microsoft App Id
- Analytics with Application Insights



Deploy to Azure

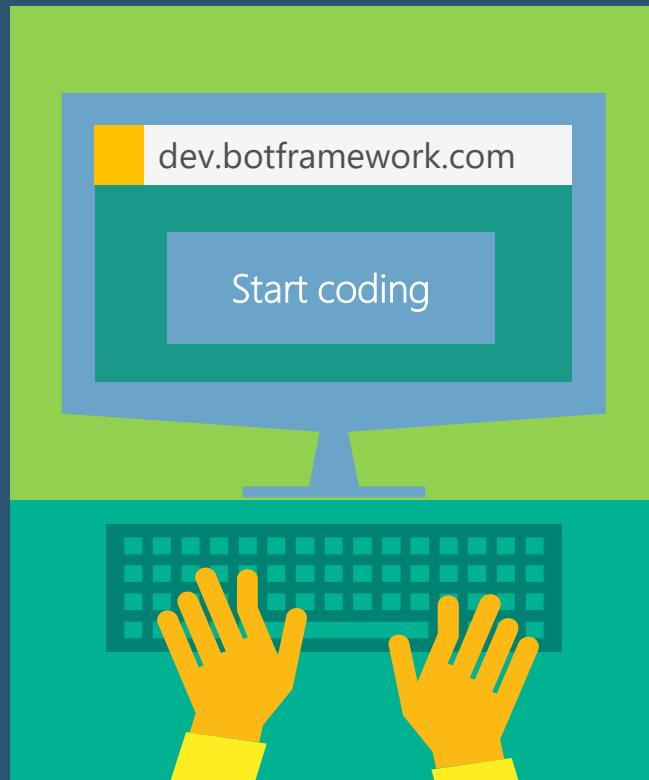
Deploy to Azure

- A Bot is a web application
- There are some options to build a Bot and deploy to Azure
- Bot Service can handle automatically CI/CD scenarios
 - The integration uses App Service Kudu Deployment Engine
- Can be directly from Visual Studio
 - Build and publish
- Or from Visual Studio Team Services (VSTS)
 - Configure Build & Release

RM25

Demo

How to deploy to
Azure



RM25

During Bot Emulator demo we can tell them that there are these hotkeys:

- F5 - Quick conversation refresh ("you will need it" ;P)
- F6 focus the addressbar.
- F10 focus the menu.

and then adding "the only reason we know about these is that one of our devs contributed them". - framing it as a "Fun Fact". Also should have the effect of taking higher Authority upon the talked subject.

Why so? [When I am listening to someone explaining something I am constantly trying to discredit him [mentally looking for flaws], prolly all devs do the same, this way we get them from the frame of mind of evaluating us to the frame that "These guys are authority on the subject" and don't have anything to prove. If the Demo is to simplistic they miscalculated us, if it is too complex then we are too stupid]

Rinor Maloku; 13.03.2018

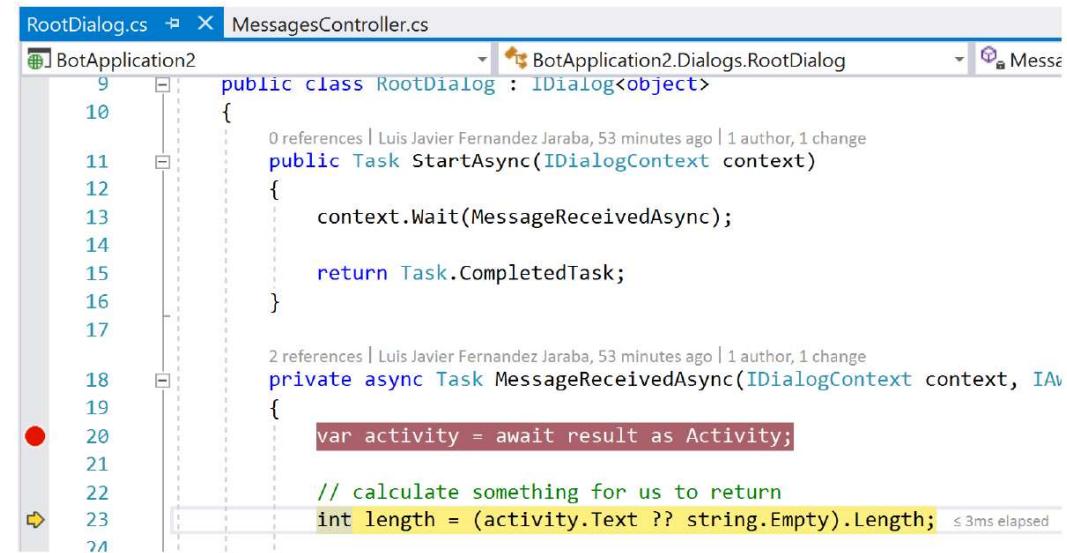
Test and Debug

Using the Bot Framework Emulator

- Is a desktop application that allows bot developers to test and debug their bots, either locally or remotely
 - If you are behind a firewall or other network boundary and want to connect to a bot that is hosted remotely, you must install and configure **ngrok** tunneling software
 - Inspect the messages that your bot sends and receives
 - You can test your bot using the emulator even if you have not yet registered it with the Bot Framework or configured it to run on any channels.

Debug a bot with Visual Studio

- Visual Studio has built-in debugging support
- IIS Express + Bot Emulator is an easy way to test and debug locally
- Additionally, you can debug Azure deployed bots



```
RootDialog.cs  X  MessagesController.cs
BotApplication2  BotApplication2.Dialogs.RootDialog  MessagesController.cs
public class RootDialog : IDialog<object>
{
    public Task StartAsync(IDialogContext context)
    {
        context.Wait(MessageReceivedAsync);

        return Task.CompletedTask;
    }

    private async Task MessageReceivedAsync(IDialogContext context, IActivity activity)
    {
        var activity = await result as Activity;

        // calculate something for us to return
        int length = (activity.Text ?? string.Empty).Length; < 3ms elapsed
    }
}
```

RM25

Demo

Debug bots with
the Bot
Framework



RM25

During Bot Emulator demo we can tell them that there are these hotkeys:

- F5 - Quick conversation refresh ("you will need it" ;P)
- F6 focus the addressbar.
- F10 focus the menu.

and then adding "the only reason we know about these is that one of our devs contributed them". - framing it as a "Fun Fact". Also should have the effect of taking higher Authority upon the talked subject.

Why so? [When I am listening to someone explaining something I am constantly trying to discredit him [mentally looking for flaws], prolly all devs do the same, this way we get them from the frame of mind of evaluating us to the frame that "These guys are authority on the subject" and don't have anything to prove. If the Demo is to simplistic they miscalculated us, if it is too complex then we are too stupid]

Rinor Maloku; 13.03.2018

RM25

Lab

Lab 1 - Creating a simple Bot



RM25

During Bot Emulator demo we can tell them that there are these hotkeys:

- F5 - Quick conversation refresh ("you will need it" ;P)
- F6 focus the addressbar.
- F10 focus the menu.

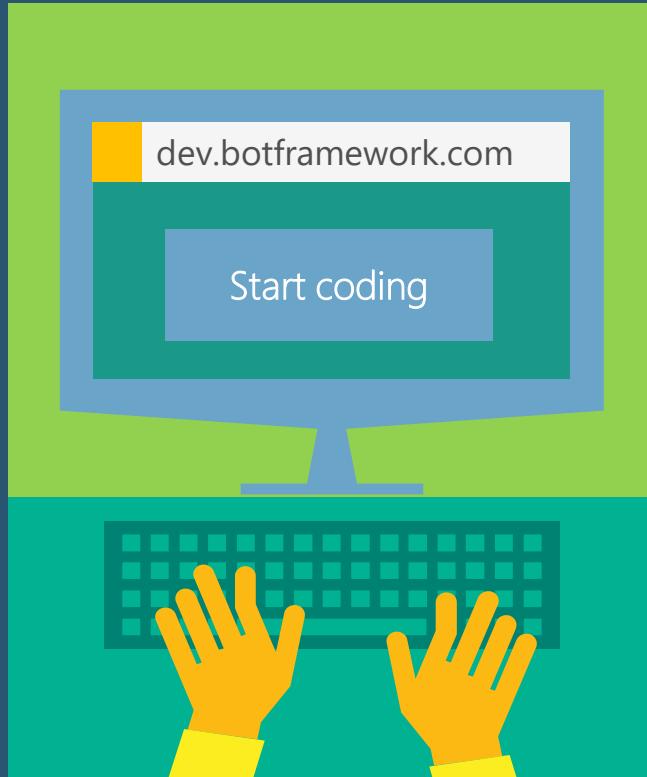
and then adding "the only reason we know about these is that one of our devs contributed them". - framing it as a "Fun Fact". Also should have the effect of taking higher Authority upon the talked subject.

Why so? [When I am listening to someone explaining something I am constantly trying to discredit him [mentally looking for flaws], prolly all devs do the same, this way we get them from the frame of mind of evaluating us to the frame that "These guys are authority on the subject" and don't have anything to prove. If the Demo is to simplistic they miscalculated us, if it is too complex then we are too stupid]

Rinor Maloku; 13.03.2018

Lab

Lab 2 - Creating a LUIS Bot using a template



RM25

During Bot Emulator demo we can tell them that there are these hotkeys:

- F5 - Quick conversation refresh ("you will need it" ;P)
- F6 focus the addressbar.
- F10 focus the menu.

and then adding "the only reason we know about these is that one of our devs contributed them". - framing it as a "Fun Fact". Also should have the effect of taking higher Authority upon the talked subject.

Why so? [When I am listening to someone explaining something I am constantly trying to discredit him [mentally looking for flaws], prolly all devs do the same, this way we get them from the frame of mind of evaluating us to the frame that "These guys are authority on the subject" and don't have anything to prove. If the Demo is to simplistic they miscalculated us, if it is too complex then we are too stupid]

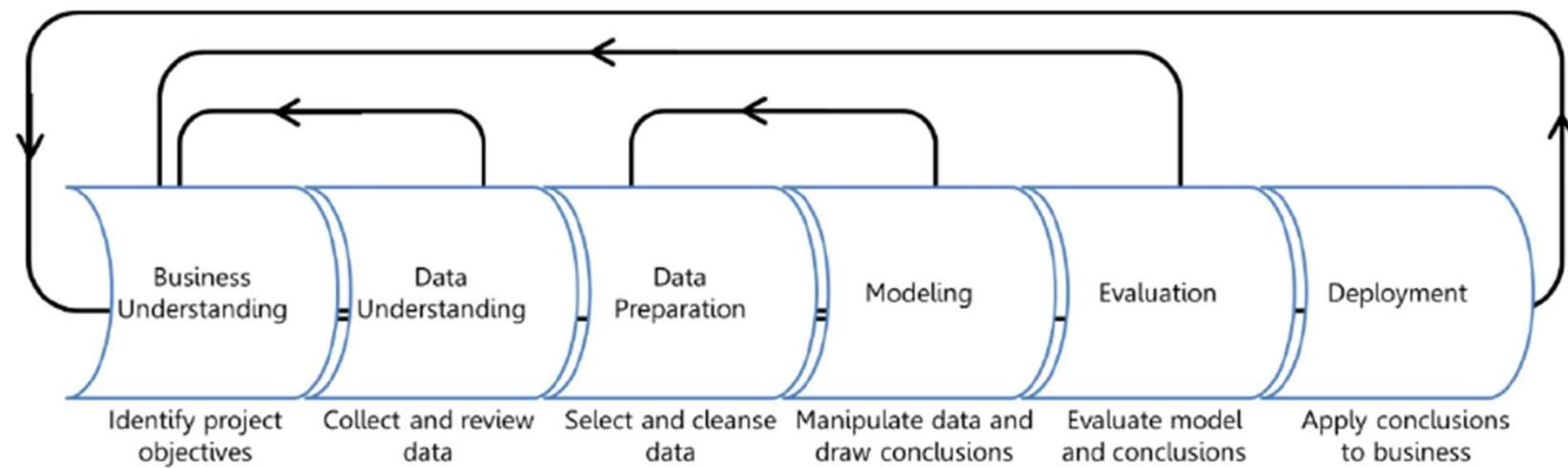
Rinor Maloku; 13.03.2018



Introducing Microsoft Cognitive Services

Stefan Volkmer

Machine Learning Steps



Why Microsoft Cognitive Services ?



Easy

Roll your own with REST APIs

Simple to add: just a few lines of code required

Flexible

Make the same API code call on iOS, Android, and Windows

Integrate into the language and platform of your choice

Tested

Built by experts in their field from Microsoft Research, Bing, and Azure Machine Learning

Quality documentation, sample code, and community support



Microsoft Cognitive Services

Give your apps a human side

Cognitive Services



Vision

From faces to feelings, allow your apps to understand images and video



Speech

Hear and speak to your users by filtering noise, identifying speakers, and understanding intent



Language

Process text and learn how to recognize what users want



Knowledge

Tap into rich knowledge amassed from the web, academia, or your own data



Search

Access billions of web pages, images, videos, and news with the power of Bing APIs

Demo

Cognitive Services



Language Understanding Intelligent Service (LUIS.ai)

Introduction & Overview

- Overview
- What is LUIS.AI?
 - Language Understanding Intelligent Service (LUIS) allows your application to understand what a person wants in their own words.
 - LUIS uses machine learning to allow developers to build applications that can receive user input in natural language and extract meaning from it.
 - A client application that converses with the user can pass user input to a LUIS app and receive relevant, detailed information back.

Why LUIS.AI



It is fast and easy

LUIS is designed to enable you to quickly deploy an HTTP endpoint that will take the sentences you send it and interpret them in terms of the intention they convey and the key entities that are present.



It learns and adapts

After your endpoint has processed a few dozen interactions, LUIS begins active learning. LUIS examines all the utterances that have been sent to it, and calls to your attention the ones that it would like you to label.



It offers pre-built domains

Prebuilt domains jump-start your application by providing intents and entities that can be mixed in and modified to create better language understanding with less effort. We include domains for things like Calendar, Music, Devices and many more.



It is a power developer tool

The over all experience of LUIS focuses on boosting developers' productivity though providing a set of powerful tools, offered through a simple user-experience & comprehensive set of APIs.



It is MultiLingual

LUIS is a multi-lingual platform and can be trained to understand over 10 different languages.



Key Concepts

01

Intents – Intents are like verbs in a sentence, it represents actions the user want to perform such as booking a flight, paying a bill

02

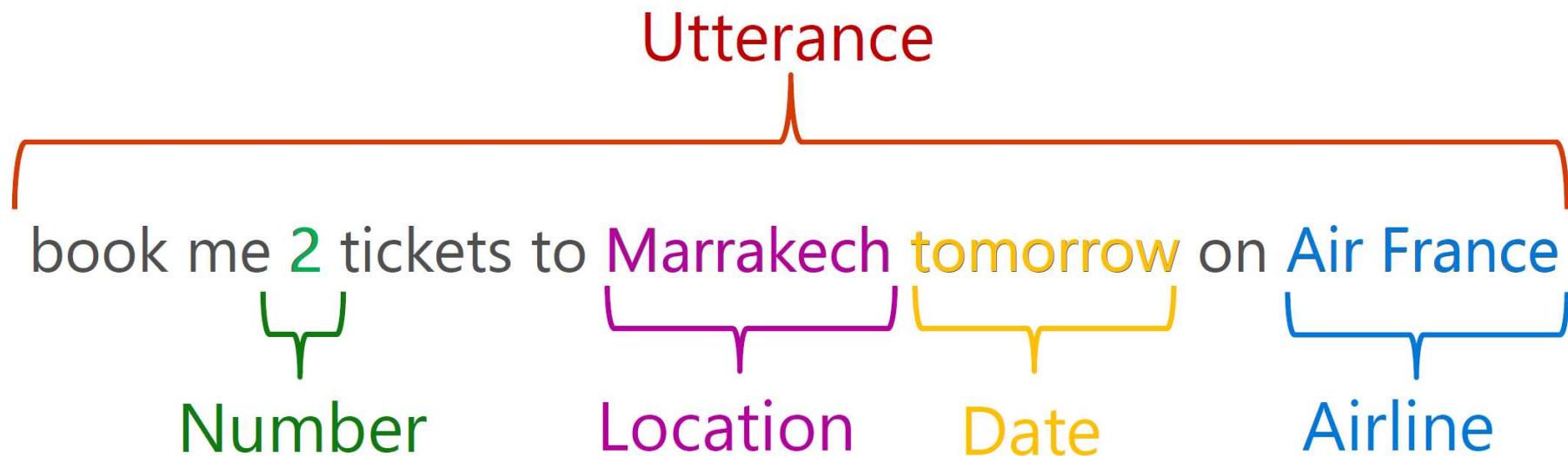
Utterance–Is the textual input from the user, that your app needs to interpret. It may be a sentence like “Book me a ticket to Paris”

03

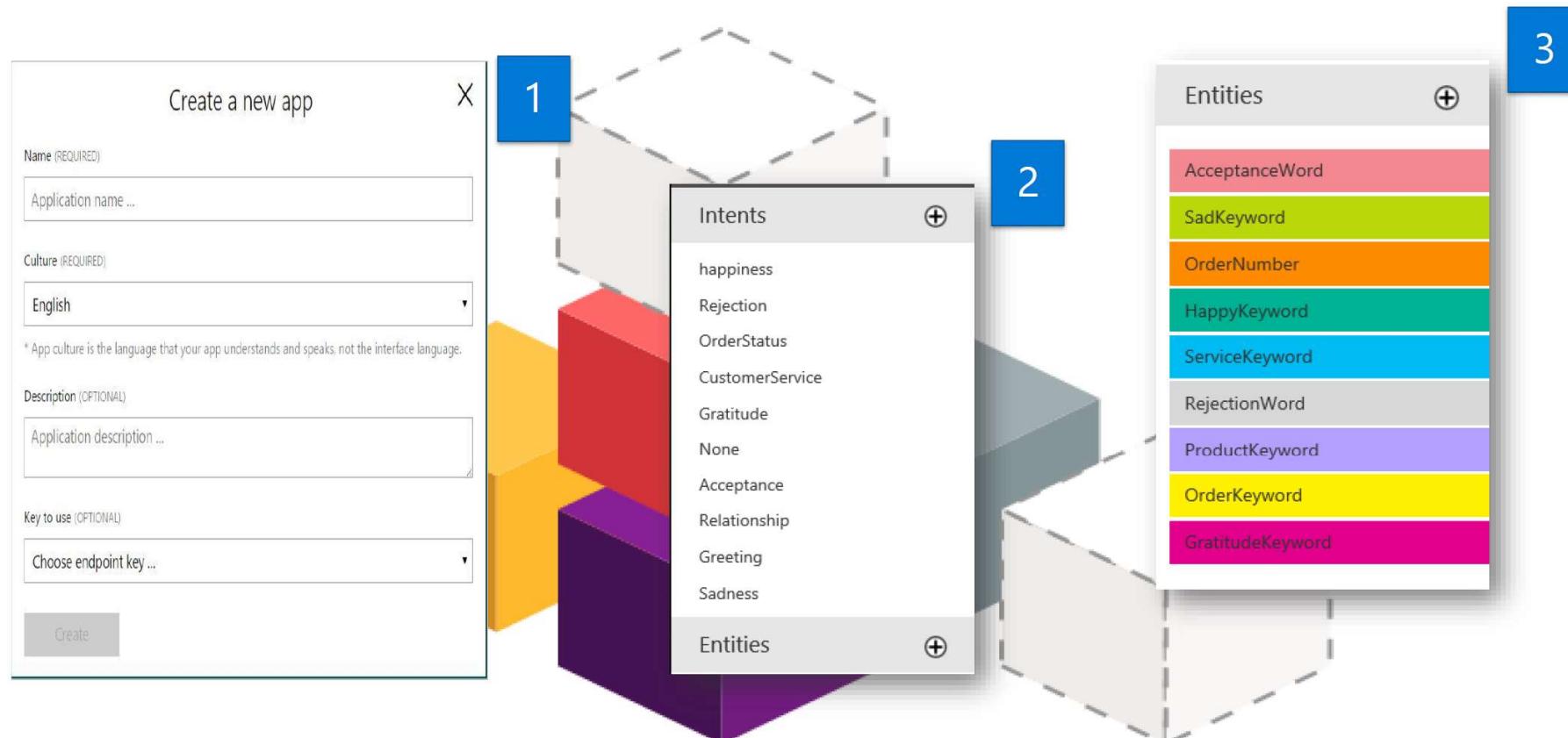
Entities – Entities are nouns, it represents an instance of a class of object that is relevant to a user’s intent. For example in the utterance “Book me a ticket to Paris”, Paris is an entity of type location



Key Concepts (Label Utterance)



Create your own Luis ... App!



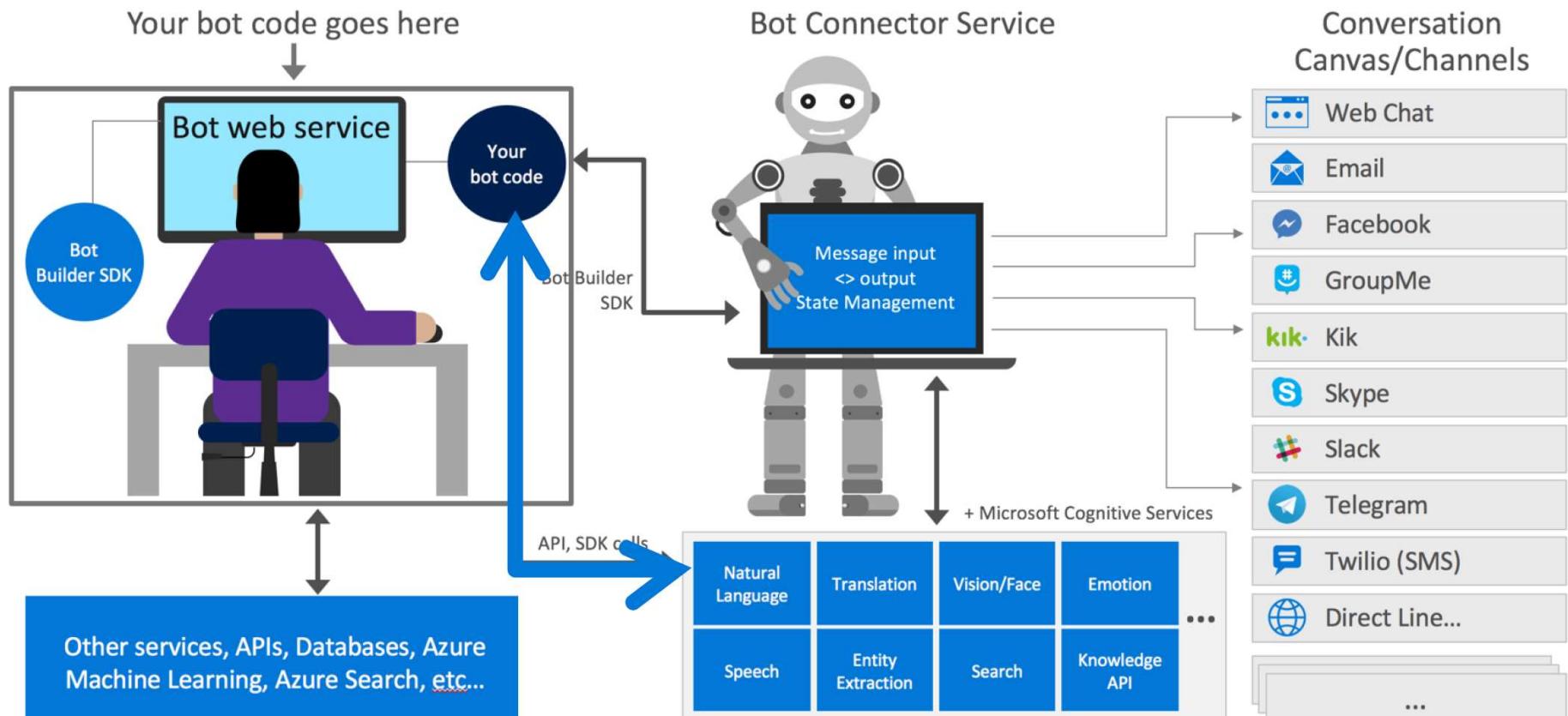
Demo

Luis.ai

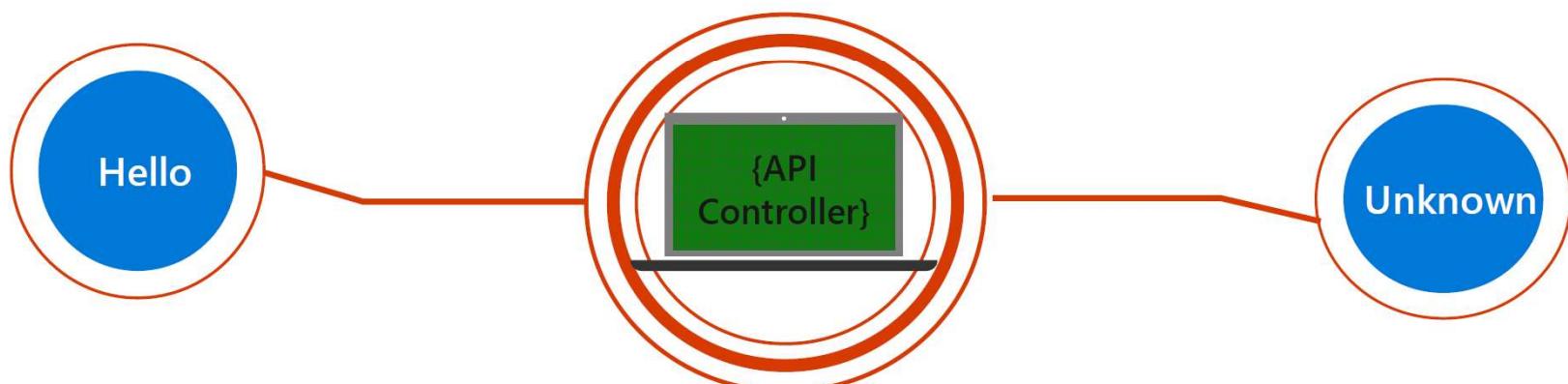
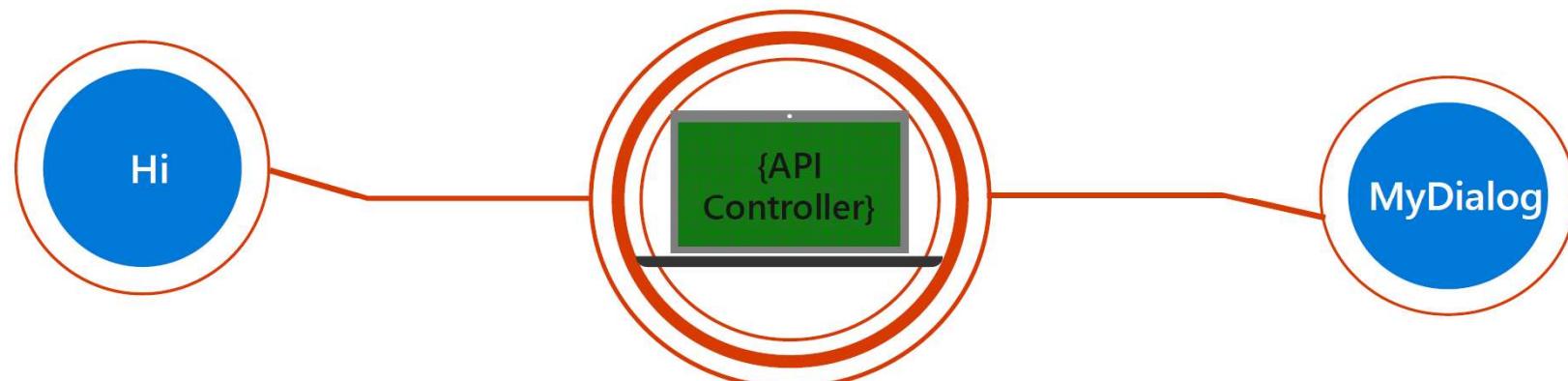


Integrating LUIS with a Bot

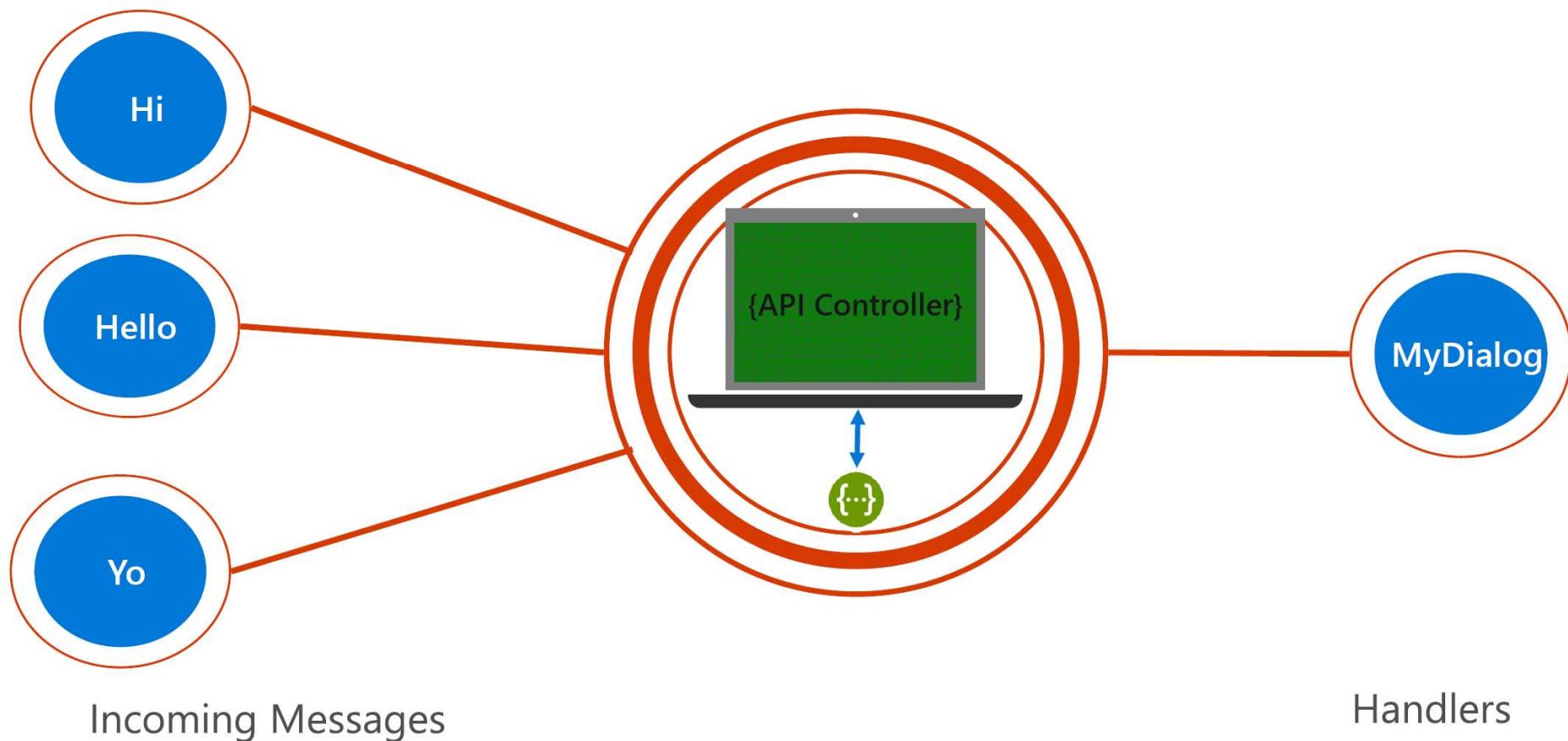
Bot Framework



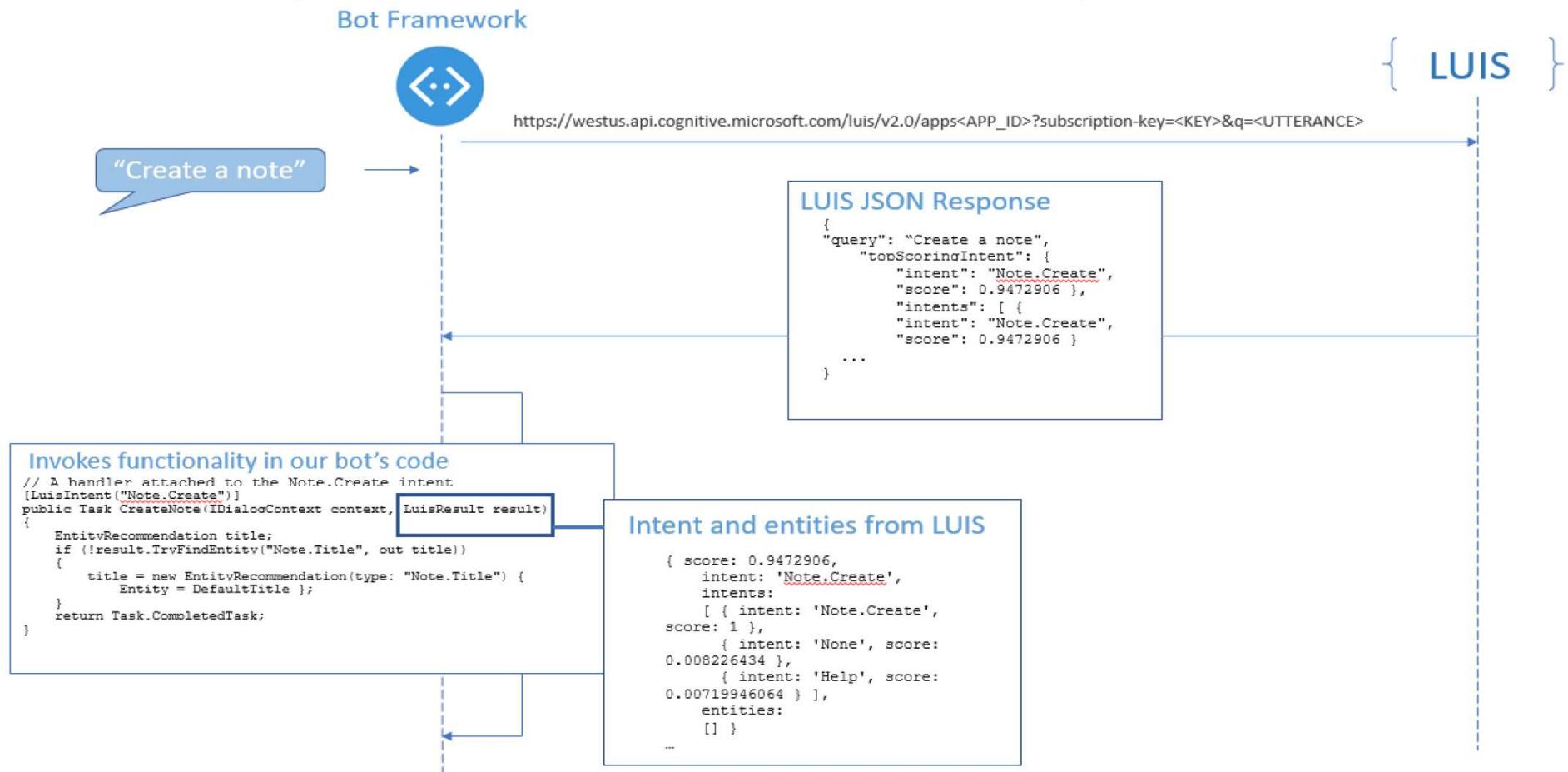
A world without LUIS



A world with LUIS



How LUIS passes intents and entities to your bot



Luis Dialog Development

- Dialog class inherits from **Microsoft.Bot.Builder.Dialogs. LuisDialog**
- Dialog class should have the **Serializable Attribute**

```
[LuisModel("<YOUR_LUIS_APP_ID>", "YOUR_SUBSCRIPTION_KEY", domain: "westus.api.cogn
[Serializable]
public class SimpleNoteDialog : LuisDialog<object>
{
    // ...
}
```

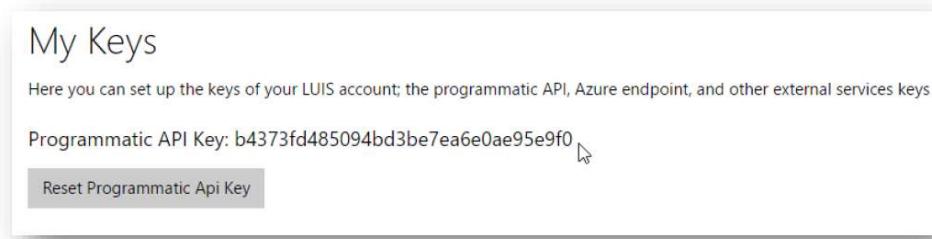
LUIS Dialog Development

LuisModelAttribute Parameters

- modelID:



- subscriptionKey: The LUIS subscription key.



- domain: Domain where LUIS model is located.

<https://docs.microsoft.com/en-us/bot-framework/dotnet/bot-builder-dotnet-luis-dialogs#regions-and-keys>

Intent Handler

- To designate the method that runs when a specific intent is matched, specify the **LuisIntent** attribute.
- Specify the name of the intent as a parameter of the LuisIntent attribute
- Intent handler methods have **1:*** mapping with Intents

```
[LuisIntent("Note.Delete")]
public async Task DeleteNote(IDialogContext context, LuisResult result)
{
    Note note;
    if (TryFindNote(result, out note))
    {
        this.noteByTitle.Remove(note.Title);
        await context.PostAsync($"Note {note.Title} deleted");
    }
    else
    {
        // Prompt the user for a note title
        PromptDialog.Text(context, After_DeleteTitlePrompt, "What is the title of
    }
}
```

Retrieves an Entity From LuisResult

- To retrieve an entity value from the processed **LuisResult** use the extension method **TryFindEntity**

```
[LuisIntent("ShowHotelsReviews")]
0 references | 0 changes | 0 authors, 0 changes
public async Task Reviews(IDialogContext context, LuisResult result)
{
    EntityRecommendation hotelEntityRecommendation;
    if (result.TryFindEntity(EntityHotelName, out hotelEntityRecommendation))
    {
        await context.PostAsync($"Looking for reviews of '{hotelEntityRecommendation.Entity}'....");
        IMessageActivity resultMessage = Helper.CreateCarouselOfThumbnailCard(context);
        await context.PostAsync(resultMessage);
    }
    context.Wait(this.MessageReceived);
}
```

Use LuisDialog implementation

- To use the implemented LuisDialog class, update the Post method in your **MessageController** class to reference it.

```
public async Task<HttpResponseMessage> Post([FromBody]Activity activity)
{
    if (activity.Type == ActivityTypes.Message)
    {
        await Conversation.SendAsync(activity, () => new Dialogs.LuisRootDialog())
    }
    else
    {
        HandleSystemMessage(activity);
    }
    var response = Request.CreateResponse(HttpStatusCode.OK);
    return response;
}
```

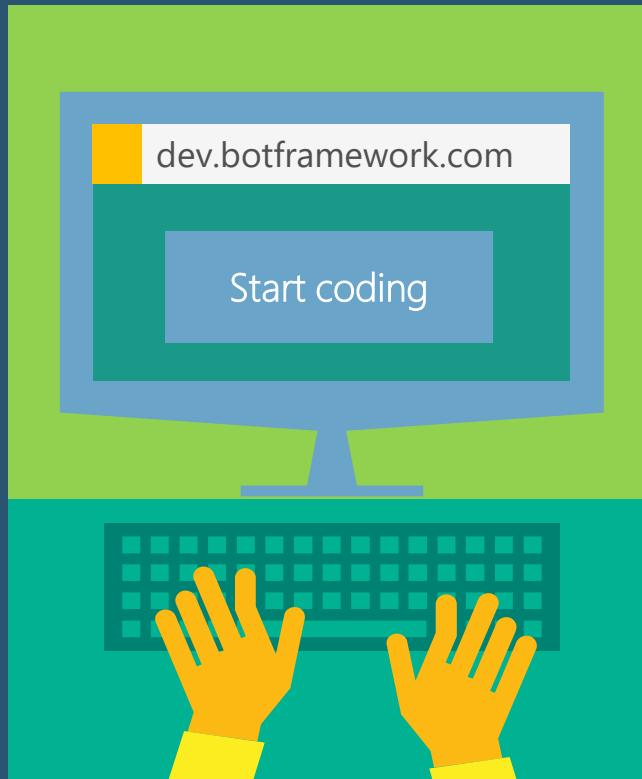
Demo

Luis Bot with
C# handlers



Lab

Lab 3 - Developing a smarter Bot



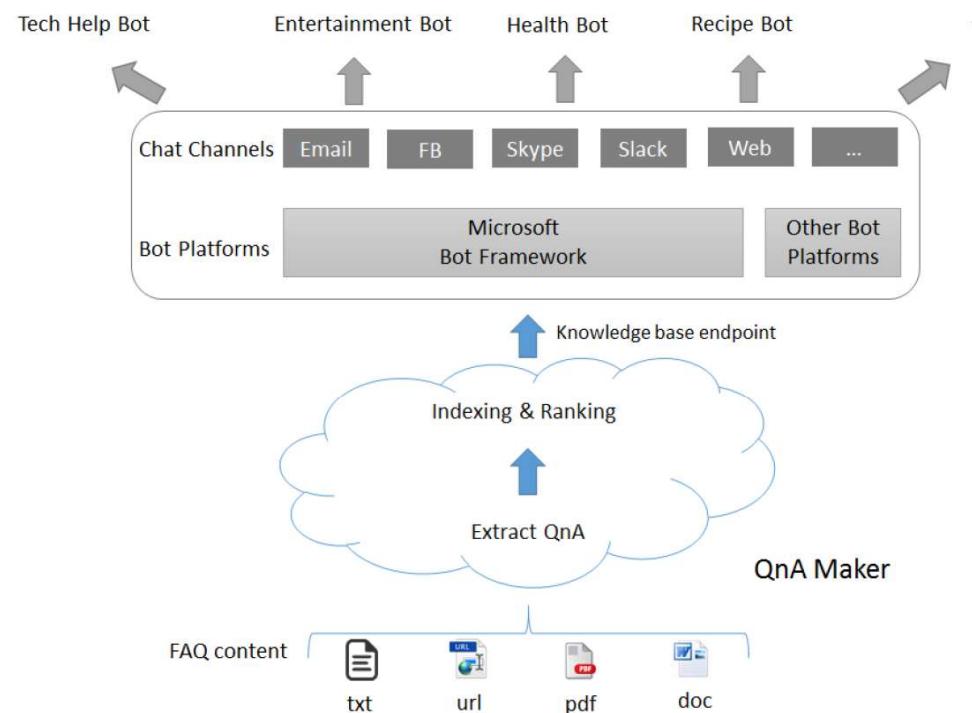
QnA Maker

What is QnA Maker?

- It's a free, easy-to-use, REST API and Web service that trains AI.
- Allows the bot to provide answers to user's questions in a more natural, conversational way.
- Compatible across development platforms, hosting services, and channels.
- QnA Maker is the only question and answer service with a graphical user interface; meaning you don't need to be a developer to train, manage, and use it for a wide range of solutions.

What is QnA Maker?

- With optimized Machine Learning logic and the ability to integrate industry-leading language processing with ease, QnA Maker distills masses of information into distinct, helpful answers.



Authentication & Subscription keys

- It is required a Microsoft account to sign in to the portal.
- You will receive a unique pair of keys.
 - The second one is just a spare.
 - Do not share the secret keys with anyone.
- These subscription keys are used to track your usage of the service and need to be part of every request.

The screenshot shows the Microsoft QnA Maker portal interface. At the top, there's a navigation bar with links for 'QnA Maker PREVIEW', 'My services', 'Create new service', 'Documentation', and 'Feedback'. On the right side, there's a user profile for 'Prashant Choudhari' (prashar) with options to 'Subscribe' (which is highlighted with a red box) and 'Sign Out'. Below the navigation bar, the main content area is titled 'My QnA services'. A table lists services based on 'Service Name', 'Last modified', 'Last published', and 'Sample Code'. To the right of this table, a large callout box is open, titled 'Subscription keys'. It contains information about the free preview terms: 'QnA Maker is provided under Cognitive Services Terms. This free preview provides up to 10 transactions per minute, up to 10,000 transactions per month.' Below this, two text fields show 'Primary key' and 'Secondary key', each followed by a 'Show' and 'Copy' link.

Service Name	Last modified	Last published	Sample Code

Subscription keys

QnA Maker is provided under Cognitive Services Terms. This free preview provides up to 10 transactions per minute, up to 10,000 transactions per month.

Primary key	Show Copy
Secondary key	Show Copy

Pricing

- Since currently the QnA Maker is a free to use tool.
- We have the following restriction of usage per Subscription key.
 - 10.000 transactions per month.
 - 10 per minute.
- Beyond this your requests will be throttled.

Create your knowledge base

- Currently the tool can auto-extract question and answer pairs from two types of input:

FAQ Pages

- We support extraction from following types of FAQ URLs:
 - Plain FAQ pages: Questions are immediately followed by answers.
 - FAQ pages with section links: Questions are aggregated together and linked to answers on the same page.
 - FAQ pages with linked answers: Questions are aggregated together and linked to answers on a different page.
- We also support extraction from offline doc types (.docx, .doc, .pdf, .xlsx and .tsv)
- Auto-extraction works best on FAQ pages with clear Q-A structure and semantics such as question ending with "?" and question contains interrogative words such as "why", "what", "how", etc.

Product manuals

- We support extraction of Q-As from PDF format product manuals.
- QnA Maker extracts the headings and sub-headings as questions and the proceeding content as the answer.
- Auto-extraction works best of manuals with a table of contents and/or an index page, and clear structure with hierarchical headings.

Update your knowledge base

- There are several ways you might want to update your knowledge base:

Editorial QnA updates:

- Edit the QnA table directly.
- You can:
 - Add/delete a QnA Pair.
 - Add an alteration of an existing QnA Pair.
- This is ideal for quick editorial fixes to your knowledge base.

Update the sources

- You could add/delete/modify existing sources.

Download and upload

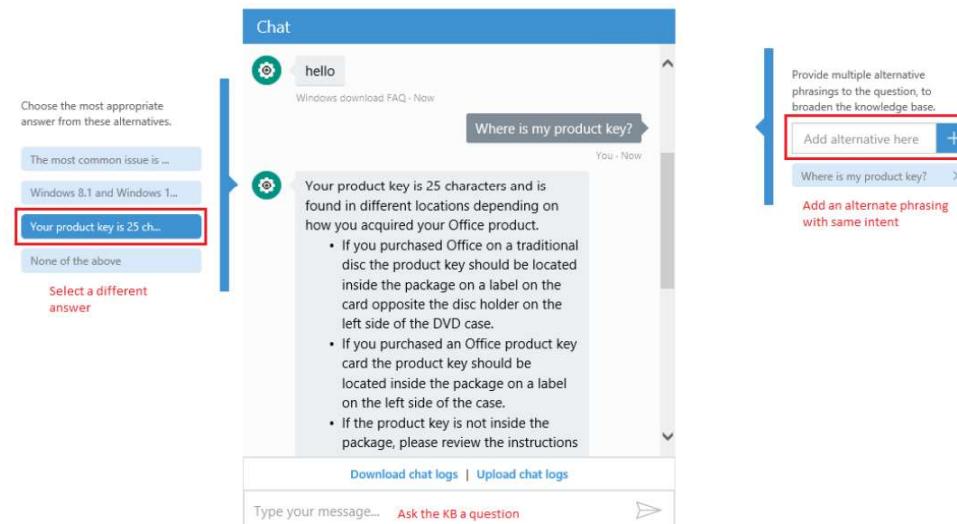
- You could replace your entire knowledge base at one go.
- Ideal for bulk updates to your knowledge base.
- Upload KB expects file format of tab separated columns of Question, Answer and Source.
- You could download the entire KB, make changes, and then upload the knowledge base.

Train your knowledge base

- The train feature lets you evaluate the correctness of the responses and correct them and re-train the KB.
- There are two ways you can improve the relevance of the responses.

Chat with your KB

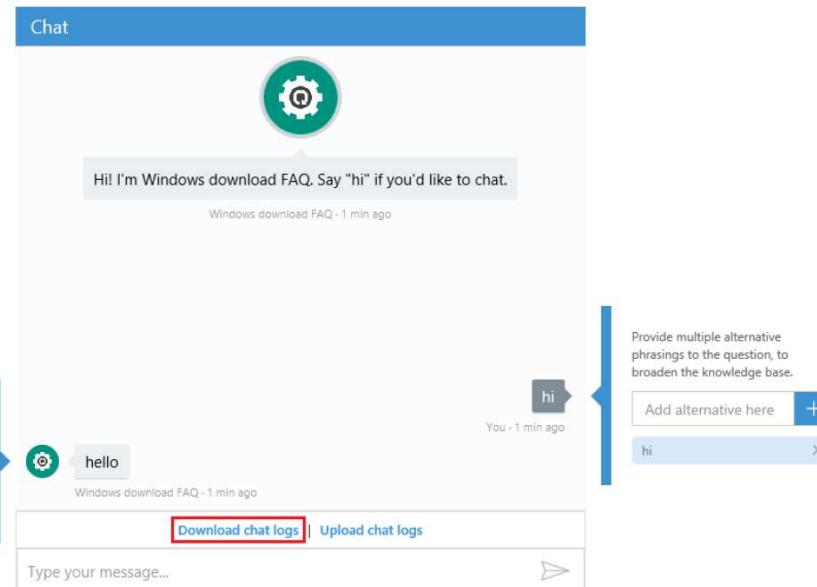
- While chatting with your KB, you can see the relevance of the responses.
- You can add a variation to an existing question as well as choose a different answer for a question.



Train your knowledge base

Replay live chat logs

- A very useful feature is to see what responses the service returns for live traffic, and the train it appropriately.
- You can download the live chat traffic hitting your published end-point.
 - This will download all the questions hitting your end-point in descending order of frequency.
- Looking at the chat logs, you can decide which questions you want to test and train your KB base on.



Publish your knowledge base

- Before the final publish, you can preview the changes that will affect the KB on final publish.

Windows download FAQ

Review your changes

Source	QnA in production	QnA in current draft	QnA added	QnA deleted
https://www.microsoft.com/en-us/software-download/faq	26	26	0	0
Editorial	6	6	0	
upload - Copy.tsv	0	29	29	

[Download Diff File](#)

[Cancel](#) [Publish](#)

Success! Your service has been deployed. What's next?

You can always find the deployment details in your service's settings.

Use the below HTTP request to build your bot. [Learn how.](#)

Sample HTTP request	POST /knowledgebases/321595eaaf0347d08f5856df0adb006e/generateAnswer Host: https://westus.api.cognitive.microsoft.com/qnamaker/v1.0 Ocp-Apim-Subscription-Key: 5f7e25duoLdr43b505845d155866858 Content-Type: application/json {"question": "hi"}
---------------------	--

Need to fine-tune and refine? Go back and keep editing your service.

[Edit Service](#)



Share your knowledge base

- You could add other users to your KB, so they can contribute to it as well.

Add users to your KB

- Go to the settings tab and add users

SETTINGS

Service name	Bitlocker FAQ
--------------	---------------

URLs

<https://support.microsoft.com/en-us/help/17133/windows-8-bitlocker-recovery-keys-frequently-asked-questions>

<http://> [+ Add another](#)

Files

Select file...

Share this service

Enter your team member email to grant them access to this service [+ Invite](#)

user1@outlook.com (Owner) [X](#)

user2@outlook.com [X](#)

Share your knowledge base

Merge changes

- In case you have unsaved changes, and another user updates the KB, you will be asked to refresh before you can save your changes.
- When you refresh, you will be able to see the merged changes in orange.

KNOWLEDGE BASE 6 QnA pairs		
	Question	Answer
<small>^ Original source: Editorial</small>		
1	Hi	Hello
2	Wassup?	Merged changes Not much.
<small>^ Original source: https://support.microsoft.com/en-us/help/17133/windows-8-bitlocker-recovery-keys-frequently-asked-questions</small>		
1	Why am I locked out?	There are several reasons that might make a PC go into recovery mode. For example, your organization might have a password security policy that locks you out after a certain number of failed attempts to sign in. Or perhaps your PC encountered a hardware malfunction, an unexpected configuration change, or another security event. Requiring a recovery key helps ensure that only an authorized person can unlock your PC and restore access to your encrypted data
2	How can I get my BitLocker recovery key?	Depending on how your PC is set up, there are different ways to get your recovery key. If your PC is connected to a domain: Contact your administrator to get your recovery key. If your PC isn't connected to a domain: There are several locations in which your BitLocker recovery key might have been saved. Here are some places to check: Your Microsoft account online. This option is only available on non-domain-joined PCs. To get your re-

RM7
RM24

Demo

qnamaker.ai



RM7 Demonstrate LUIS setup and updating the RootDialog.

Rinor Maloku; 13.03.2018

RM24 In the notes: Demo 8 - LUIS.AI

Rinor Maloku; 13.03.2018

RM7
RM24

Demo

Bot + QnAMaker



Folie 71

RM7 Demonstrate LUIS setup and updating the RootDialog.

Rinor Maloku; 13.03.2018

RM24 In the notes: Demo 8 - LUIS.AI

Rinor Maloku; 13.03.2018

Lab

Lab 4 - Connect to QnA Maker



Dialog Stack

Maintaining Conversation State with Dialogs

Conversations rely heavily on the context.

Dialogs and State work together to provide context to conversations with chatbots



RM14 In the Demo on the previous slide the attendees set up the application and they got to know a Dialog, i.e. the default RootDialog.
The theory here will improve their understanding of the dialog

Rinor Maloku; 13.03.2018

RM26 Cartoon is meant to show the importance of context (w.o. the context their conversation is misleading) and lighten the tone a little.

-
I couldn't find any better cartoon :(.

Would much rather find something that is really funny, any ideas?

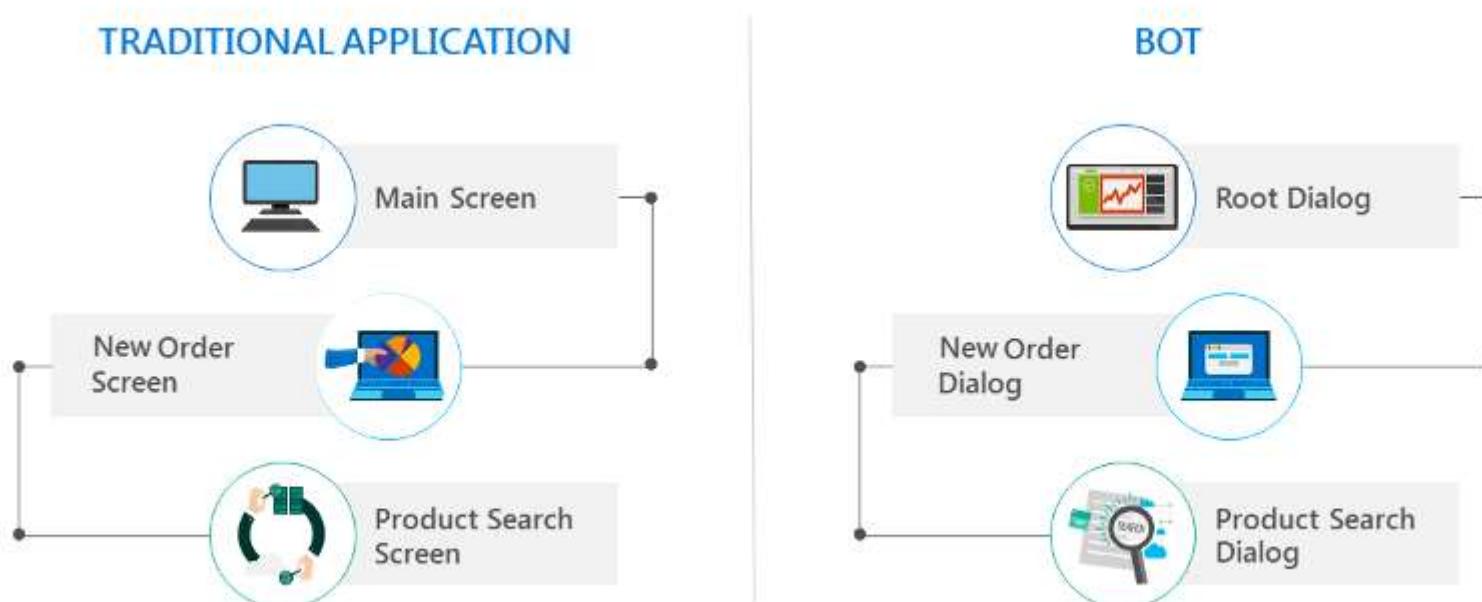
Rinor Maloku; 13.03.2018

Maintaining Conversation State with Dialogs

- What Are Dialogs
 - Way of Wrapping an entire “experience” into easily managed interaction based on a “chained” and “conversational” paradigm
 - Dialogs make the UI of a Bot
 - Dialogs may or may not have graphical interfaces
 - They may contain buttons, text, and other elements, or be entirely speech-based
 - Dialogs also contain actions to perform tasks such as invoking other dialogs or processing user input.

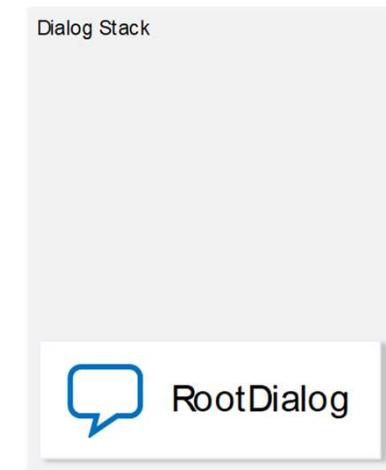
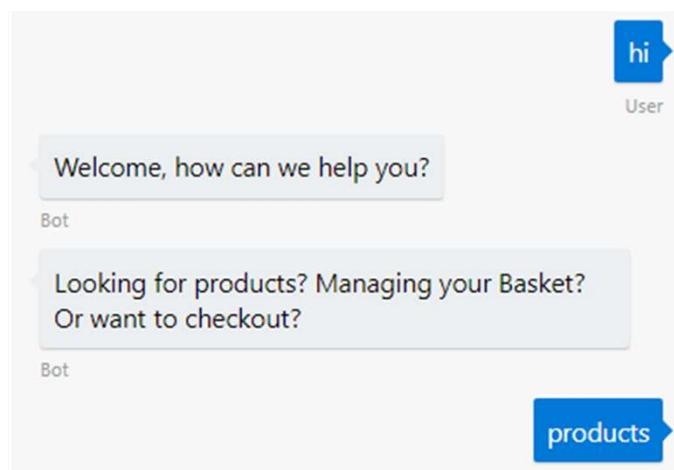
Dialog Stack

- Dialogs and the flow of conversation behind the scenes are managed via the DialogStack.
- The DialogStack keeps track in which Dialog a user currently is.



Dialog Stack - Details

Chat



RM2

After we complete the example we have two dialogs:

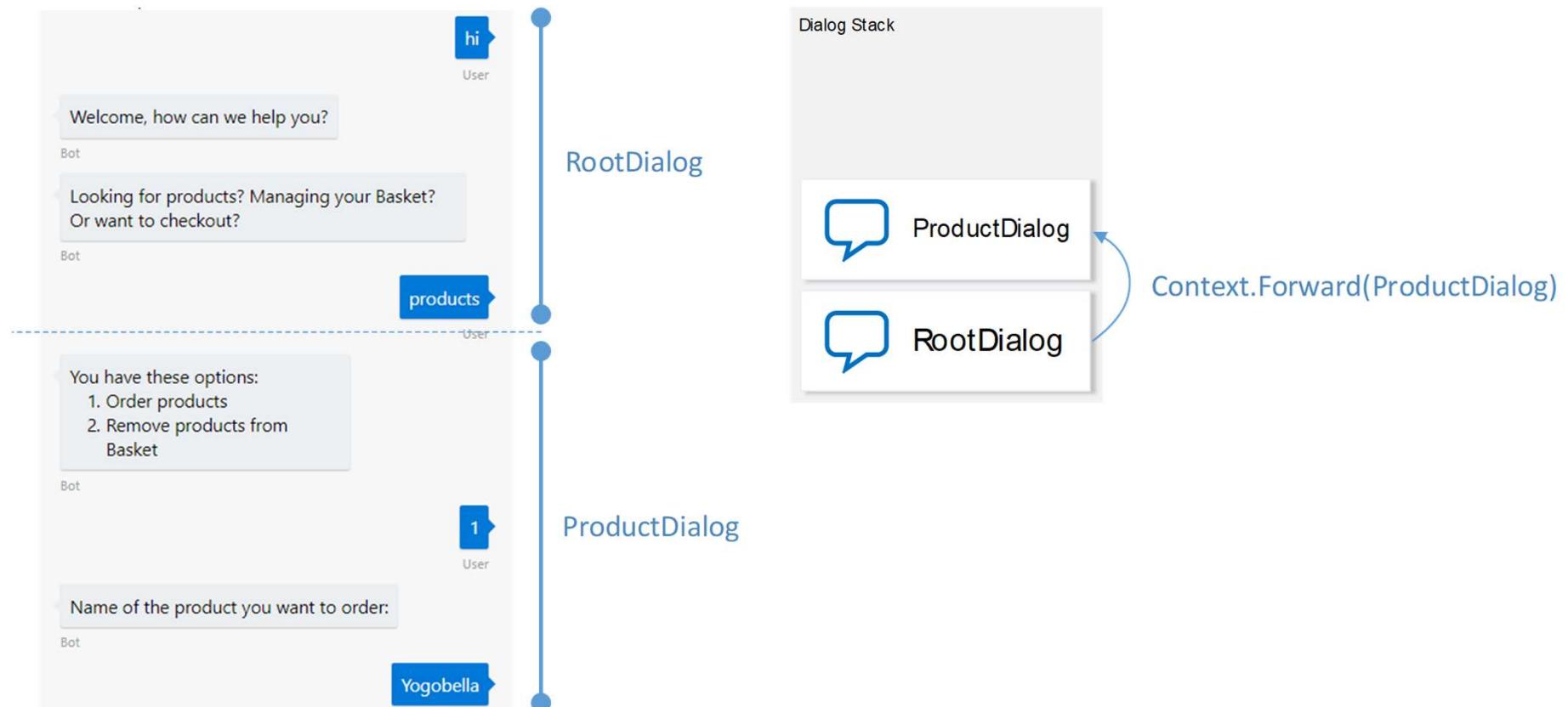
1. RootDialog and
2. ProductDialog.

In this slide we explain how the DialogStack keeps track of in which Dialog the user is in the conversation

Rinor Maloku; 12.03.2018

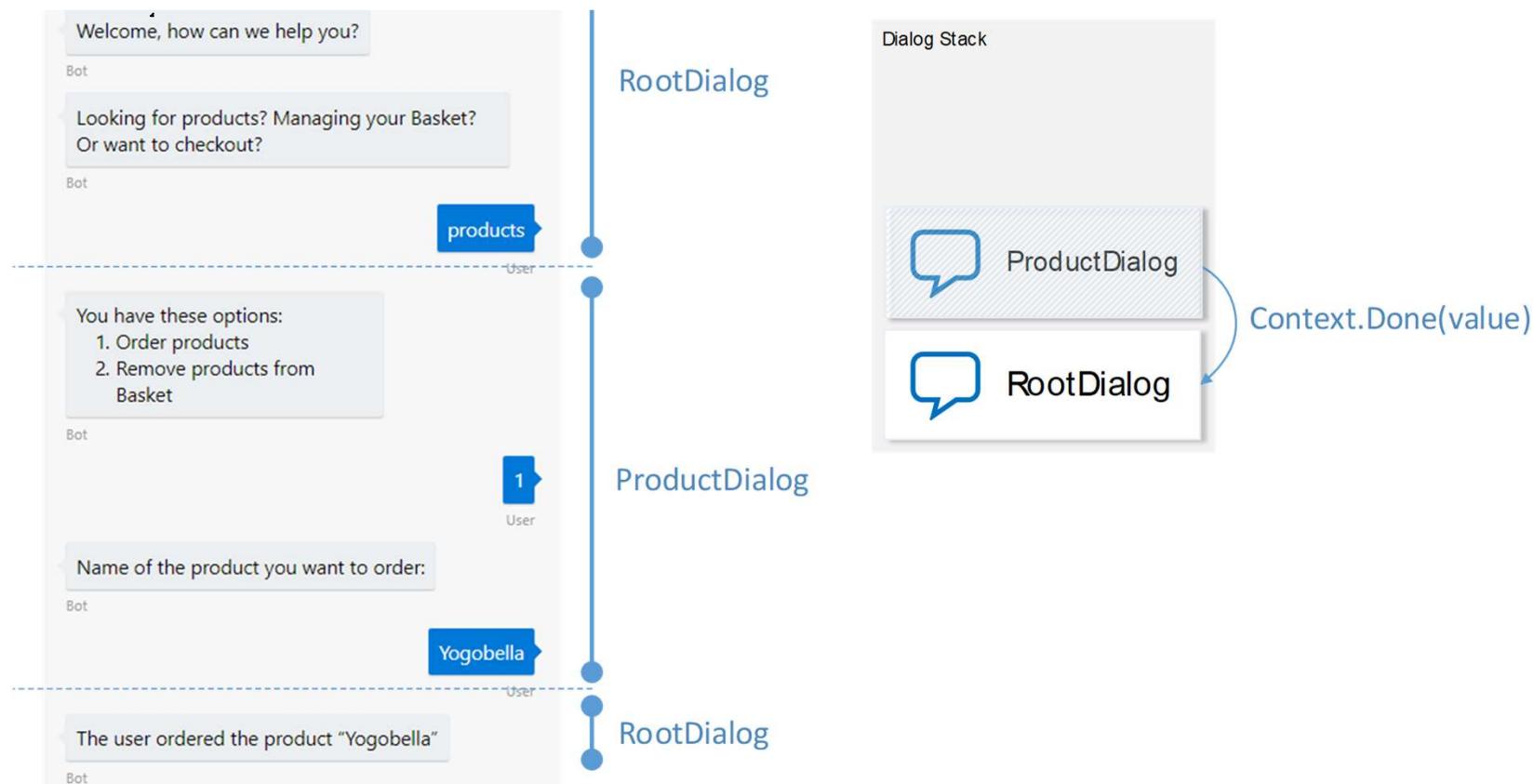
Dialog Stack - Details

Chat



Dialog Stack - Details

Chat



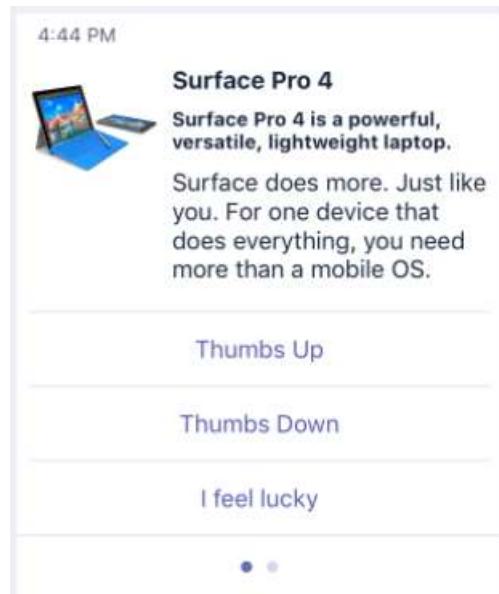
Dialog – C#

- context.Wait() -> specify the callback to invoke the next time the user sends a message
- context.Done() -> close a dialog and remove it from the stack
- context.Forward() or context.Call() -> redirect to other dialog. Dialog will be put in the DialogStack
- Every method must be ended with context.wait(), context.fail(), context.done() or redirection (Forward, Call)

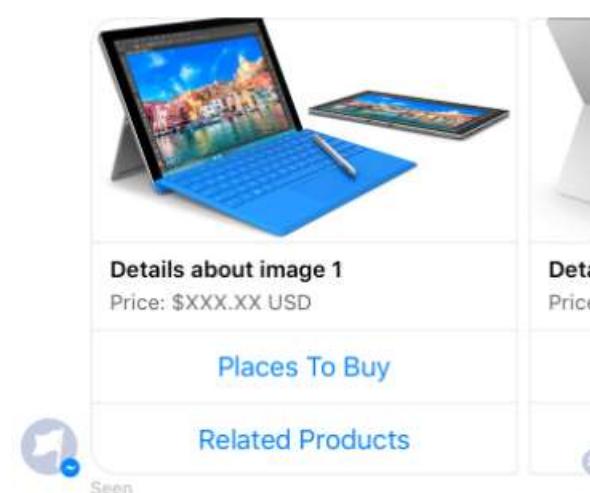
Attachments & Adaptive Cards

Attachments

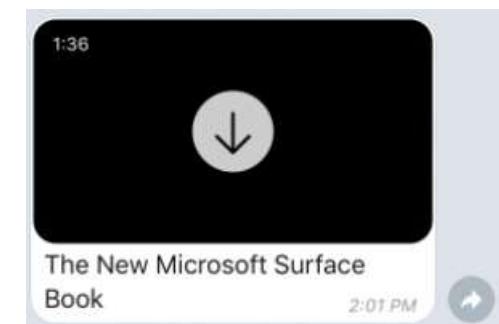
- Text Messages are not always the best medium to transfer information.
- Information exchange can be aided with Images, Videos and Audio Files.
- Bot Framework supports those media using Attachments.



Thumbnail Card



Carousel Card



Video Card

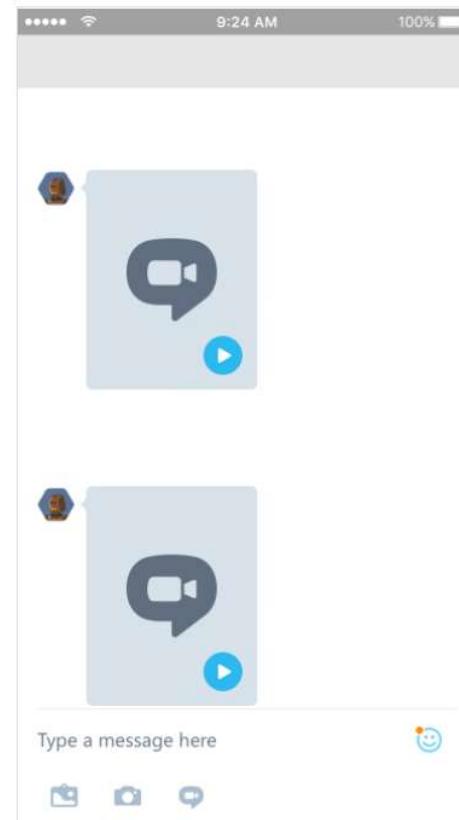
Attachments

Not all media types are supported by all channels.

Channel Inspector

Shows what media attachment are supported and how they look in each channel.

Channel Inspector



Channel:

Skype

Feature:

Videos

Videos

Playing of video attachments.

[Bot Framework Documentation](#)

Notes

Appearance Native Control.

Max Video Size 100 MB

Thumbnail Image Should be set in thumbnailUrl

See something wrong? [Send Feedback](#)

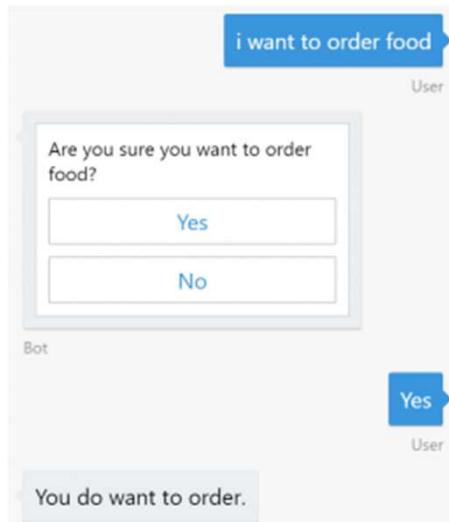
Rich Card Types

HeroCard	A card that typically contains a single large image, one or more buttons, and text.
AudioCard	A card that can play an audio file.
AnimationCard	A card that can play animated GIFs or short videos.
ThumbnailCard	A card that typically contains a single thumbnail image, one or more buttons, and text.
ReceiptCard	A card that enables a bot to provide a receipt to the user. It typically contains the list of items to include on the receipt, tax and total information, and other text.
SignInCard	A card that enables a bot to request that a user sign-in. It typically contains text and one or more buttons that the user can click to initiate the sign-in process.
VideoCard	A card that can play videos.
AdaptiveCard	A card that can contain any combination of text, speech, images, buttons, and input fields.

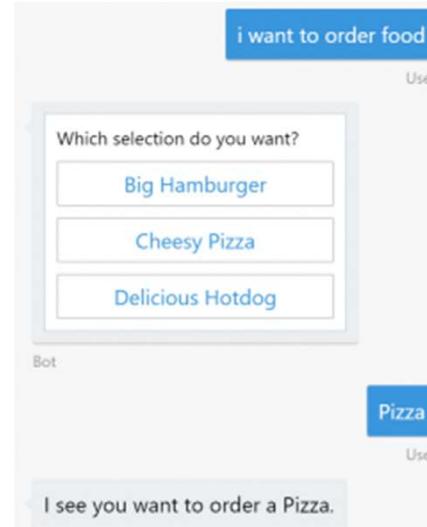
Prompt Dialogs

Simple way to provide the user with different type of dialogs

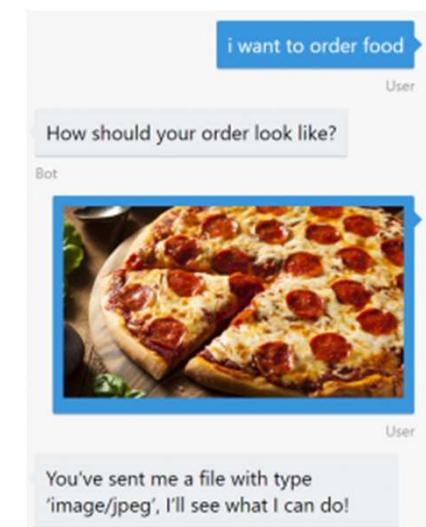
Confirm



Choice



Attachment



Adaptive cards



OPEN FRAMEWORK
MULTIPLE CANVASES

Card created: Publish Adaptive Card schema

Miguel Garcia
Created Monday, February 13, 2017 10:08:39 ...

Now that we have define the main rules and features of the format, we need to produce a schema and ...

Board: Adaptive Card
List: Backlog
Assigned to: David Claux
Due date: Not set

[Set due date](#) [Comment](#) [View](#)

Card created: Publish Adaptive Card schema

Miguel Garcia
Created Monday, February 13, 2017 10:08:39 PM

Now that we have define the main rules and features of the format, we need to produce a schema and ...

Board: Adaptive Card
List: Backlog
Assigned to: David Claux
Due date: Not set

[Set due date](#) [Comment](#) [View](#)

Card created: Publish Adaptive Card schema

Miguel Garcia
Created Monday, February 13, 2017 10:08:39 PM

Now that we have define the main rules and features of the format, we need to produce a schema and ...

Board: Adaptive Card
List: Backlog
Assigned to: David Claux
Due date: Not set

[Set due date](#) [Comment](#) [View](#)

Notification Microsoft Teams Skype Android iOS

Payments

Works across most major platforms

Fast and simple payment experience

Bring commerce capabilities to your bot

Stripe is currently supported

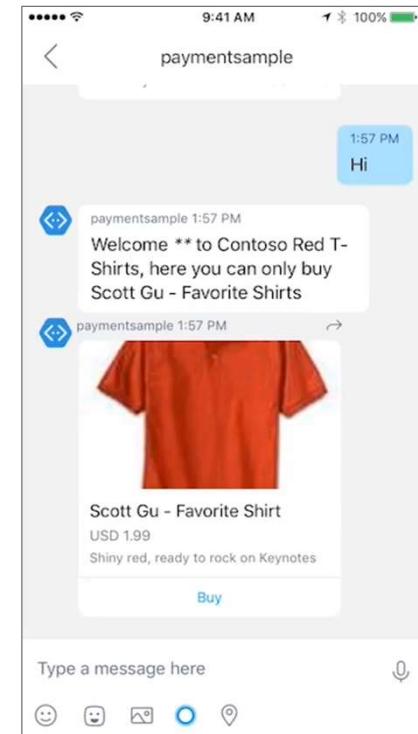
Visit Microsoft seller center

Obtain merchant ID

Start building

Learn more on [Microsoft Seller Center!](#)

Read the [integration guide](#) for the bot framework





Adaptive Cards

- Adaptive Cards enable:
 - Rich presentation experience,
 - Improve the Bot UX, and
 - SDKs for:
 - JavaScript
 - Android
 - iOS
 - UWP
 - .NET

I



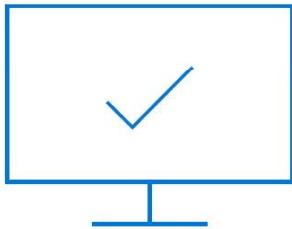
Adaptive Cards

- Adaptive Cards enable:
 - Rich presentation experience,
 - Improve the Bot UX, and
 - SDKs for:
 - JavaScript
 - Android
 - iOS
 - UWP
 - .NET

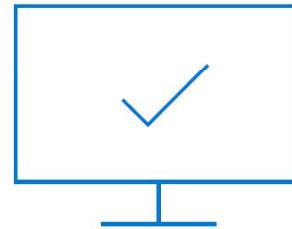
I



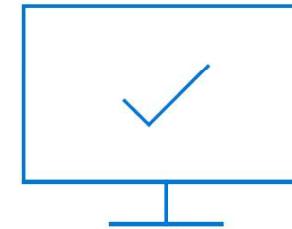
Adaptive Cards



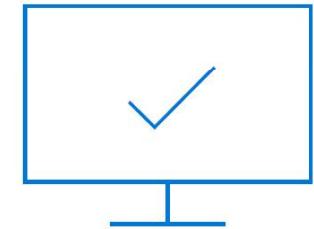
Rich schema
for layout
content



Actions to
open URLs,
show other
cards, etc

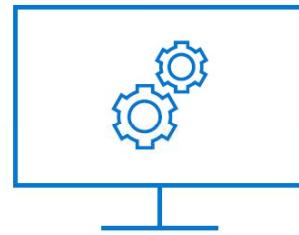


Input to
gather
information
from users

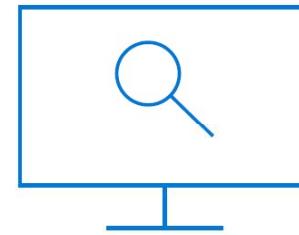


Speech
enabled from
day-one

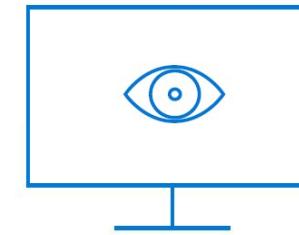
Tooling



Interactive
Visualizer



Schema
Explorer



VS Code
Previewer

Create and send an Adaptive Card from your bot

Adaptive Cards offer much more than just customizable text:

- Add Images to your card
- Organize your content with Containers and Columns
- Add multiple types of Actions
- Collect Input from your users
- Have one card show another card

Create and send an Adaptive Card from your bot

Property	Type	Required	Description
type	"AdaptiveCard"	Yes	Must be AdaptiveCard
actions	Action[]	No	The Actions to show in the card's action bar
body	array[]	No	The Card Elements to show in the primary card region
version	string	Yes	version of schema that this card was authored
minVersion	string	No	if a client doesn't support the minVersion the card should be rejected and return the fallbackText. If it does, then the elements that are not supported are safe to ignore
fallbackText	string	No	if a client is not able to show the card, show fallbackText to the user. This can be in markdown format.
speak	string	No	Specifies what should be spoken for this entire Item. This is simple text or SSML fragment

Mapping Json -> SDK

JSON

```
{
  "$schema": "http://adaptive-card.io/schemas/json",
  "type": "AdaptiveCard",
  "version": "1.0",
  "body": [
    {
      "type": "TextBlock", ①
      "text": "PIZZA"
    },
    {
      "type": "TextBlock",
      "text": "Tom's Pizza",
      "weight": "bolder",
      "size": "extraLarge",
      "spacing": "none"
    }
  ],
  "actions": [
    {
      "type": "Action.OpenUrl", ③
      "title": "More Info",
      "url": "https://adaptivecard.io/_Qw4w9WgXcQ"
    }
  ]
}
```

C#

```
new AdaptiveCard
{
  Body = new List<CardElement>
  {
    new TextBlock ①
    {
      Text = "PIZZA"
    },
    new TextBlock ②
    {
      Text = "Tom's Pie",
      Weight = TextWeight.Bolder,
      Size = TextSize.ExtraLarge
    }
  },
  Actions = new List<ActionBase>
  {
    new OpenUrlAction ③
    {
      Title = "More Info",
      Url = "https://www.youtube.com/watch?v=dQw4w9WgXcQ"
    }
  }
}
```

Create and send an Adaptive Card from your bot

```
// Create the attachment.  
Attachment attachment = new Attachment()  
{  
    ContentType = AdaptiveCard.ContentType,  
    Content = card  
};  
  
replyToConversation.Attachments.Add(attachment);
```

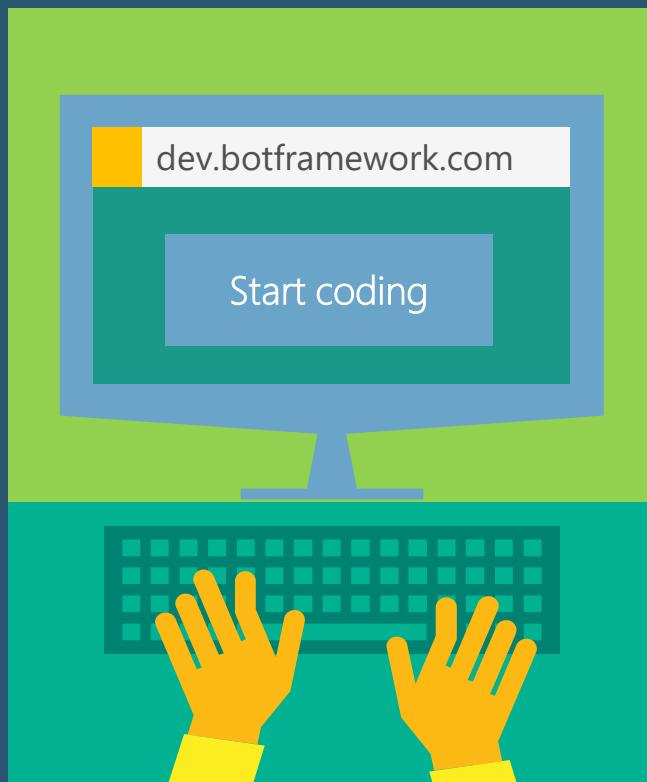
Create and send an Adaptive Card from your bot

You can also parse a JSON payload into an Adaptive Card. This makes it easy to manipulate the object model or even render Adaptive Cards inside your app by using our [renderer SDKs](#).

```
var json = card.ToJson();  
  
card = AdaptiveCard.FromJson(json).Card;
```

Demo

<http://adaptivecards.io/>



Lab

Lab 5 - Utilize Adaptive Cards with Bots



Authentication with Azure Active Directory

Bot integration

- I never want to see a user's password.
 - Don't take Username/Password into the bot and then authenticate from the bot.
 - Use some sort of delegation mechanism instead.
- Use Oauth 2.0
 - User logs in to AD directly via browser popup.
 - Browser receives authentication and provides a temporary code
 - User gives code to bot

AD has two modes

V1 endpoint

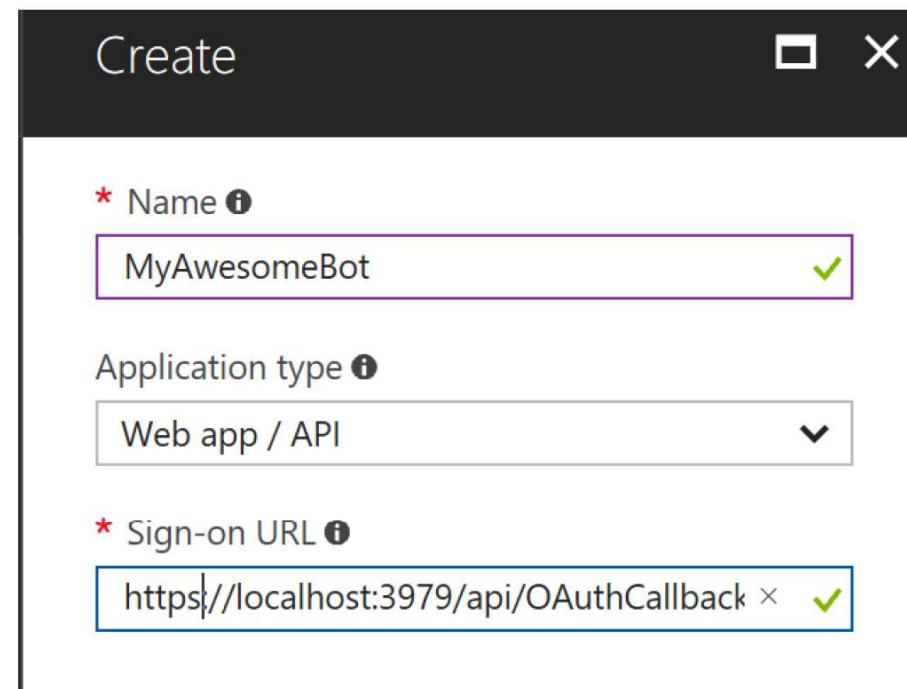
- If only supporting Microsoft work and school accounts

V2 endpoint

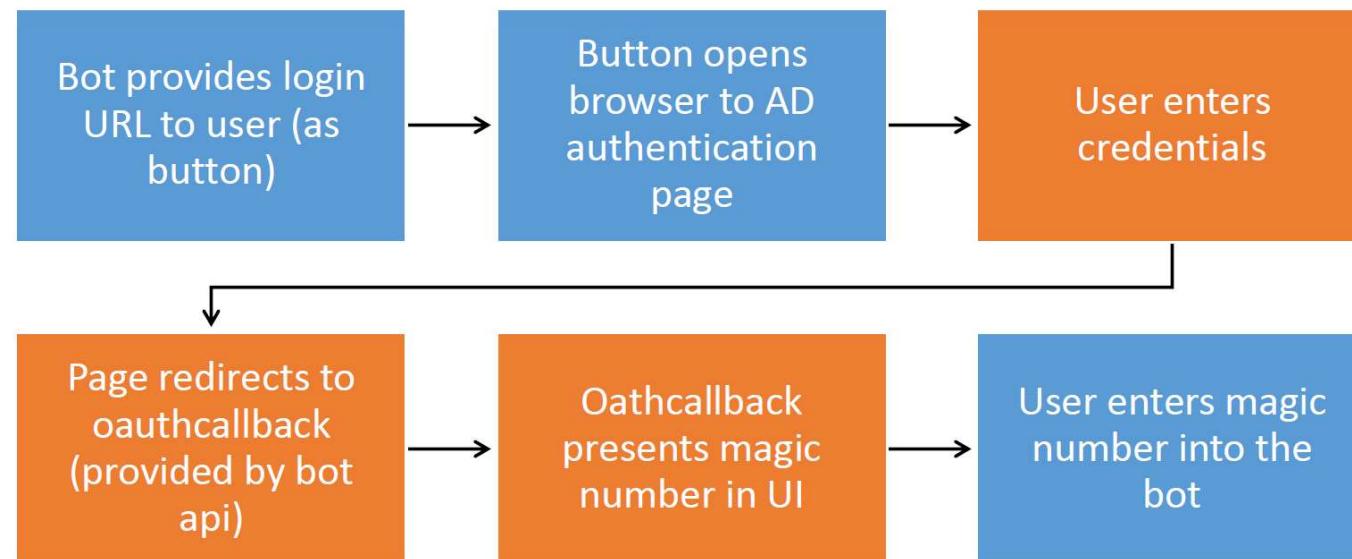
- If also supporting personal Microsoft accounts
- Supports fewer scenarios than v1 (today)

How do I set it up (.NET)?

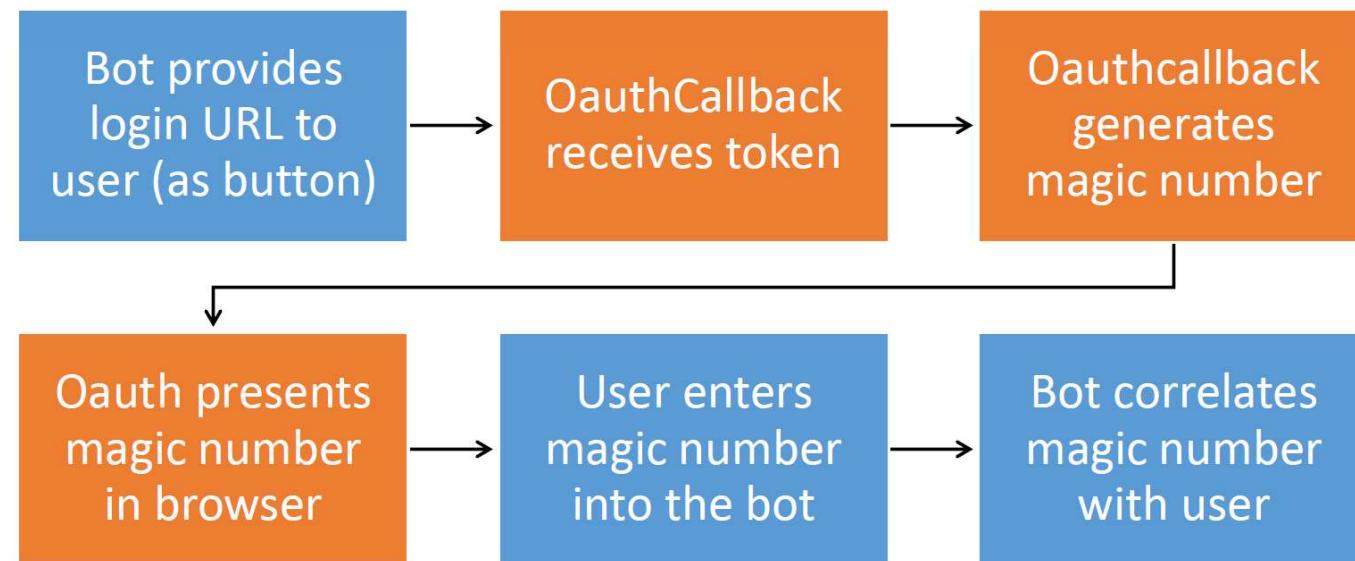
1. Register AD app (v1)
 1. Remember the app id and secret!
 2. App type = Web app/API
2. Enable SSL in bot project
3. Add Nuget package AuthBot
4. Forward call to AzureAuthDlg
(has 2 overloads)
 1. "resourceID" is for v1
 2. "scopes" overload is for v2



Auth Process (user perspective)



Auth Process (bot perspective)



Demo

Authentication



Lab

Lab 6 - Adding Authentication



Vielen Dank.