

오픈 소스 및 클라우드와 함께 하는 다양한 앱 개발 여정

Microsoft AI & reinforcement Learning : Empowering Developers

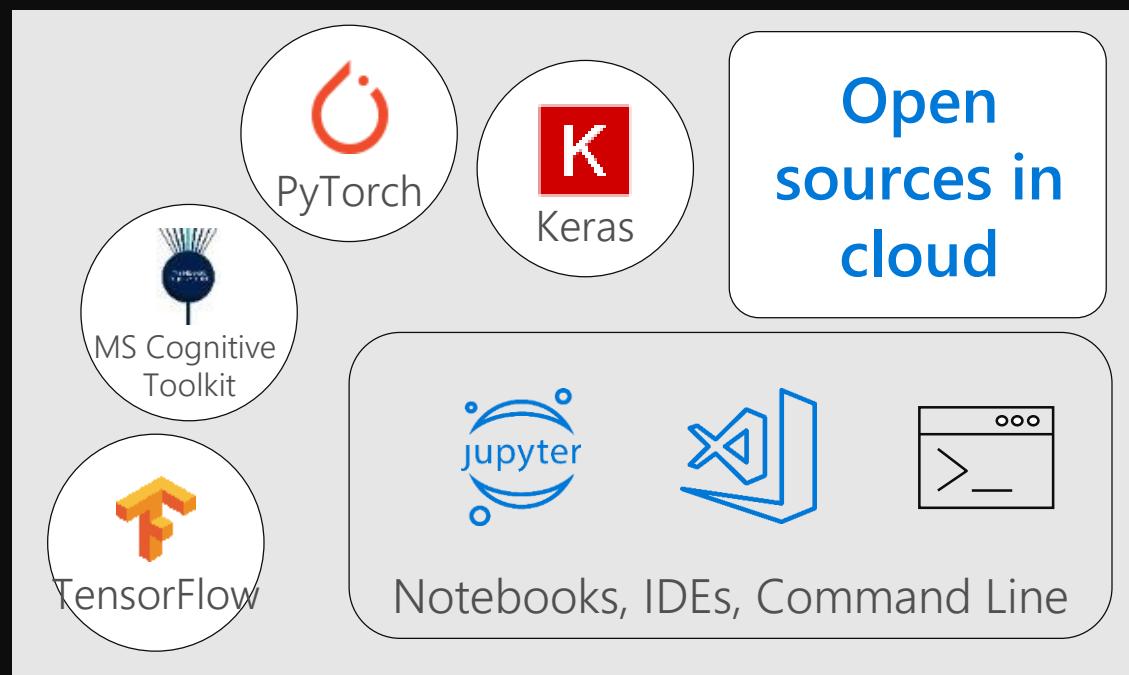
Ian Choi, Korea Developer Relations (Twitter: @ianychoi)



목차

1. 오픈 소스에 대한 이야기
2. 마이크로소프트 이야기로 살펴보는
AI with Open Source, Cloud
3. 마이크로소프트와 강화 학습
4. With AI developers: 공유하고 싶은 이야기

1. 오픈 소스에 대한 이야기(로 잠깐..)



같이하면 즐겁다



당신의 장비:

〈조명〉	검은 수정
〈머리〉	붉은 머리띠
〈목〉	핸슨의 머플러(숨겨진 장비)
〈어깨〉	아리만의 어깨 보호대
〈몸둘레〉	암흑천사의 날개(숨겨진 장비)
〈몸통〉	은회색 로브(숨겨진 장비)
〈팔〉	건더슨의 팔 보호대(숨겨진 장비)
〈방패〉	해골 방패(+)
〈손목〉	암흑천사의 팔찌(숨겨진 장비)
〈장갑〉	검은 장갑
〈반지〉	오팔 반지(마법반지)
〈오른손〉	(마법무기) 사파이어 흰표범발톱(물+)
〈왼손〉	오크의 석궁(숨겨진 장비)
〈허리〉	보석 벨트
〈다리〉	트롤 다리 보호대
〈무릎〉	검은 무릎 보호대
〈발〉	아네스의 부츠

〈8398체력 7000마법력 8264이동력〉

»*« 202호(비스크) »*«

방이 깔끔히 정돈되어 있습니다.

[출구: 동(닫힌문)]

〈물건〉 커다란 침대가 놓여 있습니다.

시민이 생업에 종사하고 있습니다.

(절망) 마녀 비비안이 요염한 포즈로 침대에 앉아 있습니다.

더 재미있는 방식으로 같이하면 즐겁다

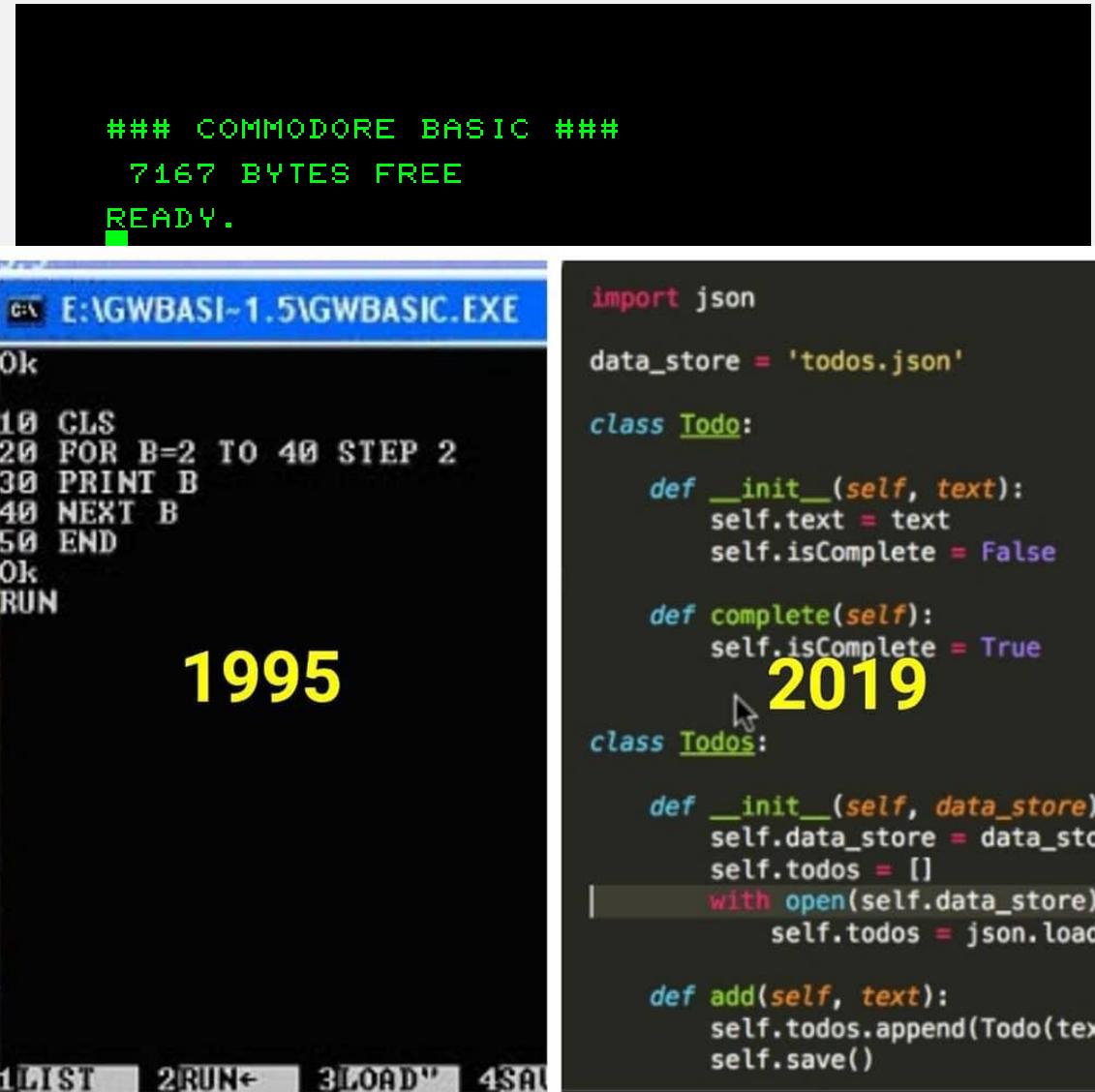


오픈소스가 더 재미있고 쉬워진 계기

- 1. 프로그래밍 언어

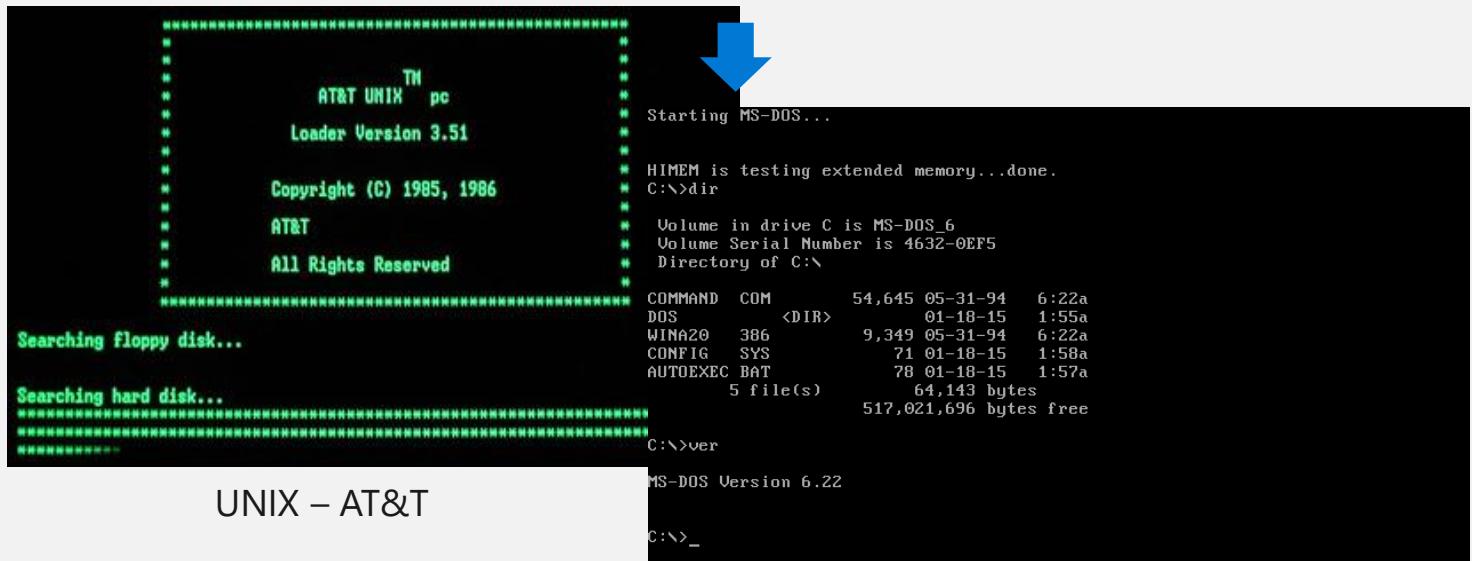
- 저수준 vs. 고수준 언어
- 컴파일러 vs. 인터프리터
- 절차적 vs. 객체지향 vs. 함수형
- ...

A screenshot of a Commodore 64 assembly debugger. The screen shows assembly code in the top half and a memory dump in the bottom half. The assembly code includes instructions like CLS, FOR, PRINT, NEXT, and END. The memory dump shows hex values for the program's memory. At the bottom, there are status indicators: Console=0, Dynamic=0, Tmp=0, Printer=0, and a timestamp of 0:00:56.



오픈소스가 더 재미있고 쉬워진 계기

- 2. 공유가 편해진 방식 매체, 인터넷



다양한 휴대용 스토리지



Tape → Floppy Disk → Hard Disk → SSD

응용프로그램 소프트웨어
- 워드프로세서, 스프레드시트,
유틸리티, 게임, ...

• 소프트웨어 소유자는?

- 회사 또는 개인
- S/W 저작권 (Copyright)

오픈소스가 더 재미있고 쉬워진 계기

- 3. 공유가 가능해진 저작권

- GPL vs. LGPL
- AGPL license
- Apache License
- MIT License
- ...



리차드 스톤먼



GNU is Not UNIX 로고

• 왜 이러한 운동이 일어났는가?

- 1980년대, 컴퓨터 하드웨어에 포함된 전용 운영체제들은 회사에 소유된 소스였음
- UNIX의 경우, 오픈 소스는 아니나 개방형 그룹을 통해 자유로운 의사소통으로 발전된 운영체제임에도, 기업들은 규정 등을 통해 자유로운 소스를 통한 개발자간 의사소통 제한
- 자유 소프트웨어를 강조하는 철학 하에, GPL (GNU 공개 라이선스) 및 Copyleft 용어가 널리 확대됨

2. 마이크로소프트 이야기로 살펴보는 AI with Open Source, Cloud

Microsoft의 시작



HBIOS 버전 1.20a.
Copyright (C) Microsoft Corp 1993

GW-BASIC 3.20
<C> Copyright Microsoft 1983,1984,1985,1986,1987,1988
한글 GW-BASIC 3.20 Release 1.12
저작권자 <주>한국마이크로소프트 1986-1988
60411 바이트를 사용할 수 있습니다
Ok

1LIST 2RUN▼ 3LOAD" 4SAVE" 5CO

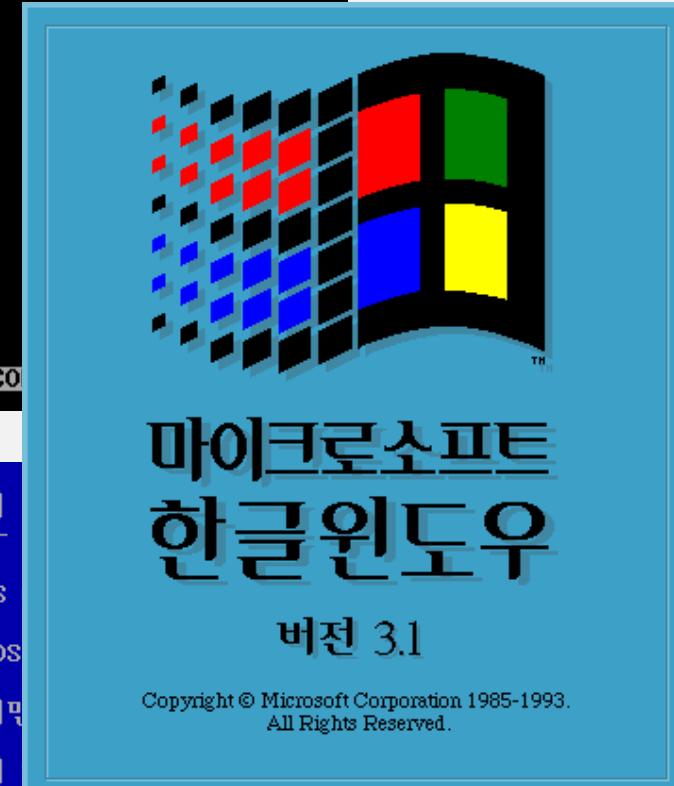
마이크로소프트 한글 MS-DOS 6.2 설치

마이크로소프트 한글 MS-DOS
설치 프로그램은 한글 MS-DOS

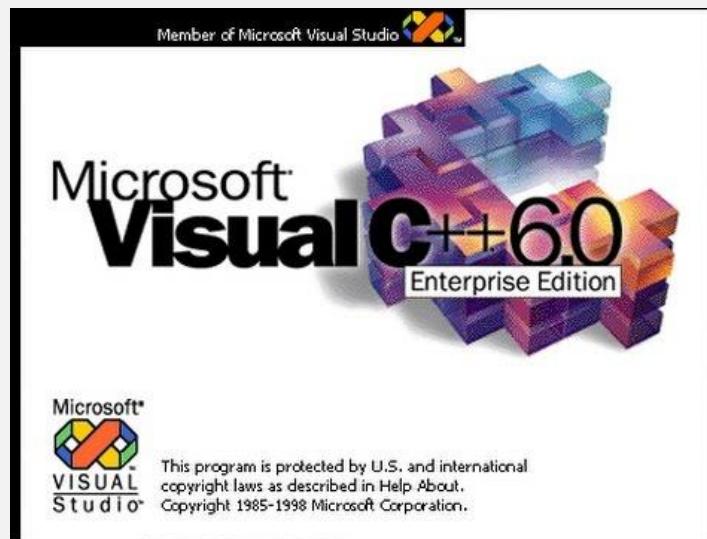
- 한글 MS-DOS를 설치하려면 F3키를 누르십시오.
- 설치 프로그램에 대해서는 F1키를 누르십시오.
- 한글 MS-DOS를 설치하지 않고 종료하려면 F7키를 누르십시오.

주의: 백업을 하려면 한글 MS-DOS를 설치하기 전에 하십시오.
파일을 백업하려면 F3키를 눌러 설치를 종료한 다음
백업 프로그램을 사용하여 파일을 백업합니다.

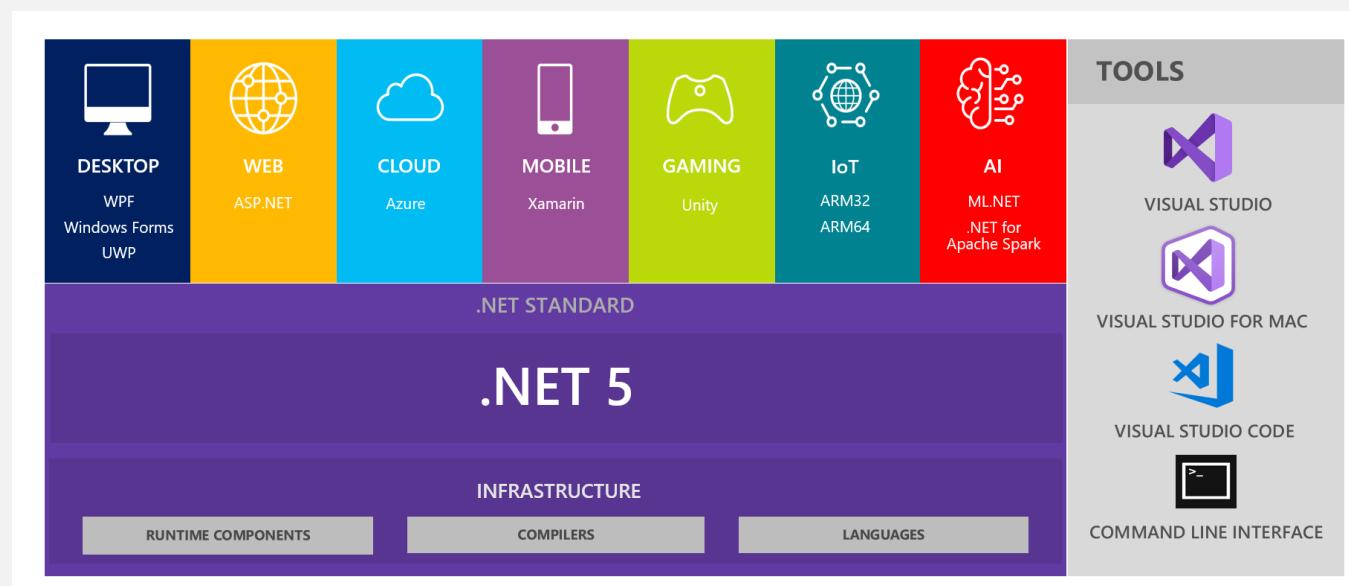
설치 프로그램을 계속하려면 Enter키를 누르십시오.



Microsoft와 개발자 & Visual C++/Studio & .NET



```
C:\Administrator: Command Prompt - powershell
C:\Users\Administrator>powershell
Copyright © 2006 Microsoft Corporation. All rights reserved.
PS C:\Users\Administrator>
```



Microsoft의 여정



Azure, 클라우드의 등장 (2009)

Windows Azure Platform



Windows Azure™



Microsoft
SQL Azure™



Windows Azure™ platform
AppFabric

- Compute
- Storage
- Fabric Controller
- CDN
- Connect

- Relational DB in cloud

- Service Bus
- Access Control
- Caching
- Integration
- Composite App

Windows Azure? Microsoft Azure?



Microsoft Azure

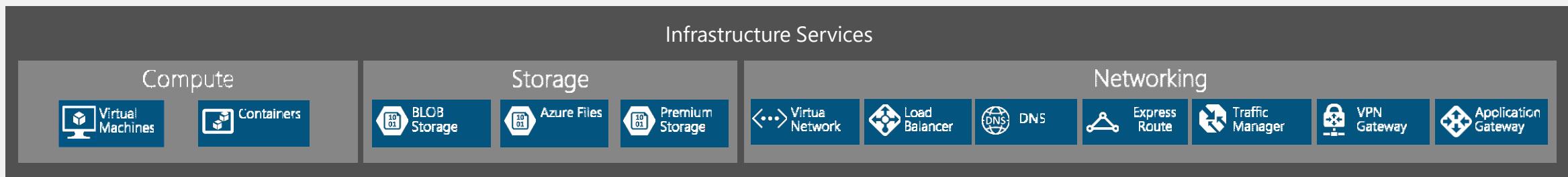
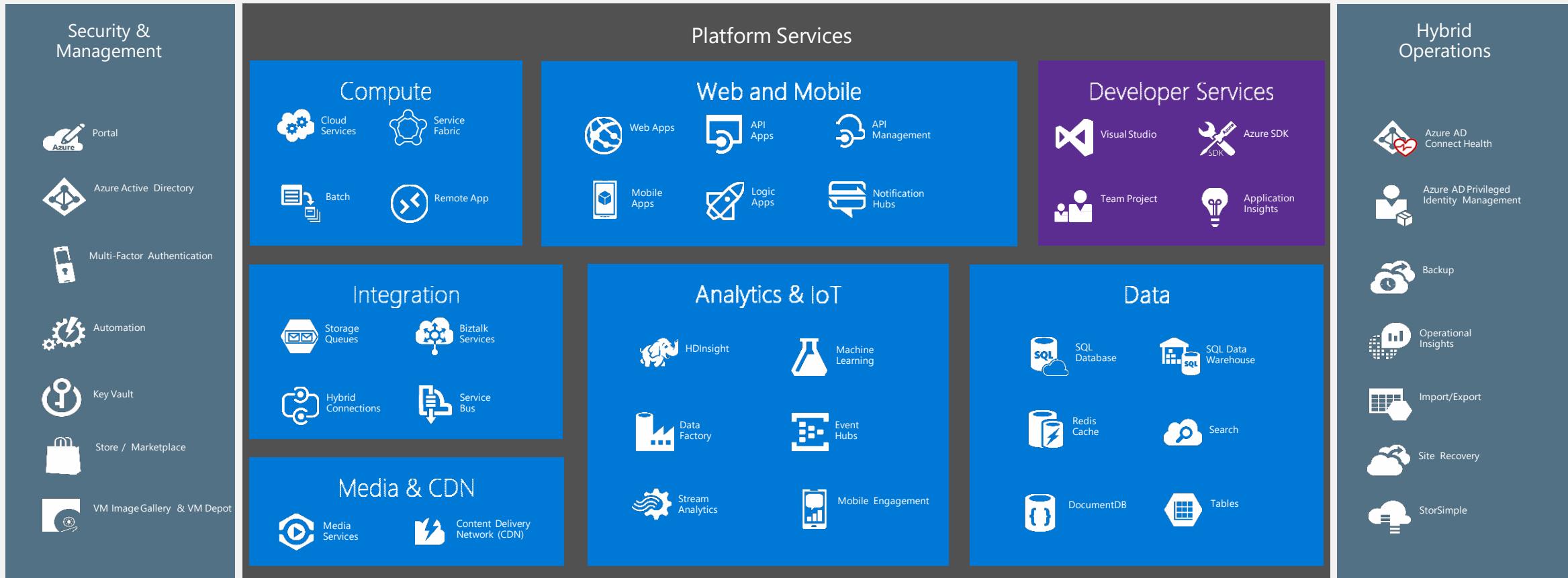
클라우드 기반 Microsoft 서비스



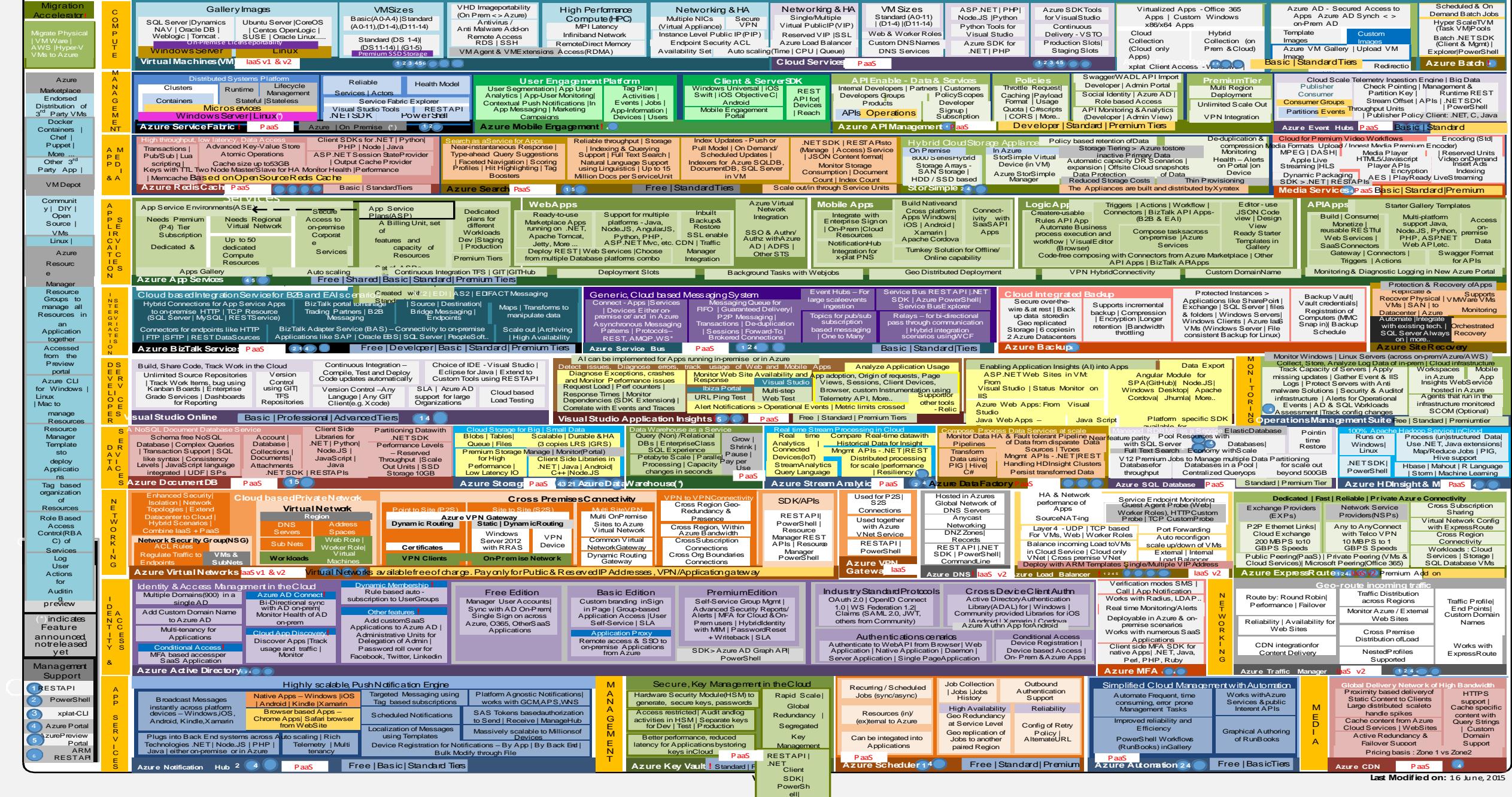
200+ Cloud Services

1+ billion customers · 20+ million businesses · 90+ markets worldwide

다양한 Azure 서비스



Microsoft Azure - Services Overview



선택과 유연성에 따라 다양하고 창의적인 구축 가능

DevOps	Nagios®							
Management								
Applications								
App frameworks & tools								
Databases & middleware								
Infrastructure								

대중적인 도구 및 프레임워크와의 높은 호환성

Microsoft ❤️ Open Source

관리



데이터베이스 & 미들웨어



다양한 응용 프로그램



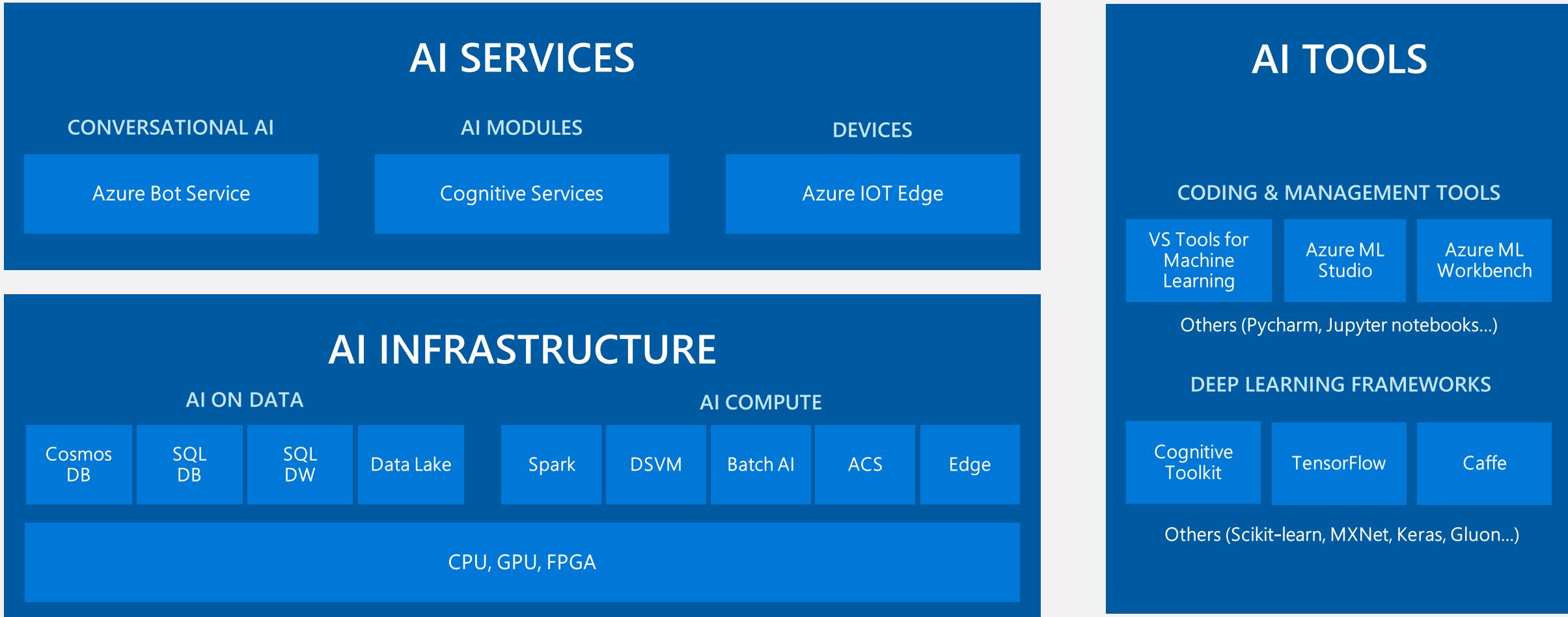
인프라



앱 프레임워크



Microsoft AI Platform: Azure + AI



Microsoft AI Platform: Azure + AI

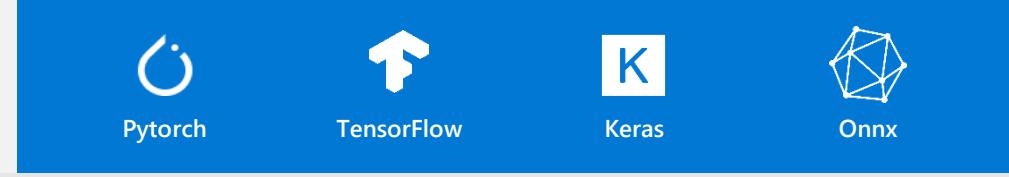
정교하게 미리 학습된 모델

솔루션 개발을 손쉽게 구현하기 위한 방법



유용한 프레임워크 활용

고급 딥 러닝 솔루션을 구축하기 위한 방법



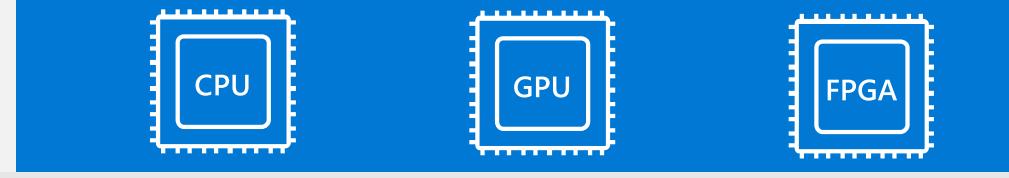
다양한 서비스 활용을 통합 생산성 향상

Data science와 개발팀을 위한 역량 강화



강력한 인프라스트럭처

효율적이고 원활한 딥 러닝 환경제공



유연하고 다양한 배포 모델

인텔리전트 클라우드와 Edge환경 모두 적용가능한 배포 및 관리 지원



3. 마이크로소프트와 강화 학습

마이크로소프트: AI 스타트업 인수

Microsoft to acquire Bonsai in move to build
'brains' for autonomous systems

Jun 20, 2018 | Gurdeep Pall - Corporate Vice President, Business AI



Bonsai's team members. Photo courtesy of Bonsai.

[뉴스]

MS, 기계학습 전문 기업 '본사이(Bonsai)' 인수

마이크로소프트(이하 MS)가 기계학습(머신러닝) 전문 기업 '본사이(Bonsai)'를 인수하였다.

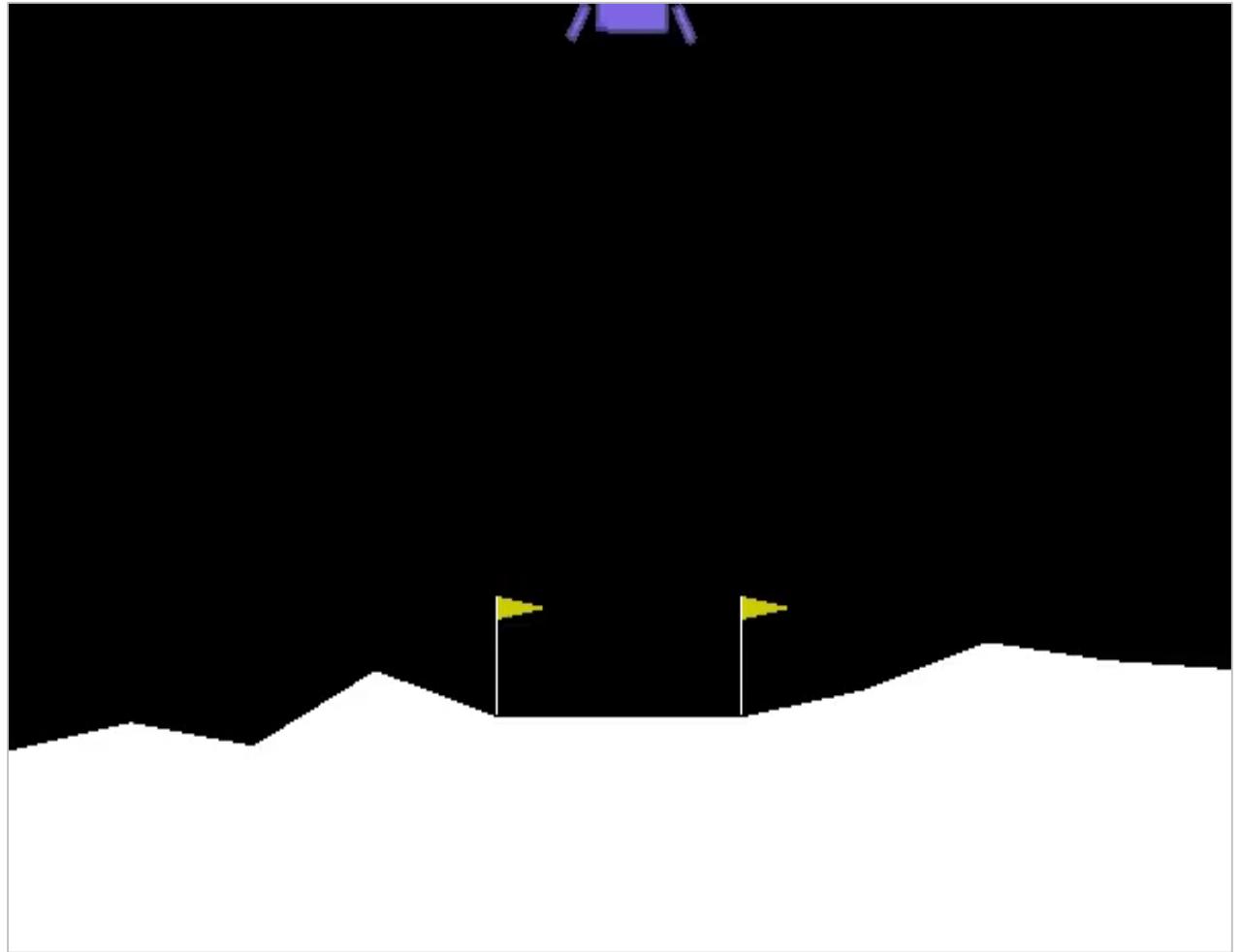
마이크로소프트

마이크로소프트는 '에듀테크'와 '인공지능' 등 경쟁 IT 업체들에 대응하기 위한 차원의 스타트업 인수를 추진했다. 2018년 6월에 인수한 미국 미네아폴리스 소재 '플립그리드(Flipgrid)'는 2015년 설립된 온라인 교육/학습 플랫폼 운영 스타트업이다. 앞서 구글이 인수한 '워크벤치'와의 차별점은 아동에서 성인에 달하는 다양한 연령대의 이용자들이 다양한 주제에 대해 짧은 동영상을 활용하여 교육, 학습한다는 점이다. 구글과의 에듀테크 분야 경쟁에 대응하고, 교육 서비스/솔루션 개발 아이디어로 활용하기 위해 인수했으며, 인수 이후에도 별도 서비스로 그대로 운영 중이다.

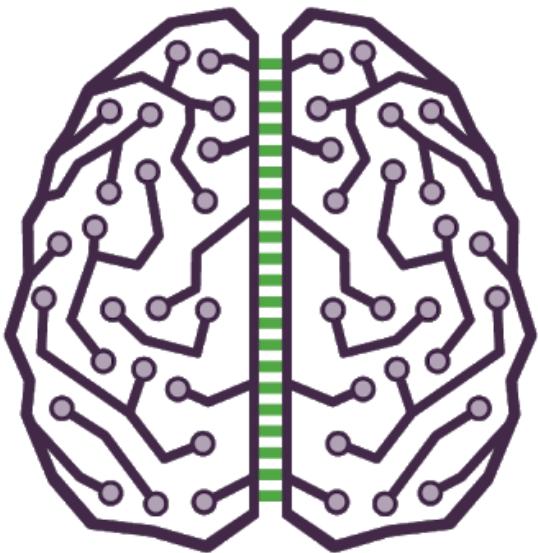
인공지능 분야에서는 2018년 9월에 인수한 'Lobe'와 'Bonsai' 2개 업체 인수가 주목된다. Lobe는 샌프란시스코 소재 스타트업으로서, 공학 전공자가 아닌 시각적 툴(tool)을 활용해 머신러닝 시스템을 개발하도록 도와주는 비주얼 툴을 개발했다. 6월에 인수한 **Bonsai는 산업 자동화용 인공지능 시스템 개발 업체**이다. Lobe와 Bonsai는 마이크로소프트에 흡수되지 않고, 별도 법인으로서 인공지능 시스템 개발을 진행하고 있다.

We know it
works in games.

What about the
real world?



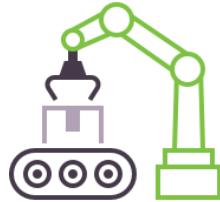
Building BRAINs for Autonomous Systems



- Bonsai founded 2014, acquired by Microsoft on June 2018
- Combining state-of-the-art techniques in machine teaching and **deep reinforcement learning**
- Providing a complete toolchain for building deep reinforcement learning models (aka BRAINs) to optimize and/or control autonomous systems

What kinds of autonomous systems?

Motion control



Machine tuning & calibration



Smart buildings/ homes



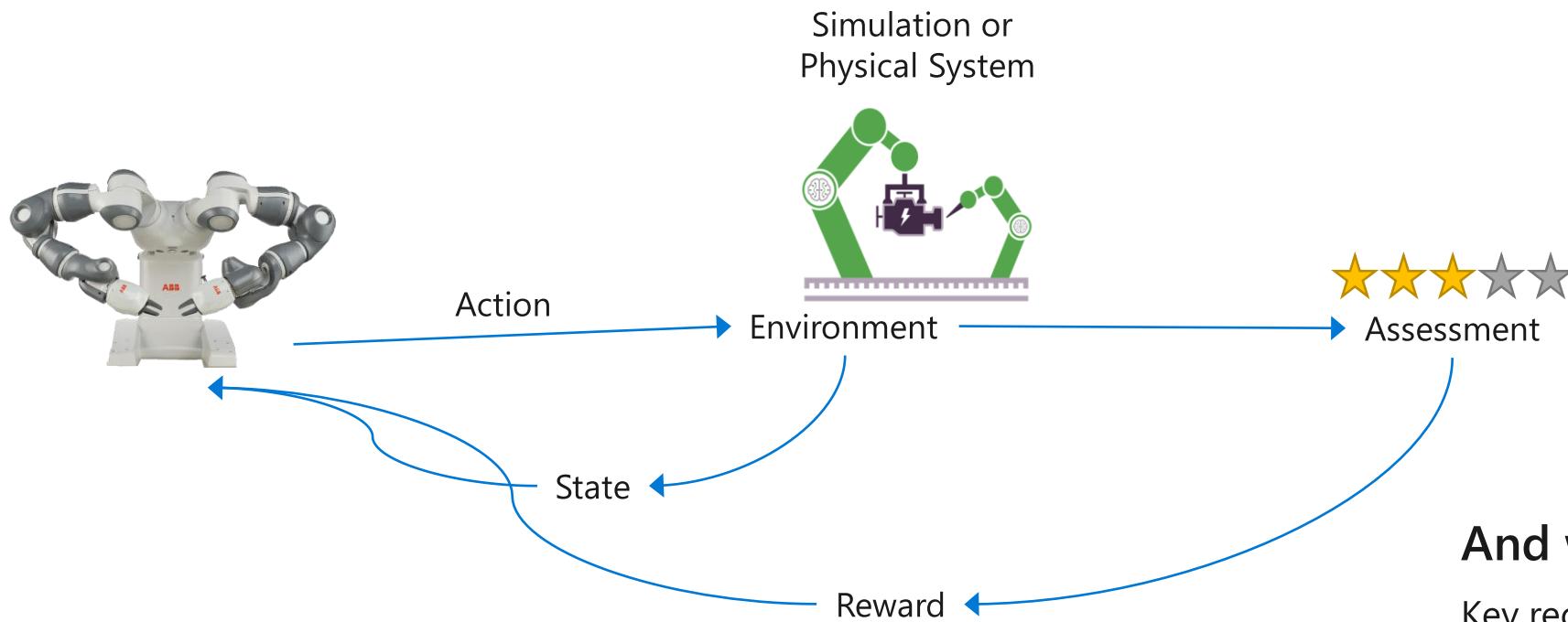
Industrial robotics



Process control & automation



What is deep reinforcement learning?



And why now?

Key recent advancements in reinforcement learning: neural networks, compute infrastructure and algorithms

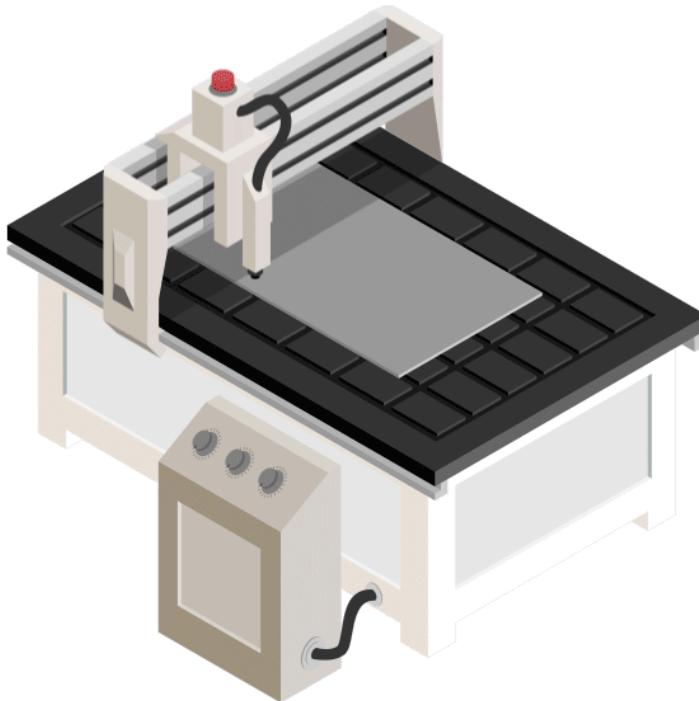
DRL is well suited for complex and dynamic environment (unlike other ML techniques)

Proliferation of connected physical devices and sensors

Applications



Advanced
motion control



Machine
Calibration



Smart Buildings

Microsoft Research activities: TextWorld (open source)

TextWorld

our quest: the yellow basket in the kitchen must be locked with the yellow key. You are in the contemporary bedroom. You cannot miss an open modern coffer within reach. You can see nothing in it. You see a locked suitcase, which looks elegant with each. A locked safe is within reach. Upon examination, you see that it is also green. You can go south from here. You are carrying: a blue latchkey, a red latchkey, a green latchkey, a dry broccoli and an immaculate pencil. Your quest: the yellow basket in the kitchen must be locked with the yellow key. You are in the contemporary bedroom. You cannot miss an open modern coffer within reach. You can see nothing in it.



Unlock the door to text-based adventure



our quest: the yellow basket in the kitchen must be locked with the yellow key. You are in the contemporary bedroom. You cannot miss an open modern coffer within reach. You can see nothing in it. You see a locked suitcase, which looks elegant with each. A locked safe is within reach. Upon examination, you see that it is also green. You can go south from here. You are carrying: a blue latchkey, a red latchkey, a green latchkey, a dry broccoli and an immaculate pencil. Your quest: the yellow basket in the kitchen must be locked with the yellow key. You are in the contemporary bedroom. You cannot miss an open modern coffer within reach. You can see nothing in it.



Try TextWorld today

Play a TextWorld-Generated Game



You are hungry! Let's cook a delicious meal. Check the cookbook in the kitchen for the recipe. Once done, enjoy your meal!

-- Backyard --

You find yourself in a **backyard**.

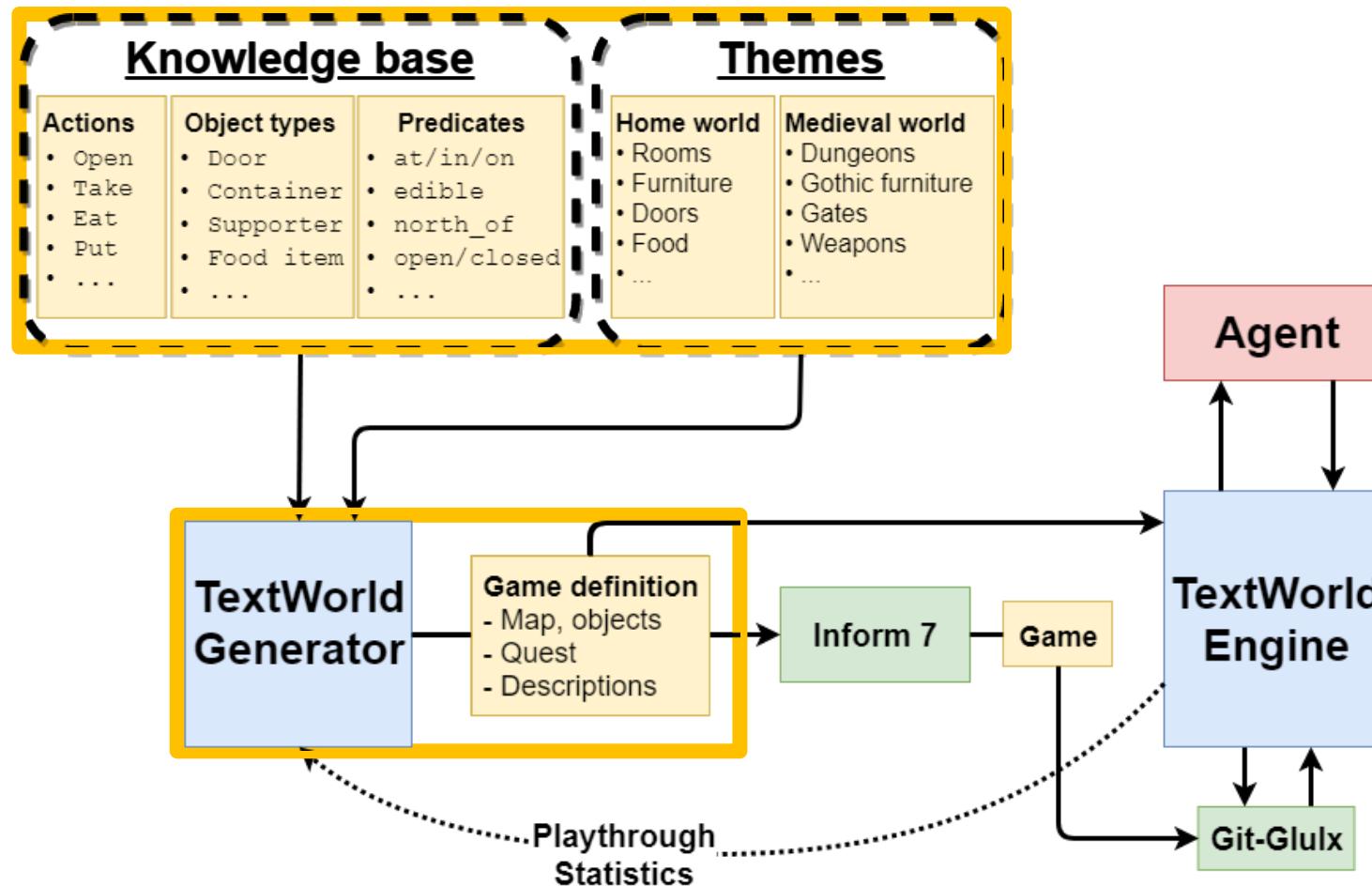
You make out a **patio table**. But the thing is empty. You see a **patio chair**. Wow, isn't TextWorld just the best? The **patio chair** is stylish. But there isn't a thing on it. You see a gleam over in a corner, where you can see a **BBQ**.

There is a closed **screen door** leading south. There is a closed **wooden door** leading west. There is an exit to the east. Don't worry, there is no door.

>

Try it!
aka.ms/textworld

Auto-Generate Text-Based Games Using TextWorld



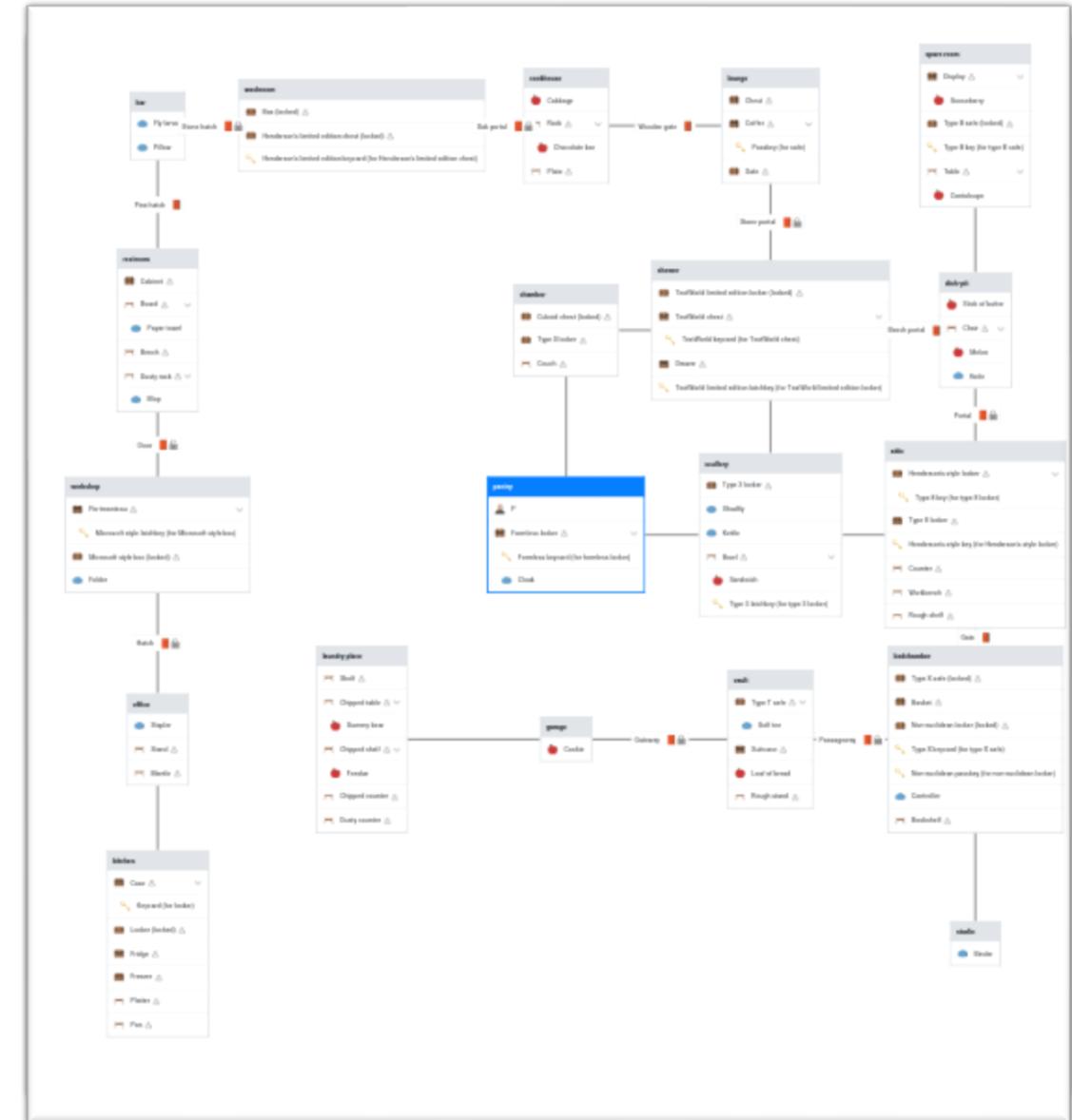
Auto-Generate Text-Based Games Using TextWorld

```
> tw-make custom  
          --quest-length 5  
          --world-size 1  
          --nb-objects 10
```

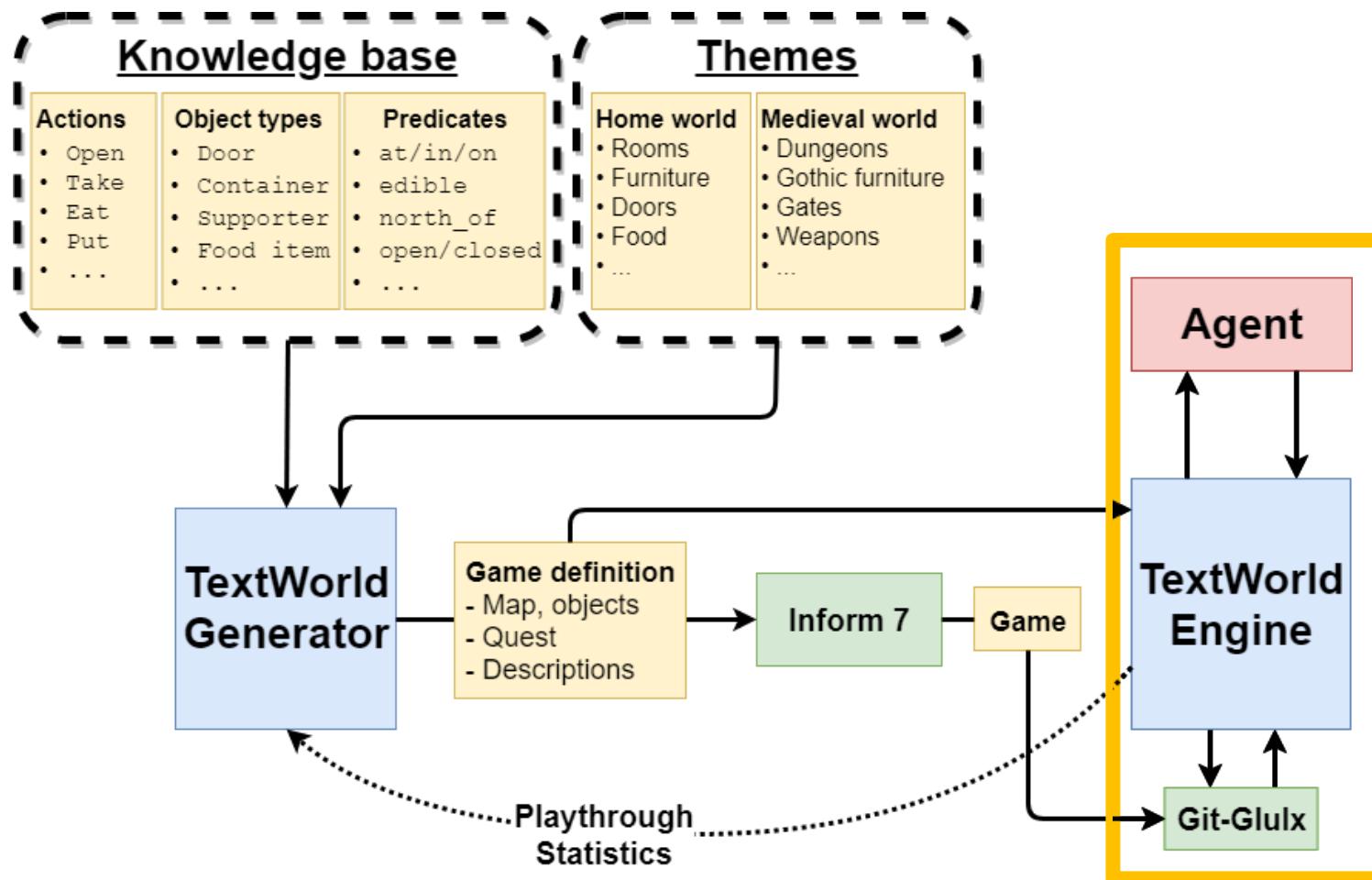
```
> tw-make custom  
          --quest-length 5  
          --world-size 5  
          --nb-objects 10
```

```
> tw-make custom  
          --quest-length 5  
          --world-size 20  
          --nb-objects 10
```

```
> tw-make custom  
          --quest-length 5  
          --world-size 20  
          --nb-objects 100
```



Evaluate An Agent's Skills Using TextWorld



Playing TextWorld Games Using OpenAI Gym

Register the game with Gym

In order to call to `gym.make`, we need to create a valid `env_id` for our game.

```
import textworld.gym  
env_id = textworld.gym.register_game('tw_games/game.ulx')
```

Make the gym environment

With our `env_id` we are ready to use `gym` to start the new environment.

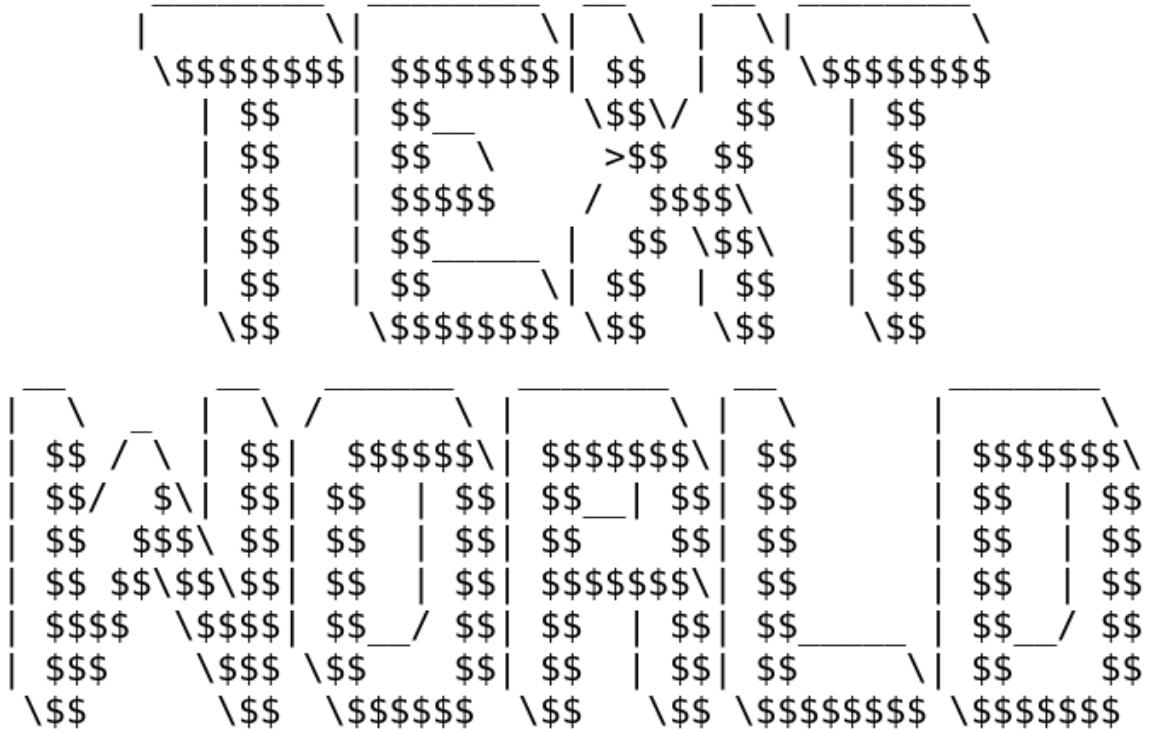
```
import gym  
env = gym.make(env_id)
```

Start the game

Like for other Gym environments, we start a new game by calling the `env.reset` method. It returns the initial observation text string as well as a dictionary for additional informations (more on that later).

```
obs, infos = env.reset()  
print(obs)
```

Playing TextWorld Games Using OpenAI Gym



Welcome to another fast paced game of TextWorld! Here is your task for today. First, it would be great if you could try to go east. With that accomplished, take the shoe that's in the attic. With the shoe, place the shoe on the shelf. Alright, thanks!

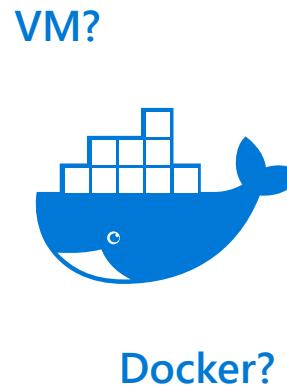
4. With AI developers

: 공유하고 싶은 이야기

Building your own AI models

배포할 때

Machine Learning
결과



- On-prem deployment
- Azure IoT edge
- Azure Container Instance
- Azure Managed Kubernetes Service
- IoT accelerated targets
- Project Brainwave (FPGAs)

Building your own AI models

더 많은 데이터를 활용할 때

SQL DB

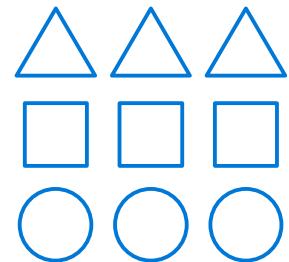
Cosmos DB

Datawarehouse

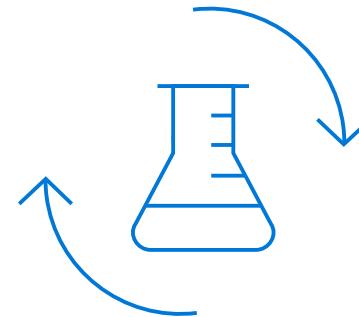
Data lake

Blob storage

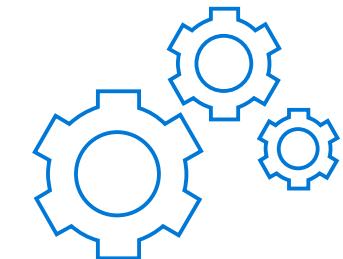
...



Prepare data

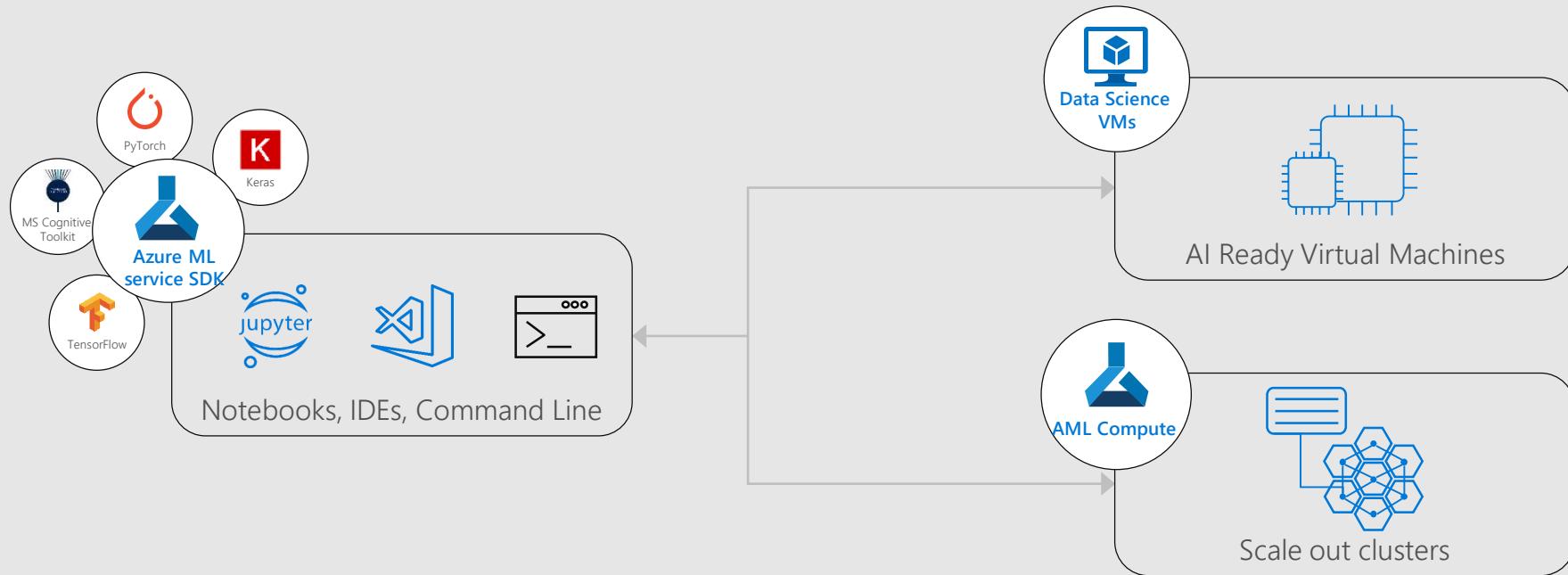


Build and train



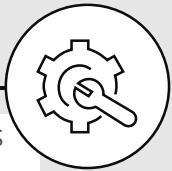
Deploy

Build and deploy deep learning models



Streamline AI development efforts

- Leverage popular deep learning toolkits
- Develop your language of choice



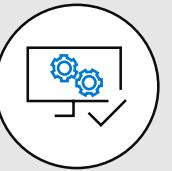
Scale compute resources in any environment

- Choose VMs for your modeling needs
- Process video using GPU-based VMs

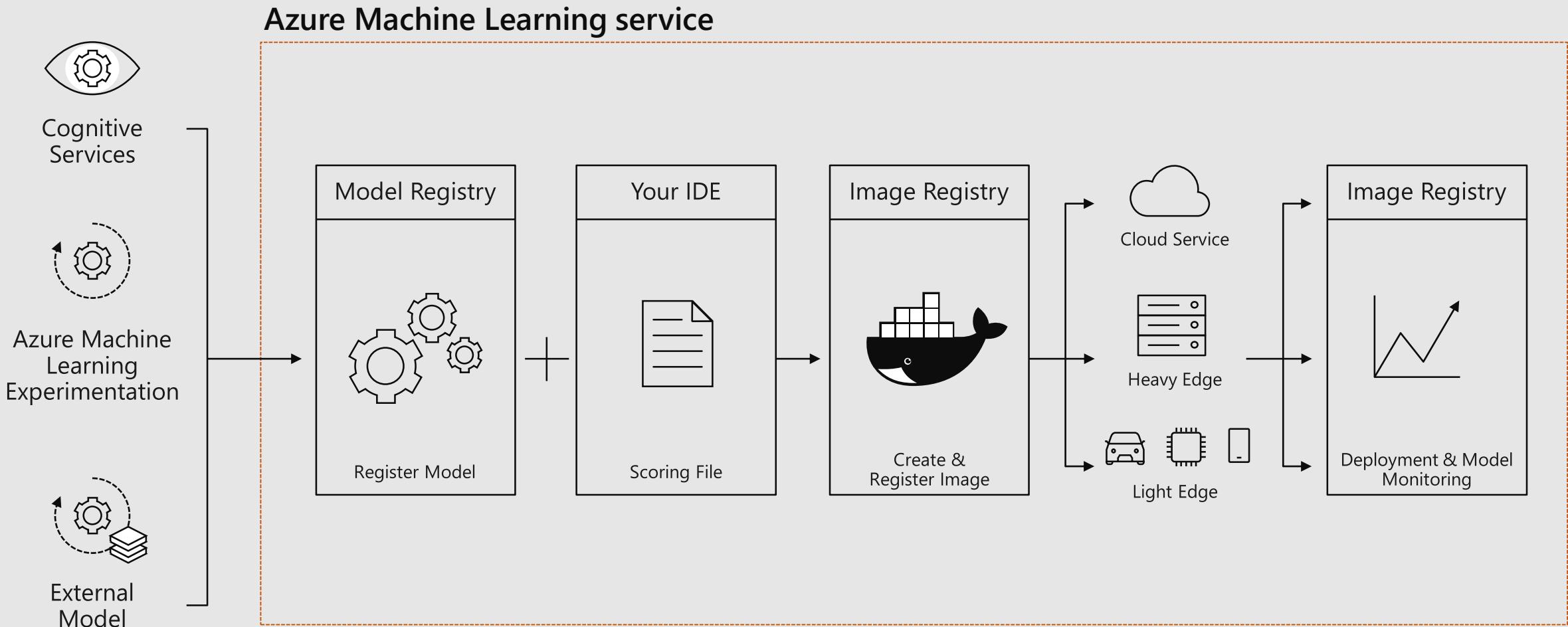


Quickly evaluate and identify the right model

- Run experiments in parallel
- Provision resources automatically



Deploy Azure ML models at scale



REST API & SDKs: open to all developers!

Azure Cognitive Services Documentation

Learn how to build intelligent algorithms into apps, websites, and bots so that they see, hear, speak, and understand your user needs through natural methods of communication. Tutorials, quickstarts, API references, and examples show you how.



What are Cognitive Services?

Learn about Azure Cognitive Services and how to use the different APIs to build intelligent apps.



Create a Cognitive Services account

Learn how to create an Azure Cognitive Services account.



Authenticate a request

Learn how to authenticate requests to Azure Cognitive Services.



Language support in Cognitive Services

Learn about the human languages supported by Cognitive Services APIs.



Using Cognitive Services in Containers

Learn how to use Azure Cognitive Services in Containers.

5-Minute Quickstarts



Analyze image using Computer Vision

Using C#
Using Java
Using Node.js
Using Python



Recognize speech using Speech Service

Using C#
Using JavaScript
Using Python
Using Java



Get intent using LUIS

Using C#
Using Java
Using Node.js
Using Python



Detect language, analyze sentiment using Text Analytics

Using C#
Using Java
Using Node.js
Using Python



Get relevant web queries using Bing Web Search

Using C#
Using Java
Using Node.js
Using Python



Get relevant image queries using Bing Image Search

Using C#
Using Java
Using Node.js
Using Python

- ✓ Using the REST API
- ✓ Analyze a remote image
 - cURL
 - Go
 - Java
 - JavaScript
 - Node.js
 - PHP
 - Python
 - Ruby
- > Analyze a local image
- ✓ Generate a thumbnail
 - C#
 - cURL
 - Go
 - Java
 - JavaScript
 - Node.js
 - PHP
 - Python
 - Ruby

Familiar Data Science tools

Choose any python development environment



PyCharm



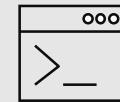
Jupyter



Visual Studio Code



Zeppelin



Command line

And improve data science productivity

Run Properties		Output Logs	
Status	Running	[2018-08-31T10:59:35.289459][INFO][!][OSSOURCE] Generated 10 jobs - 100% completed for request [100].	
Start Time	8/31/2018 10:45:45 AM	[2018-08-31T10:59:56.025779][INFO][!][WARNING] Jobs	
Duration	5:18:43	[2018-08-31T10:59:56.025779][INFO][!][WARNING] Run [RunKeyrun_id=1535705145692_19] has been submitted to account net, subscription_id=[15ae0fc6-95c1-48d3-aed3-1b1af1ab0f32], resource_group=[DanielSc], workspace_name=[Daniel1C], project_name=[tensorflow-hypervideo], run_id=[tensorflow-hypervideo, 1535705145692_19] do not contain any metrics named 'epoch_val_acc' at this moment, policy cannot be applied.	
Run Id	tensorflow-hypervideo_1535705145692_19	[2018-08-31T11:00:44.765888][INFO][!][WARNING] Job [RunKeyrun_id=1535705145692_19] has been submitted to account net, subscription_id=[15ae0fc6-95c1-48d3-aed3-1b1af1ab0f32], resource_group=[DanielSc], workspace_name=[Daniel1C], project_name=[tensorflow-hypervideo], run_id=[tensorflow-hypervideo, 1535705145692_19] do not contain any metrics named 'epoch_val_acc' at this moment, policy cannot be applied.	
Max concurrent runs	4		
Max total runs	100		

		Completed (66)		Cancelled (33)	
Run	Best Metric*	Status	Started	Duration	Run Id
22	0.9801333546638489	Completed	Aug 31, 2018 10:55 AM	0:07:47	tensorflow-hypervideo_1535705145692_12
29	0.9799333214758827	Completed	Aug 31, 2018 11:07 AM	0:02:35	tensorflow-hypervideo_1535705145692_19
21	0.9798666834831238	Completed	Aug 31, 2018 10:54 AM	0:10:29	tensorflow-hypervideo_1535705145692_11
52	0.979733478927612	Completed	Aug 31, 2018 11:31 AM	0:08:52	tensorflow-hypervideo_1535705145692_42
12	0.97966666502952576	Completed	Aug 31, 2018 10:46 AM	0:06:53	tensorflow-hypervideo_1535705145692_3

Interactive widgets for Jupyter Notebooks

The screenshot shows a Jupyter Notebook interface with two main panes. The left pane contains a code cell with the following Python code:

```
Run Cell | Run All Cells
1 %%markdown
2 ## This is markdown.
3
4 Run Cell | Run All Cells
4 %% 
5 # Let's load and review some data
6 import pandas as pd # pandas is a dataframe
7 df = pd.read_csv("./data/pima-data.csv")
8 df.head(5)
9
10 Run Cell | Run All Cells
10 %% Let's plot the correlation between data
11 import matplotlib.pyplot as plt # matplotlib
12 import matplotlib.style as style
13 style.use('dark_background')
14
15 def draw_corr(df, size = 11):
16     corr = df.corr() # data frame correlate
17     fig, ax = plt.subplots(figsize=(11, 11)
18     ax.matshow(corr) # color code the rect
19     plt.xticks(range(len(corr.columns)), c
20     plt.yticks(range(len(corr.columns)), c
21
22 draw_corr(df)
```

The right pane shows a "Python interactive" window titled "This is markdown." containing the text "This is markdown." and a code cell starting with "%%". Below this is a heatmap visualization of the correlation matrix for the Pima Indians dataset, showing a strong positive correlation along the diagonal and some negative correlations off-diagonal.

Azure Machine Learning for Visual Studio Code extension

Popular frameworks

Build advanced deep learning solutions

Use your favorite machine learning frameworks



TensorFlow



PyTorch



Scikit-Learn



MXNet



Chainer



Keras



without getting locked into one framework



ONNX

Community project created by Facebook and Microsoft

Use the best tool for the job. Train in one framework
and transfer to another for inference



NVIDIA



QUALCOMM



Windows



vespa



SOPHON



intel



skymizer



MathWorks

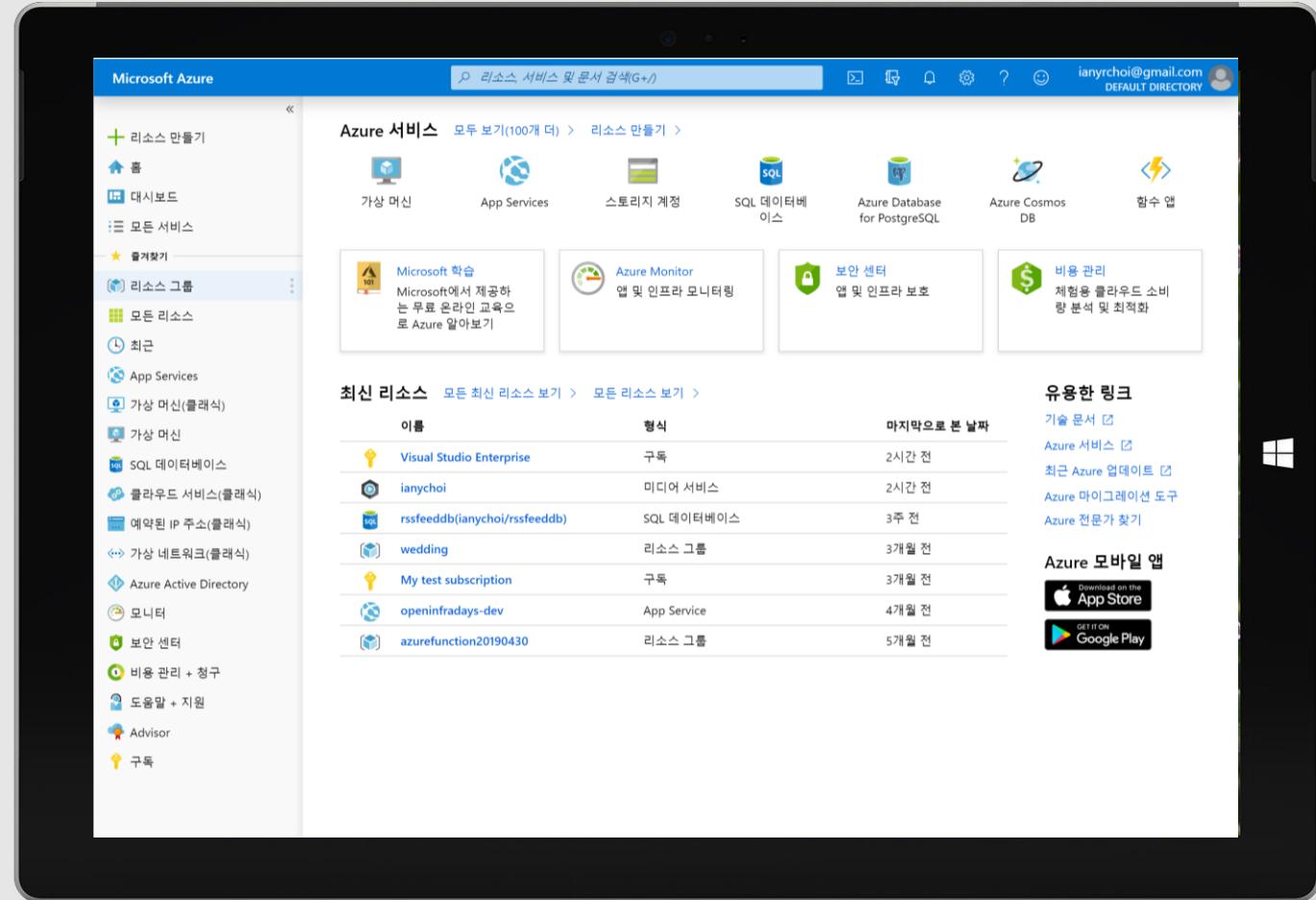


NETRON



Visual DL

Azure를 시작해 봅시다



시작하기
azure.microsoft.com



<http://aka.ms/do-azure>

 800 XP

Data Science Virtual Machine에서 Jupyter, Docker 및 PyTorch를 통한 대화형 딥 러닝

1시간 14분 • 모듈 • 7 단위

★★★★★ 4.7 (35)

중급

개발자

Azure

Virtual Machines

Jupyter, PyTorch 및 Data Science Virtual Machine에서 딥 러닝 모델 학습을 살펴봅니다.

이 모듈에서는 다음을 수행합니다.

- 심층 학습 및 Azure DSVM(Data Science Virtual Machine)에 대한 설명
- Azure CLI를 사용하여 DSVM 만들기
- Docker를 사용하여 사용자 지정 심층 학습 환경을 설정하는 방법 알아보기
- DSVM에서 Docker를 통해 Jupyter Notebook에 연결
- Jupyter를 사용하여 첫 번째 딥러닝 실험 실행

