

Preparation to exam AI 100

« Designing and Implementing an Azure AI Solution »

Module 2 – Design AI Solution (40-45%)

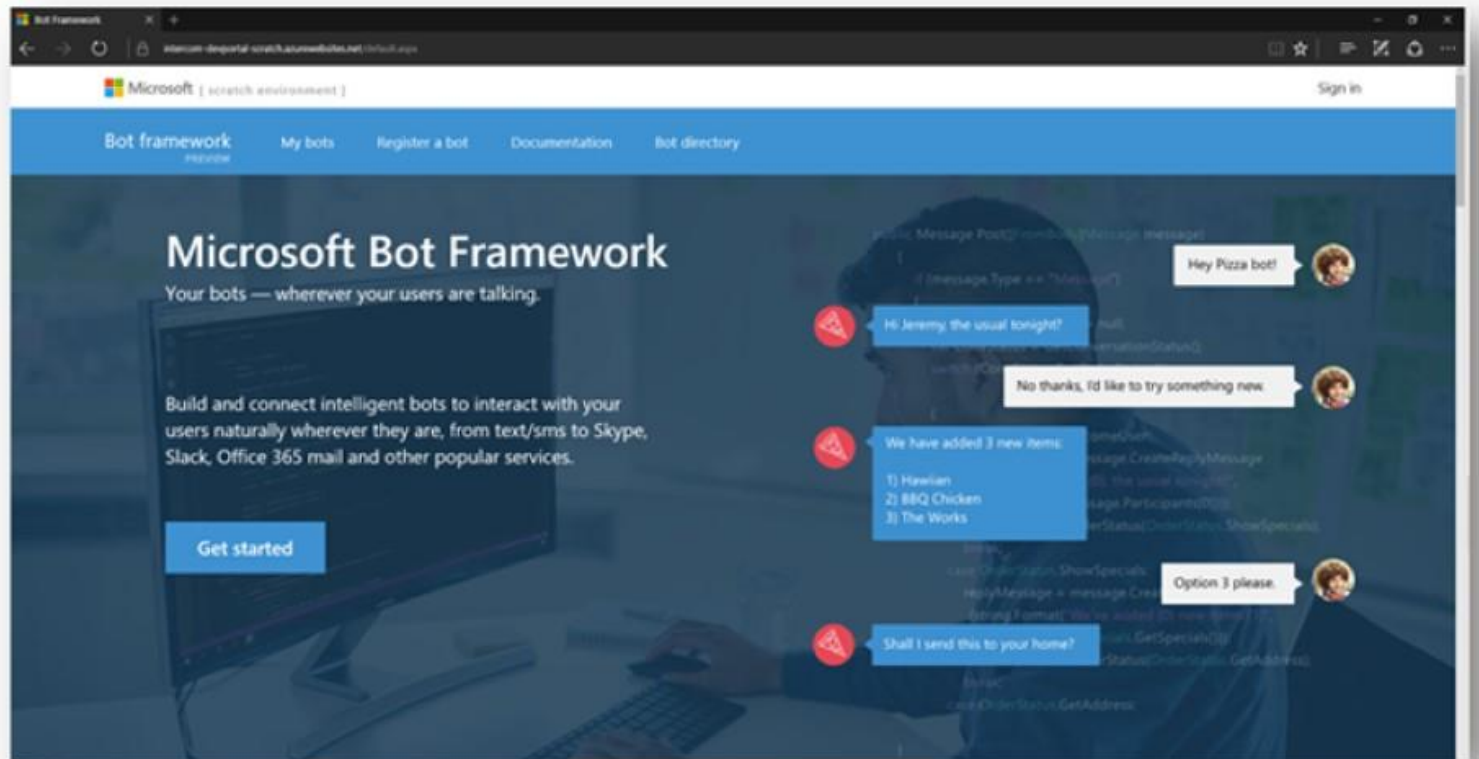
Mohammed Arif
Senior Technical Consultant



Design solutions that
implement the Bot Framework

Conversational AI

- Azure Bot Services enables to build intelligent, Enterprise-grade bots



What is a bot ?

- bot can be used to shift simple, repetitive tasks, such as taking a dinner reservation or gathering profile information, on to automated systems that may no longer require direct human intervention
 - What makes bots unique is their use of mechanisms generally reserved for human-to-human communication
- Users converse with a bot using text, interactive cards, and speech. A bot interaction can be a quick question and answer, or it can be a sophisticated conversation that intelligently provides access to services
- Bots can do the same things other types of software can do - read and write files, use databases and APIs, and do the regular computational tasks.

Create and integrate bots

- Azure Bot Service provides tools to build, test, deploy, and manage intelligent bots all in one place
- Through the use of modular and extensible framework provided by the SDK, tools, templates, and AI services developers can create bots that use speech, understand natural language, handle questions and answers, and more

Building a bot

- Azure Bot Service offers an integrated set of tools and services to facilitate this process
- It provides tools for various stages of bot development to help you design and build bots



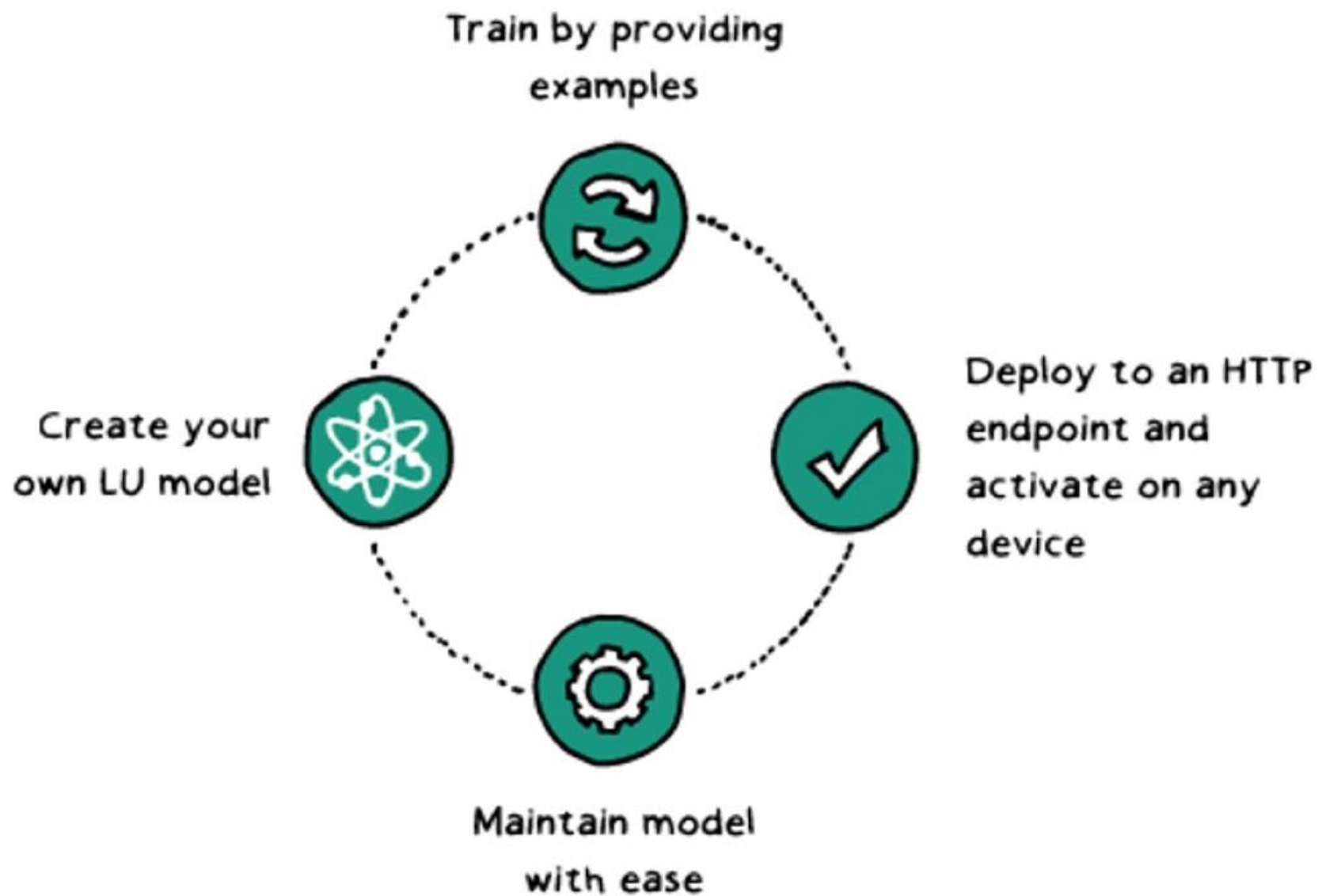
In the **build** phase of a bot

- Add natural language processing with LUIS (Language Understanding)
- Prerequisites
 - Luis.ai account
- Create a LUIS app in the LUIS portal
 - **Obtain values to connect to your LUIS app**
 - **Configure your bot to use your LUIS app**
 - **Get the intent by calling LUIS**

LUIS (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.

LUIS



Welcome to the Language Understanding Intelligent Service (LUIS)!

Country/Region (Required)

France ▼

- Contact me with promotional offers and updates about Cognitive Services.
- I agree that this service is subject to [the same terms under which I subscribe to Cognitive Services through Azure](#). If I do not subscribe to Cognitive Services through Azure, I agree that this service is subject to the [Microsoft Online Subscription Agreement](#). In each case, the terms include the [Online Services Terms](#). I acknowledge the [Privacy & Cookies statement](#)

Continue



Welcome to Microsoft's Language Understanding (LUIS)

LUIS enables you to integrate natural language understanding into your chatbot or other application without having to create the complex part of machine learning models. Instead, you get to focus on your own application's logic and let LUIS do the heavy lifting.

A typical LUIS app goes through the following 3 steps:

- 1 Design & Build
- 2 Train & Test
- 3 Publish & Improve



Scroll down to learn more or

[Create a LUIS app now](#)

LUIS Terms

- Intents

- Intents are how LUIS determines understands what a user wants to do. If your client application is for a travel agency, then you will need the intents "**BuyPlaneTicket**" and "**RentHotelRoom**" in order to identify when your users want to perform these different tasks

- Utterances

- An utterance is the textual input that LUIS will interpret. LUIS first uses example utterances that you add to an intent to teach itself how to evaluate the variety of utterances that users will input

- Entities

- An entity is used like a variable in algebra, it will capture and pass important information to your client app. In the utterance, "I want to buy a ticket to Seattle", you would want to capture the city name, Seattle, with the entity, like **destination_city**. Now LUIS will see the utterance as, "I want to buy a ticket to {**destination_city**}". This information can now be passed on to your client application and used to complete a task

Utterance, Entity examples

Utterance

Entity

Data

Buy 3 tickets to New York

Prebuilt number
Location.Destination

3
New York

Buy a ticket from New York to London on March 5

Location.Origin
Location.Destination
Prebuilt datetimeV2

New York
London
March 5, 2018

Different types of Entities

- Prebuilt entity

- Prebuilt entities are built-in types that represent common concepts such as email, URL, and phone number

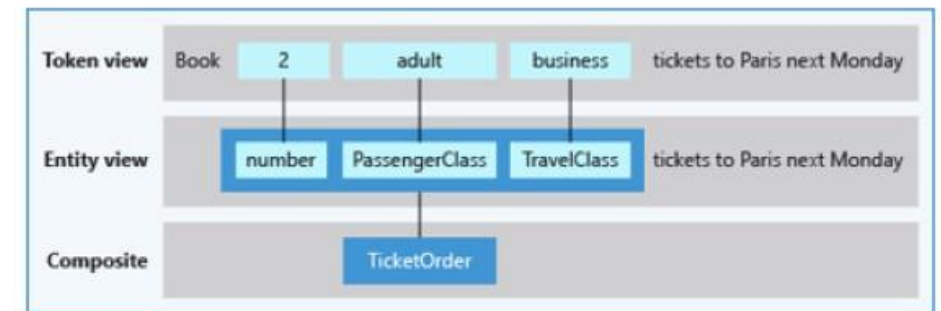
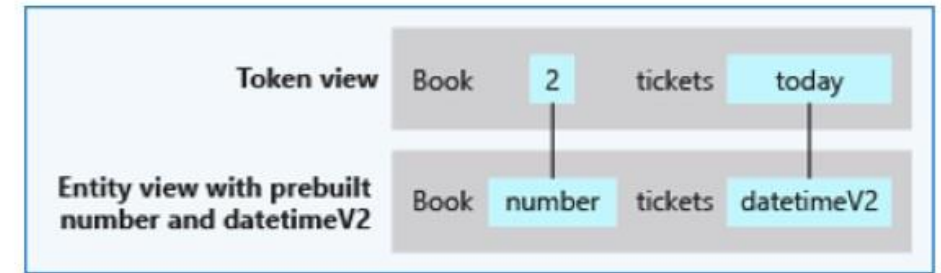
- Composite entity

- A composite entity is made up of other entities, such as prebuilt entities, simple, regular expression, and list entities. The separate entities form a whole entity

- List entity

- List entities represent a fixed, closed set of related words along with their synonyms. LUIS does not discover additional values for list entities

List item	Item synonyms
Seattle	sea-tac, sea, 98101, 206, +1
Paris	cdg, roissy, ory, 75001, 1, +33



Different types of Entities

- Hierarchical entity
 - Is a category of contextually learned simple entities called children
- Pattern.any entity
 - Pattern.any is a variable-length placeholder used only in a pattern's template utterance to mark where the entity begins and ends
- Regular expression entity
- Simple entity
 - is a machine-learned value. It can be a word or phrase

Design

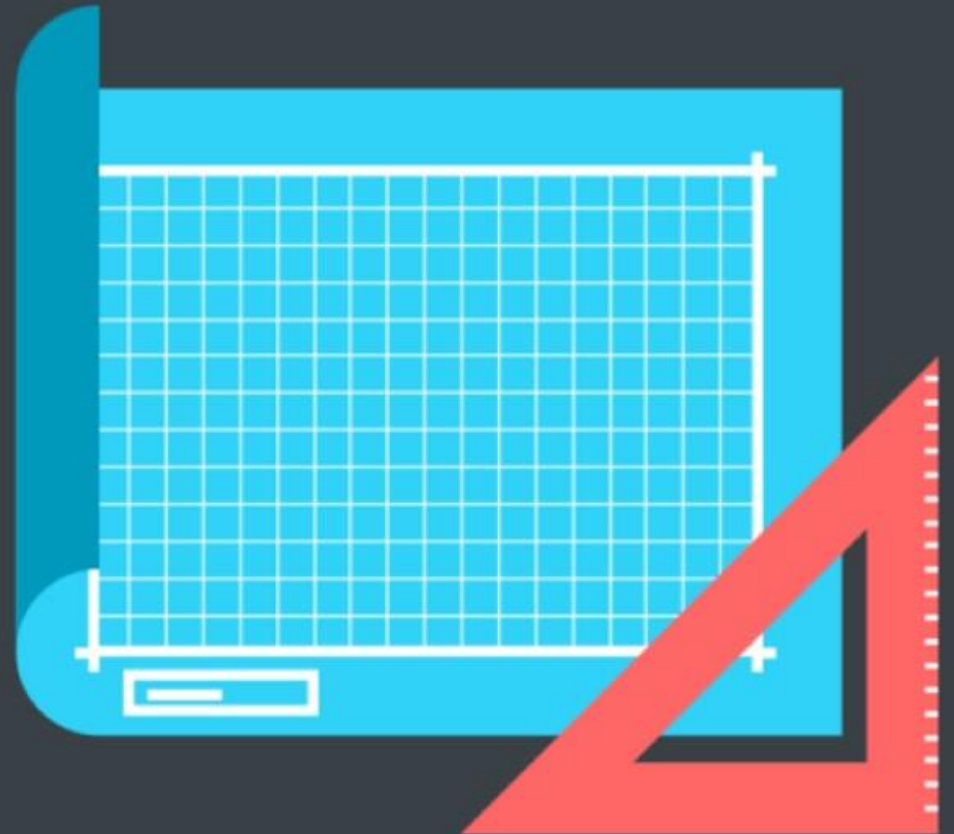
Start by identifying what you want to do with your LUIS app, for example, help users book a hotel room.

Keep things simple

Rather than trying to build a LUIS app that can understand any kind of input, focus on a few goals that you want your end-users to accomplish, such as book a room or change a reservation. Those goals are the **intents** you will define in your LUIS app.

Finally, identify any information you need to capture to fulfill a user's request, such as location, check-in and check-out dates, and the number of people. These translate to **entities** in your LUIS app.

See [Plan your LUIS app](#) for more details.



Machine-learned	Can Mark	Tutorial	Example Response	Entity type	Purpose
✓	✓	✓	✓	Composite	Grouping of entities, regardless of entity type.
		✓	✓	List	List of items and their synonyms extracted with exact text match.
Mixed		✓	✓	Pattern.any	Entity where end of entity is difficult to determine.
		✓	✓	Prebuilt	Already trained to extract various kinds of data.
		✓	✓	Regular Expression	Uses regular expression to match text.
✓	✓	✓	✓	Simple	Contains a single concept in word or phrase.

Consideration for LUIS

- Understand your domain
 - Ex: Selling airplane tickets
- Plan your Intents
 - Ex: buy a ticket, modify a ticket
- Create example Utterances for each Intent
 - Create 10 or 15 example utterance

Hands on lab - Add conversational intelligence to your apps by using Language Understanding Intelligent Service (LUIS)

To access the lab go here - <https://docs.microsoft.com/en-us/learn/modules/create-and-publish-a-luis-model/1-introduction>

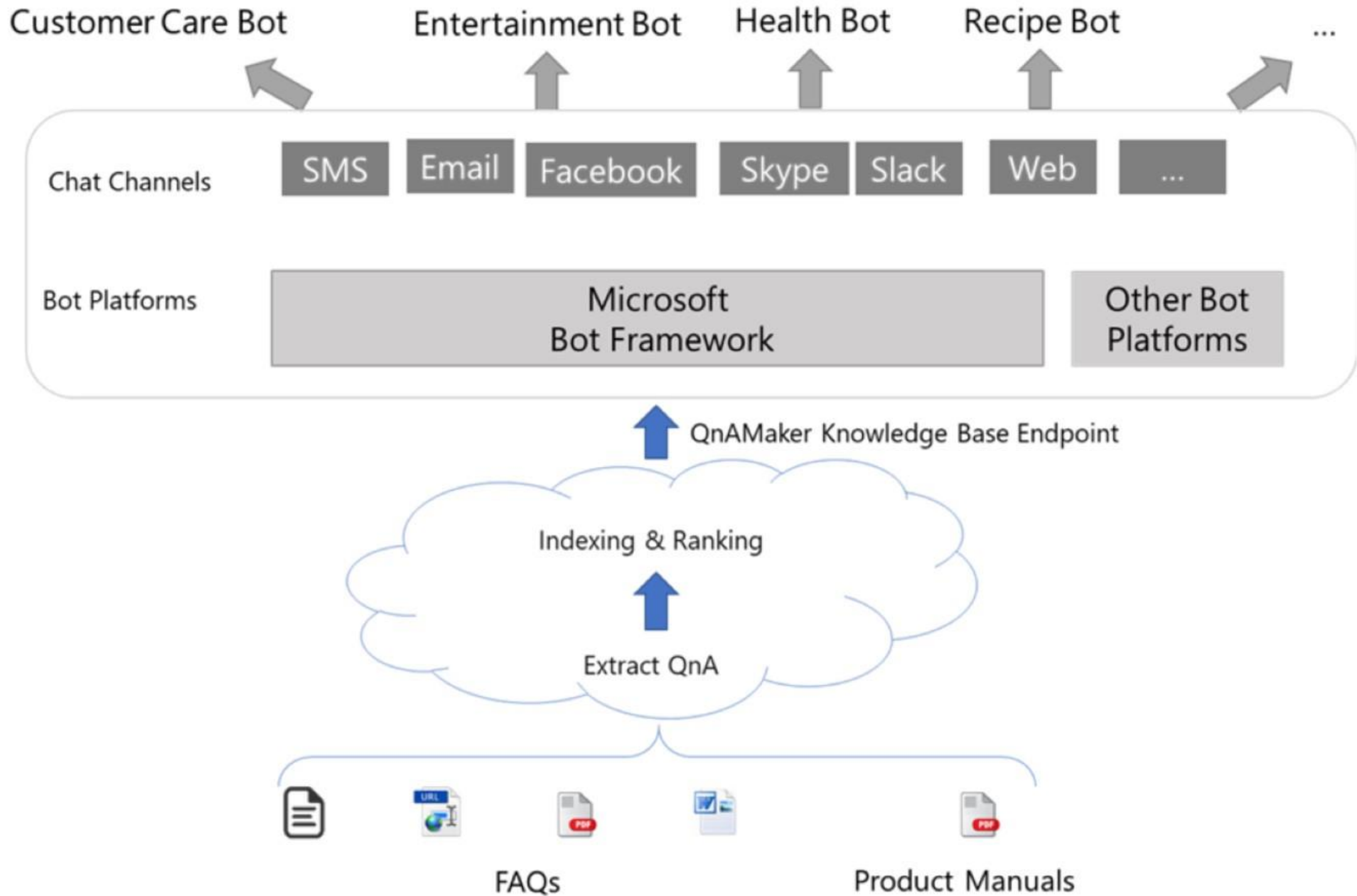
Let me know if you have any questions

Enhance bot with LUIS

- The developer of a bot is responsible for creating the logic that makes the bot intelligent
- LUIS can help the bot to understand the user's intentions
- First you need to publish LUIS app so it is accessible to applications as an endpoint

Use QnA Maker to answer questions

- From FAQ to Bot in minutes
- Build, train and publish a simple question and answer bot based on FAQ URLs, structured documents, product manuals or editorial content in minutes
- <https://www.qnamaker.ai/>



QnA Maker Architecture

- The content that you have added into the knowledge base is stored in Azure Search while the endpoint for access by client applications (including your Bot) are provided as a deployed App Service
- It is also possible to integrate Application Insights into the service for analytics

Create a knowledge base

1. Create a QnA service in Azure
2. Connect your QnA service to your KB
3. Name your knowledge base
4. Populate your KB
 - Extract question-and-answer pairs from an online FAQ, product manuals, or other files. Supported formats are .tsv, .pdf, .doc, .docx, .xlsx, containing questions and answers in sequence
5. Create your KB

Hands on lab - Build an FAQ chatbot with QnA Maker and Azure Bot Service

To access the lab go here - <https://docs.microsoft.com/en-us/learn/modules/build-a-faq-chat-bot-with-qna-maker-and-azure-bot-service/1-introduction>

Let me know if you have any questions

Data sources for QnA Maker content

Source Type	Content Type	Examples
URL	FAQs (Flat, with sections or with a topics homepage) Support pages (Single page how-to articles, troubleshooting articles etc.)	Plain FAQ, FAQ with links, FAQ with topics homepage Support article
PDF / DOC	FAQs, Product Manual, Brochures, Paper, Flyer Policy, Support guide, Structured QnA, etc.	Structured QnA.doc, Sample Product Manual.pdf, Sample semi-structured.doc, Sample white paper.pdf
Excel	Structured QnA file (including RTF, HTML support)	Sample QnA FAQ.xls
TXT/TSV	Structured QnA file	Sample chit-chat.tsv

Integrate QnA Maker and a bot

- One of the main reasons you create a QnA Maker Service along with an associated Knowledge Base is to serve as the foundation of a chat bot
- You need to connect the bot to this service
- You need for this 3 pieces of information:
 - QnAAuthKeys
 - QnAEndpointHostName
 - QnAKnowledgeBaseID

Easily design complex multi-turn conversations with follow-up prompts. [Learn more.](#)

My knowledge bases

Knowledge base name



PrepaAI-100-Stan-KB

Sample HTTP Request

Postman

Curl

```
POST /knowledgebases/924ac075-5edb-448f-95a3-dbb908de5d91/generateAnswer
Host: https://prepaai100-qna-stan1.azurewebsites.net/qnamaker
Authorization: EndpointKey 5d149ca5-7520-4803-ab32-03380fe95d1b
Content-Type: application/json
{"question": "<Your question>"}
```

Copy

Close

Integrate QnA Maker and a Bot

- One you create a Web App Bot you will need to enter the 3 informations (QnAAuthKeys, QnAEndpointHostName, QnAKnowledgeBaseID) in the application settings of the bot

PrepaAI100-stan1 - Configuration

Web App Bot

Search (Ctrl+/)

- Activity log
- Access control (IAM)
- Tags

Bot management

- Build
- Test in Web Chat
- Analytics
- Channels
- Settings
- Speech priming
- Bot Service pricing

App Service Settings

- Configuration
- All App service settings

Save Discard

Application settings General settings Default documents Path mappings

Application settings

Application settings are encrypted at rest and transmitted over an encrypted channel. You can choose to display them in plain text in your browser by using the controls below. Application Settings are exposed as environment variables for access by your application at runtime. [Learn more](#)

+ New application setting Show values Advanced edit Filter

Name	Value	Deployment slot setting	Delete	Edit
LuisAPIHostName	Hidden value. Click show values button			
LuisAPIKey	Hidden value. Click show values button			
LuisAppId	Hidden value. Click show values button			
MicrosoftAppId	Hidden value. Click show values button			
MicrosoftAppPassword	Hidden value. Click show values button			
WEBSITE_NODE_DEFAULT_VERSION	Hidden value. Click show values button			

PrepaAI100-stan1 - Configuration

Web App Bot

Search (Ctrl+/)

- Activity log
- Access control (IAM)
- Tags

Bot management

- Build
- Test in Web Chat
- Analytics
- Channels
- Settings
- Speech priming
- Bot Service pricing

App Service Settings

- Configuration**
- All App service settings

Support + troubleshooting

- New support request

Add/Edit application setting

Name

QnAAuthKey

Value

|

Deployment slot setting

OK

Cancel

PrepaAI100-stan1 - Configuration

Web App Bot

Search (Ctrl+/)

- Activity log
- Access control (IAM)
- Tags

Bot management

- Build
- Test in Web Chat
- Analytics
- Channels
- Settings
- Speech priming
- Bot Service pricing

App Service Settings

- Configuration**
- All App service settings

Support + troubleshooting

- New support request



App

App

A

u



Na

L

L

L

M

M

C

V

Add/Edit application setting

Name

QnAEndpointHostName

Value

https://prepai100-qna-stan1.azurewebsites.net/qnamaker

Deployment slot setting

OK

Cancel

PrepaAI100-stan1 - Configuration

Web App Bot

Search (Ctrl+*f*)

- Activity log
- Access control (IAM)
- Tags

Bot management

- Build
- Test in Web Chat
- Analytics
- Channels
- Settings
- Speech priming
- Bot Service pricing

App Service Settings

- Configuration**
- All App service settings

Support + troubleshooting

- New support request

App

+

+

Name

L

L

L

M

M

C

C

V

Con

Add/Edit application setting

Name	<input type="text" value="QnAKnowledgebaselid"/>
Value	<input type="text" value="924ac075-5edb-448f-95a3-dbb908de5d91"/>

Deployment slot setting

PrepaAI100-stan1 - Configuration

Web App Bot

Search (Ctrl+)

Save Discard

Application settings are encrypted at rest and transmitted over an encrypted channel. You can choose to display them in plain text in your browser by using the controls below. Application Settings are exposed as environment variables for access by your application at runtime. [Learn more](#)

+ New application setting Show values Advanced edit Filter

Name	Value	Deployment slot setting	Delete	Edit
LuisAPIHostName	Hidden value. Click show values button			
LuisAPIKey	Hidden value. Click show values button			
LuisAppId	Hidden value. Click show values button			
MicrosoftAppId	Hidden value. Click show values button			
MicrosoftAppPassword	Hidden value. Click show values button			
QnAAuthKey	Hidden value. Click show values button			
QnAEndpointHostName	Hidden value. Click show values button			
QnAKnowledgebaselId	Hidden value. Click show values button			
WEBSITE_NODE_DEFAULT_VERSION	Hidden value. Click show values button			

Bot management

- Build
- Test in Web Chat
- Analytics
- Channels
- Settings
- Speech priming
- Bot Service pricing

App Service Settings

- Configuration
- All App service settings

Support + troubleshooting

- New support request

Testing a bot

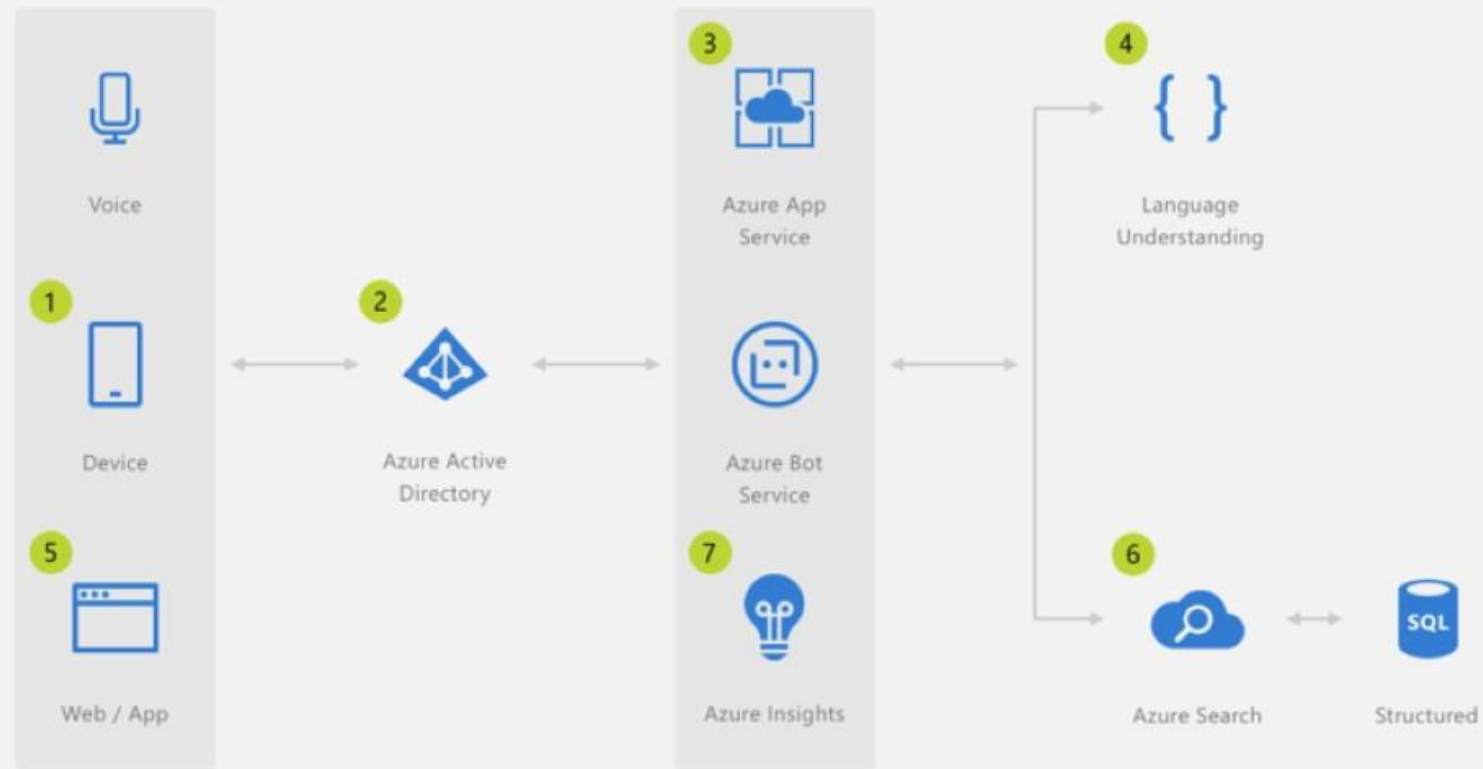
- The Bot Framework Emulator is a desktop application that allows bot developers to test and debug bots built using the [Bot Framework SDK](#)
- You can use the Bot Framework Emulator to test bots running either locally on your machine or connect to bots running remotely through a tunnel
 - Available on Windows, Linux & Mac
 - <https://github.com/microsoft/BotFramework-Emulator>

Integrate machine learning solutions in an app

- Learn how to build intelligent algorithms into apps and websites. Use Scikit-learn, Tensorflow, Pytorch or any other Python-based framework to build your machine learning model and train it locally or in the cloud
- An Azure Machine Learning workspace is the foundational block in the cloud that you use to experiment, train, and deploy machine learning models with Machine Learning

Information Chatbot

This Informational Bot can answer questions defined in a knowledge set or FAQ using Cognitive Services QnA Maker and answer more open-ended questions using Azure Search. [Learn more](#)



1 Employee starts the Application Bot

2 Azure Active Directory validates the employee's identity

3 The employee can ask the bot what type of queries are supported

4 Cognitive Services returns a FAQ built with the QnA Maker

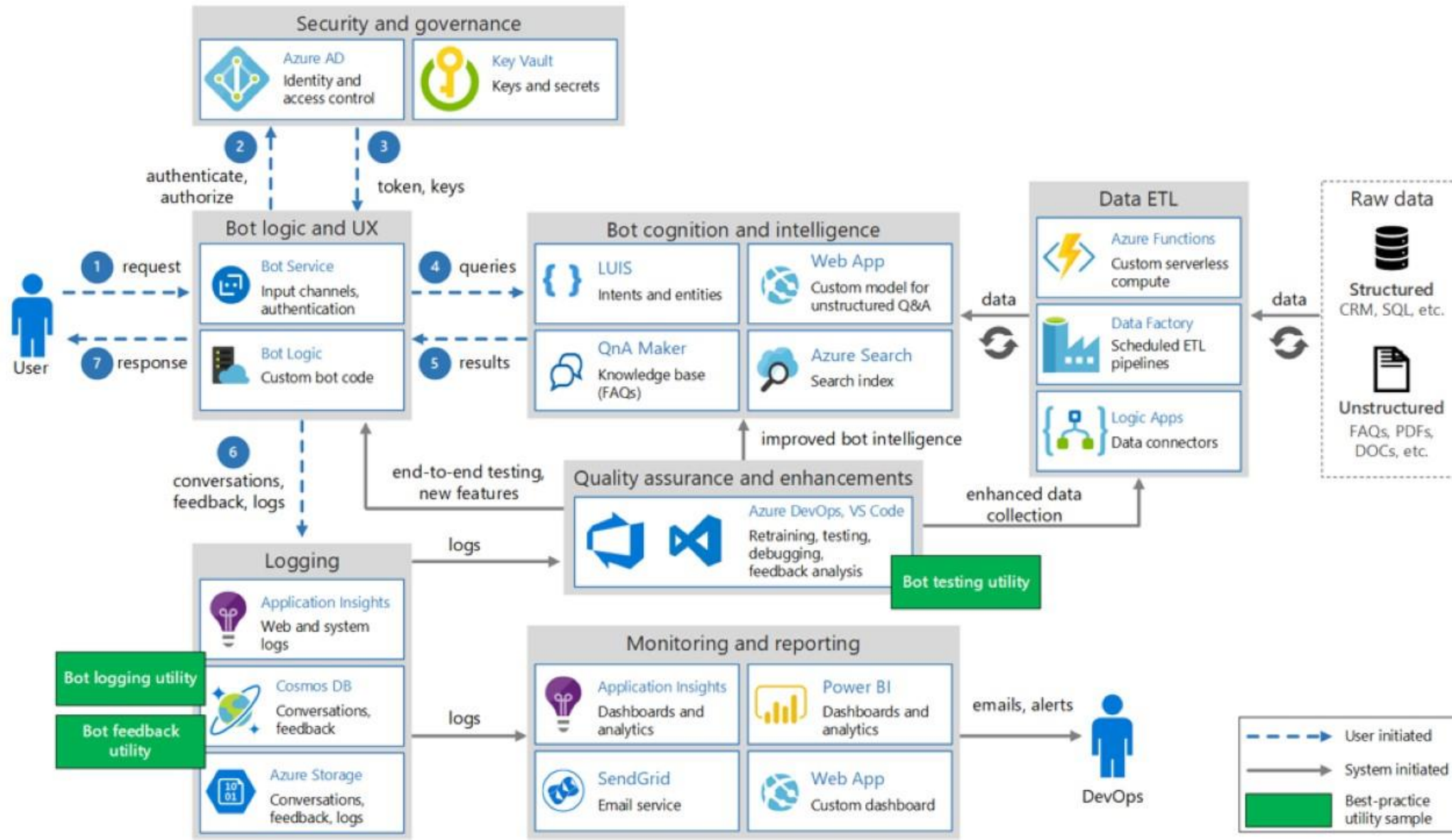
5 The employee defines a valid query

6 The Bot submits the query to Azure Search which returns information about the application data

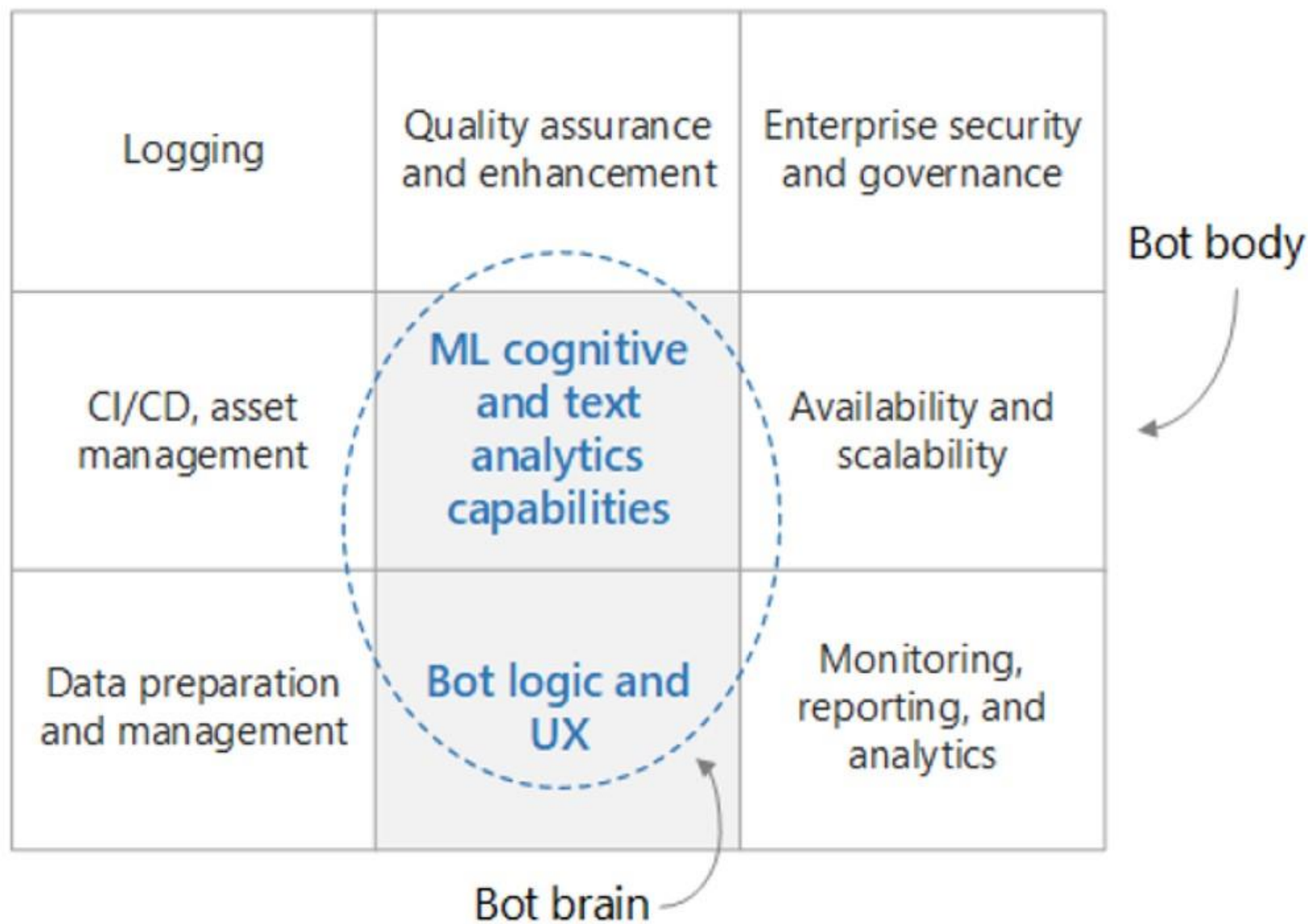
7 Application insights gathers runtime telemetry to help development with Bot performance and usage

"Conversation as a platform is the future, so it's great that we're already offering it to our customers using the Bot Framework and Azure."

Enterprise-grade conversational bot



Design Considerations



Machine learning on Azure

NEW UPDATES

Sophisticated pretrained models

Simplify solution development



Vision



Speech



Language



Search

Popular frameworks

Build advanced deep learning solutions



Pytorch



TensorFlow



Keras



Onnx

Productive services

Empower data science and development teams



Azure Databricks



Azure Machine Learning

Powerful infrastructure

Accelerate deep learning



CPU



GPU



FPGA

Flexible deployment

Deploy, manage models on intelligent cloud & edge



On-premises



Cloud



Edge

Azure Machine Learning service

- Azure Machine Learning service is a cloud service that you use to train, deploy, automate, and manage machine learning models, all at the broad scale that the cloud provides
- Machine learning is a data science technique that allows computers to use existing data to forecast future behaviors, outcomes, and trends. By using machine learning, computers learn without being explicitly programmed

Azure Machine Learning service

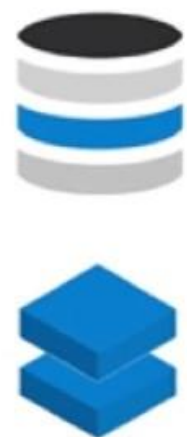
- Start training on your local machine and then scale out to the cloud
- The service fully supports open-source technologies such as **PyTorch**, **TensorFlow**, and **scikit-learn** and can be used for any kind of machine learning, from classical ml to deep learning, supervised and unsupervised learning

Azure Machine Learning Service

- For Custom AI
- Can run locally, remote, in the cloud
- Full lifecycle and continuous improvement
 - ML Ops (DevOps Pipeline for Machine Learning)
- Industry class deployment
 - Using Docker image running on Docker server (ex: ACI) or Kubernetes Cluster (ex: AKS)
- Enterprise Class Experience
 - Authentication, Logging
 - Scaling, Monitoring

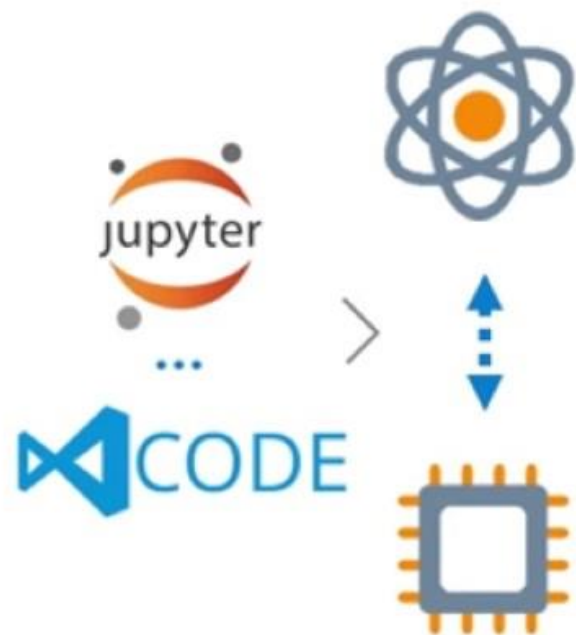
End to end workflow in Azure Machine Learning Service

Prepare



Prepare data

Experiment



Build model

Train & test model

Deploy



Register & manage model



Build image



Deploy & monitor service

Azure Machine Learning Visual Experience

MLServiceWorkspaceAI100-stan - Visual interface
Machine Learning service workspace

Search (Ctrl+/)

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

Authoring (Preview)

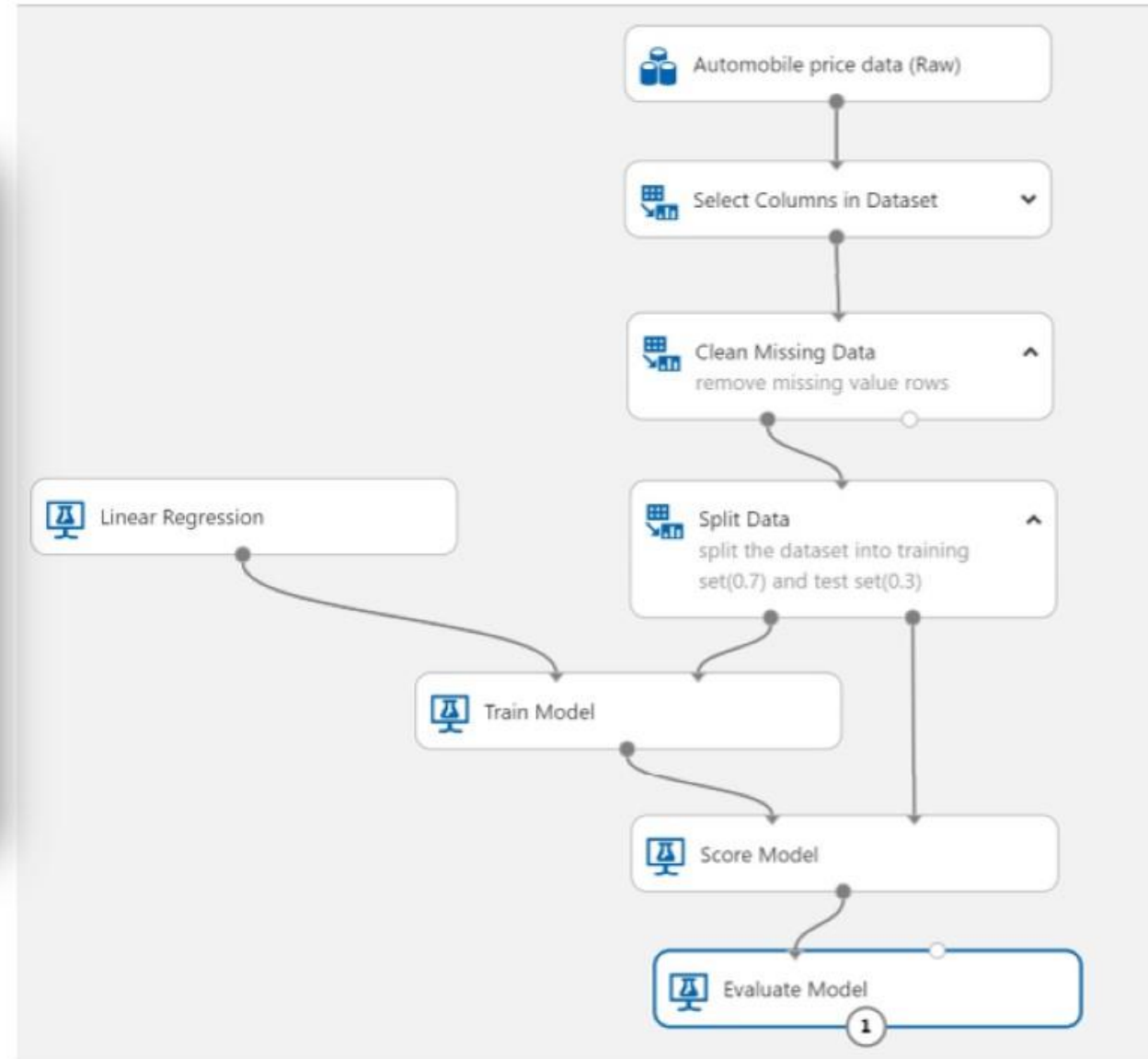
- Automated machine learning
- Notebook VMs
- Visual interface**

Visual interface (preview)

What's possible with visual interface

- ✓ Drag-n-Drop to build machine learning models
- ✓ No limit to data size or compute capacity for model training
- ✓ Intrinsic and powerful Python support
- ✓ One click to deploy your web service
- ✓ Rich and fast-growing modules support

[Launch visual interface](#) [View Documentation](#)



Machine Learning Studio versus Azure Machine Learning

	Machine Learning Studio	Azure Machine Learning service: Visual interface
	Generally available (GA)	In preview
Modules for interface	Many	Initial set of popular modules
Training compute targets	Proprietary compute target, CPU support only	Supports Azure Machine Learning compute, GPU or CPU. (Other computes supported in SDK)
Deployment compute targets	Proprietary web service format, not customizable	Enterprise security options & Azure Kubernetes Service. (Other computes supported in SDK)
Automated model training and hyperparameter tuning	No	Not yet in visual interface. (Supported in the SDK and Azure portal.)

Machine Learning service workspace

Microsoft



Machine Learning service workspace [Save for later](#)

Microsoft

Create

Azure Machine Learning is a secure and powerful cloud-based offering for rapidly building, deploying, and monitoring advanced machine learning and analytics solutions.

Use this template to create an Azure Machine Learning service workspace. This workspace contains the tools to train, manage, and deploy machine-learning experiments and web services for Azure Machine Learning service.

This workspace is different from and not compatible with the Machine Learning Studio Workspace, which offers users a serverless, drag-n-drop environment.

Useful Links

[Documentation](#)

[Pricing Details](#)

[Azure AI Gallery](#)



Machine Learning service workspace

Create

[* Main](#) [Tags](#) [* Review](#)

* Workspace Name

MLServiceWorkspaceAI100-stan

Subscription

Conso interne de la plateforme Windows Azure 2

Resource group

RG-Prepa-AI-100

[Create new](#)

Location

West US



For your convenience, these resources are added automatically to the workspace, if regionally available: [Azure storage](#), [Azure Application Insights](#) and [Azure Key Vault](#).

MLServiceWorkspaceAI100-stan

Machine Learning service workspace

Search (Ctrl+/)

- Overview
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

Authoring (Preview)

- Automated machine learning
- Notebook VMs
- Visual interface

Assets

- Experiments
- Pipelines
- Compute
- Models
- Images
- Deployments
- Activities

Settings

Download config.json Delete

Resource group : [RG-Prepa-AI-100](#)
Location : West US
Subscription : [Conso interne de la plateforme Windows Azure 2](#)
Subscription ID : f885b031-4059-40e4-9240-eb77ae16cc26

Storage : [mlserviceworks1982686034](#)
Registry : ...
Key Vault : [mlserviceworks2513673009](#)
Application Insights : [mlserviceworks5762448042](#)

Getting Started



Get Started with Sample Notebooks (Preview)

Quickly get started with the Python SDK and run sample experiments with Azure Machine Learning Notebook VMs.



Create a new Automated Machine Learning Model (Preview)

Automatically create a model from your existing data.



Build a model using the Visual Interface (Preview)

Drag and drop existing components to create new models.



View Documentation

Learn how to use Azure Machine Learning.



View more samples at GitHub

Get inspired by a large collection of machine learning examples.



View Forum

Join the discussion of Azure Machine Learning.

Hands on Lab: Create an Experiment in Azure Machine Learning Studio

Go to this link: <https://docs.microsoft.com/en-us/learn/modules/create-machine-learning-studio-account/1-introduction/>

Let me know if you have any questions

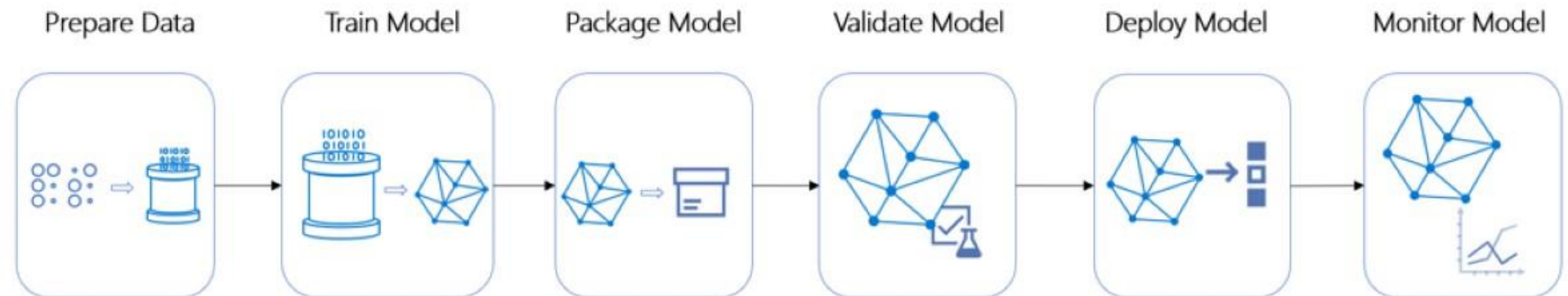
Hands on Lab: Create an Experiment in Azure Machine Learning Studio

Go to this link: <https://docs.microsoft.com/en-us/learn/modules/create-an-experiment-in-ml-studio/1-introduction>

Let me know if you have any questions

Azure machine learning pipelines

- Using machine learning (ML) pipelines, data scientists, data engineers, and IT professionals can collaborate on the steps involved in:
 - Data preparation, such as normalizations and transformations
 - Model training
 - Model evaluation
 - Deployment



Azure machine learning pipelines

- With Machine Learning Pipelines, you can collaborate on each step from data preparation, model training and evaluation, through deployment
- Pipelines allow you to:
 - Automate the end-to-end machine learning process in the cloud
 - Reuse components and only re-run steps when you need to
 - Use different compute resources in each step
 - Run batch scoring tasks

Azure machine learning pipelines

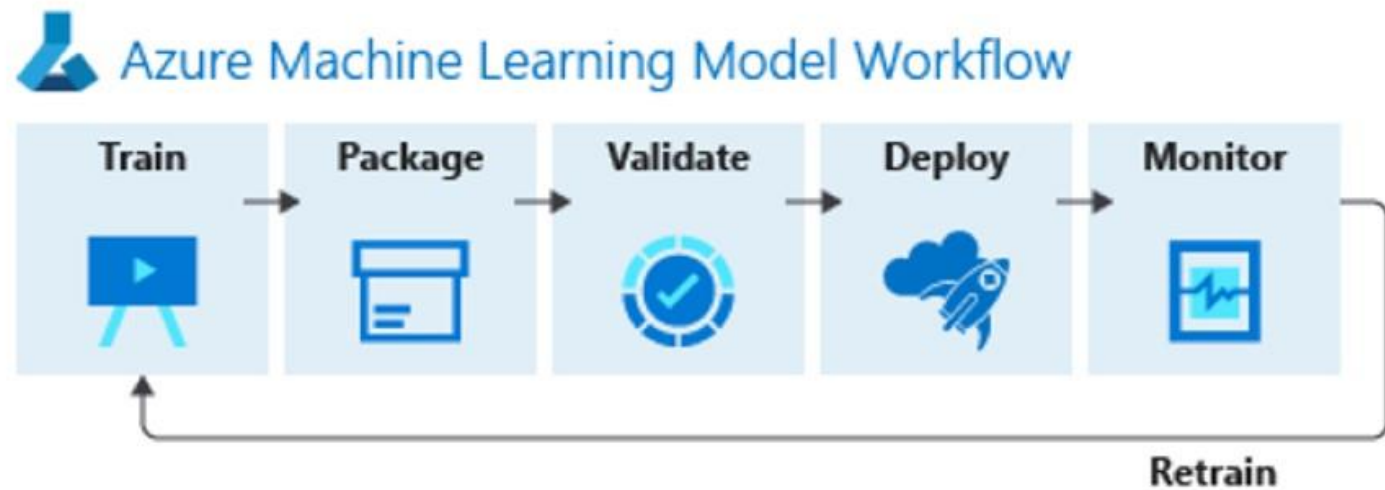
- A typical pipeline would have multiple tasks to prepare data, train, deploy and evaluate models. Individual steps in the pipeline can make use of diverse compute options (for example: CPU for data preparation and GPU for training) and languages

Key advantages of using pipelines for your machine learning workflows

The key advantages of using pipelines for your machine learning workflows are:


Key advantage	Description
Unattended runs	Schedule steps to run in parallel or in sequence in a reliable and unattended manner. Data preparation and modeling can last days or weeks, and pipelines allow you to focus on other tasks while the process is running.
Heterogenous compute	Use multiple pipelines that are reliably coordinated across heterogeneous and scalable compute resources and storage locations. Run individual pipeline steps on different compute targets, such as HDInsight, GPU Data Science VMs, and Databricks. This makes efficient use of available compute options.
Reusability	Create pipeline templates for specific scenarios, such as retraining and batch-scoring. Trigger published pipelines from external systems via simple REST calls.
Tracking and versioning	Instead of manually tracking data and result paths as you iterate, use the pipelines SDK to explicitly name and version your data sources, inputs, and outputs. You can also manage scripts and data separately for increased productivity.
Collaboration	Pipelines allow data scientists to collaborate across all areas of the machine learning design process, while being able to concurrently work on pipeline steps.

AI Pipelines










- Use the [Azure Machine Learning Python SDK](#) with open-source Python packages, or use the [visual interface \(preview\)](#) to build and train highly accurate machine learning and deep-learning models yourself in an Azure Machine Learning service Workspace

Authoring (Preview)

-  Automated machine learning
-  Notebook VMs
-  Visual interface

Assets

-  Experiments
-  Pipelines
-  Compute
-  Models
-  Images
-  Deployments
-  Activities

Settings

-  Properties
-  Locks
-  Export template

Monitoring

-  Metrics

Support + troubleshooting

-  Usage & quotas

- Experiments
- Pipelines**
- Compute
- Models
- Images
- Deployments
- Activities

Working with Pipelines

1. What are Pipelines?

Pipelines are used to create and manage workflows that stitch together machine learning phases. Various machine learning phases including data preparation, model training, model deployment, and inferencing.

[View Documentation](#)

2. Getting Started

Create your first pipeline in Azure Notebooks.

[Open Azure Notebooks](#)

What's possible with Pipelines?

With pipelines, you can optimize your workflow with simplicity, speed, portability, and reuse. When building pipelines with Azure Machine Learning, you can focus on your expertise, machine learning, rather than on infrastructure



Unattended Execution

Schedule a few steps to run in parallel or in sequence in a reliable and unattended manner.



Diverse Compute

Individual pipeline steps can be run on different compute targets such as Data Science VMs, Databricks, or Azure Data Lake Analytics.



Reusability

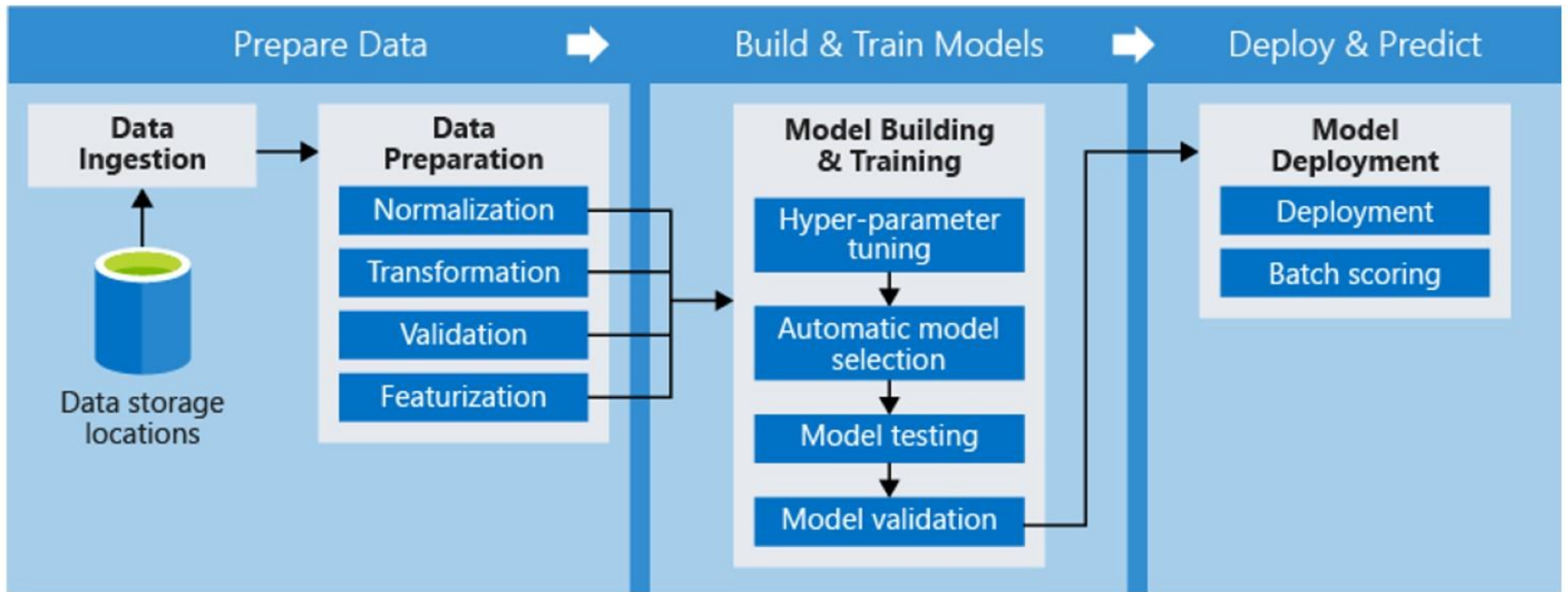
Pipelines can be templated and can be triggered from external systems via simple REST calls for retraining and batch scoring.



Tracking

Instead of manually tracking data and results as you iterate, use Pipelines SDK to name and version your data sources, inputs, and outputs.

Azure Machine Learning Pipeline



Inference

- Inference, or model scoring, is the phase where the deployed model is used for prediction, most commonly on production data

Deploy a model

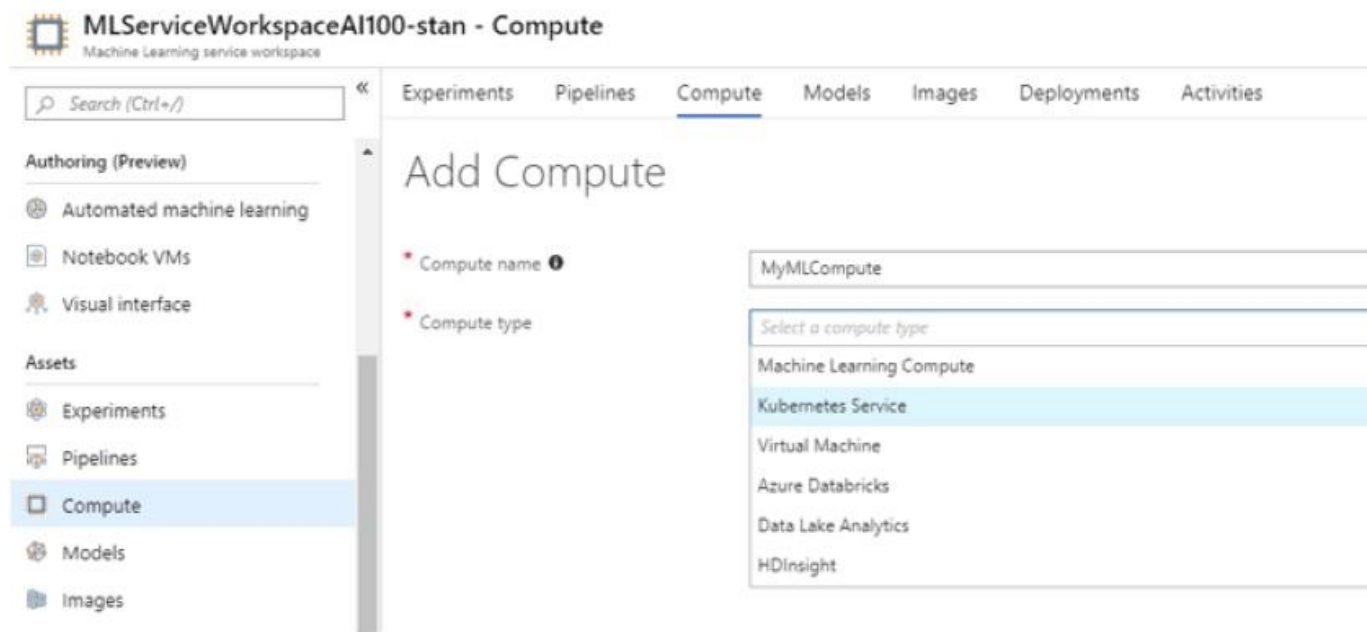
- You deploy a model as a Web Service (is a Docker image in that case). Models in Azure Machine Learning Services can be deployed on :
 - Local Web Service, Notebook VM Web Service
 - Azure IoT Edge device, Azure Databox Edge
 - Azure Container Instance
 - **Azure Kubernetes Service**

Choose a compute target

Compute target	Usage	GPU support	FPGA support	Description
Local web service	Testing/debug	maybe		Good for limited testing and troubleshooting. Hardware acceleration depends on using libraries in the local system.
Notebook VM web service	Testing/debug	maybe		Good for limited testing and troubleshooting.
Azure Kubernetes Service (AKS)	Real-time inference	yes	yes	Good for high-scale production deployments. Provides fast response time and autoscaling of the deployed service. Cluster autoscaling is not supported through the Azure Machine Learning SDK. To change the nodes in the AKS cluster, use the UI for your AKS cluster in the Azure portal. AKS is the only option available for the visual interface.
Azure Container Instances (ACI)	Testing or dev			Good for low scale, CPU-based workloads requiring <48-GB RAM
Azure Machine Learning Compute	(Preview) Batch inference	yes		Run batch scoring on serverless compute. Supports normal and low-priority VMs.
Azure IoT Edge	(Preview) IoT module			Deploy & serve ML models on IoT devices.
Azure Data Box Edge	via IoT Edge		yes	Deploy & serve ML models on IoT devices.

Deploy a model to an Azure Kubernetes Service cluster

- Deploy a model to an Azure Kubernetes Service cluster if you need :
 - Fast response time
 - Autoscaling of the deployed service
 - Hardware acceleration options such as GPU and field-programmable gate arrays (FPGA)



Deploy a model using a Custom Docker image

- When you deploy a trained model to a web service or IoT Edge device, a Docker image is created
 - This image contains the model, conda environment, and assets needed to use the model
 - It also contains a web server to handle incoming requests when deployed as a web service, and components needed to work with Azure IoT Hub

Machine Learning Pipelines



Azure AI Pipelines

- Defines reusable machine learning workflows that can be used as a template for your template learning scenarios
- Rerun only the steps you need
- Use Various Toolkits and Frameworks such as Pytorch or TensorFlow
- Track metrics in Azure Portal

Azure AI Pipelines

- Unattended runs
- Mixed and diverse compute
 - HDInsight
 - Databricks
 - GPU Data Science VM
- Tracking and versioning

Data Science Virtual Machine



- Data Science Virtual Machine

- The Data Science Virtual Machine (DSVM) is a customized VM image on Microsoft's Azure cloud built specifically for doing data science. It has many popular data science and other tools pre-installed and pre-configured to jump-start building intelligent applications for advanced analytics

- Geo AI Data Science Virtual Machine

- The Azure **Geo AI Data Science VM** (Geo-DSVM) delivers geospatial analytics capabilities from Microsoft's Data Science VM. Specifically, this VM extends the AI and data science toolkits in the [Data Science VM](#) by adding ESRI's market-leading [ArcGIS Pro](#) Geographic Information System

- Deep Learning Data Science Virtual Machine

- Utilizing VM scaling capabilities of Azure cloud, DSVM helps you use GPU-based hardware on the cloud as per need. One can switch to a GPU-based VM when training large models or need high-speed computations while keeping the same OS disk

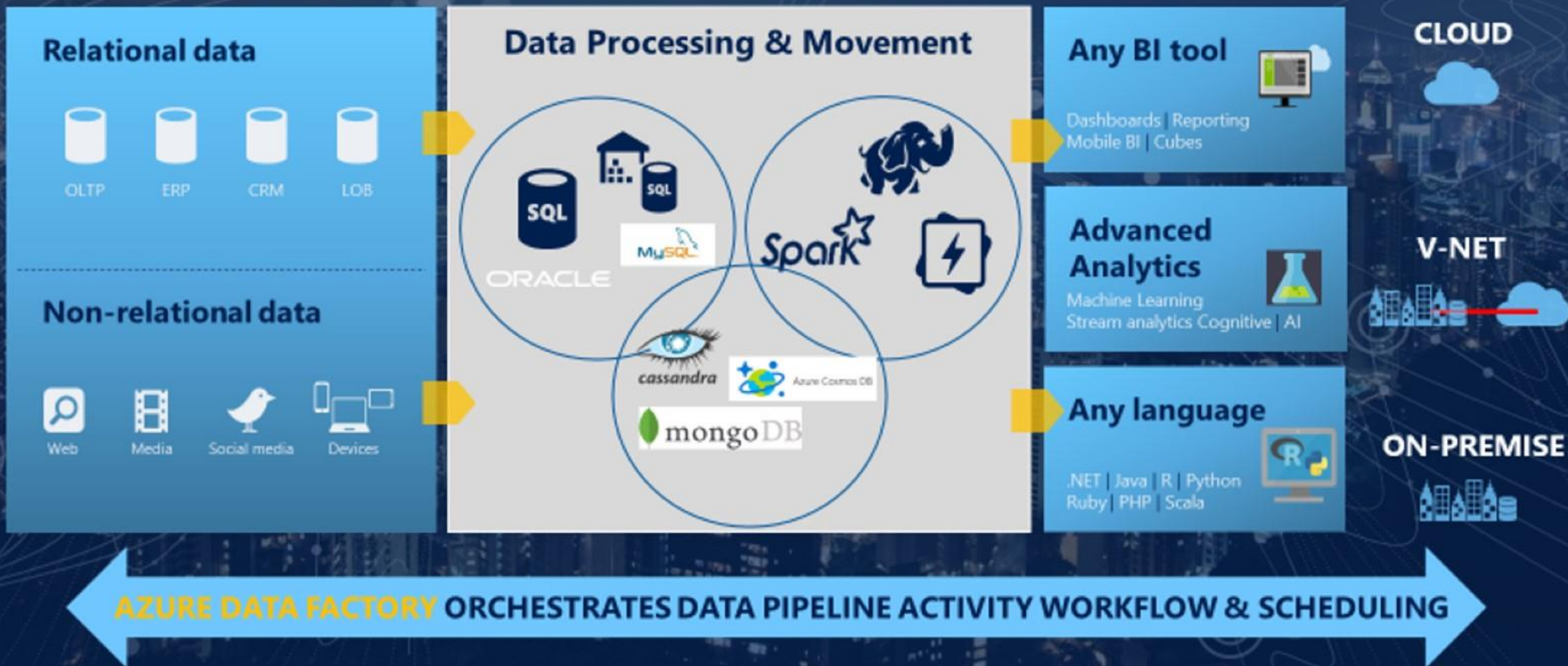
Azure AI Pipeline –model deployment (model is a Docker image)

- Local Deployment on a Docker engine
- Azure Kubernetes Services → for **production**
- Azure Container Instance → for dev & test
- Azure Machine Learning Compute → For batch (prediction or scoring) processing of large collection of data. Cost effective inference for asynchronous application
- Azure IoT Edge

Azure Datafactory

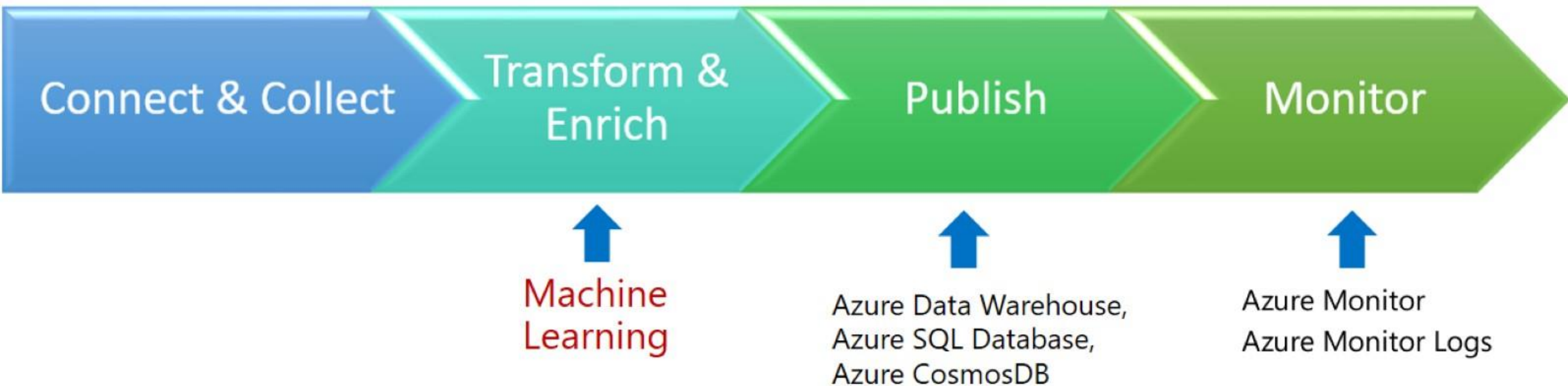
- Azure Data Factory is the platform that solves many data scenarios
- It is a cloud-based data integration service that allows you to create data-driven workflows in the cloud for orchestrating and automating data movement and data transformation
- Using Azure Data Factory, you can create and schedule **data-driven workflows (called pipelines)** that can ingest data from **disparate data stores**. It can process and transform the data by using compute services such as Azure HDInsight Hadoop, Spark, Azure Data Lake Analytics, and Azure Machine Learning

HYBRID DATA INTEGRATION AT SCALE



Design pipelines that call Azure Machine Learning models

- The pipelines (data-driven workflows) in Azure Data Factory typically perform the following four steps:



Azure Data Factory pipeline

- A data factory might have one or more pipelines
- A pipeline is a logical grouping of activities that performs a unit of work. Data Factory supports 3 types of activities:
 - data movement activities
 - data transformation activities
 - control activities
- Together, the activities in a pipeline perform a task

Select an AI solution that meet cost constraints

- It is widely accepted that for deep learning training, GPUs should be used due to their significant speed when compared to CPUs
- However, due to their higher cost, for tasks like inference which are not as resource heavy as training, it is usually believed that CPUs are enough and are more attractive due to their cost savings
- However, when inference speed is a bottleneck, using GPUs or FPGA provide considerable gains both from financial and time perspectives

Integrate machine learning solutions in an app

- Learn how to build intelligent algorithms into apps and websites. Use Scikit-learn, Tensorflow, Pytorch or any other Python-based framework to build your machine learning model and train it locally or in the cloud
- An Azure Machine Learning workspace is the foundational block in the cloud that you use to experiment, train, and deploy machine learning models with Machine Learning

Part 4

Design the compute
infrastructure to support a
solution

Deep Learning with CPU, GPU or FPGA

- It is widely accepted that for deep learning training, GPUs should be used due to their significant speed when compared to CPUs
 - However, due to their higher cost, for tasks like inference which are not as resource heavy as training, it is usually believed that CPUs are enough and are more attractive due to their cost savings
 - However, when inference speed is a bottleneck, using GPUs or FPGA provide considerable gains both from financial and time perspectives

Identify whether to create a GPU, FPGA, or CPU-based solution

- FPGAs contain an array of programmable logic blocks, and a hierarchy of reconfigurable interconnects
 - The interconnects allow these blocks to be configured in various ways after manufacturing
 - Compared to other chips, FPGAs provide a combination of programmability and performance
- FPGAs make it possible to achieve low latency for **real-time inference** (or model scoring) requests

FPGAs vs. CPU, GPU, and ASIC

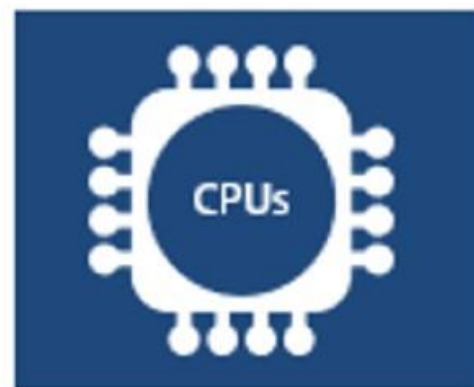
Silicon alternatives

TRAINING

CPUs and GPUs, limited FPGAs,
ASICs under investigation

EVALUATION

CPUs and FPGAs,
ASICs under investigation



FLEXIBILITY

EFFICIENCY

FPGAs vs. CPU, GPU, and ASIC

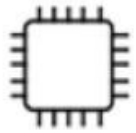
Processor	Description
Application-specific integrated circuits	ASICs Custom circuits, such as Google's TensorFlow Processor Units (TPU), provide the highest efficiency. They can't be reconfigured as your needs change.
Field-programmable gate arrays	FPGAs FPGAs, such as those available on Azure, provide performance close to ASICs. They are also flexible and reconfigurable over time, to implement new logic.
Graphics processing units	GPUs A popular choice for AI computations. GPUs offer parallel processing capabilities, making it faster at image rendering than CPUs.
Central processing units	CPUs General-purpose processors, the performance of which isn't ideal for graphics and video processing.

FPGA in Azure

- Azure supports:
 - Image classification and recognition scenarios
 - TensorFlow deployment
 - DNNs (Deep Neural Network) : ResNet 50, ResNet 152, VGG-16, SSD-VGG, and DenseNet-121
 - Intel FPGA hardware
- Azure FPGAs are integrated with Azure Machine Learning. Microsoft uses FPGAs for DNN evaluation, Bing search ranking, and software defined networking (SDN) acceleration to reduce latency, while freeing CPUs for other tasks

Identify whether to use a cloud-based, on-premises, or hybrid compute infrastructure

Select a compute solution that meets cost constraints



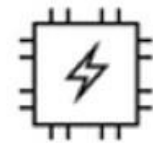
CPUs

General purpose machine learning
D, F, L, M, H Series



GPUs

Deep learning
N Series



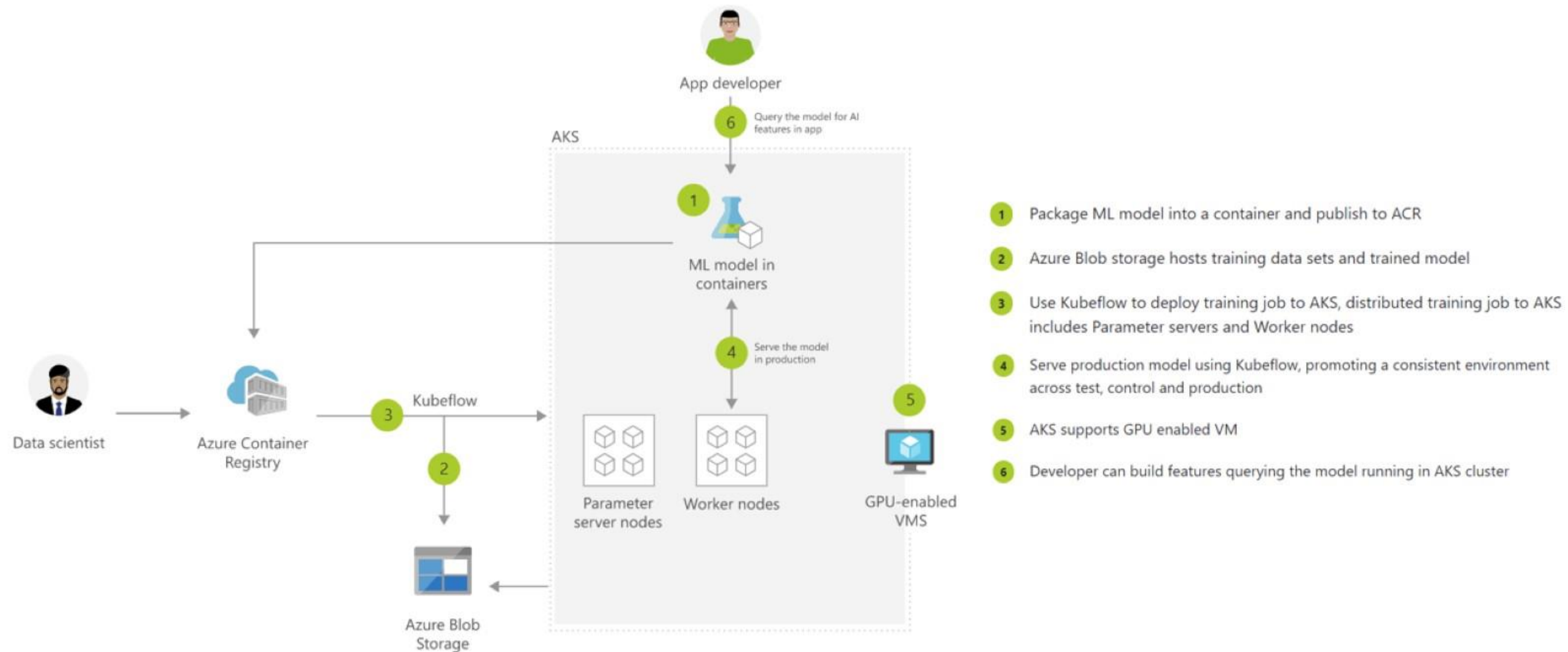
FPGAs

Specialized hardware accelerated deep learning
Project Brainwave

Optimized for flexibility

Optimized for performance

Machine Learning model training with AKS



Part 4

Design for data governance, compliance, integrity, and security

Ensure that data adheres to compliance requirements defined by your organization

- Use Azure Policy to define your corporate policies

Design strategies to ensure the solution meets data privacy and industry standard regulations

- → GDPR
- Use Azure Policy, Azure Compliance Manager