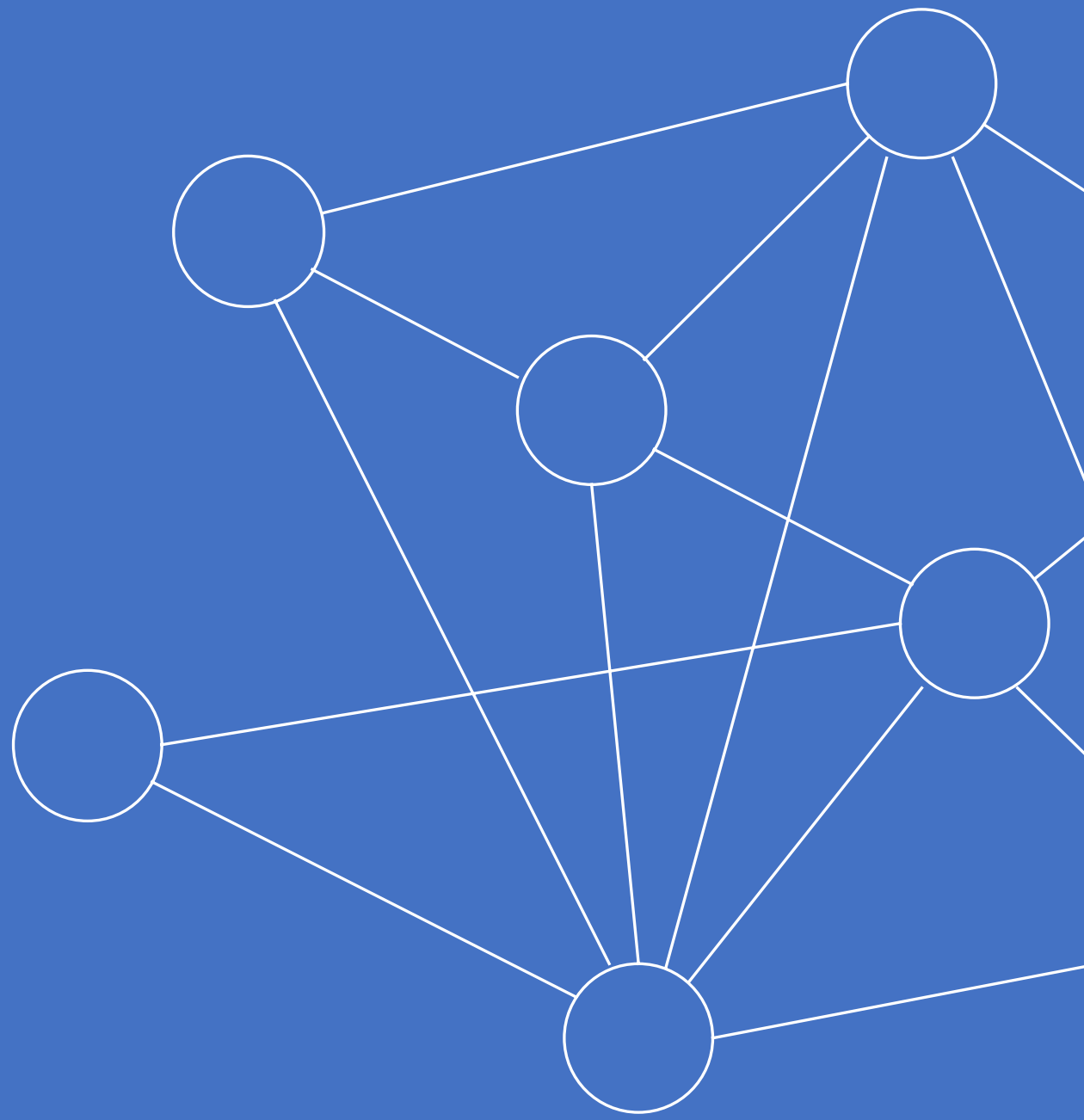
Microsoft

Conversational AI

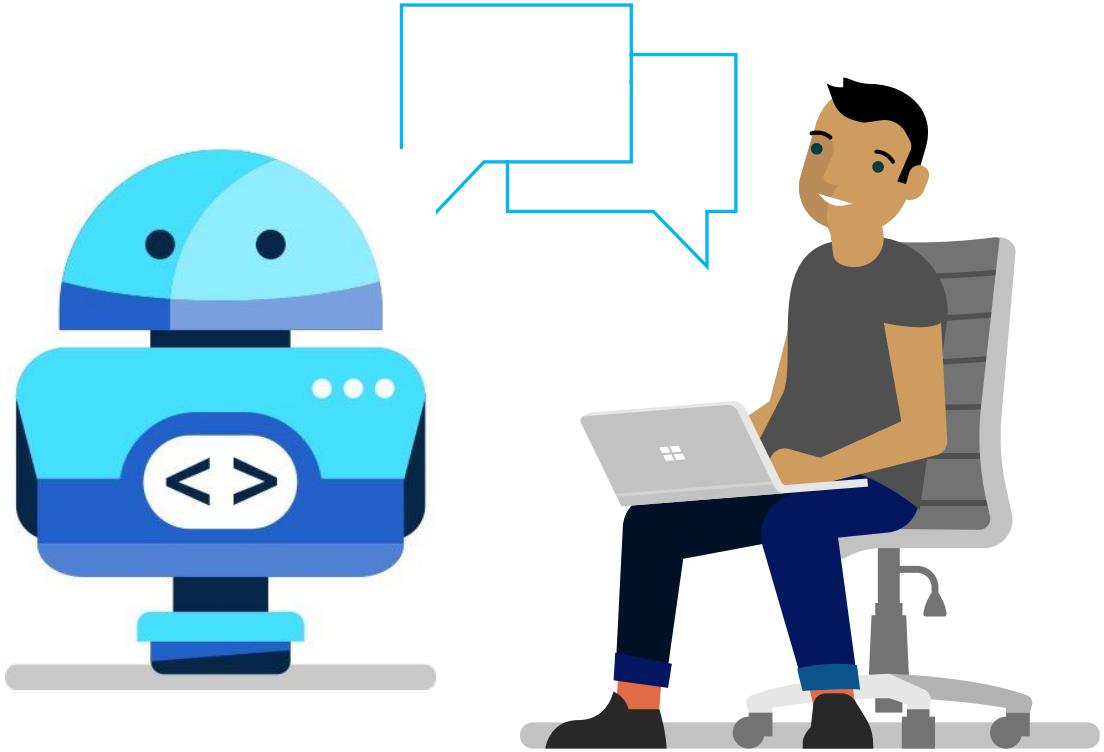


Conversational AI

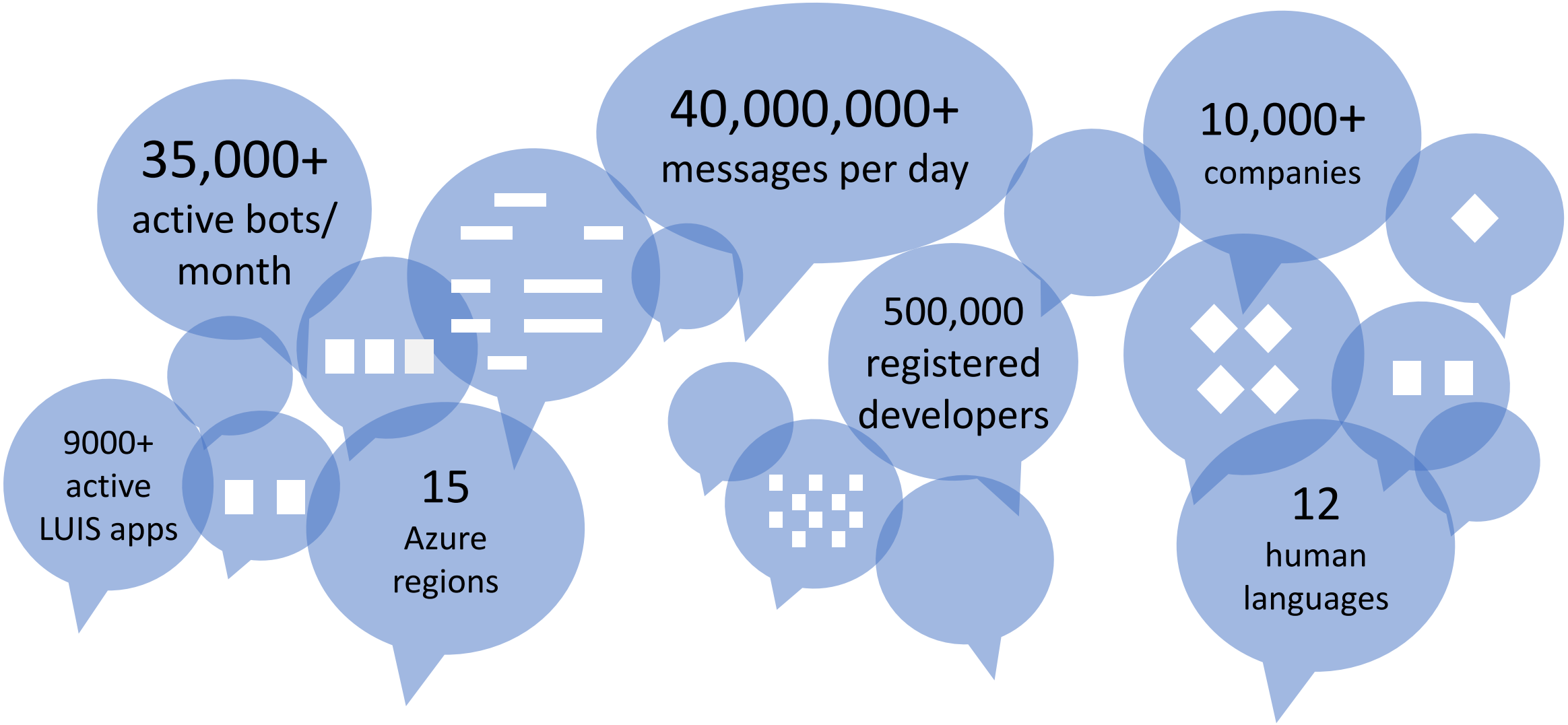
What is Conversational AI?

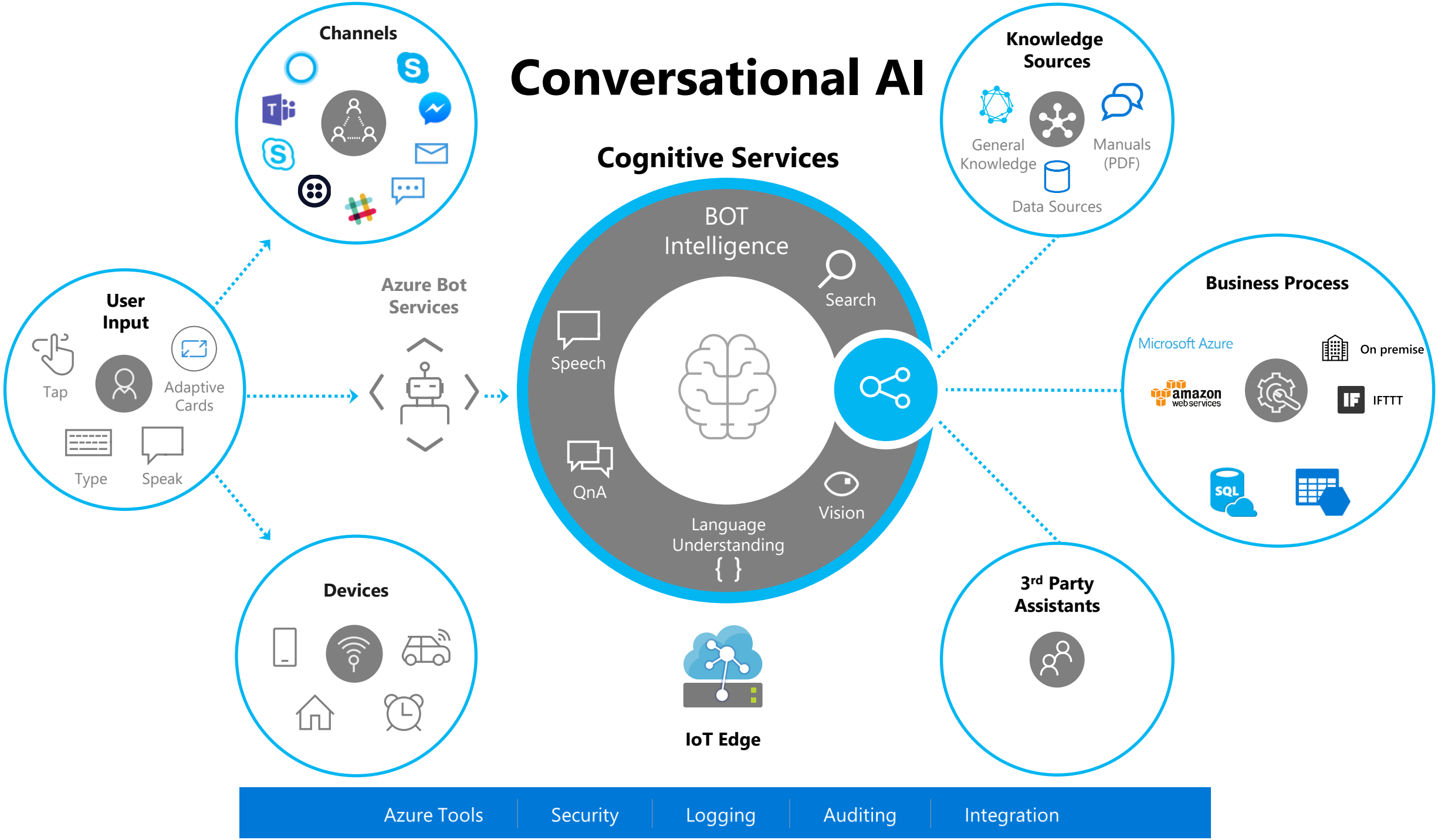
Conversational AI is a subfield of artificial intelligence focused on producing **natural and seamless conversations** between humans and computers

- A new **user interface paradigm**
- Exposes software services through **conversational interface**
- In apps, browser and on **IoT devices**



Conversational AI: Growth




Conversational AI
Channels
Knowledge Sources
General Knowledge
Manuals (PDF)
Data Sources
Cognitive Services
BOT Intelligence
Search
Speech
QnA
Vision
Language Understanding
User Input
Tap
Adaptive Cards
Type
Speak
Azure Bot Services
Business Process
Microsoft Azure
On premise
amazon web services
IFTTT
SQL
Devices
3rd Party Assistants
IoT Edge
Azure Tools
Security
Logging
Auditing
Integration

Our strengths…

AI and natural language

Get started with AI using cognitive services and make your bot see, hear, understand and learn.

Write it once, embed it anywhere

Connect with customers on Web chat, Facebook, Skype, IoT devices and more.

Enterprise-grade solutions

Build secure, global and scalable solutions, integrated into your own technology ecosystem.

Complete ownership and control

Create an experience to extend your brand and remain in control of your own data.

Platform for Conversational AI

Proven and reliable platform

Make data scientists and developers extremely productive using cloud + AI

Pretrained AI services

Powerful tools

Comprehensive platform

Innovate Anywhere

Enable organizations to innovate with AI anywhere – in the cloud, on-premises and the edge

Cloud

Edge

On-premises

Integrate, extend or customize

Empower development teams to use the languages and tools of their choices

python

PYTORCH

ONNX

TensorFlow

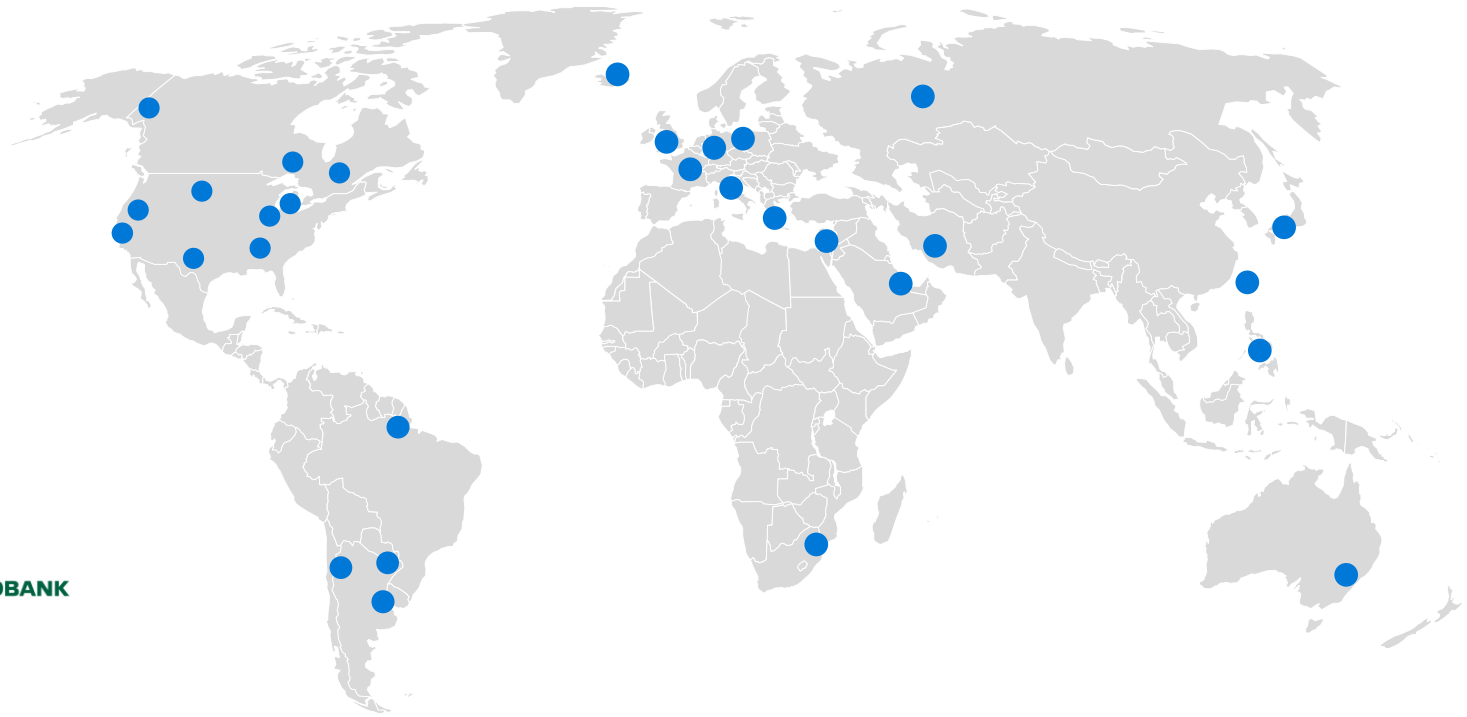
Spark

Enterprise Grade

Provide the best level of security, privacy and compliance standards to apply AI with confidence

>90% of Fortune 500 companies use Microsoft Cloud

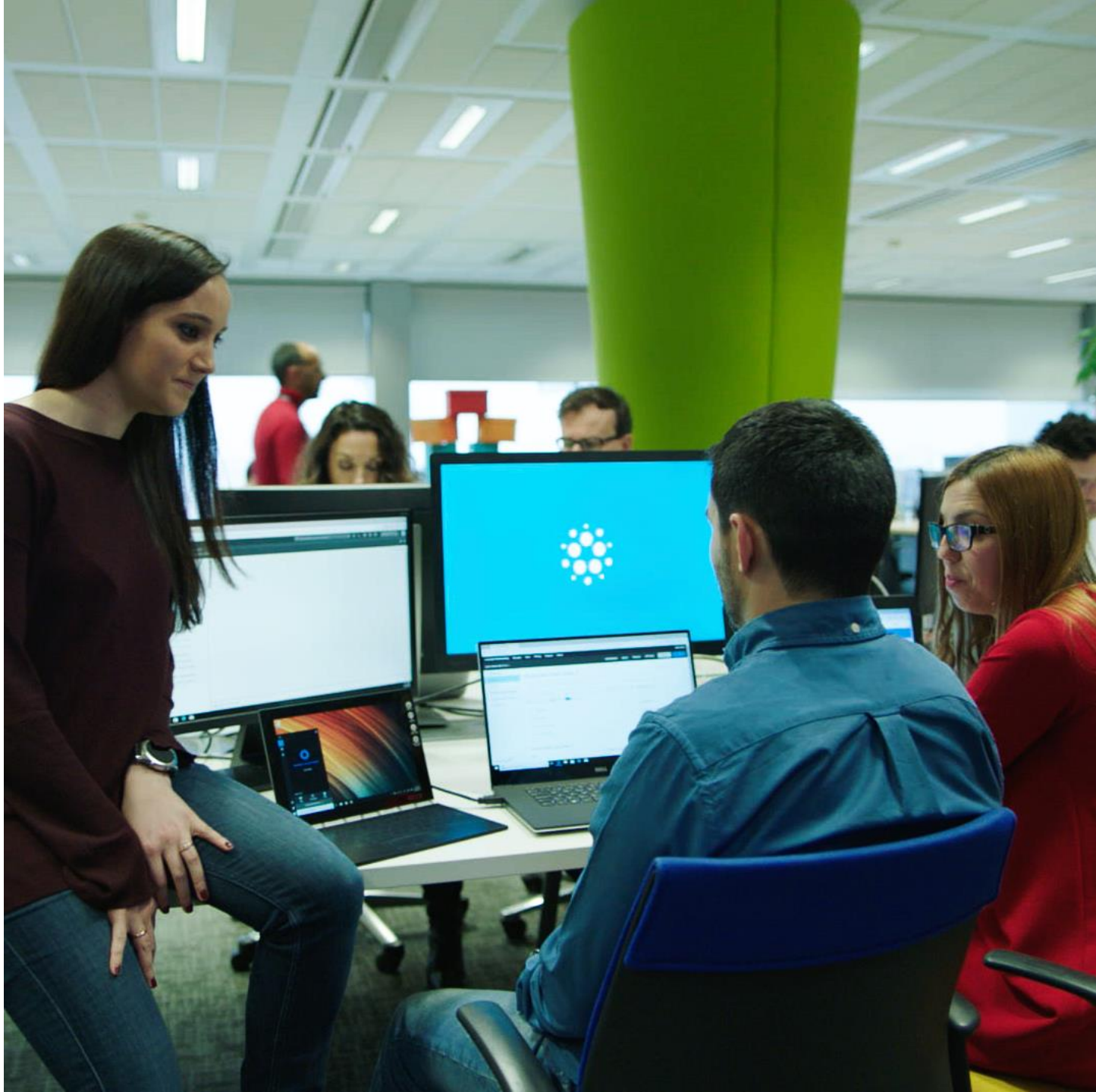
Our customers



Telefónica

"We use Bot Service and Cognitive Services to help people interact simply and naturally with technology."

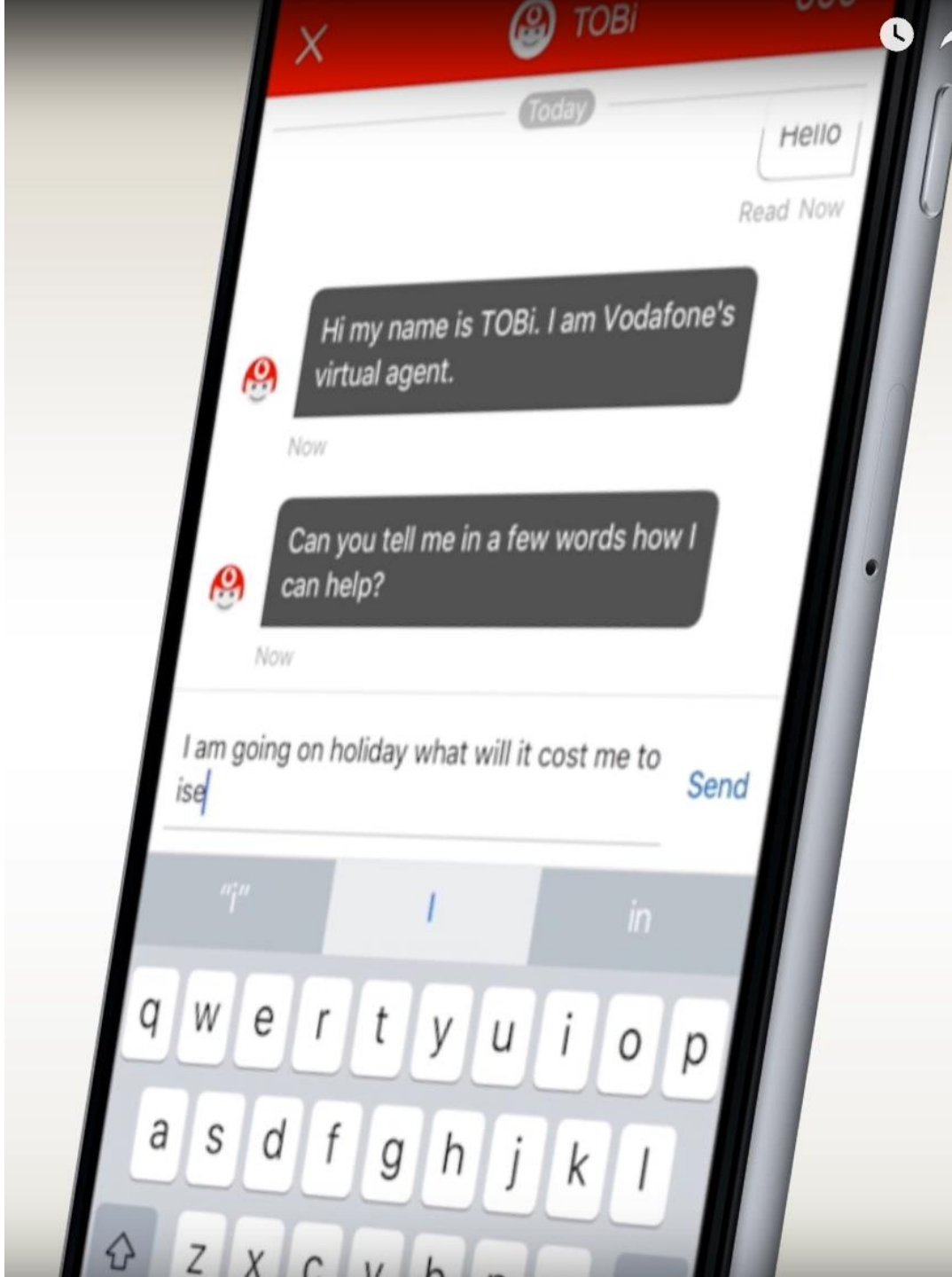
Irene Gomez, Aura Global Director



vodafone

"We work every day to ensure our customers have the best customer experience. This is one of many reasons behind the introduction of our digital assistant, TOBi. The conversational interface has been a revolution making every interaction even simpler and faster.
'And our customers loved it."

Claudio Raimondi, Head of Digital and AI,
Vodafone Italia



"Within five weeks, we had developed a chatbot prototype with the Microsoft Bot Framework…. Our Chief Information and Engineering Officer loved it and asked that we get a version into production in just two months…and that's just what we did"

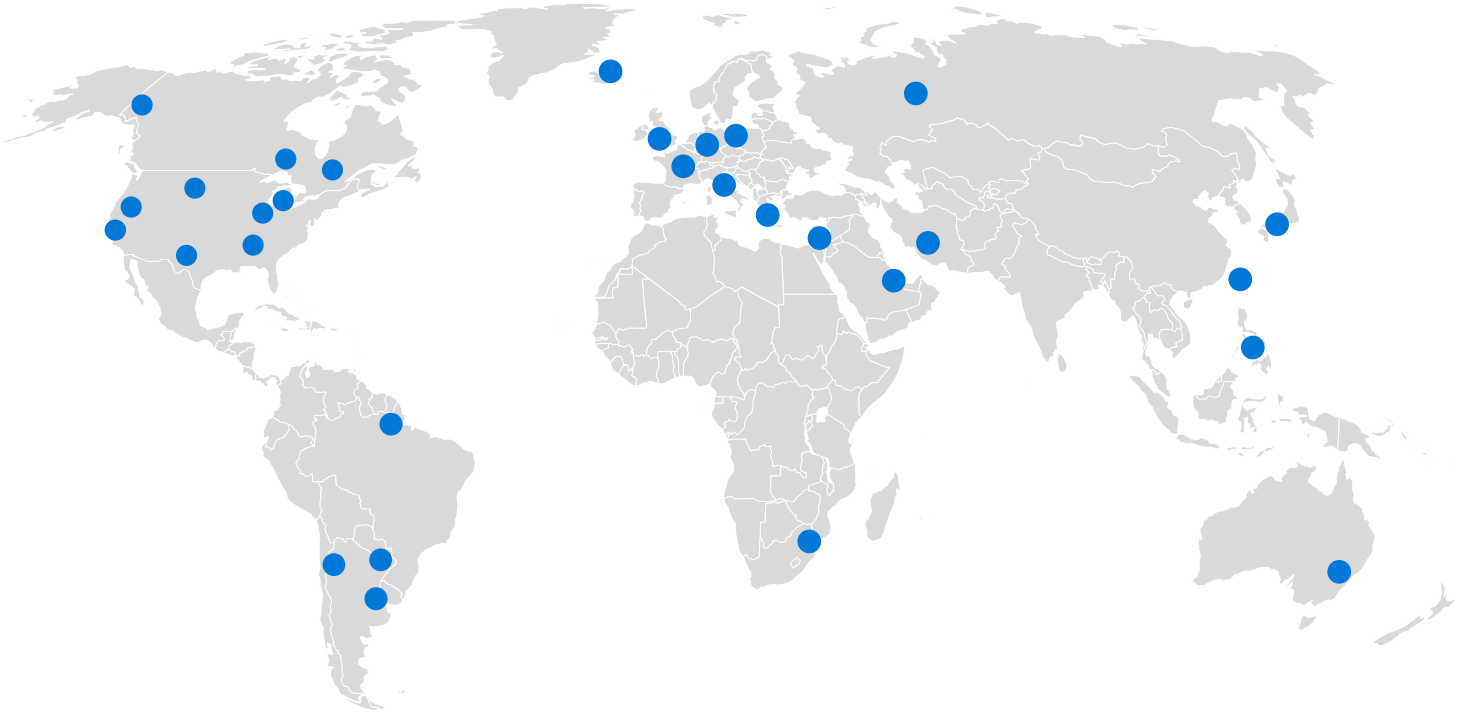
Kumar Athreya
Senior Applications Development Manager of Shipping Systems, UPS

accenture

DiPA is a digital personal assistant that serves the employees "from hire to retire"

OneHR Bot Resolves queries per user location. Proactive (reminders, dates, notices, approvals). Human handoff and is available via: Web, Mobile, Skype for Business, and Cortana.

Conversational AI: Customers



Conversational AI Principles

Complete control of the user experience

Your Brand, Your Personality, Your Voice, **Your Assistant**

Complete ownership and control of data

GDPR, Data Processor and monetization opportunity

Your Assistant, anywhere..

Inside your device/service, Mobile App, Web-Site, Stores

Integrated and Context Aware

Integrated with your ecosystem and infused with intelligence about your customers

External assistant integration

Integrate with your customers chosen assistant to hand off certain questions

Flexible integration

Integrate with your existing architecture and partners.

In the Cloud, On The Edge

Microsoft

Language Understanding Intelligence Service (LUIS)

Language Understanding Cognitive Service (LUIS)

The most comprehensive cloud-based service for Natural Language Understanding (NLU), and the simplest to use for developers with no AI expertise

- Improve recognition through Machine Learning and Teaching
- Extract intent and entities from utterances

What is NLU?

- NLU directly enables Human-Computer Interaction (HCI).
- NLU understanding of natural human languages enables computers to understand commands without the formalized syntax of computer languages and for computers to communicate back to humans in their own languages.
- NLU is tasked with communicating with untrained individuals and understanding their ***intent***, and ***entities*** meaning that NLU goes beyond understanding words and interprets meaning. NLU is even programmed with the ability to understand meaning despite common human errors like mispronunciations or transposed letters or words.
- NLU is fundamental to enable ***dialogues/conversations***

Definitions: Intents, Entities and dialogues

- **Intent:** An intent represents the purpose of a user's input. You define an intent for each type of user request you want your application to support.
- **Entity:** An entity represents a term or object that is relevant to your intents and that provides a specific context for an intent.
- **Dialog:** A dialog is a branching conversation flow that defines responses to the defined intents and entities.

Dialogues: What we Need

"I want to chat"	Turing Test (Talk like a human)	Open Domain Dialogue
"I have a question"	Information Consumption	Task-oriented Dialogue
"I need to get this done"	Task Completion	
"What should I do?"	Decision Support	

What is today's agenda?
What does MSFT stand for?

Book me a flight from Cairo to Seattle
Reserve me a table at The Kitchen for 5 people at 7pm

Is Build conference good to attend?

Chit-Chat vs Task-Oriented

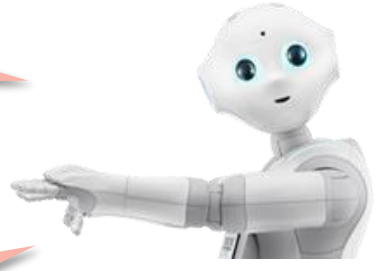
Social Chat
Engaging, Human-Like Interaction
(Ungrounded)

Task-Oriented
Task Completion, Decision Support
(Grounded)

The weather is so depressing these days.

I know, I dislike rain too.
What about a day trip to eastern Washington?

Any recommendation?



Try Dry Falls, it's spectacular!

Machine Teaching

- Interleaves ML algorithms with HCI in the form of human teachers who provide interactions on the utterances to achieve improved performance
- Leverage huge volumes of unlabeled utterances to improve modeling
- Active learning cycle of training, scoring and machine-driven feedback.
  - Cross fertilization of the human expert with the machine learning algorithms
  - Select data points that would be labeled to improve efficiency of the models developed.

Language Understanding (LUIS)

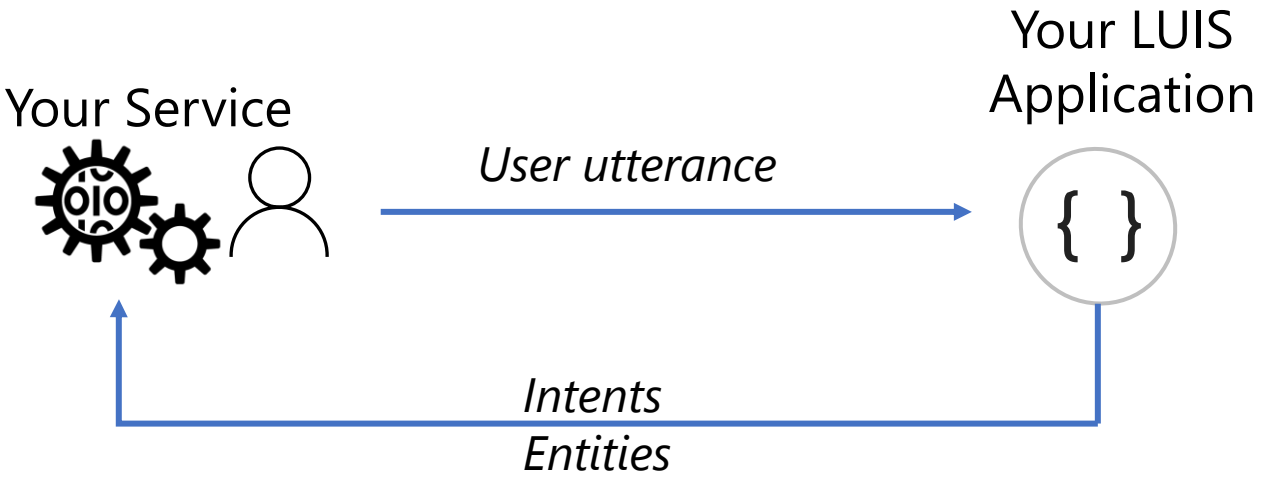
Overview

A web service that enables building your own custom intelligent models using **utterances** in identifying **intents** and **entities**

Utterances: Short text up to 500 characters

Intents: The objective and goal of a sentence

Entities: Valuable information relating to the sentence's objective



Language Understanding (LUIS)

Overview

A web service that enables building your own custom intelligent models using **utterances** in identifying **intents** and **entities**

Utterances: Short text up to 500 characters

"Find me a wireless mouse at $30"

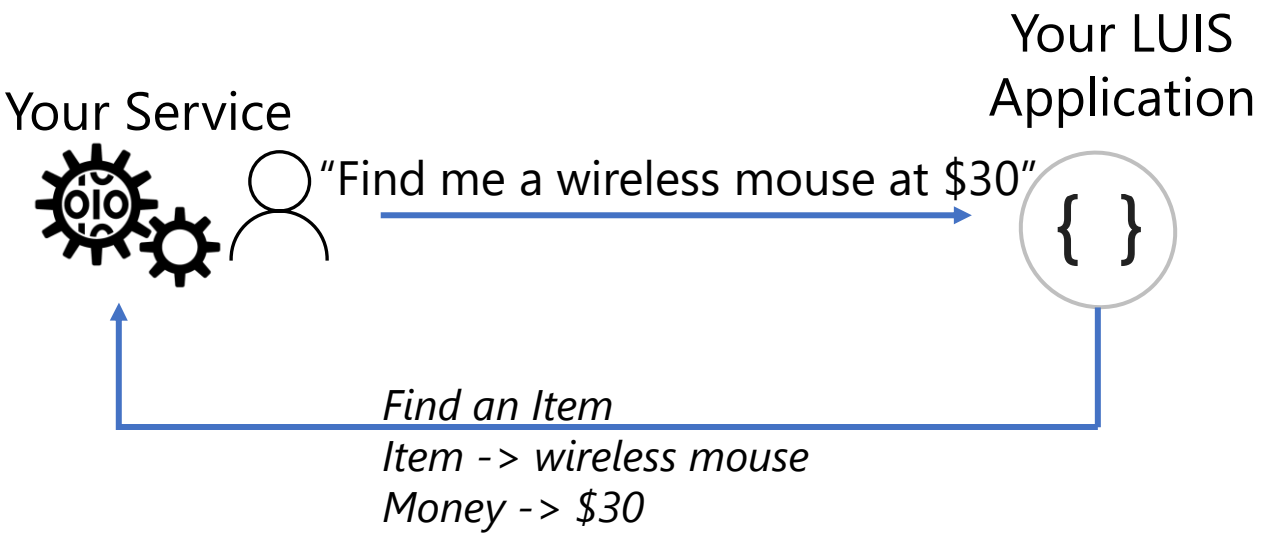
Intents: The objective and goal of a sentence

Find Item

Entities: Valuable information relating to the sentence's objective

Item → wireless mouse;

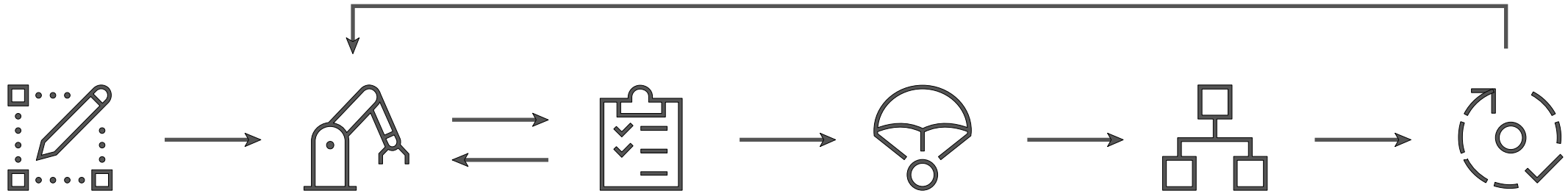
Money → $30




```
{
  "query": "find me a wireless mouse for $30",
  "topScoringIntent": {
    "intent": "FindItem",
    "score": 0.973758459
  },
  "entities": [
    {
      "entity": "wireless mouse",
      "type": "Item",
      "startIndex": 10,
      "endIndex": 23,
      "score": 0.9827788
    },
    {
      "entity": "$ 30",
      "type": "Money",
      "startIndex": 29,
      "endIndex": 31,
      "score": 0.9104368
    }
  ]
}
```

Language Understanding (LUIS)

The Lifecycle of your LUIS Application



Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

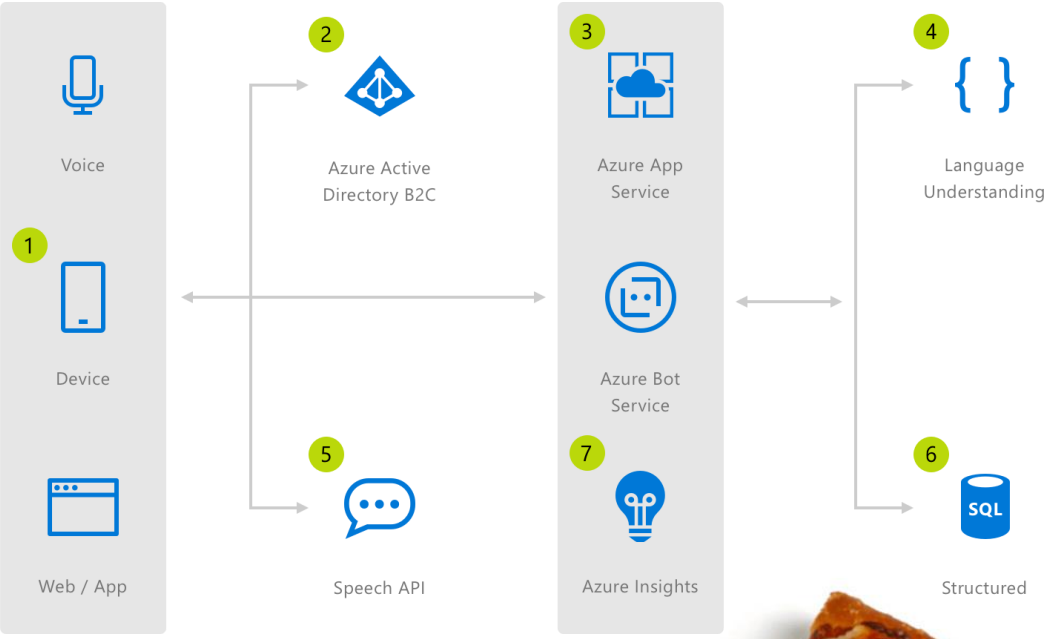
Microsoft

Let’s try a model

Build a model for a Pizza ordering application

- Ordering from the full menu
  - It should handle adding and removing and modifying the contents of a cart
  - Should be able to handle products and product modifications
- Offer different delivery/payment
- Handle “niceties” and other general conversational elements
- Should handle different input modes (text and speech) and work with an E2E bot scenario

Let’s jump in….



Language Understanding (LUIS)

The Lifecycle of your LUIS Application

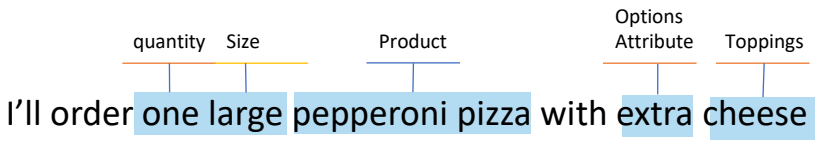
Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

Example Schema

How can we create a scheme for this

- Entities include product, size and quantity etc..
- Map actions to intents (such as adding, removing, modifying, canceling etc..) *is that a good idea?*



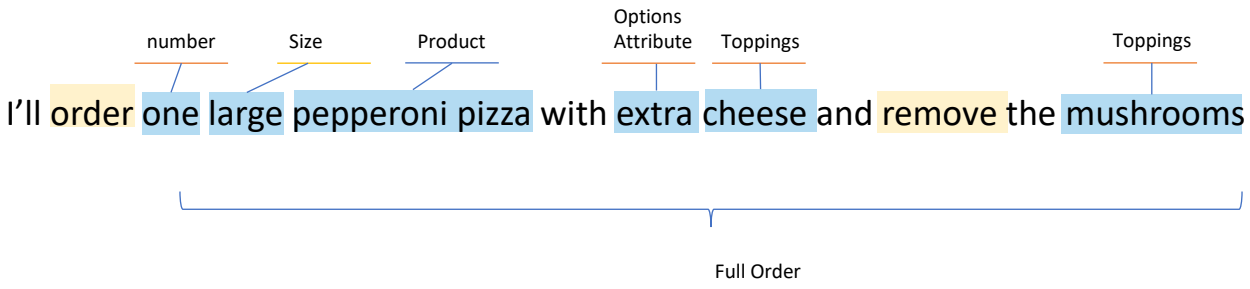

```
"entities":

Simple: "Product“

ClosedLists: "CrustType","CutOptions","Dips", “OrderType”,
"paymentType","PizzaType”,"Sauces","Sizes", "Toppings”, “OptionsAttribute”


PrebuiltEntities: "datetimeV2", "number"
```

```
"intents": [
    {"name": “MakeOrder"},
    {"name": “RemoveOrder"},
    {"name": “EditOrder"},
    {"name": “CancelOrder"},
    {"name": "Confirmation"},
    {"name": "Greetings"},
    {"name": "None"}
  ],
```

Example Utterance

Customers offload multiple elements in the same utterance

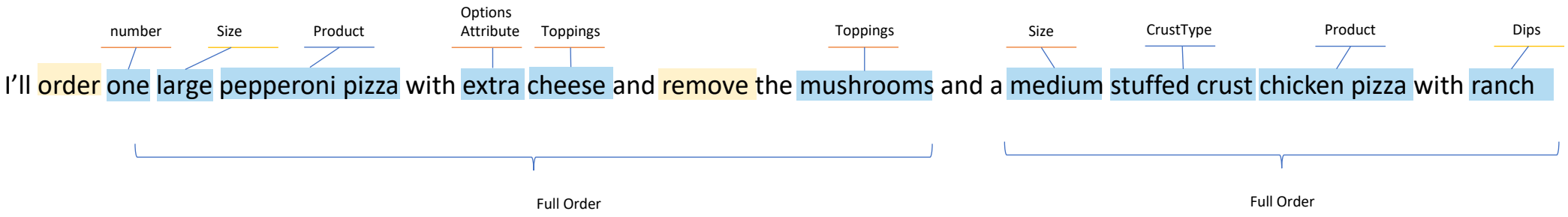
- How do we identify different intents in the same utterance?



Example Utterance

Customers offload multiple elements in the same utterance

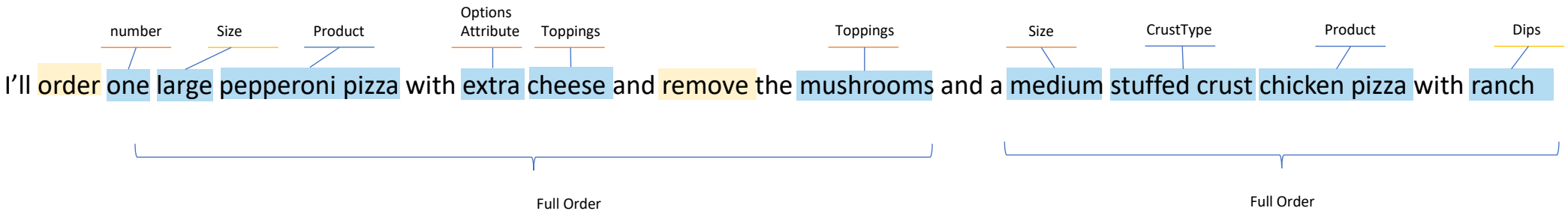
- How do we identify different intents in the same utterance?
- How do we perform multi-level parse in an utterance?



Example Utterance

Customers offload multiple elements in the same utterance

- How do we identify different intents in the same utterance?
- How do we perform multi-level parse in an utterance?



Tip

- Intent driven (Command and control) models not suitable, we need <u>entity driven models</u>
- We need to drive more structure/decomposition in the model

Language Understanding (LUIS): Entity Types

Simple Entity: Describes a single concept. They learn from context using examples through machine learning

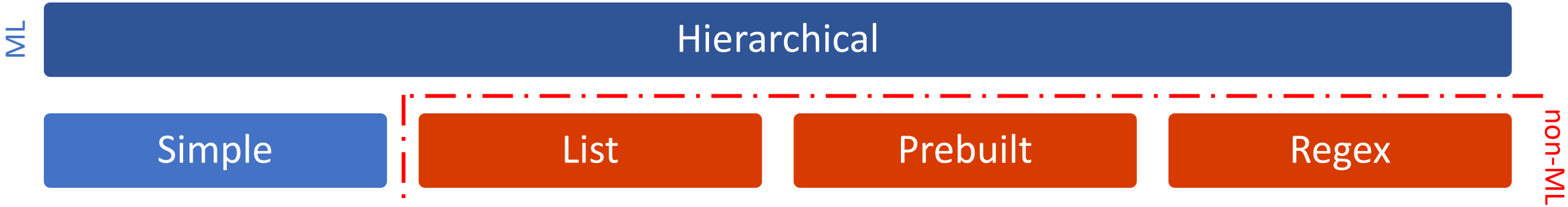
Roles: Named, contextual subtypes of an entity. Extract more information by adding roles to your entities. Roles can be applied to any kind of entity.

Hierarchical: Represents an entity that has parts. It is made up of one or more entities, including supplementary text.

List Entity: A fixed list of words, phrases, and their synonyms that get exact matched in your utterances.

Prebuilt Entity: Common entities provided by LUIS through open source text recognizers available through GitHub, expanded through the community and driven by Microsoft.

Regex Entity: An entity that exact matches based on the regular expression defined.



Language Understanding (LUIS): Entity Usage

Simple Entity: Should be the entity type to use by default

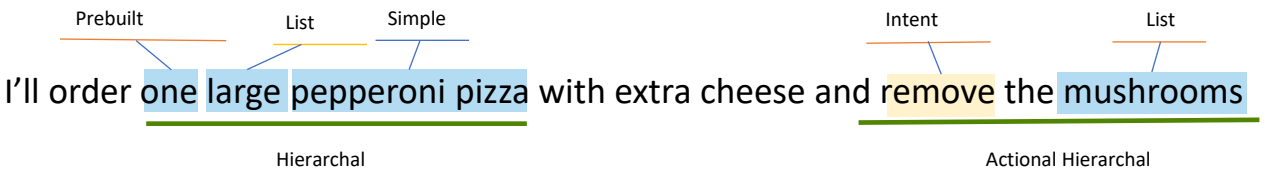
Roles: When the same entity could be used in different contexts (e.g. number could be quantity “three pizzas” or type “four cheese pizza”).

Hierarchical : Used to encode structure, could be multiple entities, or intent/entity elements (e.g. remove {topping})

List Entity: Suitable for a bounded set of elements that rarely change (e.g. pizza size, payment method, etc…)

Prebuilt Entity: Well formed entities that should be leveraged when needed (e.g. datetime, number etc…)

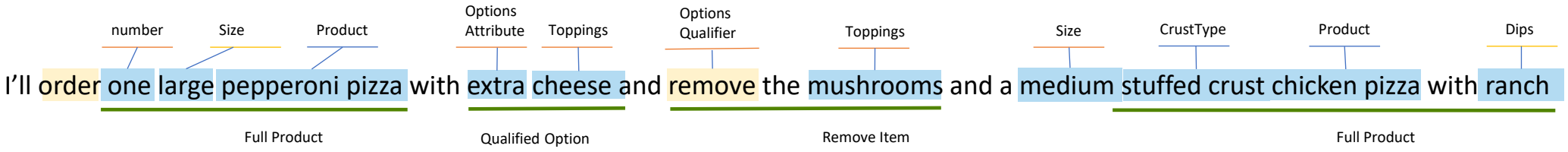
Regex Entity: Well formed alpha-numeric elements (e.g. product codes)



Decomposition

Actionable composites (“Int-ent-ities”)

- Map actions to composite entities (such as adding, removing, modifying, canceling etc..)



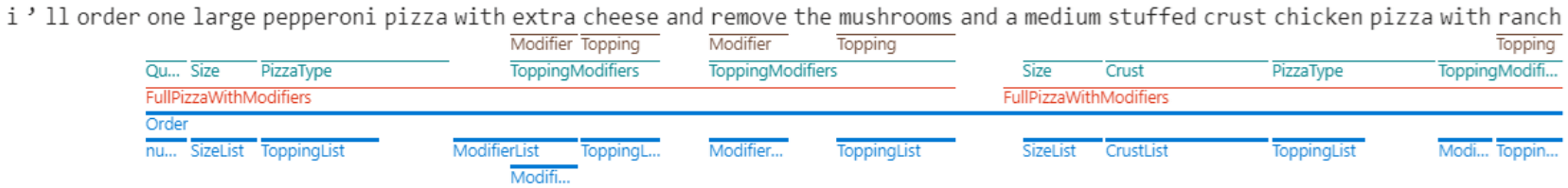

```
"intents“:

    {"name": “ModifyOrder"},
    {"name": “CancelOrder"},
    {"name": "Confirmation"},
    {"name": "Greetings"},
    {"name": "None"}
```

```
"entities":

Simple: "Product“

Composites: "FullProduct", “QualifiedDelivery", "QualifiedOptions", "RemoveItem",
"ReplaceItem", "SideOrder”

ClosedLists: "CrustType","CutOptions","Dips","ModifyOrderDecor", "OPtionsQualifier”,
“OrderType”,"paymentType","PizzaType”,"Sauces","Sizes", "Toppings

PrebuiltEntities: "datetimeV2", "number"
```

Decomposition

Multi-level Parsing

- Use staged hierarchy to perform the multi-level parse of an utterance



Allow decomposition of entities

- Improve the ability to conceptualize the decomposition
- Reduce the amount of rework required with the addition of scheme edits
- Allow to build more sophisticated solutions with less effort

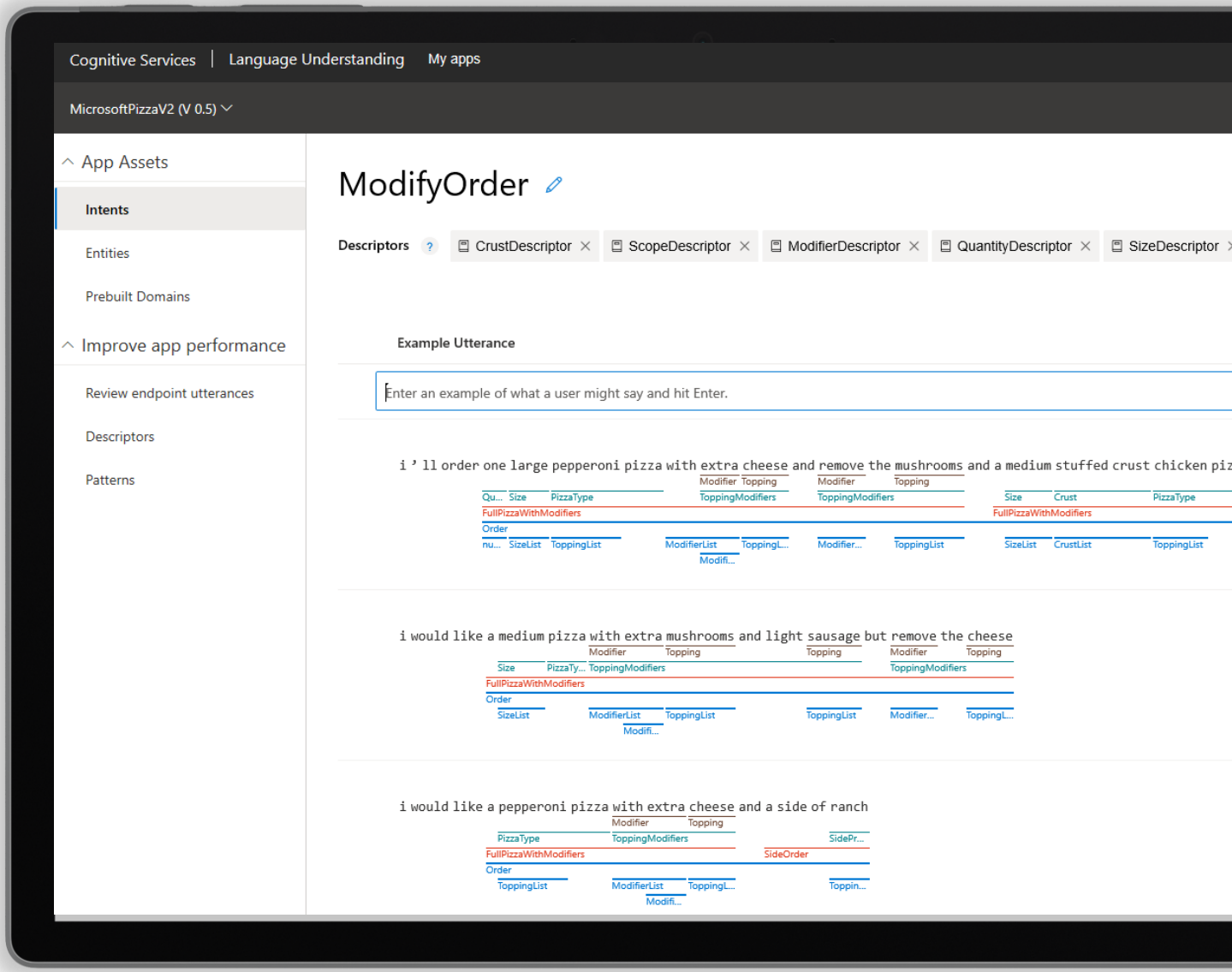
Language Understanding (LUIS)

The Lifecycle of your LUIS Application

Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

http://preview.luis.ai



Languages/Pre-built models

Supporting more languages for pre-built entities

Full support (All Pre-built entities supported)	Major Support (All pre-built entities except Person & Geo.)	Partial Support (Some pre-built entities supported, Person & Geo not supported)
English (US)	Chinese, French (FR,CA), German, Italian, Portuguese (BR), Spanish (ES, MX), Turkish	Dutch, Japanese, Korean, **Hindi, Gujarati, Telugu, Arabic**

New Pre-built domains

English (US), Chinese, Dutch, French (FR), German, Italian, Japanese, Korean, Portuguese (BR), Spanish (ES), Turkish

More coverage upcoming

- Calendar
- Communication
- Email
- Note
- Places
- Restaurant
- Weather
- To Do
- Utilities
- Web
- Home Automation
- **More to come**

Building a Custom Virtual Assistant: Language Understanding & Bot Framework Skills

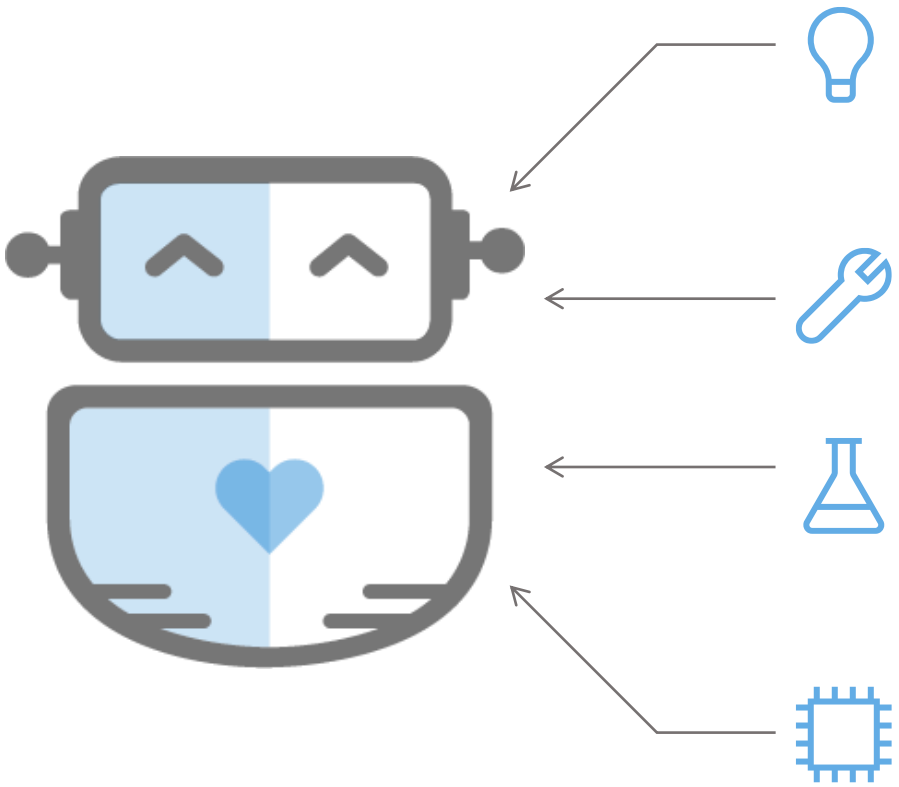
Bot Framework: Building conversational apps & experiences

Bot Framework Skills:

- Uses LUIS & Cognitive Services (QnA, Speech, etc.)
- Include pre-built domains/language models, dialogs, and adaptive cards (GUI)
- Dispatch into prebuilt and custom skills
- Skills contains a self-describing Skill Manifest
- Integrated analytics

Skills are open source, fully customizable

- Skill Template
- C# and Typescript support



Bot Framework Skills

Productivity

- Calendar Skill
- Email Skill
- Todo Skill

Industry

- Automotive
- Point of Interest

Azure Maps FOURSQUARE

Experimental

- News
- Weather
- Restaurant Booking
- Phone
- Hospitality
- Music
- IT Service Management

Bing AccuWeather servicenow

Custom

- Templates
- Your Bot

- Open-Source Re-usable Skills (GitHub repo)
- Skills include Language Models, Dialogs, Responses and integration where appropriate.
- Leverages Adaptive Cards throughout and customer can change design, branding and functionality as required.
- Experimental are sample implementations, limited language models, stubbed integration, etc.
- Speech friendly.

Language Understanding (LUIS) Limits

Limits exist to enable users to leverage **machine teaching** and avoid bloating applications with **too much data.**

Limits per application	
Intents	**500**
List entities	**50** parents + **20,000** children/parent
Patterns	**500**
Machine-learned entities	**100**
Regular expression entities	**20**
Roles per entity	**10**
Utterance character length	**500**
Utterances	**15,000**

Language Understanding (LUIS)

The Lifecycle of your LUIS Application

Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

Language Understanding (LUIS)

Test: Improve

Add user defined signals to boost and explain your models

- **Patterns** are user defined template utterances with placeholders for entities and optional words
  - You can have an intents that are defined through **patterns only** or **mixed with trained data**

- ***Please, add a jacket to cart***
- ***Add the necklace to my cart***
- ***Add album to cart***

"[Please,] add [(a | the | an)] **{Item}** to [my] cart"

- **Descriptors/Dictionaries** are a list of words or phrases that are relevant to your application and domain.
  - Example: "**Retail Terms**: Customer, Inventory, Delivery, Shopping, Incentives, Stock, Pricing"
  - Descriptors augment the language model to **emphasize** words related to your application's intents and entities as opposed to nothing at all

LUIS Bag of Words

↓

Language Model + Out of Vocabulary Words

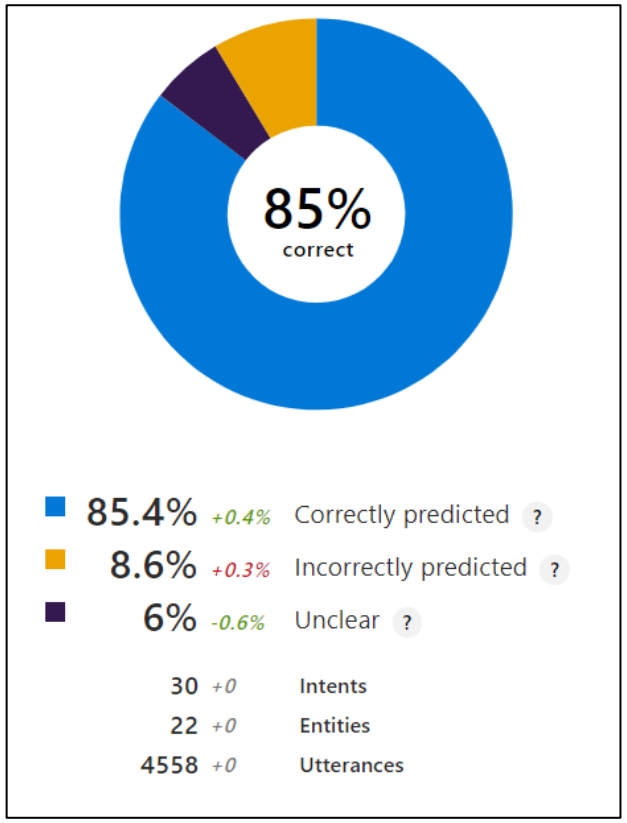
Language Understanding (LUIS)

Test: Improve

Automatically generated insight is provided through a detailed analytics dashboard

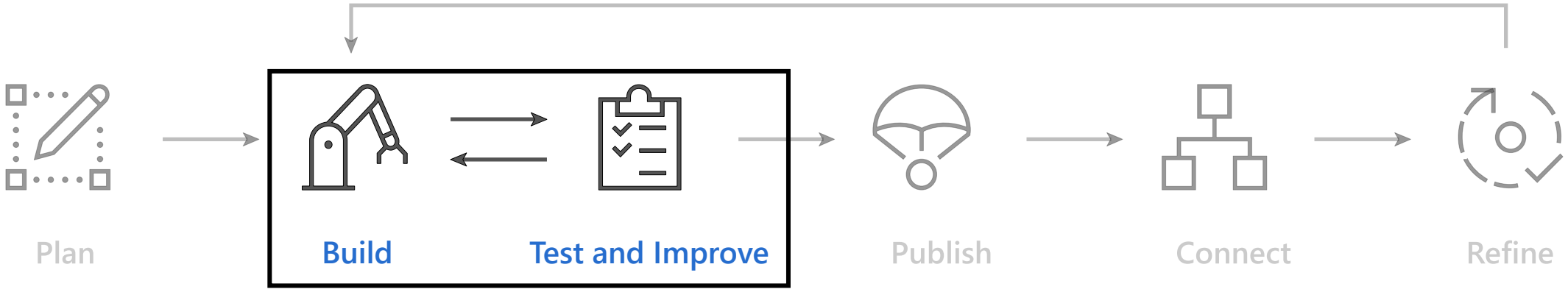
Create more confident models by identifying weak points in your application such as:

- Intents that have much fewer utterances than average
  - Adding more utterances to weak intents
- Intents with high incorrect predictions
  - Adding patterns and phrase lists when necessary
- Intents with consistently similar scores
  - Creating more distinct utterances
- Utterances that are unclear



Language Understanding (LUIS)

The Lifecycle of your LUIS Application



Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

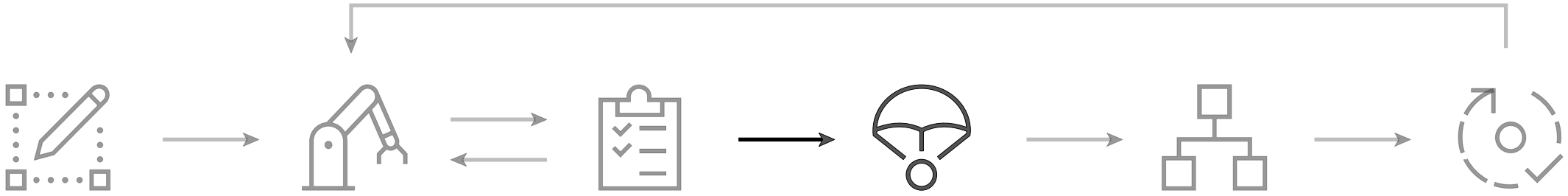
Language Understanding (LUIS)

Authoring Regions

- Build your application using authoring APIs that are hosted in 3 separate regions + Sovereign Clouds
- APIs can be consumed through LUIS' Portal, SDK, or CLI tools

Language Understanding (LUIS)

The Lifecycle of your LUIS Application



Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

Language Understanding (LUIS)

Publish: Shared Runtime Deployment

1. Obtain A Subscription Key	2. Publish	3. Consume Endpoint URL
Create a subscription key by assigning your LUIS application to an **Azure resource** tied to a subscription to keep track of billing	Publish & deploy your application to be used in either a **staging** or **production** environment	An **endpoint URL** is provided when you publish your application to be queried

"find me a wireless keyboard for $30"

*https://<**AzureRegion**>.api.cognitive.microsoft.com/luis/v2.0/apps/<**AppID**>?subscription-key=<**SubscriptionKey**>&q=*

Returns


```
{
  "query": "find me a wireless mouse for $30",
  "topScoringIntent": {
    "intent": "FindItem",
    "score": 0.973758459
  },
  "entities": [
    {
      "entity": "wireless mouse",
      "type": "Item",
      "startIndex": 10,
      "endIndex": 23,
      "score": 0.9827788
    },
    {
      "entity": "$ 30",
      "type": "Money",
      "startIndex": 29,
      "endIndex": 31,
      "score": 0.9104368
    }
  ]
}
```

Language Understanding (LUIS)

Publish: Shared Runtime Deployment: Global Availability

Secure, Compliant, Expanding

- Disaster recovery happens within regional boundaries
- Constantly expanding with Azure and Cognitive Services

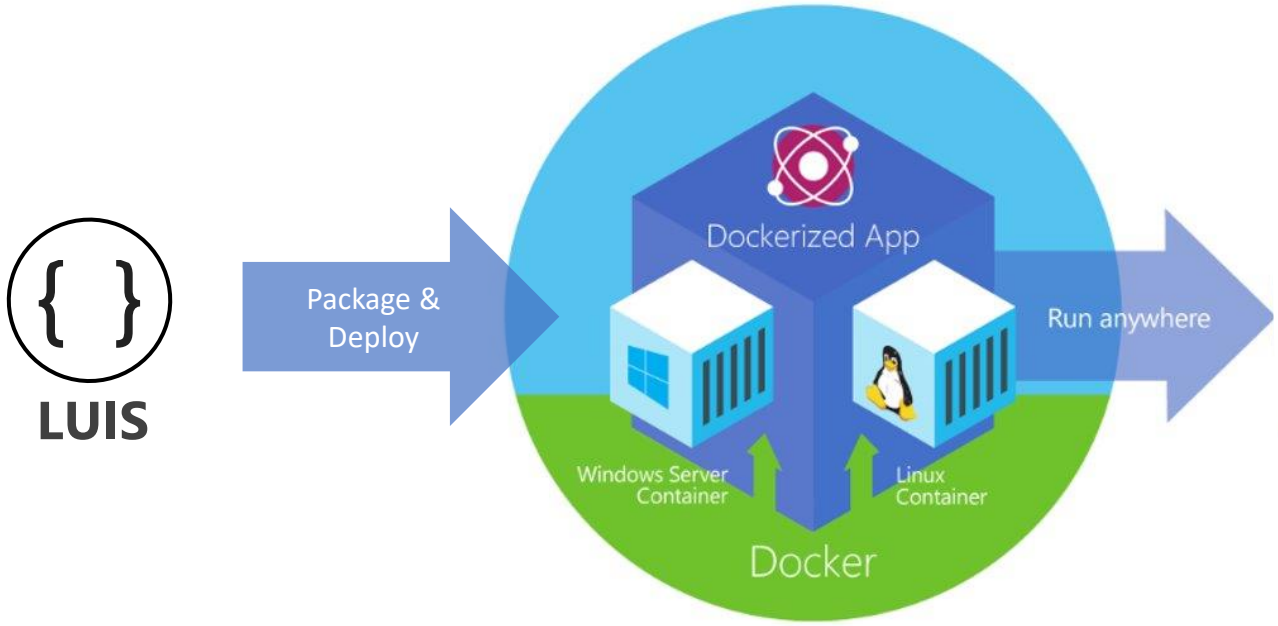
Language Understanding (LUIS)

Publish: Connected Container Runtime Deployment

Package your models into a **connected Docker container** to be deployed **on your premises** or a **docker hosting service in Azure.**

Why Containers?

- Granular control
- Scalable
- Hosting Location
- Enables IoT device integration



Language Understanding (LUIS)

The Lifecycle of your LUIS Application

Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

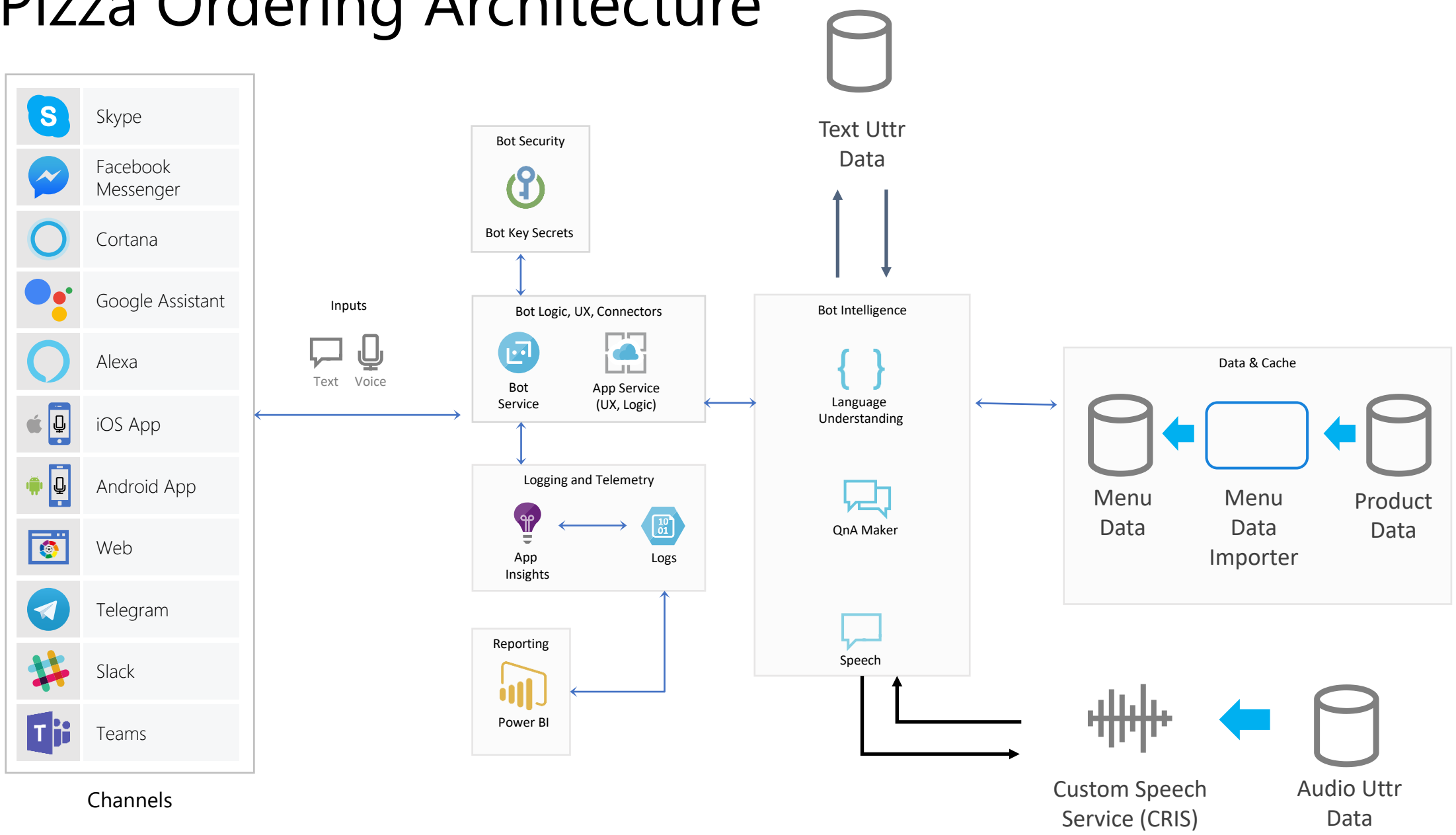
Microsoft

Language Understanding (LUIS)

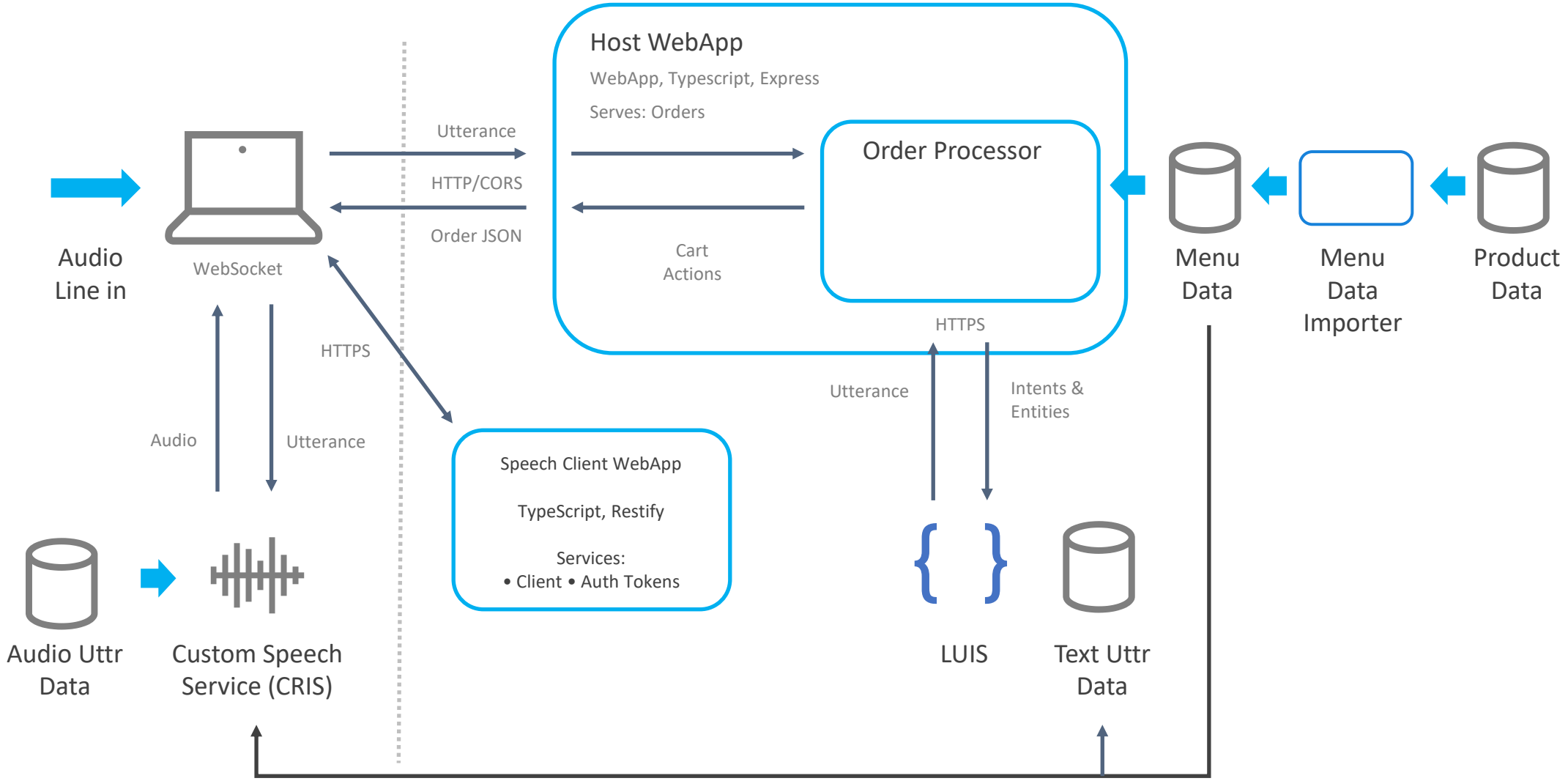
Connect: Bot Framework + Cognitive Services

- Use **Speech Service** through speech priming in LUIS to recognize spoken words
- Enable **Text Analytics** to extract key phrases and leverage sentiment analysis info
- Seamlessly integrate with **Azure Search,** to find and deliver the results you want
- Use a customized **Speller Service** to get better results despite users making errors
- Create complex end to end bots using LUIS within **Bot Framework's** rich ecosystem

Pizza Ordering Architecture

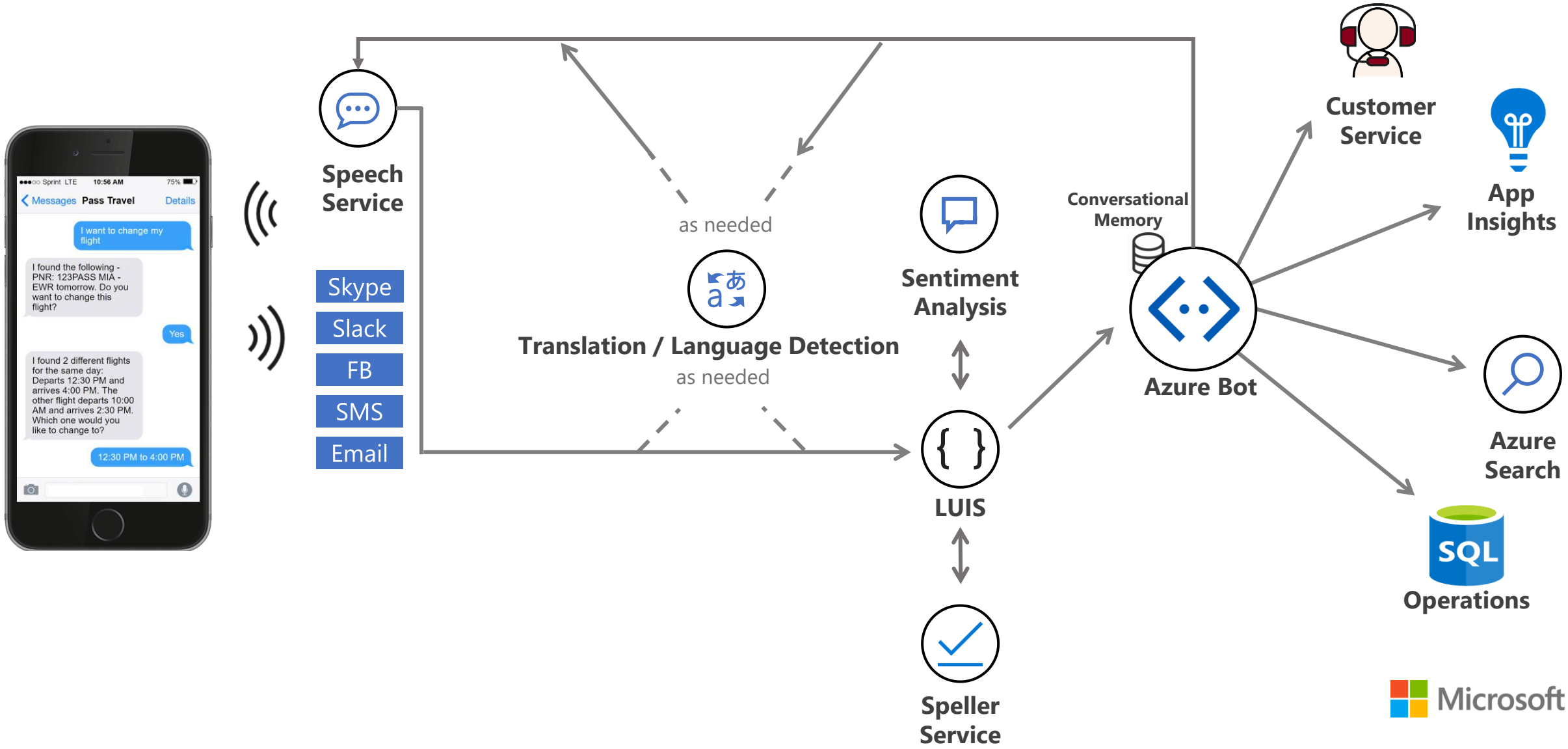


Pizza Ordering Architecture



Language Understanding (LUIS)

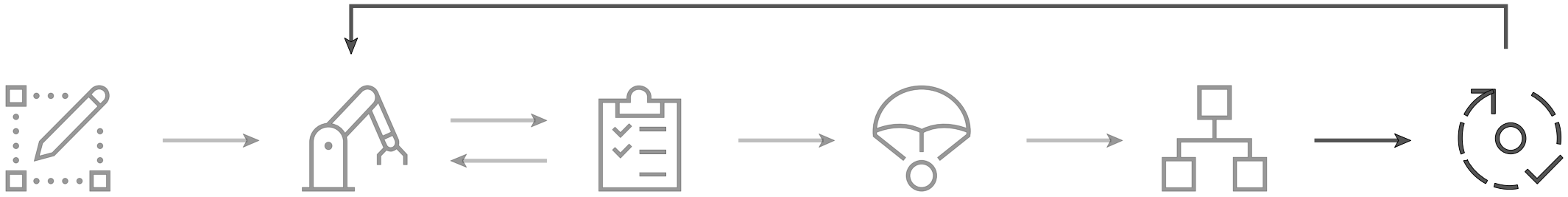
Connect: Bot Framework + Cognitive Services



Microsoft

Language Understanding (LUIS)

The Lifecycle of your LUIS Application

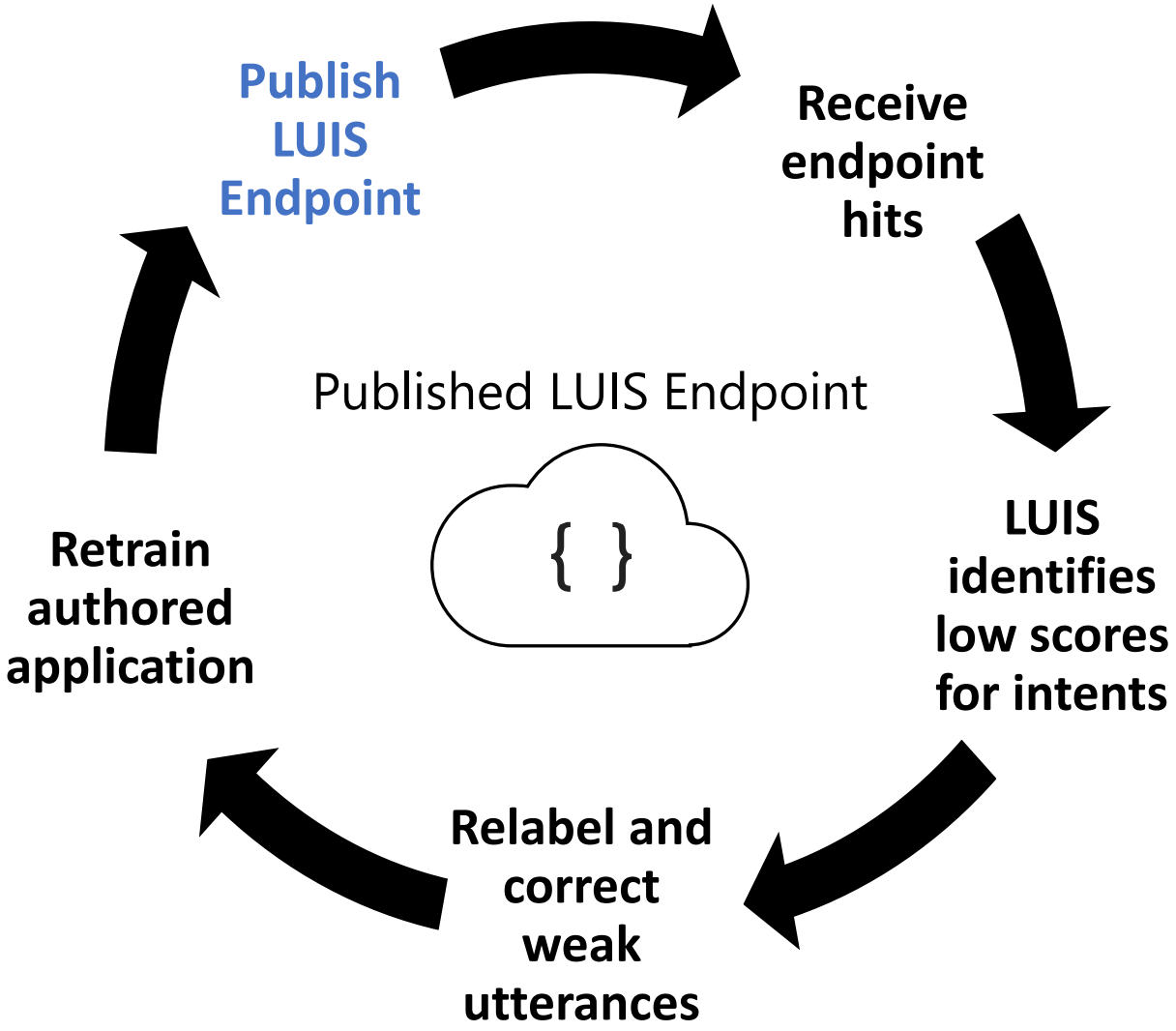


Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Define Intents and Entities	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Training Examples		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

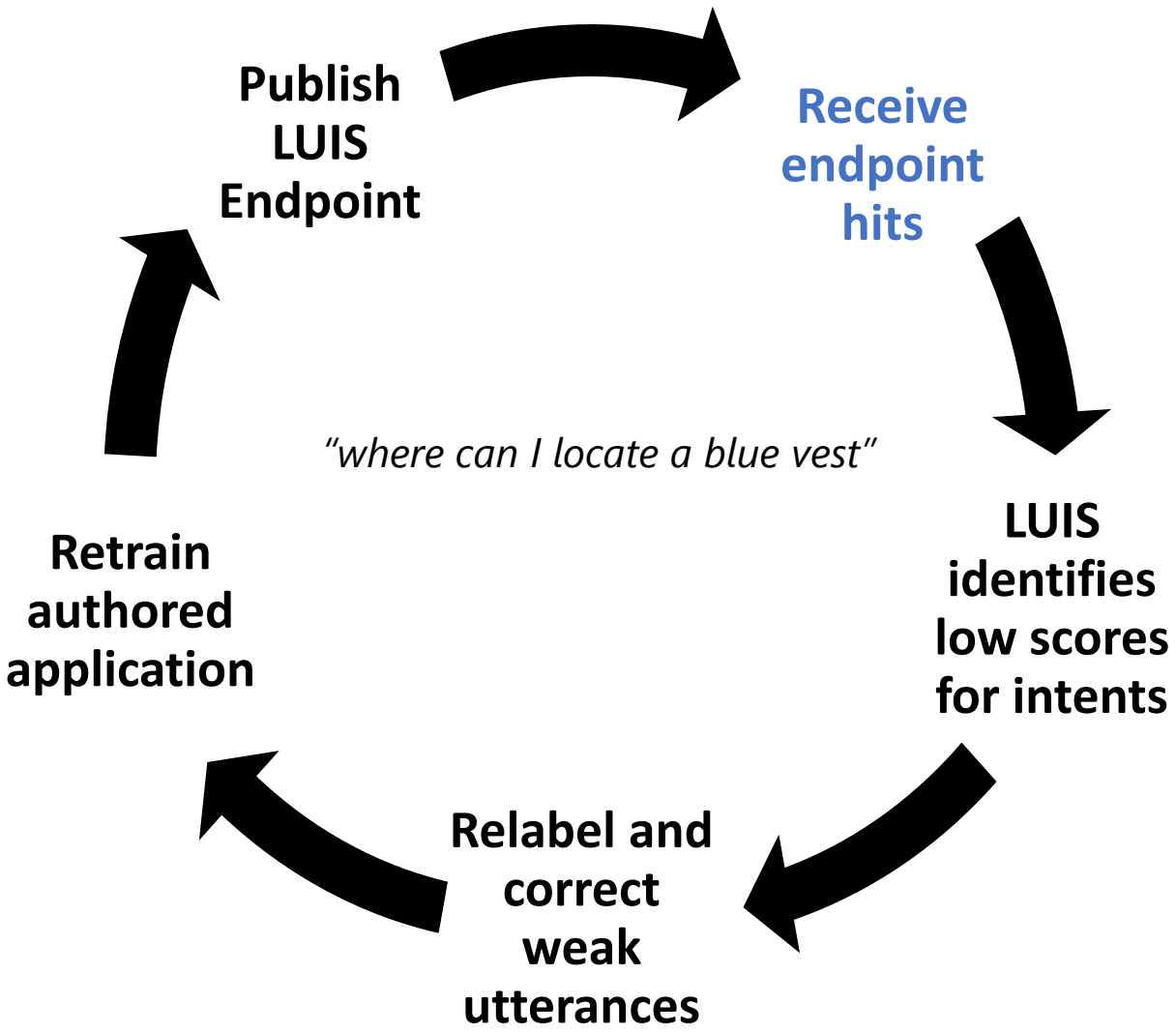
Language Understanding (LUIS)

Evaluate: Active Learning



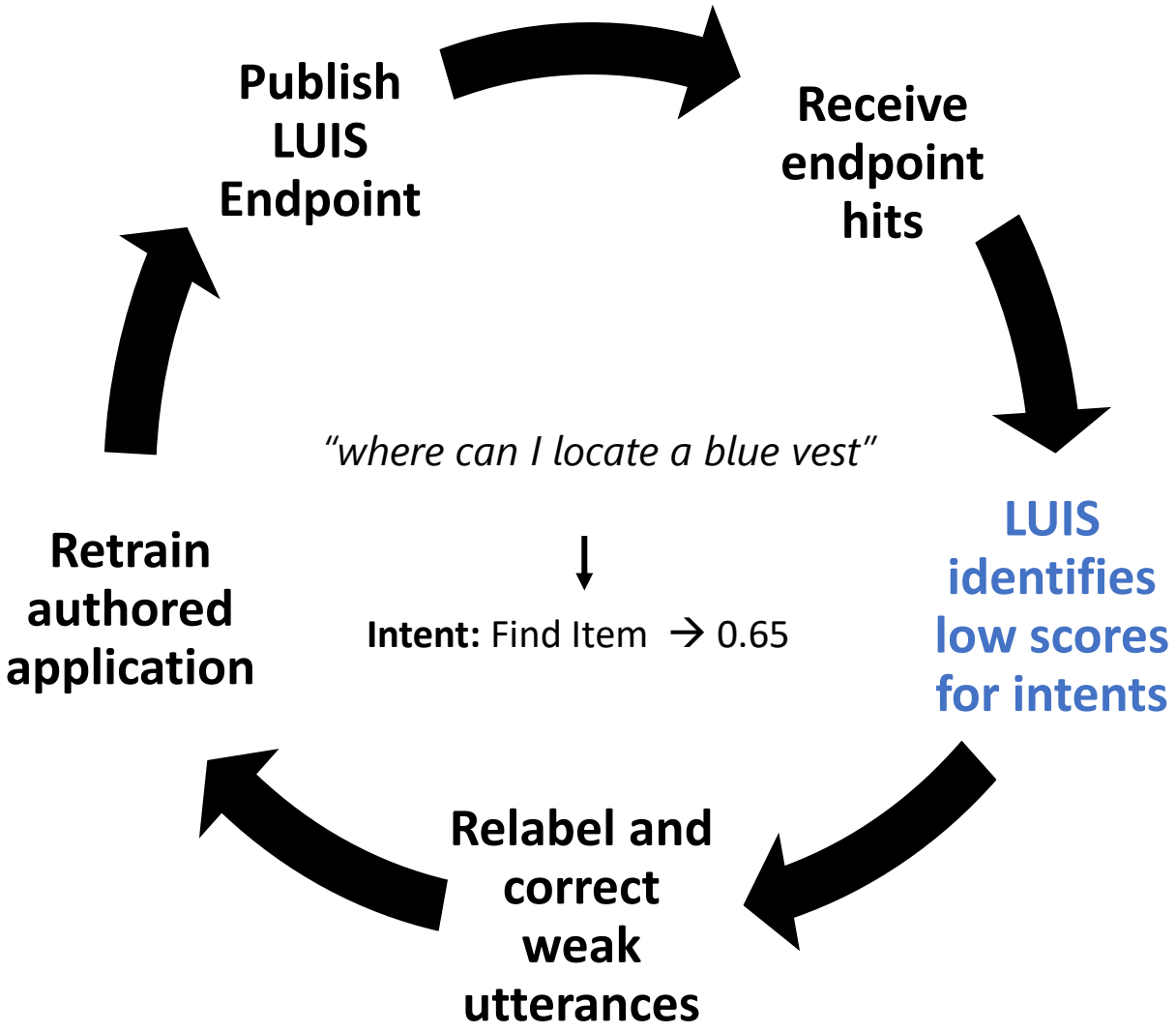
Language Understanding (LUIS)

Evaluate: Active Learning



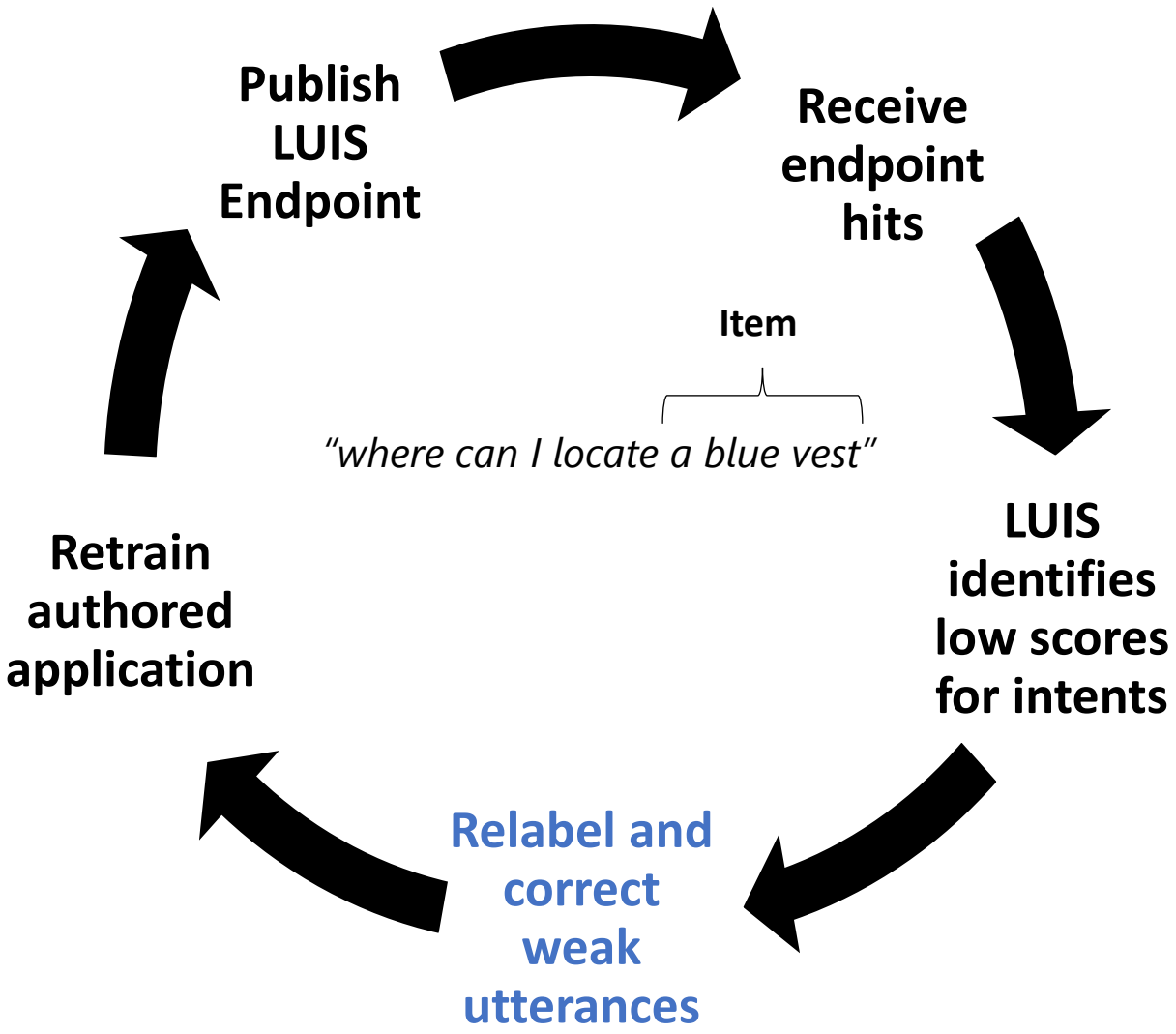
Language Understanding (LUIS)

Evaluate: Active Learning



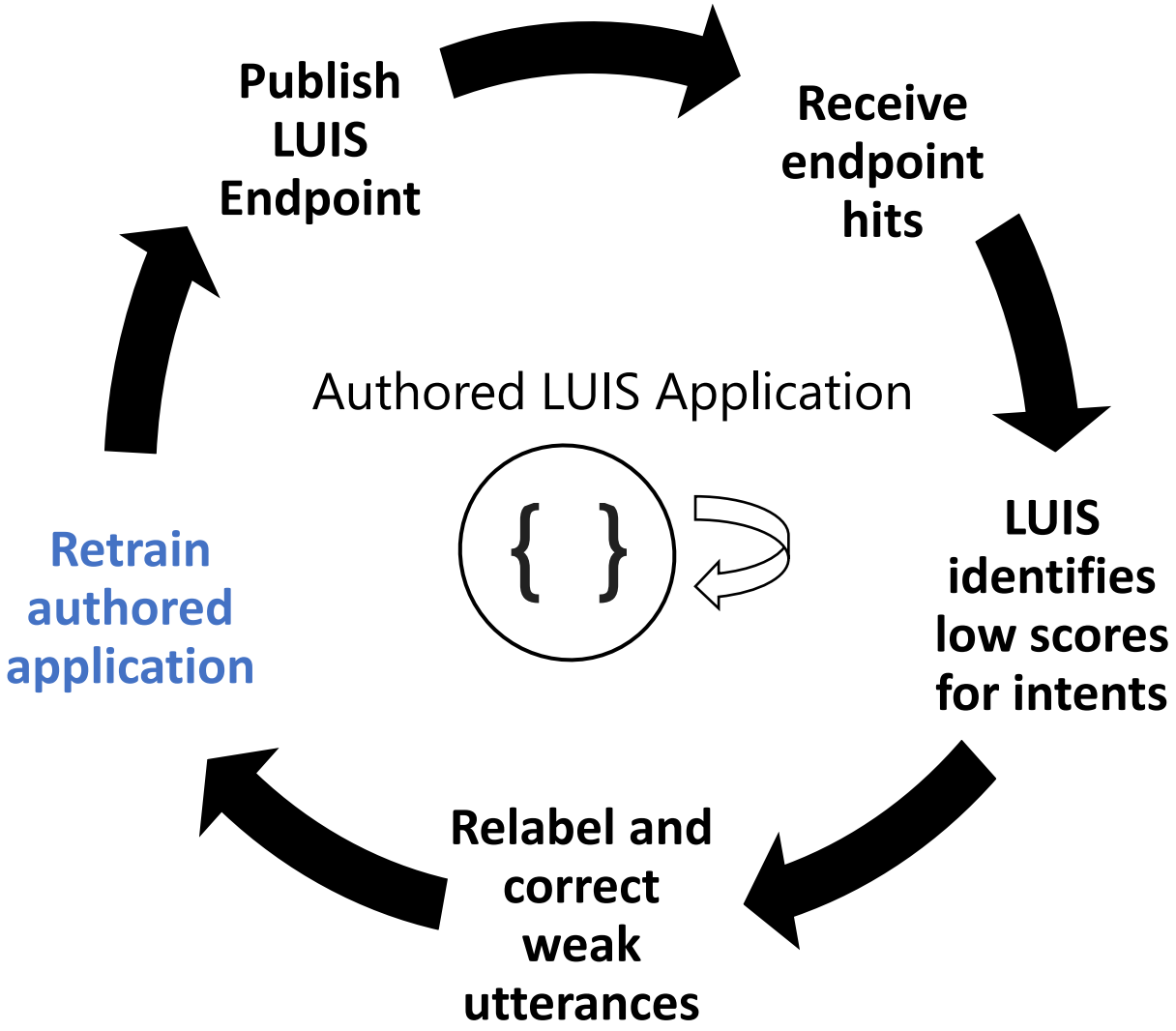
Language Understanding (LUIS)

Evaluate: Active Learning



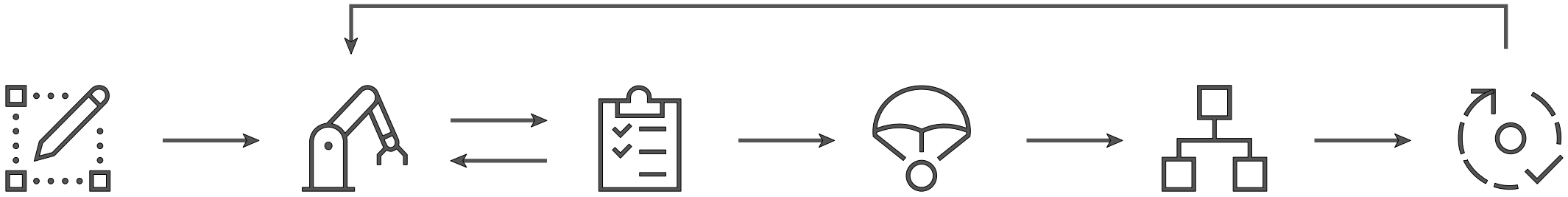
Language Understanding (LUIS)

Evaluate: Active Learning



Language Understanding (LUIS)

The Lifecycle of your LUIS Application

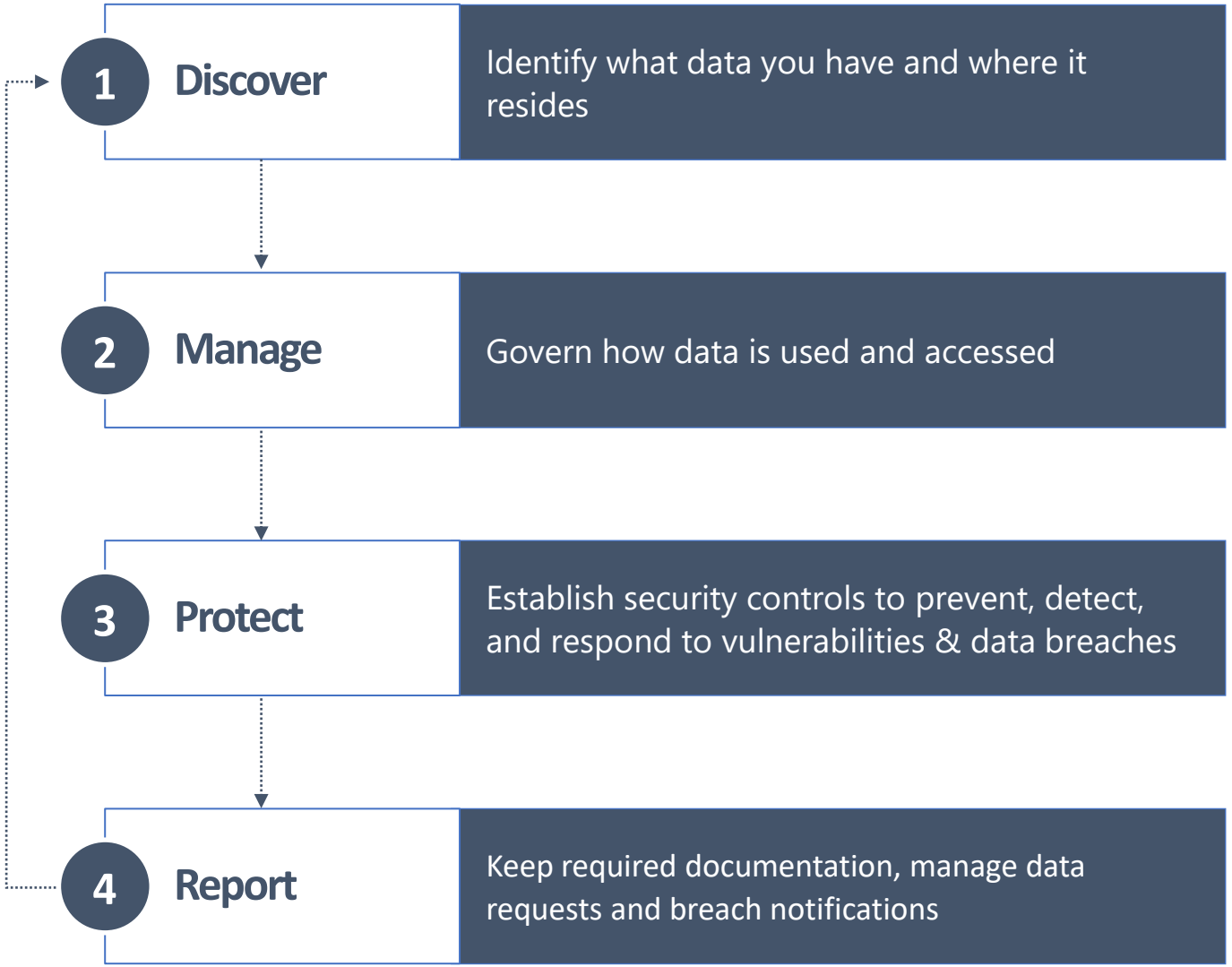


Plan	Build	Test and Improve	Publish	Connect	Refine
Define Actions	Add Training Examples	Evaluate and Improve Model using best practices	Shared Runtime Deployment	Bot Framework	App Insights
Define Information	Add Features		Container Deployment	Integrate with other Cognitive Services	Active Learning
Identify Scenarios					

Microsoft

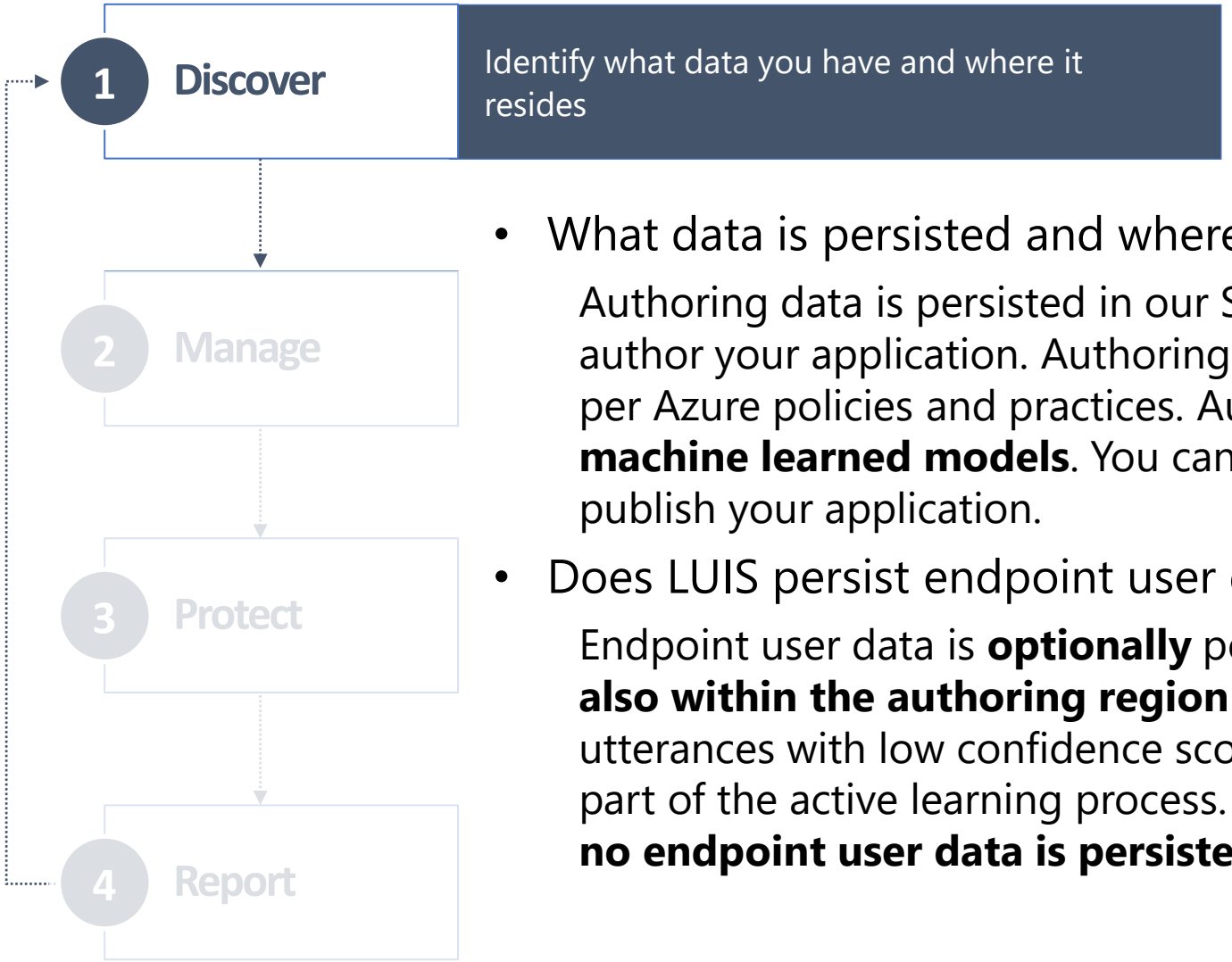
Language Understanding (LUIS)

Security



Language Understanding (LUIS)

Security: Discover



- What data is persisted and where?

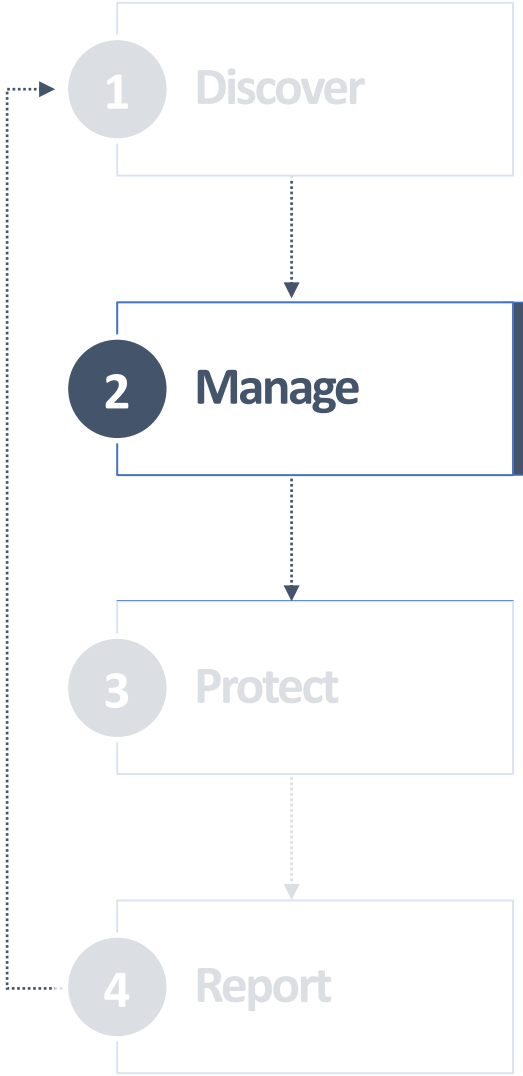
  Authoring data is persisted in our SQL Azure databases in the region you author your application. Authoring data is **encrypted in transit and at rest** as per Azure policies and practices. Authoring data is used to **create your machine learned models**. You can **delete** this data at any time, even after you publish your application.

- Does LUIS persist endpoint user data?

  Endpoint user data is **optionally** persisted for **30 days in your endpoint logs also within the authoring region of your application**. LUIS may suggest utterances with low confidence scores to be added to your authoring data as part of the active learning process. **This feature can be turned off to ensure no endpoint user data is persisted at any time.**

Language Understanding (LUIS)

Security



What happens to my data when I publish my application?

When you publish your application to a certain region, **your machine learned models are compressed as pure binaries** and copied into the specified region. The binaries are exposed via an API that you consume for prediction. The authoring data itself is **not** present in those binaries in any shape or form.

What happens to my data when I publish my application in a container?

LUIS provides a **containerized version of your application**. It compresses your models and exposes it via a container image that you can consume if you prefer to use LUIS **on premises**. This allows for more granular control over where your predictions are processed. Endpoint user queries can still **be persisted within your container** if you enable logging. You can **export your endpoint logs** and import them into your LUIS authored application if you would like to take part in active learning.

Language Understanding (LUIS)

Security

1 Discover

2 Manage

3 **Protect**

Establish security controls to prevent, detect, and respond to vulnerabilities & data breaches

4 Report

Does LUIS use my data to improve its own service?

LUIS does **NOT** use your data to improve its own services. We provide enterprise grade security and compliance.

Does my data, authoring or endpoint user queries, cross regional boundaries?

All data processing happens **within the same regional boundaries** your application is authored or published in as per GDPR compliance.

How much control do I have over my data?

You have **full control** over your data; meaning you can **view, modify, append, export and delete your data at will.**

Language Understanding (LUIS)

Key Differentiators

State Of The Art Platform

The most customizable and feature rich language understanding tool in the market

Pretrained AI services

Powerful tools

Comprehensive platform

Innovate Anywhere

Global endpoint availability to access your application anywhere – in the cloud, on-premises and the edge

Cloud

Edge

On-premises

Integrate, Extend, Customize

Create advanced architectures using LUIS and connect with a variety of other services.

Enterprise Grade

Provide the best level of security and compliance. GDPR, ISO, FedRamp, FOC, PCI, HITRUST, HIPAA

99.5% SLA Uptime for paid tiers

Microsoft

LUIS for Documents

LUIS for Documents

Document classification and entity extraction from unstructured data

During LUIS-D preview you **can** build custom models to:

- Classify digital documents
- Extract entities and clauses from sections of digital documents

During LUIS-D preview you **cannot**:

- Classify scanned documents or content
- Extract content based on document layout

Already know LUIS-C? What’s new in LUIS-D

- Expanding LUIS to handle longer documents and not limiting to 500 characters
- Improved entity extraction through hierarchical schemas
- Decomposition of models into other models that can be shared and re-used
- More intelligent algorithms to select the samples to label from a pool of unlabeled documents that the user uploads or accumulates after the model is deployed
- Pre-built feature repository for domain-specific machine learning features that assist the developer through the model building process
- Cracking of different types of documents to extract text including pdf, docx, pptx, html, eml, msg

Some scenarios with unstructured data in documents

Chat log analysis for customer support

Classification of Documents and its content

Improve QNA for User Manuals

Improve search accuracy of Cognitive search

- Email triaging

Extract information for Auditing documents

Extract information for Backoffice automation

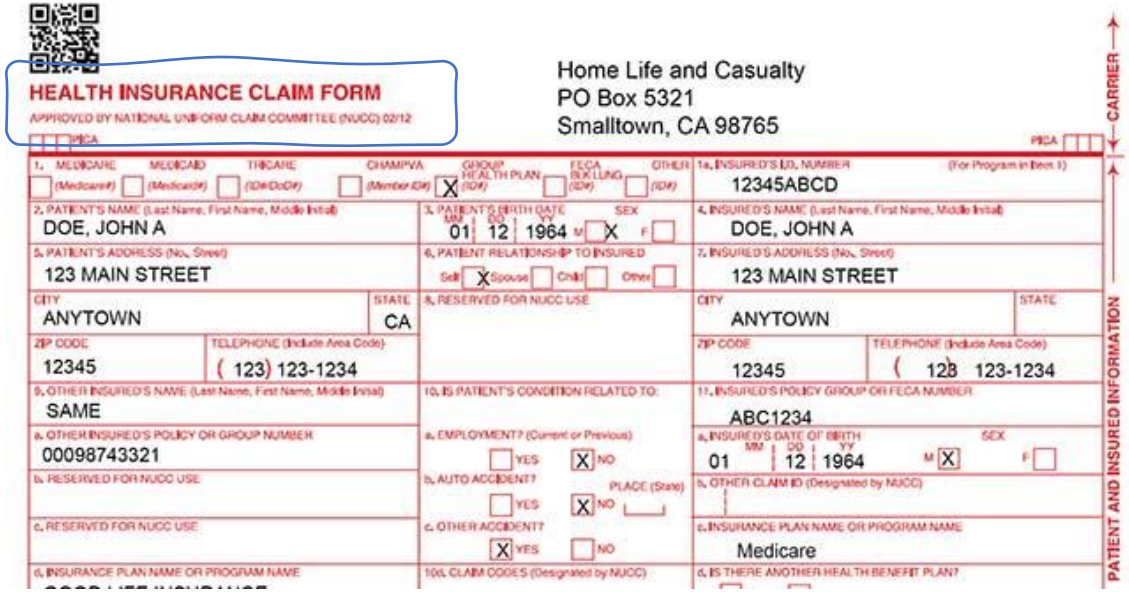
...

Classification of documents

Currently, enterprises have to manually look at the forms and documents to classify them into types. Similar use case applies for customer support where triaging of email tickets is needed.

With LUIS-D, enterprises classify forms, emails or other types of digital documents.

This can be used in triaging to route to different teams or RPA (Robotic Process Automation).



HEALTH INSURANCE CLAIM FORM

APPROVED BY NATIONAL UNIFORM CLAIM COMMITTEE (NUCC) 02/12

Home Life and Casualty
PO Box 5321
Smalltown, CA 98765

CARRIER

PICA

1. MEDICARE (Medicare#) MEDICAID (Medicaid#) TRICARE (ID#/DoD#) CHAMPVA (Member ID#) GROUP HEALTH PLAN (ID#) X FECA BLK LUNG (ID#) OTHER (ID#)

1a. INSURED'S I.D. NUMBER (For Program in Item 1): 12345ABCD

2. PATIENT'S NAME (Last Name, First Name, Middle Initial): DOE, JOHN A

3. PATIENT'S BIRTH DATE MM 01 DD 12 YY 1964 SEX M X F

4. INSURED'S NAME (Last Name, First Name, Middle Initial): DOE, JOHN A

5. PATIENT'S ADDRESS (No., Street): 123 MAIN STREET

6. PATIENT RELATIONSHIP TO INSURED: Self X Spouse Child Other

7. INSURED'S ADDRESS (No., Street): 123 MAIN STREET

CITY: ANYTOWN STATE: CA

8. RESERVED FOR NUCC USE

CITY: ANYTOWN STATE:

ZIP CODE: 12345 TELEPHONE (Include Area Code): (123) 123-1234

ZIP CODE: 12345 TELEPHONE (Include Area Code): (123) 123-1234

9. OTHER INSURED'S NAME (Last Name, First Name, Middle Initial): SAME

10. IS PATIENT'S CONDITION RELATED TO:

11. INSURED'S POLICY GROUP OR FECA NUMBER: ABC1234

a. OTHER INSURED'S POLICY OR GROUP NUMBER: 00098743321

a. EMPLOYMENT? (Current or Previous) YES NO X

a. INSURED'S DATE OF BIRTH MM 01 DD 12 YY 1964 SEX M X F

b. RESERVED FOR NUCC USE

b. AUTO ACCIDENT? YES NO X PLACE (State)

b. OTHER CLAIM ID (Designated by NUCC)

c. RESERVED FOR NUCC USE

c. OTHER ACCIDENT? X YES NO

c. INSURANCE PLAN NAME OR PROGRAM NAME: Medicare

d. INSURANCE PLAN NAME OR PROGRAM NAME

10d. CLAIM CODES (Designated by NUCC)

d. IS THERE ANOTHER HEALTH BENEFIT PLAN?

PATIENT AND INSURED INFORMATION

MOTOR INSURANCE CLAIM FORM

THE ISSUE OF THIS FORM IS NOT TO BE TAKEN AS AN ADMISSION OF LIABILITY

1. Important Instructions :

a. Claim form is to be filled in capital letter & signed by the insured.
b. Please do not leave any column unanswered.
c. All facts and Statements must be factual not influenced or biased in any form.
d. The damaged vehicle must be parked at safe place to avoid any subsequent loss/theft. Company will not be responsible for the same.
e. Please read carefully the attached list of documents required to speed up processing of your claim.

2. Policy Holder Details

Policy No. : _____ Cover Note No. : _____
Period of Insurance : From _____ To : _____
Name of the Insured : _____ Phone Off. : _____
Gender : Male / Female Date of Birth : D D M M Y Y
Address (Please note - If the Claim is approved, the Claim payment Cheque shall be dispatched at the address mentioned herein) Phone Res. : _____
Mobile : _____
PIN : _____ E mail : _____

3. Vehicle Details

Regd. No. : _____ Make : _____ Date of 1st Registration : _____
Chassis No. : _____ Engine No. : _____ Date of Transfer (if applicable) : _____
Type of Fuel : _____ Colour of Vehicle : _____

Clause extraction

Currently, enterprises have to manually extract clauses from contracts, legal documents, etc.

With LUIS-D, enterprises extract a particular field or clause by training multiple custom extraction models.

This can be used in RPA (Robotic Process Automation) or as metadata to improve search results.

Major Work

Patients receiving major work (crowns, bridges, dentures) or bleaching kits must have their portions, including lab fees and parts fees, completely paid off before the work can be delivered or cemented.

Cancellation Policy

Our time is as important as yours. We attempt to schedule as efficiently as possible to reduce waiting time. We require patients to cancel appointments the day before the appointment. A broken appointment fee is charged when the patient does not show up for an appointment, or cancels the same day of the appointment. The fee is:

$50.00 for a 30 or 45 minute appointment
$60.00 for an hour appointment
$10.00 for each additional 15 minute increment

Leaving an understandable message on the answering machine is acceptable for canceling an appointment and is not subject to the cancellation charges if the appointment was scheduled after 11:00 am.

Finance Charges

Payment is expected at time of services rendered. Any outstanding balance which is overdue by more than 30 days will have a 14.5% APR finance charge or .0397% daily periodic rate added. A payment plan is available through Care Credit. A short application will need to be completed and approved before charges can be incurred. If you are interested, please ask any of the staff members for an application.

Returned Checks

There will be a returned check fee of $20 for any bounced check. This fee may increase depending on the bank's charges. This fee will be added to the outstanding balance and may incur finance charges if not paid within the 30 day grace period.

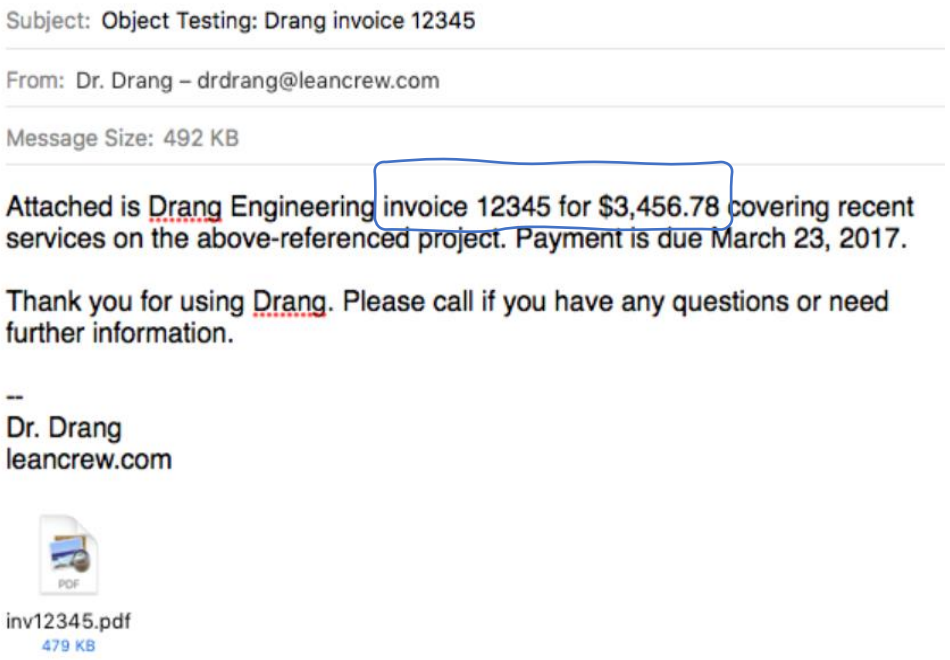
I understand to the financial policies of Marc N Rubinstein DDS LLC and agree to them.

Email triaging

Currently, enterprises have to manually go through email content to extract numbers, technical queries, or similar information.

With LUIS-D, enterprises extract information from emails by training multiple custom extraction models.

This can be used in triaging to route emails to different teams or RPA (Robotic Process Automation).


Subject: Object Testing: Drang invoice 12345

From: Dr. Drang – drdrang@leancrew.com

Message Size: 492 KB

Attached is Drang Engineering invoice 12345 for $3,456.78 covering recent services on the above-referenced project. Payment is due March 23, 2017.

Thank you for using Drang. Please call if you have any questions or need further information.

--
Dr. Drang
leancrew.com

inv12345.pdf
479 KB

Chat log analysis

Currently, enterprises can use LUIS-C to analyse utterances during a chat session.

With LUIS-D, enterprises can analyze a full chat log to audit if the agent followed the standard operating procedure (SOP).

This can help improve SOP and can improve quality assurance for better customer satisfaction.

Re: Chat on website MyWebsite.com [1519862-2282]

Agent, MyWebsite.com

Here's our chat log from today.

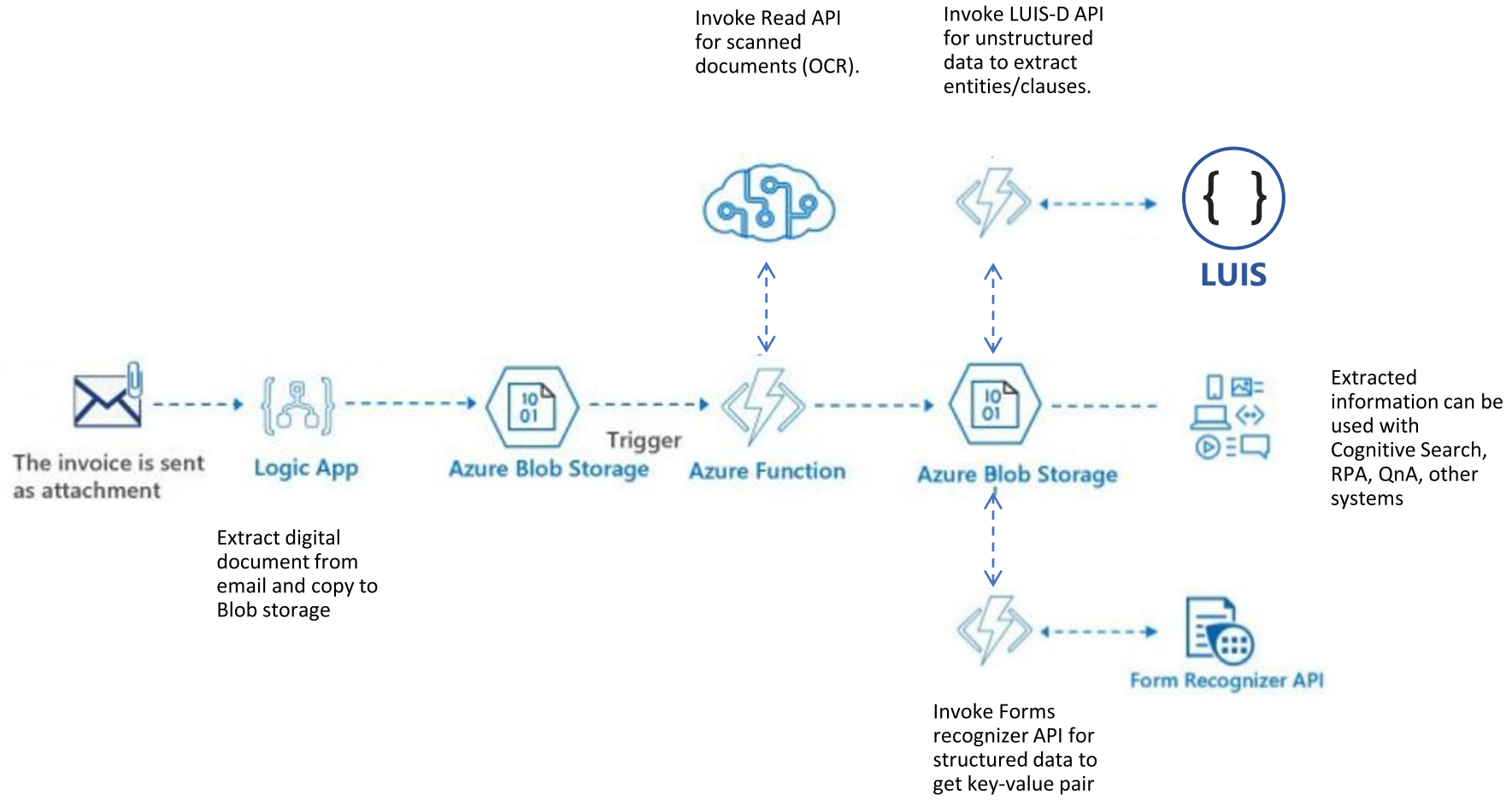
—

Agent Suporte

Chat history from website MyWebsite.com on July 24, 2019
07:56 Agent, suporte: Hello! How may I help you?
07:56 You: Hi there! I would like to know more about the product "x".
07:57 Agent, suporte: Sure! The product "x" is a, b, c, d, e, and f. It's amazing!
07:59 You: Could you please send me this chat log by email?
07:59 Agent, suporte: Absolutely!
08:01 Agent, suporte: Here's our chat log from today.

A comprehensive document processing solution

LUIS-D + Cognitive Services



Microsoft

Differentiation

Quickly start building models with unlabeled data

Unlike the competition, LUIS-D doesn't assume the existence of labeled data. It only starts with unlabeled data and guides the user through which examples to label using smart sampling techniques

Hierarchical schemas

Ability to build sophisticated models with hierarchical schemas. None of the competitors support hierarchical entities

LUIS custom machine learned entities are the best with marginal difference from the competition based on benchmark datasets

Interpretable models

Interpretable models that can be easily debugged with guidance on how to improve the model over time

Machine Learning features catalog

A catalog of machine learning features with smart feature suggestion algorithms that propose the most relevant catalog features to the user

Model as a feature for another model

Ability to decompose their problems into sub-models that can be used as features to other models. This has the added benefit of building models that can be shared and reused. For example, an address extractor can make use of a city name extractor as a feature. The city name extractor can be used in other applications

Multiple document formats

Document cracking for PDFs, docx, pptx, HTML, eml, msg

Containerized deployment

LUIS-D will be available on containers which gives unlimited QPS and lots of control to deploy cross-plat and scale

Microsoft

Thank You