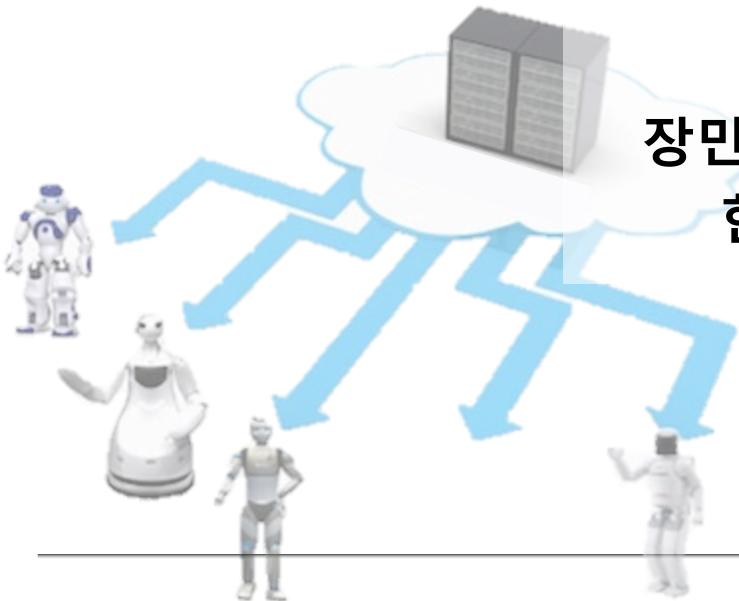


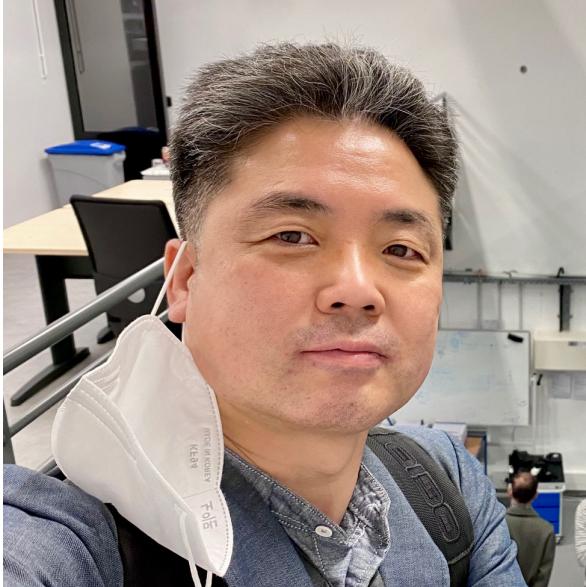
클라우드 로봇과 Robot-as-a-Service(RaaS) 소개

2022.5.31

장민수 (minsu@etri.re.kr)
한국전자통신연구원



발표자 소개



장민수

minsu@etri.re.kr

<https://zebehn.github.io>



- 소속: 한국전자통신연구원 인공지능연구소
인간로봇상호작용연구실
- 학력
 - 서강대 전산학 학·석사
 - KAIST 전산학 박사
- 전공: 인공지능 (음성인식, 제스처 인식)
- 주요 경력
 - 한국로봇학회 편집, 산학연 이사 ('16~)
 - 한국로봇학회 클라우드로봇연구회 회장 ('21~)
 - RA-L 보조편집자, RO-MAN '22/'23 조직위원
- 연구분야: 지식표현/추론, 인간-로봇 상호작용,
소셜로봇, 클라우드로봇, 체화 지능
- 비전: “Self-Aware Robots”

목 차

1. 클라우드 로봇의 개념과 동향
2. 클라우드 로봇의 주요 이슈
3. Robot-as-a-Service(RaaS) 소개
4. 클라우드 로봇 지능 연구 소개
5. Raas 정리

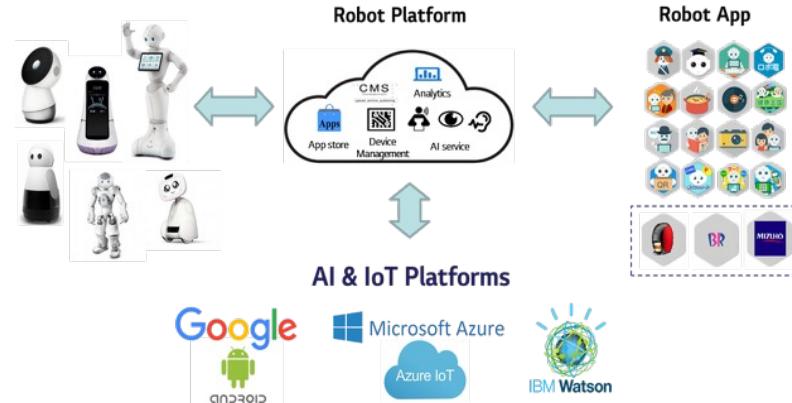
클라우드 로봇의 개념과 동향



(<https://www.rapyuta-robotics.com/2020/06/16/cloud-robotics-our-perspective/>)

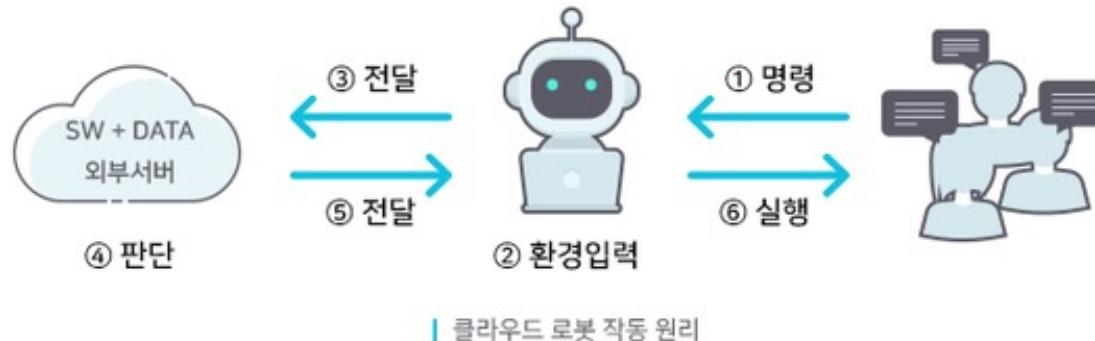
클라우드 로봇은...

- 클라우드 로봇은 클라우드 환경에 존재하는 각종 자원을 활용하여 로봇 작동에 필요한 인식-판단-표현 기능을 수행할 수 있는 로봇
- 클라우드 로봇은 원격 클라우드에 저장된 정보와 계산 자원을 상황에 맞게 활용함으로써 로봇 본체의 물리적 능력 한계를 극복
- 클라우드 로봇은 클라우드를 통해 외부의 지식을 습득하고 타 로봇, 장치, 사람과 협업하여 복잡한 임무를 수행



클라우드 로봇의 특징: 능력의 확장 Extended Brain

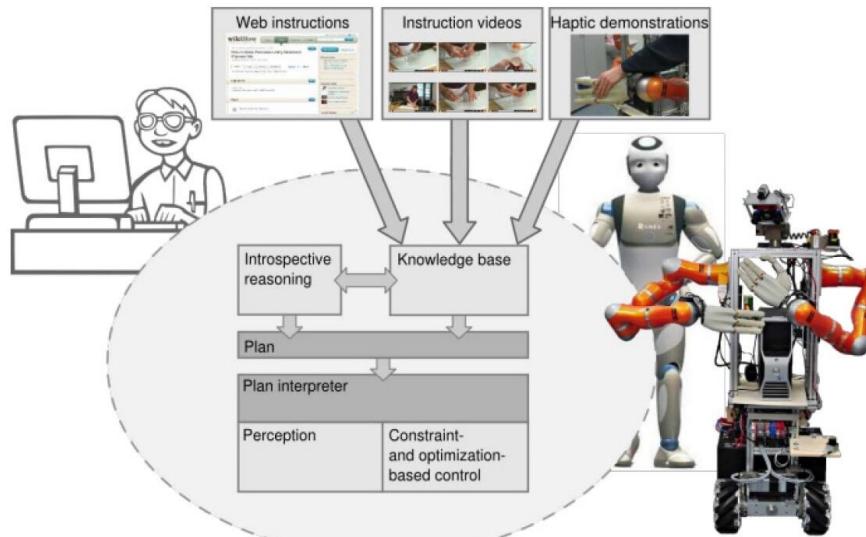
- 고성능 클라우드 서비스를 통해 복잡한 문제를 해결
 - 예: 영상 검출/인식, 음성 인식/합성 등
- 클라우드 기반 인공지능 서비스의 확대
 - 해외: Google Cloud, MS Azure, Amazon WS, IBM Watson 등
 - 국내: 네이버 클라우드, LG CNS, Saltlux ADAM, MindsLAB maum.ai 등



(출처: <https://m.blog.naver.com/sundooedu/221233869585>)

클라우드 로봇의 특징: 지능 공유 Knowledge Sharing

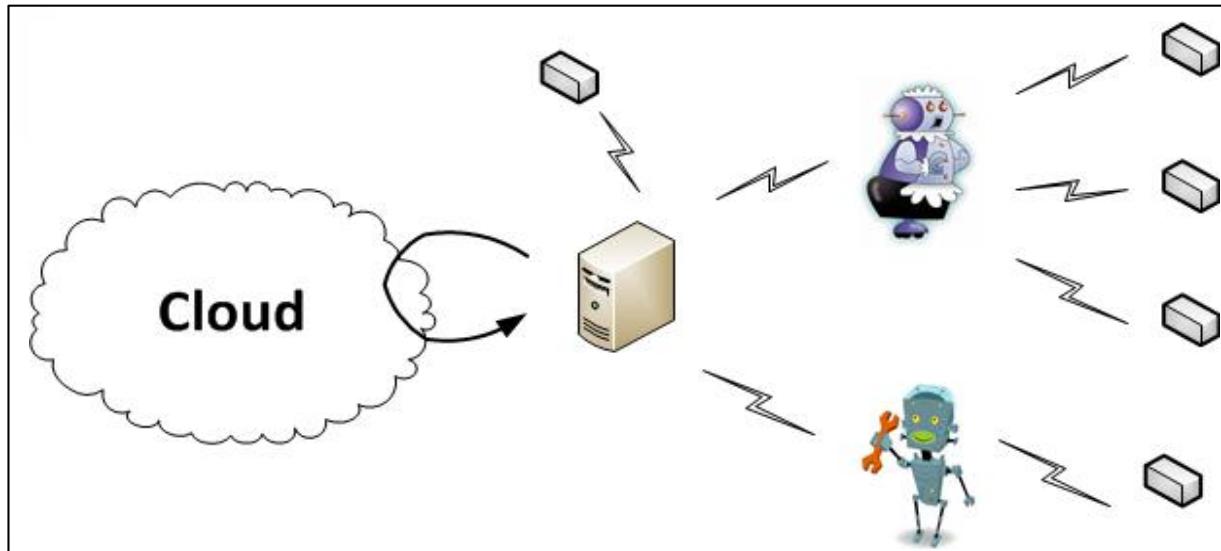
- 로봇이 수집 또는 처리한 데이터를 클라우드에 공유
- 클라우드에서는 로봇들로부터 받은 정보를 통합 가공
- 로봇은 클라우드로부터 작업 수행에 필요한 지식을 입수



(출처: <https://www.cbc.ca/news/trending/robot-learns-to-cook-pizza-pancakes-using-wikihow-youtube-1.3204724>)

클라우드 로봇의 특징: 협업 Collaboration

- 클라우드를 통해 타 로봇, IT 장치 등과 협업하여 효과적으로 목표를 달성
- 각 로봇이 특화된 기능을 통해 임무 수행

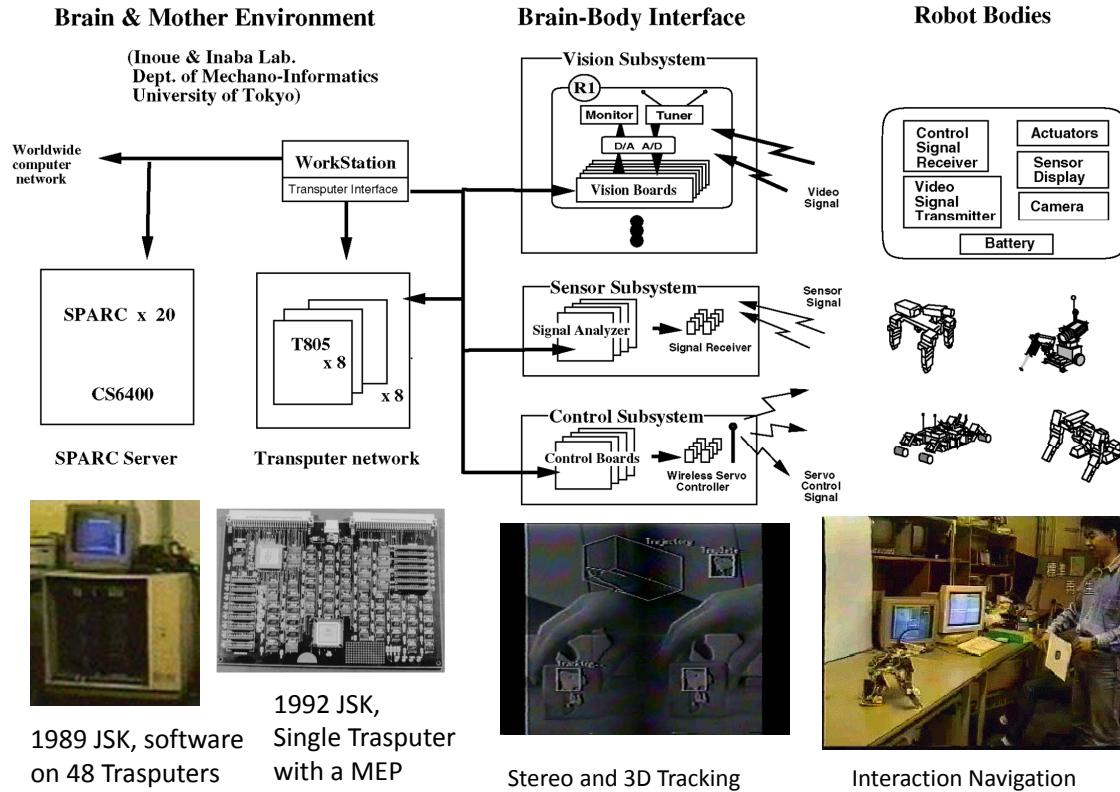


(출처: KEIT PD ISSUE REPORT November 2011 Vol.9)

클라우드 로봇의 장점

- 소형·경량화 된 로봇 개발 가능
- 로봇 솔루션 연구개발 중복 투자 절감 가능
- 로봇 관리의 집중화 구현
- 대규모 로봇 시장 창출 기대

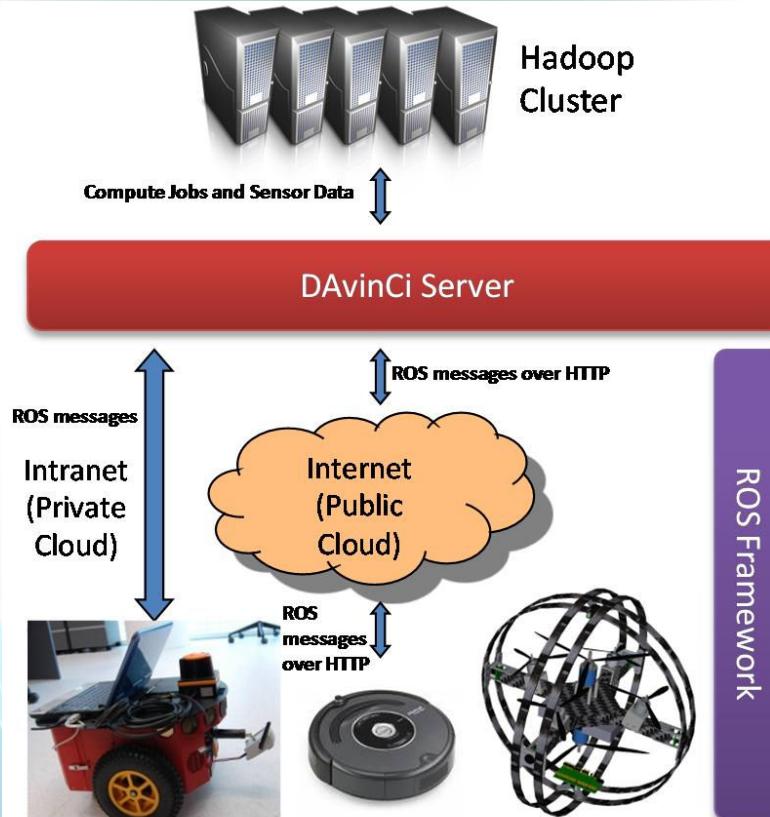
Remote-Brained Robots ('93)



Inaba, Masayuki. "Remote-brained robotics: Interfacing ai with real world behaviors." In *Proc. of 6th International Symposium of Robotics Research*, pp. 335-344. 1993.

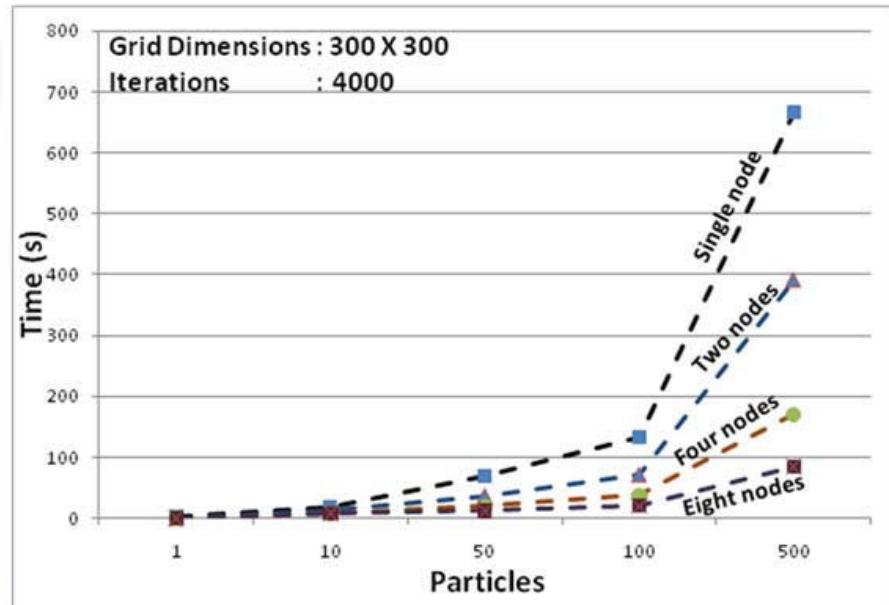
Inaba, Masayuki. "Remote-brained robots." In *International Joint Conferences on Artificial Intelligence*, pp. 1593-1606. 1997.

DAvinCi: A Cloud Computing Framework for Service Robots ('10)



SLAM 처리 속도

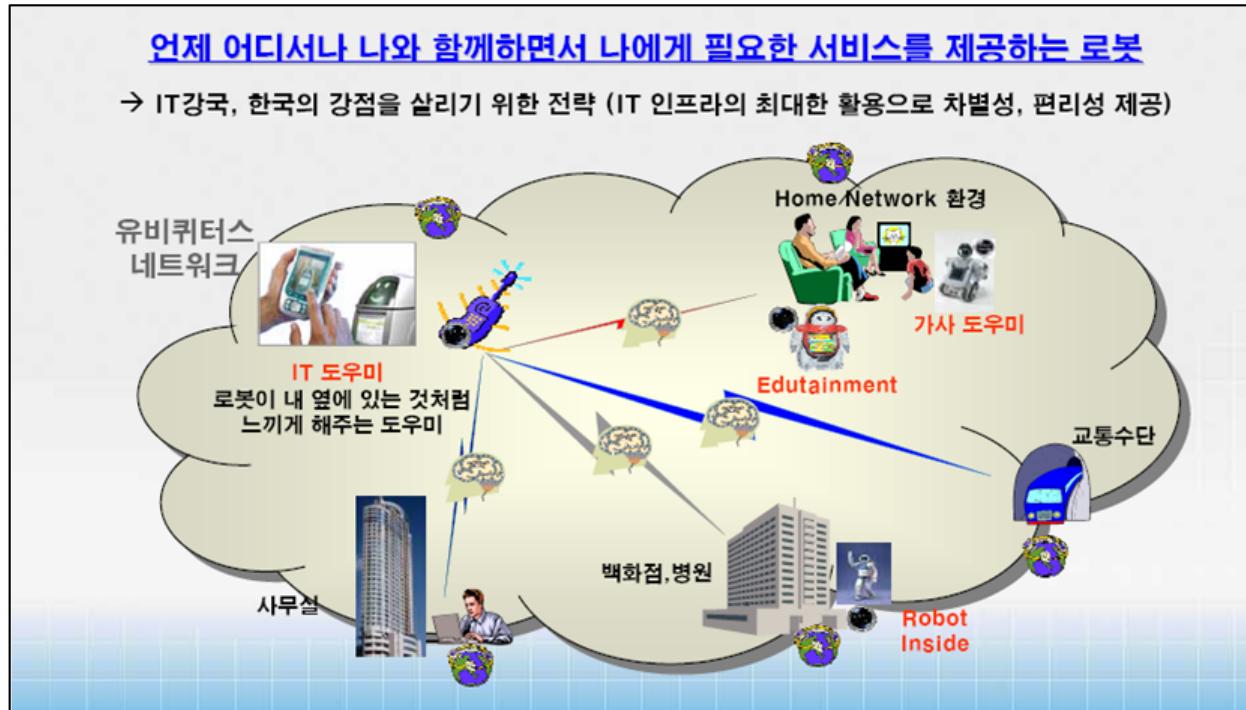
- Single node: 온보드 시스템 활용
- 2 ~ 8 nodes: 클라우드 컴퓨팅 시스템 활용



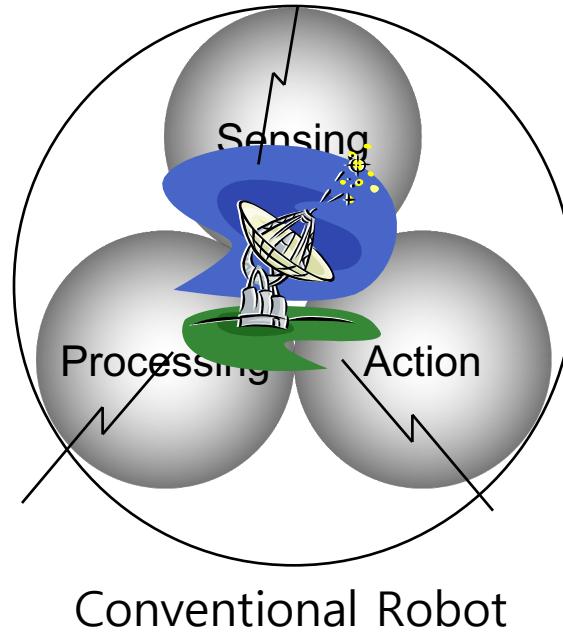
Arumugam, Rajesh, Vikas Reddy Enti, Liu Bingbing, Wu Xiaojun, Krishnamoorthy Baskaran, Foong Foo Kong, A. Senthil Kumar, Kang Dee Meng, and Goh Wai Kit. "DAvinCi: A cloud computing framework for service robots." In *2010 IEEE international conference on robotics and automation*, pp. 3084-3089. IEEE, 2010.

URC: Ubiquitous Robotic Companion ('04)

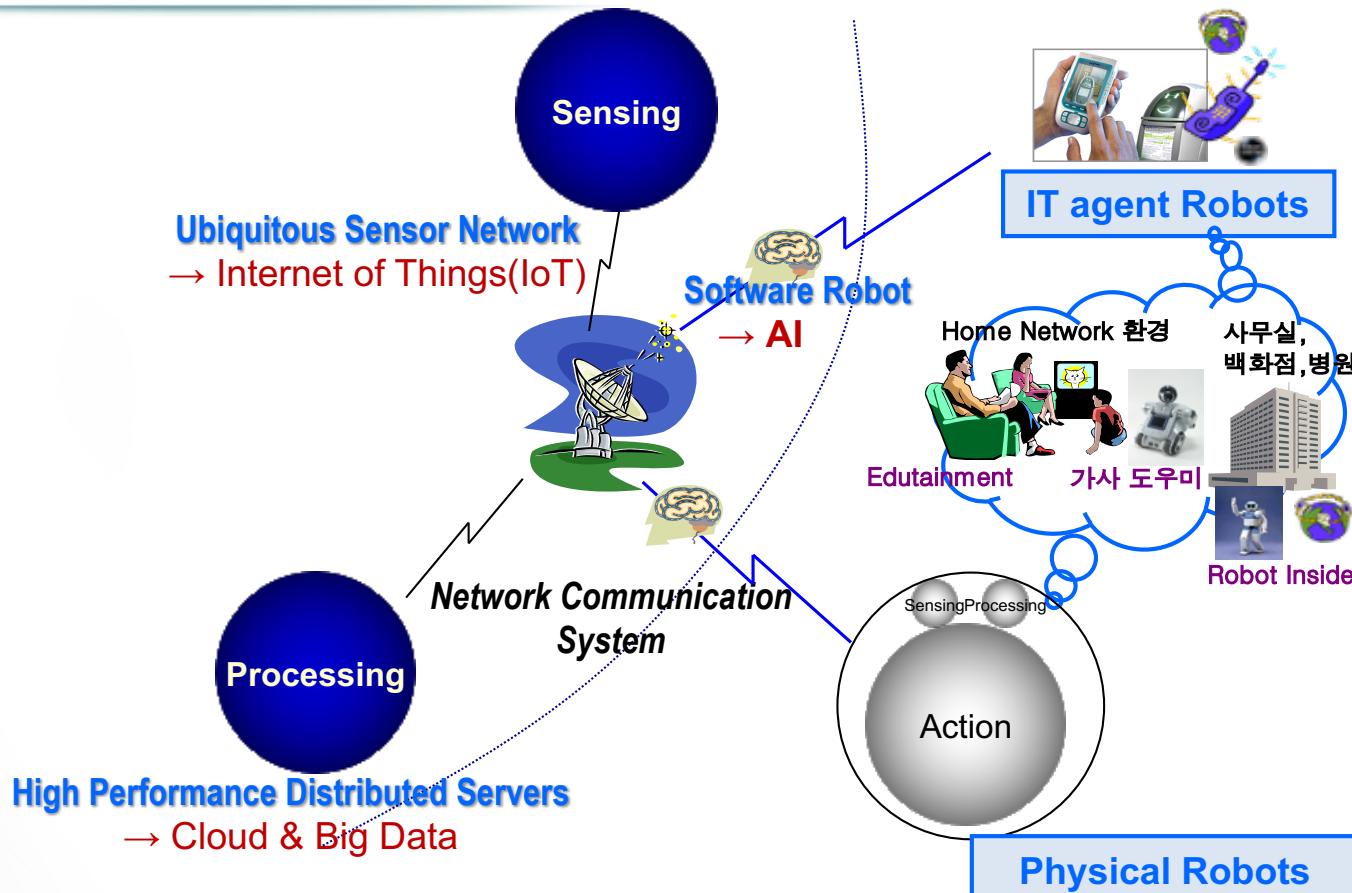
- IT 통신 인프라의 장점을 결합한 경량 로봇 모델을 개발



URC: Ubiquitous Robotic Companion ('04)



URC: Ubiquitous Robotic Companion ('04)



RoboEarth ('09): 한국이 먼저 개념 만들었으나...



쇼핑하는 로봇 PR2. 원하는 물건의 정보를 인터넷에서 확인하고 나서 매장에서 해당 물건을 찾아 쇼핑을 한다. 사람에게 서비스하는 데 필요한 콘텐츠와 소프트웨어 등을 네트워크에 연결된 외부 컴퓨터에서 가져와 쓰는 '클라우드 로봇'이다.

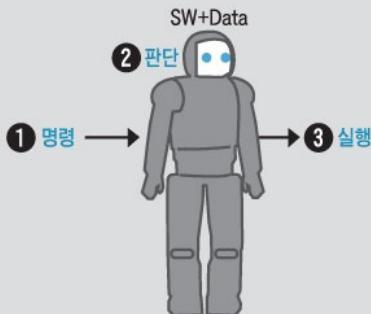
구글이 투자한 美 로봇사 개발
외부 서버 접속해 데이터 얻어

국내서 먼저 개념 만들었지만
정통부 통합되며 사업 흐지부지

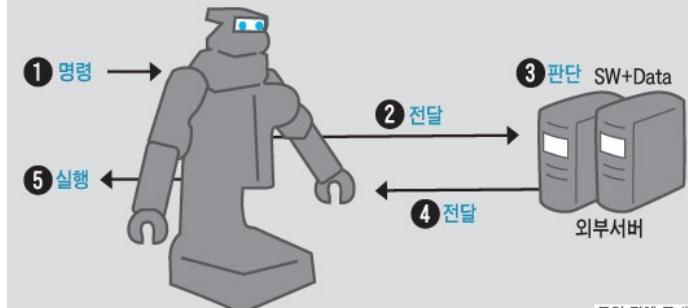
- ◆ 구글 투자 로봇회사가 주도
- ◆ 한국이 먼저 개념 만들었으나 뒤쳐져
- ◆ 우리만의 로봇 콘텐츠 고민해야

- 로봇들이 정보를 공유하고 행동과 환경에 대하여 서로 배우는 거대한 네트워크이며 DB 리파지토리, 즉, 로봇용 클라우드를 구축하여 관련 정보를 찾아 배우게 하는 시스템

기존 로봇 명령을 받은 로봇이 스스로 판단하고 행동



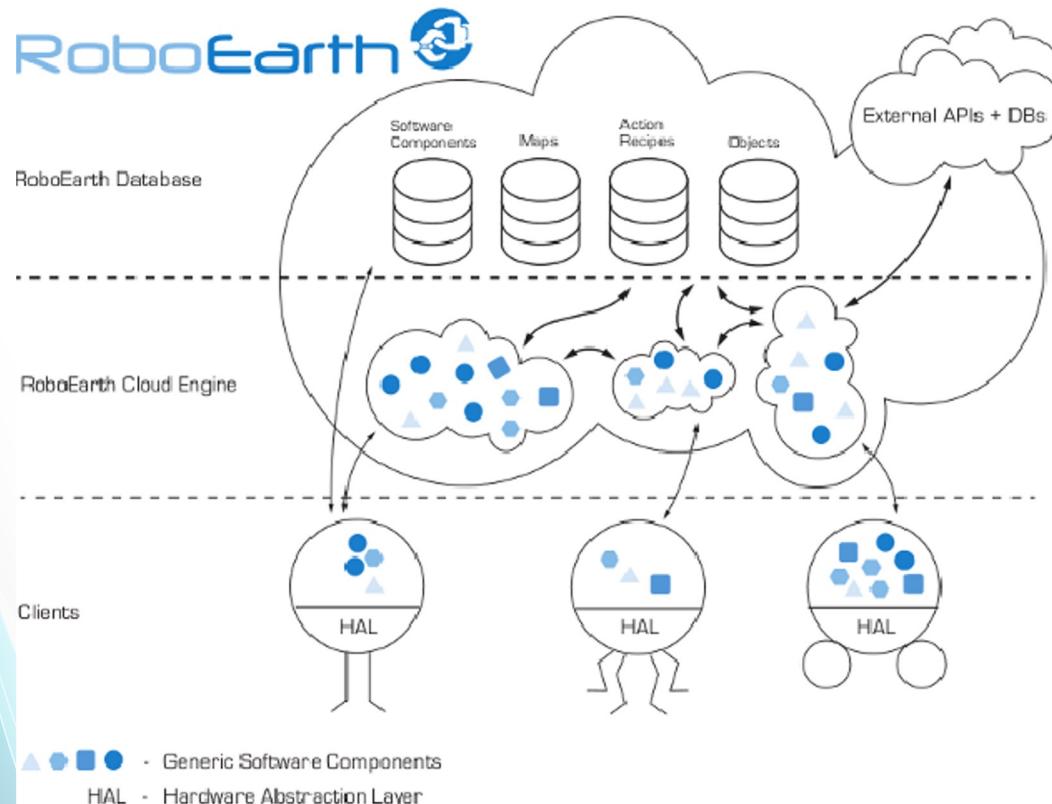
클라우드 로봇 명령을 받은 로봇이 직접 판단하지 않고 외부 서버가 인터넷상의 정보까지 이용해 판단하고 실행



독일 뮌헨 공대 제공

(조선비즈 2011년 8월 기사: http://biz.chosun.com/site/data/html_dir/2011/08/15/2011081501389.html)

RoboEarth ('09): 구조



Big Data, Data Storage

Cloud Computing

Knowledge Sharing

Collective Learning

Collaborative Tasks

***Human Computation
(Crowdsourcing)***

RoboEarth ('09): 개념과 취지 설명 영상



The RoboEarth Cloud Engine



2010-2014 <http://roboearth.ethz.ch>

RoboEarth ('09): 최종 시연 영상



Final Demonstration
January 2014



Universität Stuttgart



Universidad
Zaragoza



Universität Bremen*

RoboEarth ('09): 협업 SLAM 시연 영상



Cloud-based Collaborative 3D Mapping
in Real-Time with Low-Cost Robots
September 2013



More and more AI Services in the Clouds...



Vision

- Object, scene, and activity detection
- Face recognition and identification
- Celebrity and landmark recognition
- Emotion recognition
- Text and handwriting recognition (OCR)
- Video metadata, audio, and keyframe extraction and analysis
- Explicit or offensive content moderation
- Custom image recognition



Speech

- Speech transcription (Speech-to-text)
- Speech Synthesis (Text-to-speech)
- Real-time speech translation
- Speaker identification and verification
- Custom Speech models for transcription and translation
- Custom voice



Language

- Language detection
- Text sentiment analysis
- Key phrase extraction
- Entity recognition
- Spell checking
- Explicit or offensive text content moderation, PII detection
- Text translation
- Customizable text translation
- Contextual language understanding



Knowledge

- Q&A extraction from unstructured text
- Knowledge base creation from collections of Q&As
- Semantic matching for knowledge bases
- Customizable content personalization learning



Search

- Ad-free web, news, image, and video search results
- Trends for video, news
- Image identification, classification and knowledge extraction
- Identification of similar images and products
- Named entity recognition and classification
- Knowledge acquisition for named entities
- Search query autosuggest
- Ad-free custom search engine creation

(출처: <https://medium.com/bucharest-ai/infuse-ai-in-your-apps-with-microsoft-cognitive-services-5ee71dc96dea>)

Smart Speakers ('14 ~)

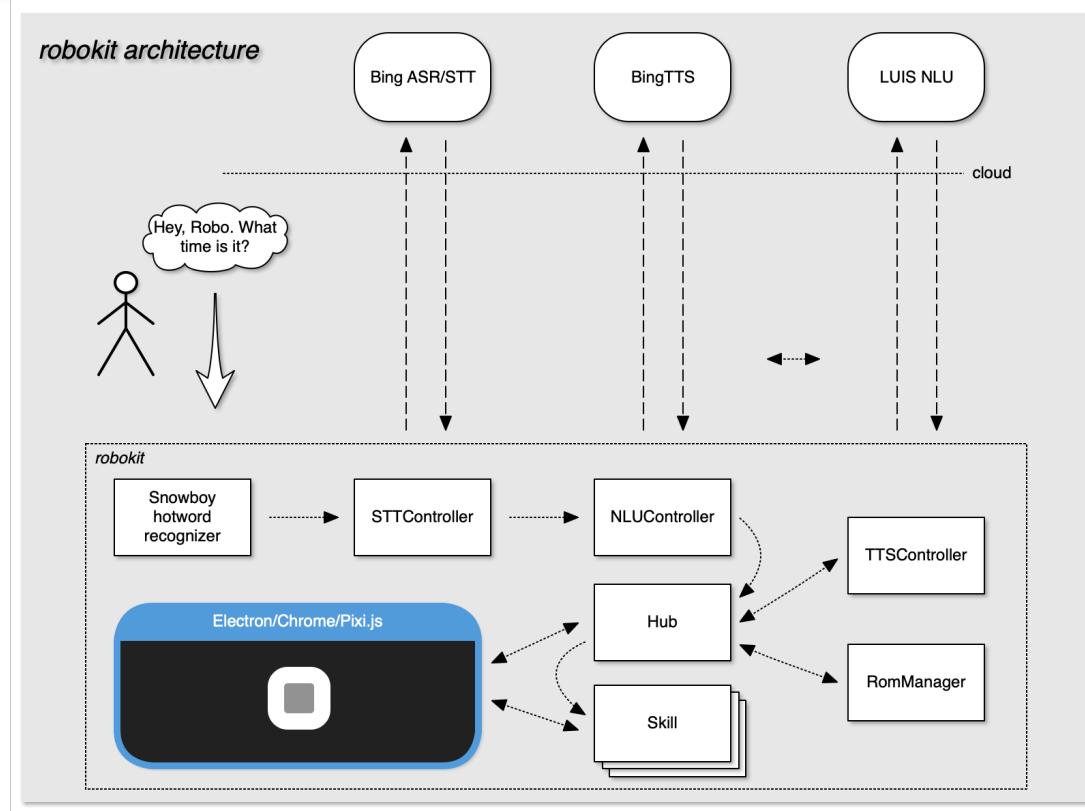


(출처: <https://developer.amazon.com/en-US/docs/alexa/alexa-gadgets-toolkit/understand-alexa-gadgets-toolkit.html>)

Jibo ('14 ~)

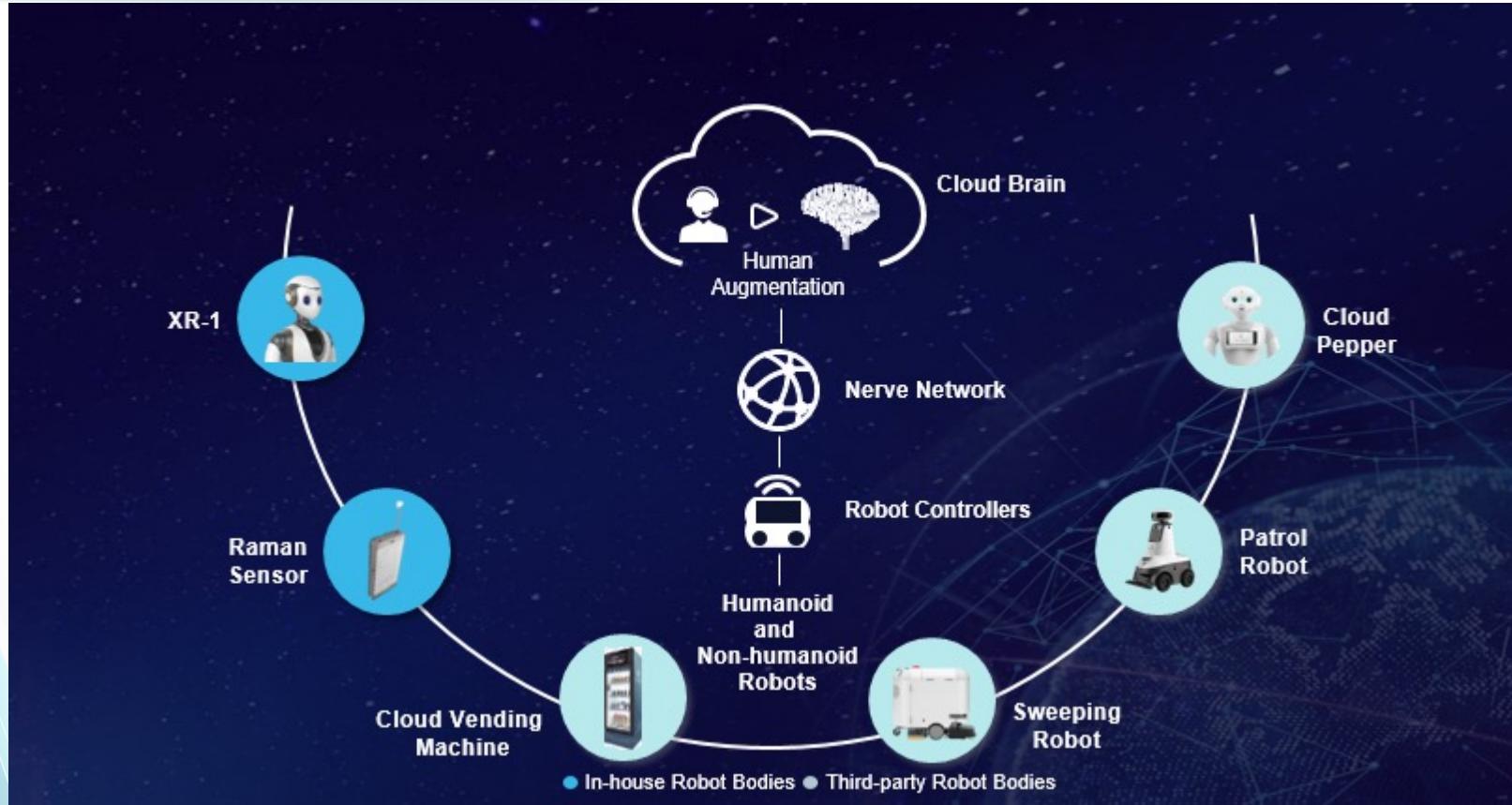


(출처: <https://www.pzdeals.com/products/jibo-the-worlds-first-social-robot-for-the-home>)



(출처: <https://medium.com/@andrew.rapo/make-your-own-conversational-ai-social-robot-with-robokit-a-crude-approximation-of-jibo-10847e9a2661>)

CloudMinds ('15 ~)



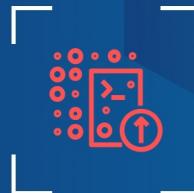
(출처: <https://www.en.cloudminds.com/home-new/cloud-robots/>)

Amazon RoboMaker ('18): 개발, 시험, 배포까지...

ROBOMAKER



Cloud Extensions
for ROS



Development
Environment

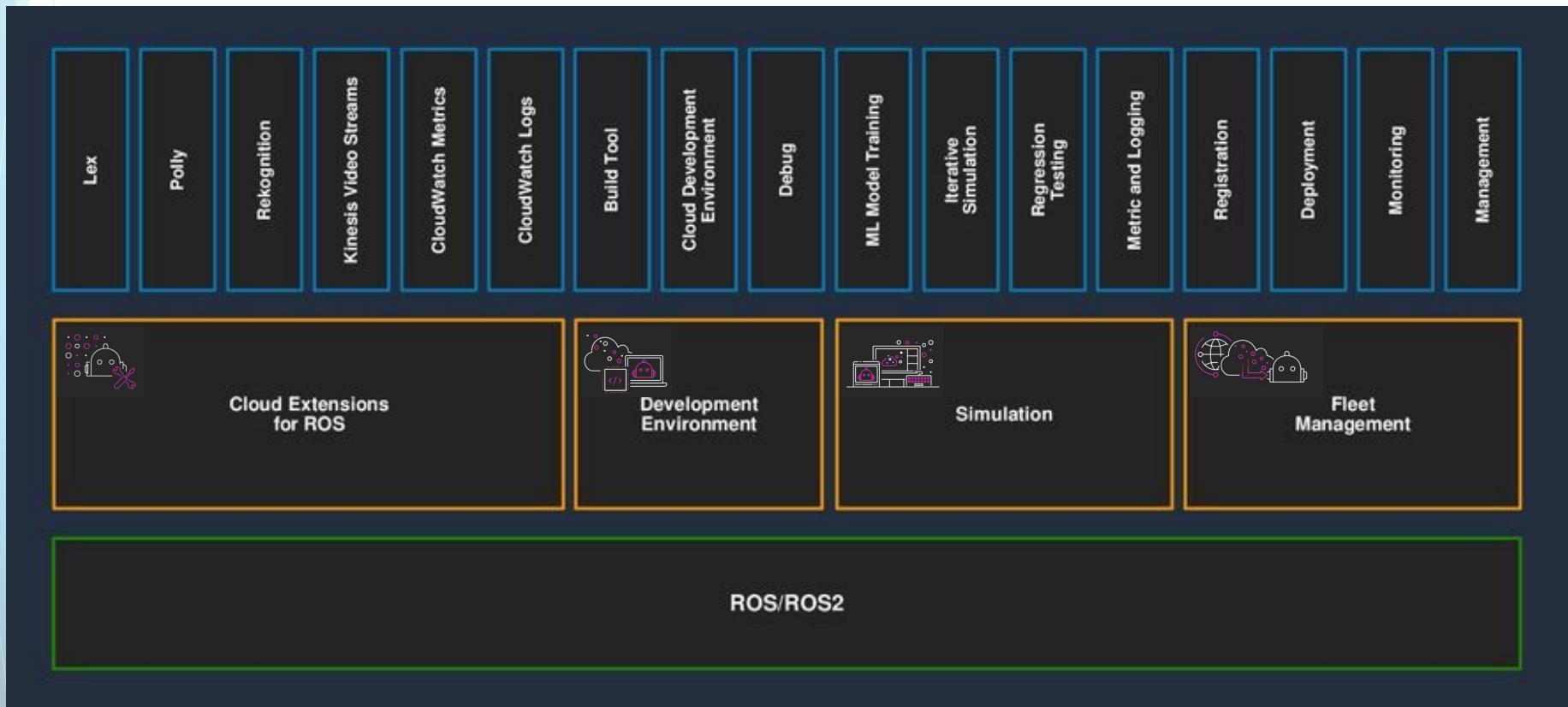


Simulation



Fleet
Management

Amazon RoboMaker ('18): 개발, 시험, 배포까지...

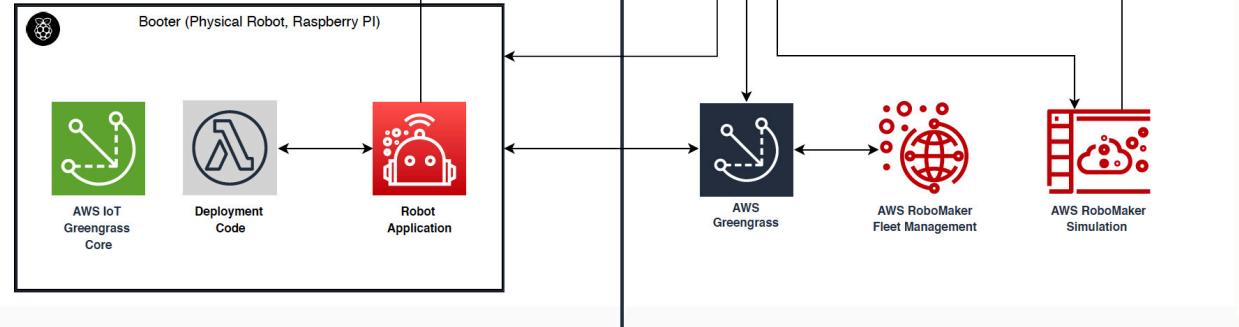


Amazon RoboMaker

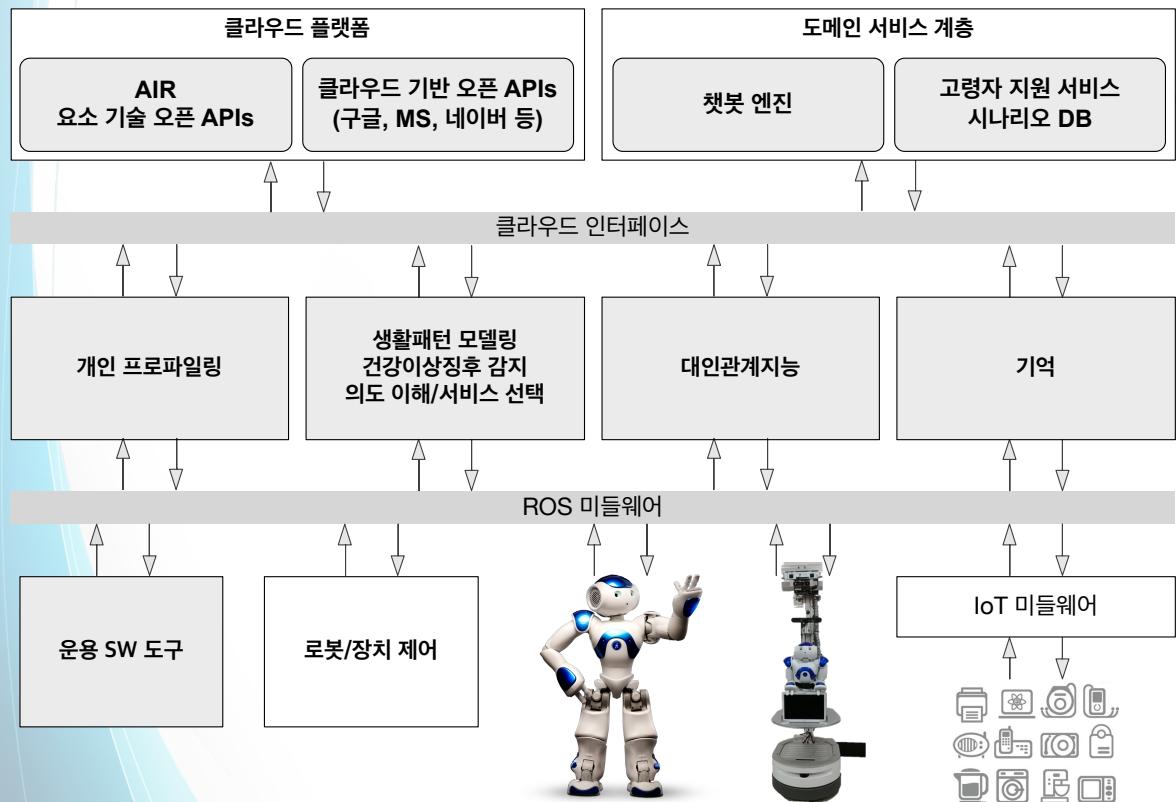
클라우드 환경에서 로봇 인식/판단/표현 기능과
응용을 개발하고 시뮬레이터를 통하여 시험



다수 로봇에 OTA 기반 응용 배포 사용과 관제



ETRI AIR ('17~): Artificial Intelligence for Robots



평가 시나리오



아파트 테스트베드 시스템 구축 운용

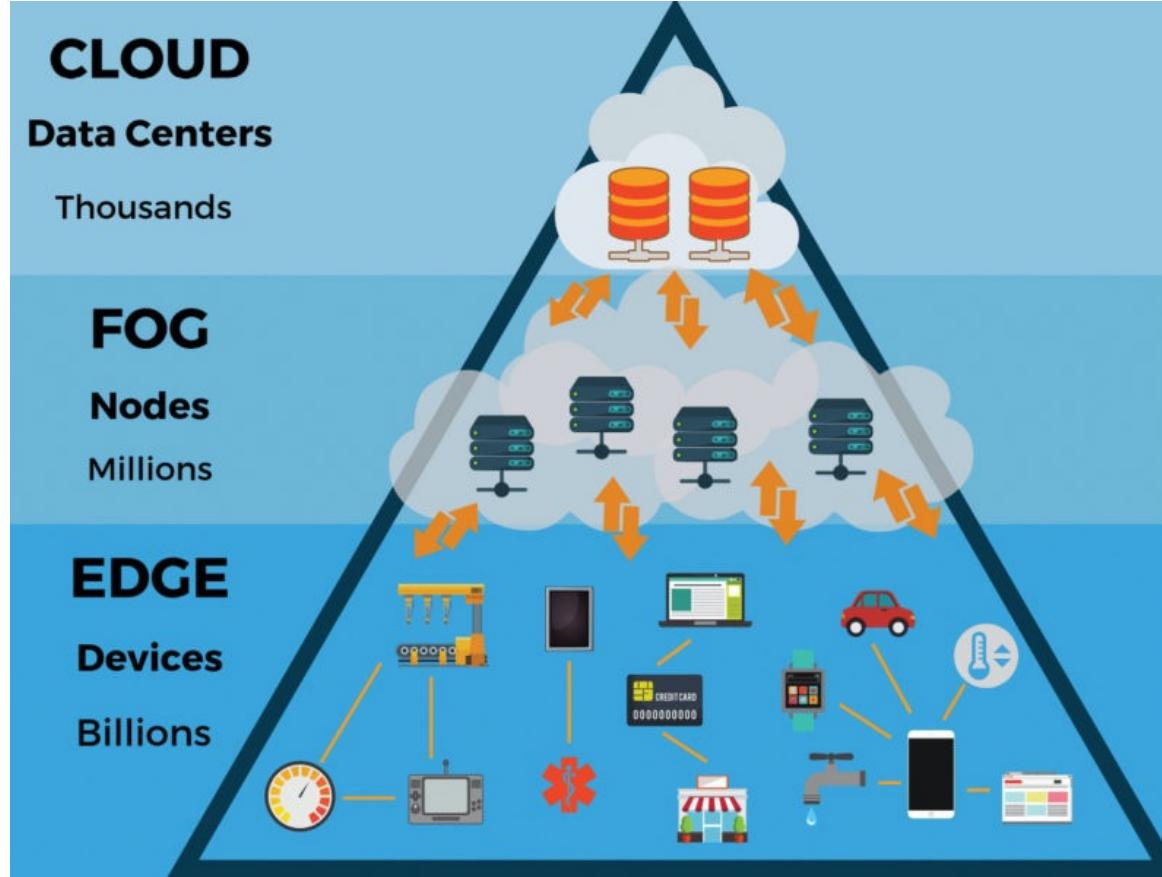
(출처: "고령사회에 대응하기 위한 실환경 휴먼케어로봇 기술 개발" 과제 '17~'21 ETRI)

클라우드 로봇의 주요 이슈

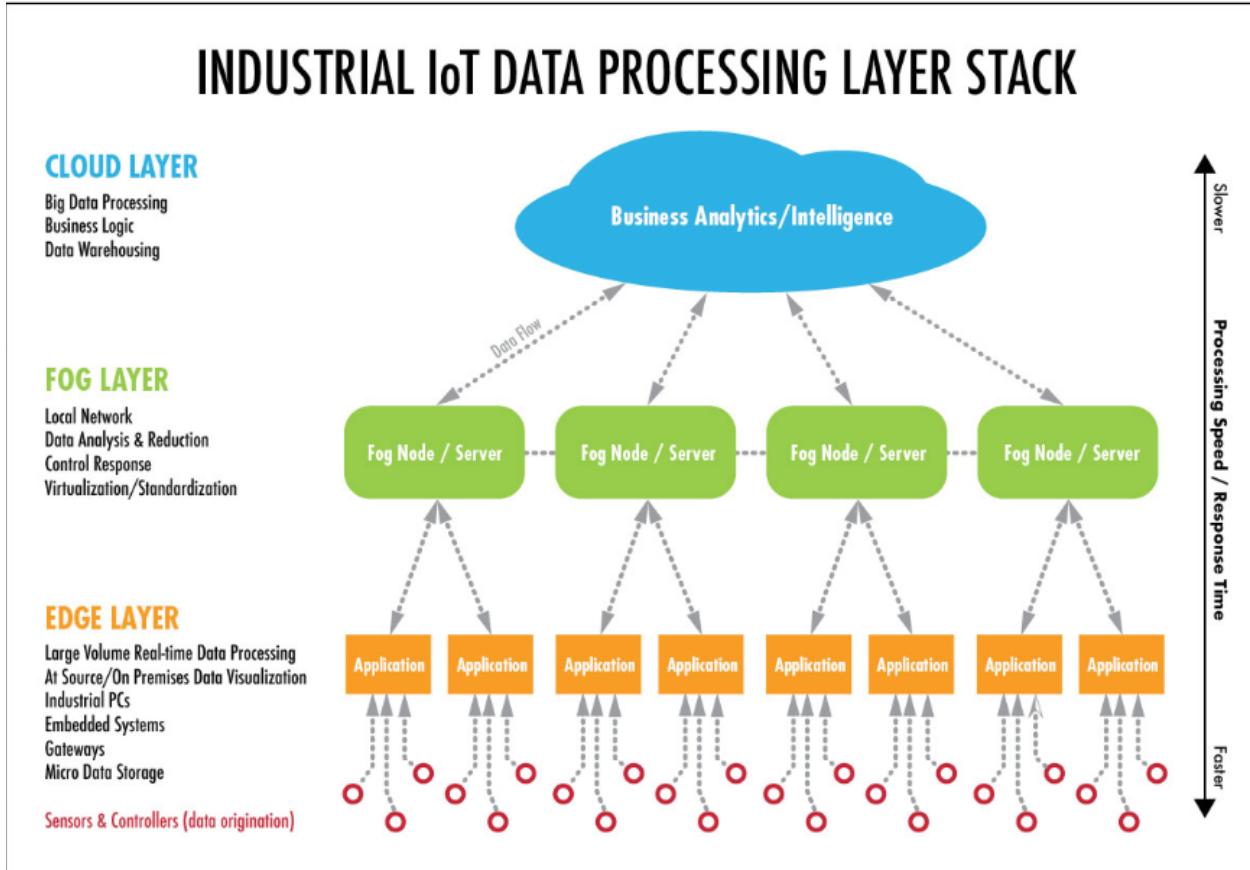
클라우드 로봇의 문제

- Latency
- Connectivity
- Privacy
- Security

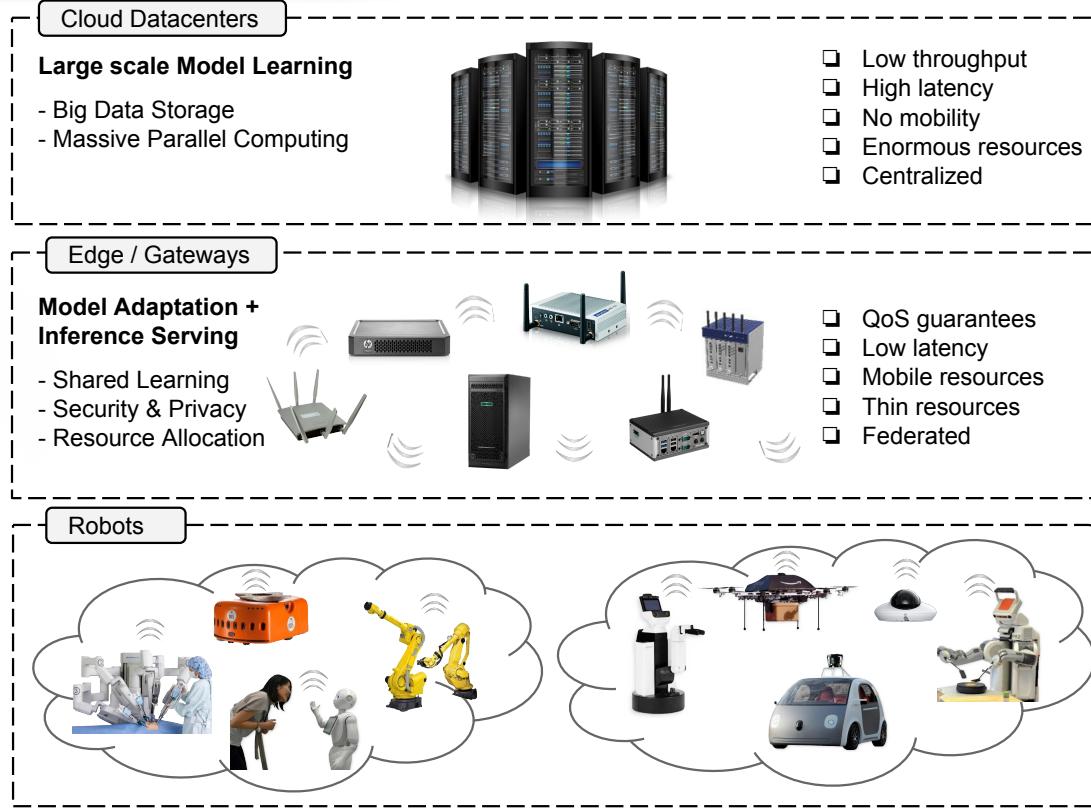
Fog Computing



Fog Computing

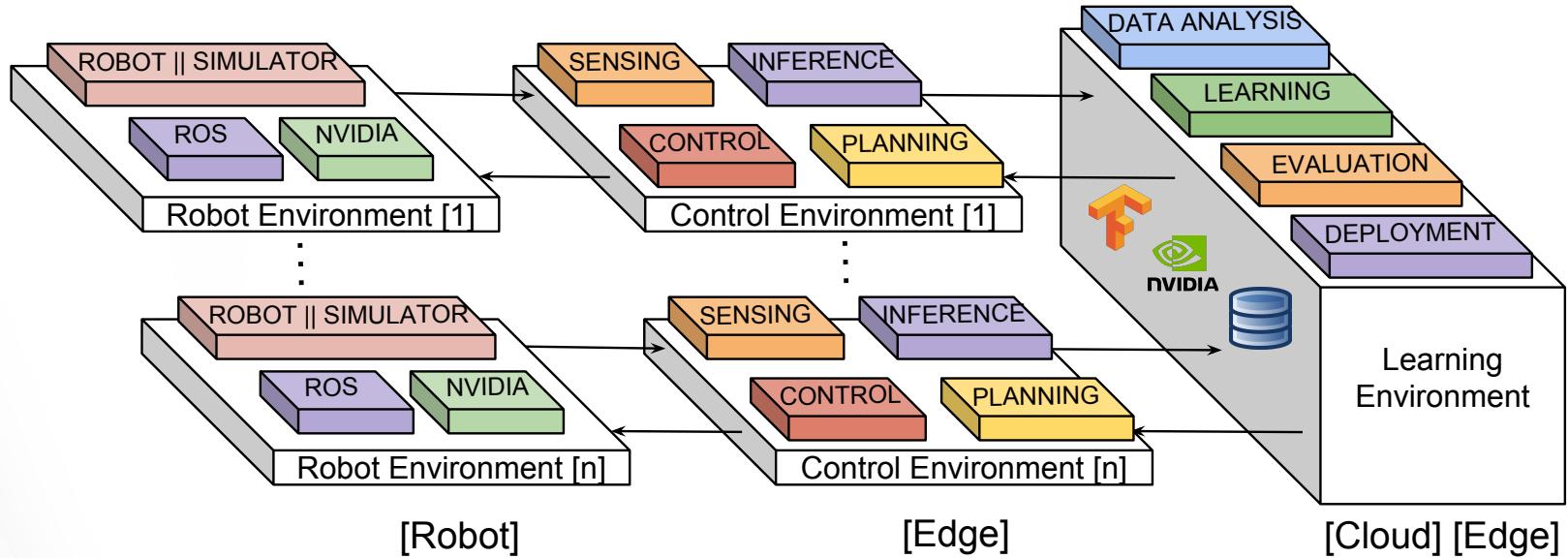


Fog Robotics



Tanwani, Ajay Kumar, Nitesh Mor, John Kubiatowicz, Joseph E. Gonzalez, and Ken Goldberg. "A fog robotics approach to deep robot learning: Application to object recognition and grasp planning in surface decluttering." In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4559-4566. IEEE, 2019.

Fog Robotics



Tanwani, Ajay Kumar, Nitesh Mor, John Kubiatowicz, Joseph E. Gonzalez, and Ken Goldberg. "A fog robotics approach to deep robot learning: Application to object recognition and grasp planning in surface decluttering." In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4559-4566. IEEE, 2019.

Fog Robotics

TABLE I: Comparative experiments for learning deep object recognition for simulation to reality transfer over Cloud, Edge and Fog. Metrics include mean Average Precision (mAP) on real images, classification accuracy on synthetic test images sim_eval, real test images real_eval and both synthetic and real test images mix_eval. Domain invariant object recognition with shared feature representation network parameters DIOR_dann model gives better performance in both simulation and real domain using Fog Robotics.

Training Set	mAP	sim_eval	real_eval	mix_eval
Cloud				
Sim	0.13	97.97	24.16	55.5
Edge				
Real	0.62	24.64	88.1	64.92
Fog				
Sim + Real	0.33	90.40	54.12	69.97
DIOR_dann	0.61	96.92	86.33	95.21
DIOR_adda	0.61	30.87	90.64	67.82

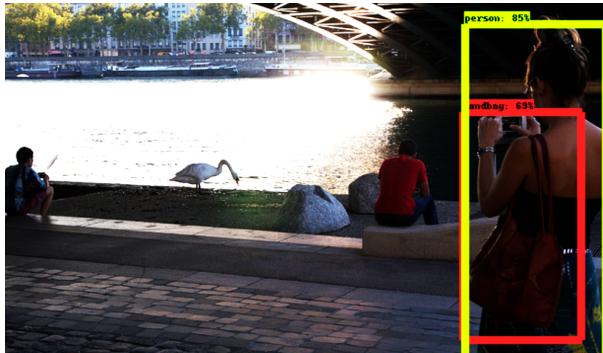
TABLE II: Computation time for inference $t^{(\text{inf})}$ vs round trip communication time $t^{(\text{rtt})}$ (in milliseconds) for inference over Edge (with and without GPU) and Cloud with EC2 (West) EC2 (East) instances. Results are averaged across 200 real images. Communication time dominates the computation time and increases as the distance to the server increases.

Location	$t^{(\text{inf})}$	$t^{(\text{rtt})}$
Object Recognition		
EC2(East)	31.93 ± 1.53	437.63 ± 100.02
EC2(West)	31.12 ± 1.28	181.61 ± 22.71
Edge(CPU)	52.34 ± 4.18	149.32 ± 21.04
Edge(GPU)	33.27 ± 3.09	119.40 ± 12.06
Grasp Planning		
EC2(East)	1906.59 ± 224.19	4418.34 ± 1040.59
EC2(West)	1880.28 ± 207.46	2197.76 ± 199.44
Edge(CPU)	3590.71 ± 327.57	3710.74 ± 214.08
Edge(GPU)	1753.65 ± 201.38	1873.16 ± 211.57

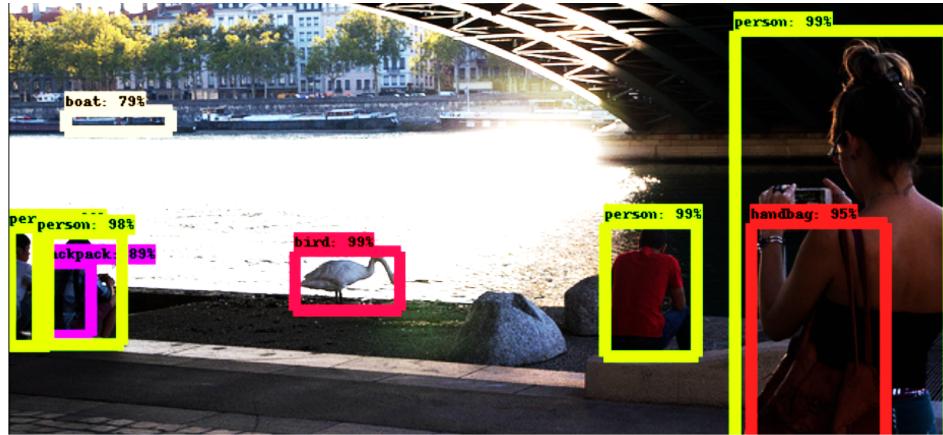
Tanwani, Ajay Kumar, Nitesh Mor, John Kubiatowicz, Joseph E. Gonzalez, and Ken Goldberg. "A fog robotics approach to deep robot learning: Application to object recognition and grasp planning in surface decluttering." In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 4559-4566. IEEE, 2019.

Robot-Cloud Offloading (1/3)

- 정확도 vs 속도 vs 전원...



Robot Model

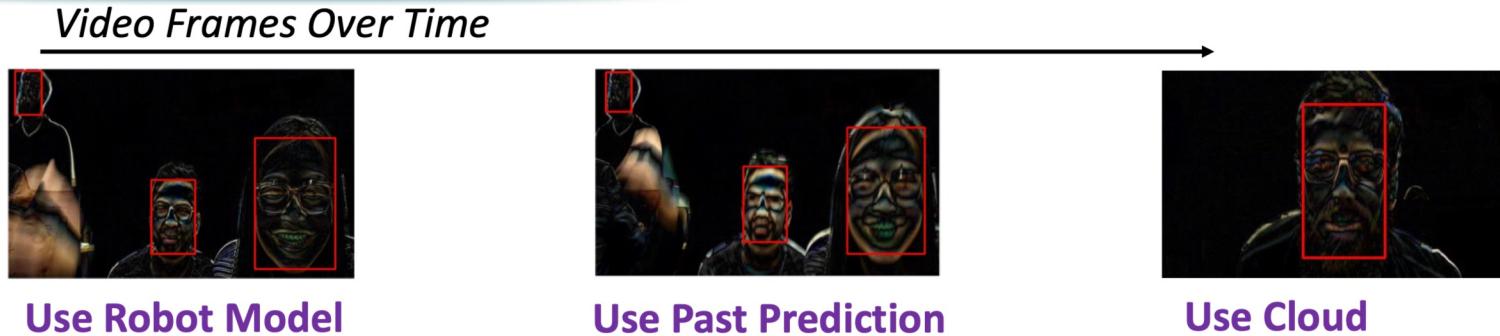


Cloud Model

Platform	Cost	Accuracy (mAP)	Inf. Time (ms)	DNN	Size (MB)
Edge TPU	\$75	0.169	6.89	MobileNet 2	5.5
ARM CPU	\$57	0.170	285.56	MobileNet 2	4.7
Cloud GPU	\$879	0.265	56.84	MobileNet 2	20
Cloud GPU	\$879	0.426	447.5	Faster R-CNN	183

10

Robot-Cloud Offloading (2/3)



State: sensory input, remaining query budget, age of past prediction, ...

Action: which model to choose?

Key Constraint: Limited Queries to the Cloud

- use past prediction**
- use robot model**
- query cloud model**

- cheap, potentially stale prediction
- more expensive, potentially low accuracy
- most expensive, highest accuracy, limited budget

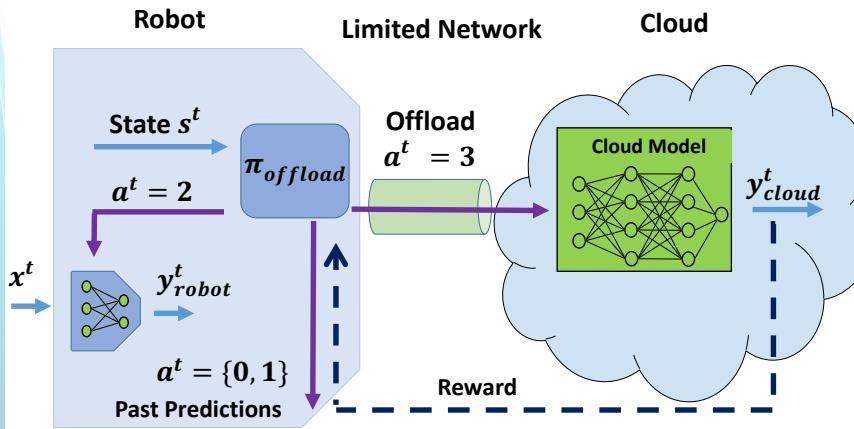
Reward: balance prediction accuracy with cost of querying model

Chinchali, Sandeep, Apoorva Sharma, James Harrison, Amine Elhafsi, Daniel Kang, Evgenya Pergament, Eyal Cidon, Sachin Katti, and Marco Pavone.
"Network offloading policies for cloud robotics: a learning-based approach." *arXiv preprint arXiv:1902.05703* (2019).

Robot-Cloud Offloading (3/3)

RL-based Offloading Policy Learning

$$a_{\text{offload}}^t = \begin{cases} 0, & \text{use past robot prediction } \hat{y}^t = f_{\text{robot}}(x^{\tau_{\text{robot}}}) \\ 1, & \text{use past cloud prediction } \hat{y}^t = f_{\text{cloud}}(x^{\tau_{\text{cloud}}}) \\ 2, & \text{use current robot prediction } \hat{y}^t = f_{\text{robot}}(x^t) \\ 3, & \text{use current cloud prediction } \hat{y}^t = f_{\text{cloud}}(x^t) \end{cases}$$



Past predictions, local model, or cloud

4 succinct actions: easy to run on hardware

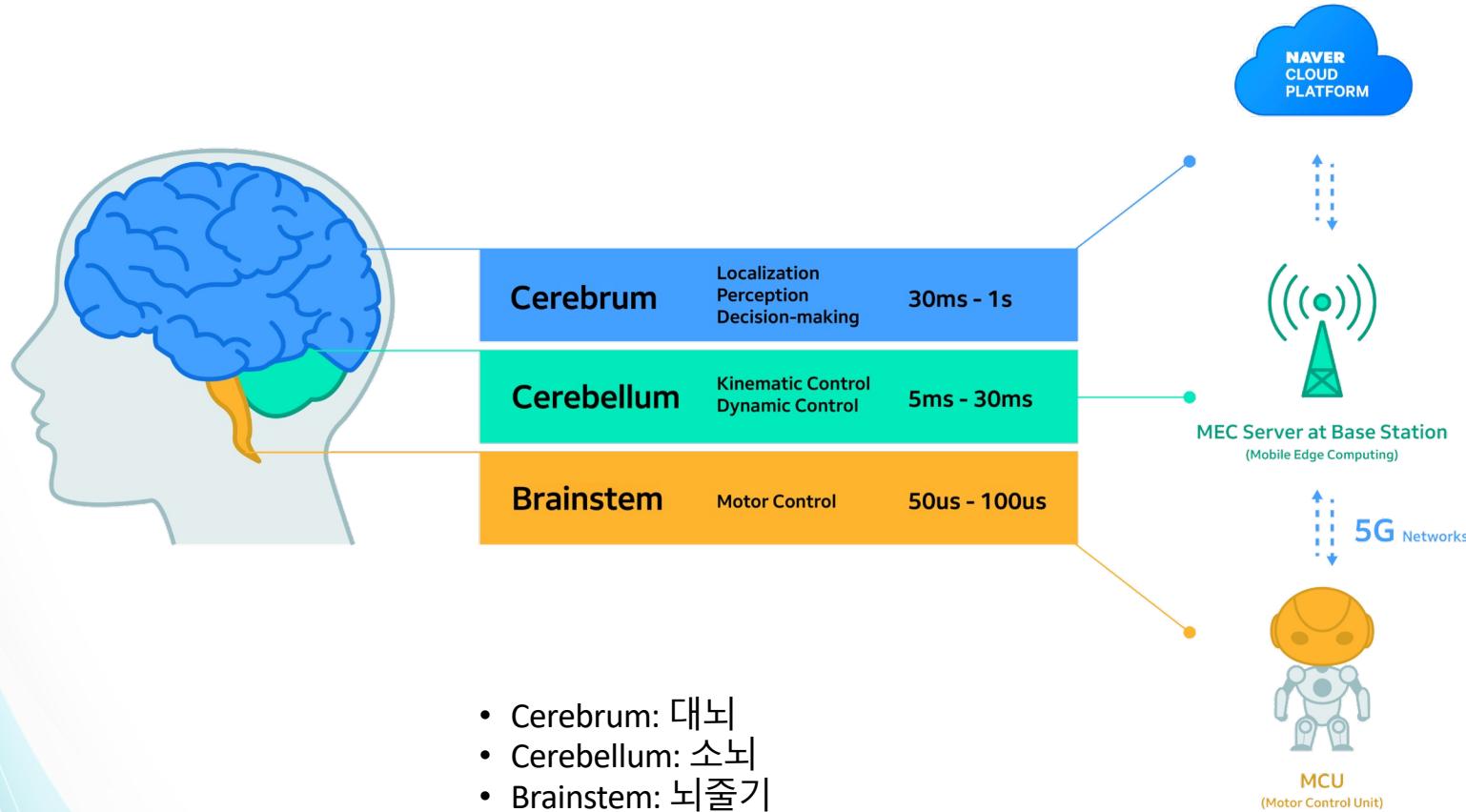
No network? Use local model (safety)

Chinchali, Sandeep, Apoorva Sharma, James Harrison, Amine Elhafsi, Daniel Kang, Evgenya Pergament, Eyal Cidon, Sachin Katti, and Marco Pavone.
"Network offloading policies for cloud robotics: a learning-based approach." *arXiv preprint arXiv:1902.05703* (2019).

5G 이동통신



네이버 Brainless Robot

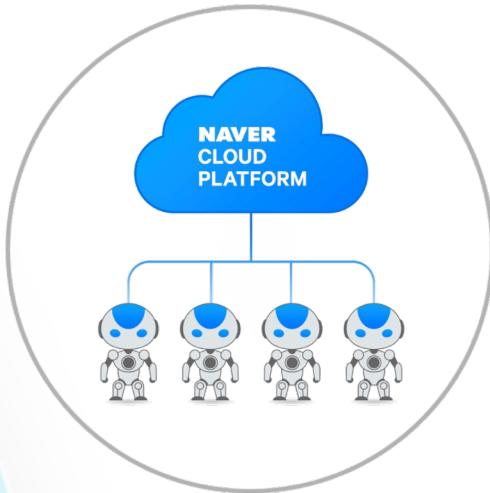


네이버 Brainless Robot

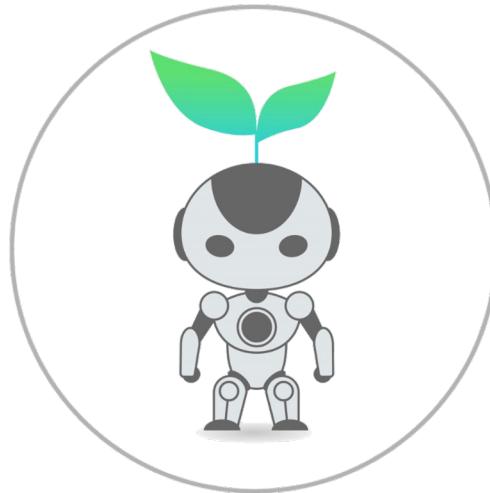
99.9"

(출처: <https://www.naverlabs.com/en/5gbrainlessrobot>)

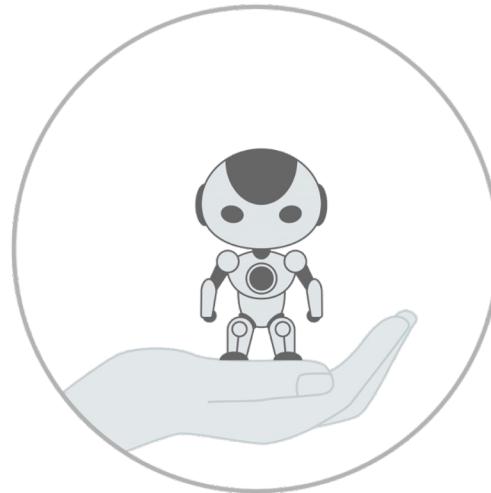
네이버 Brainless Robot



Multiple robots are
controlled simultaneously
(cloud base)

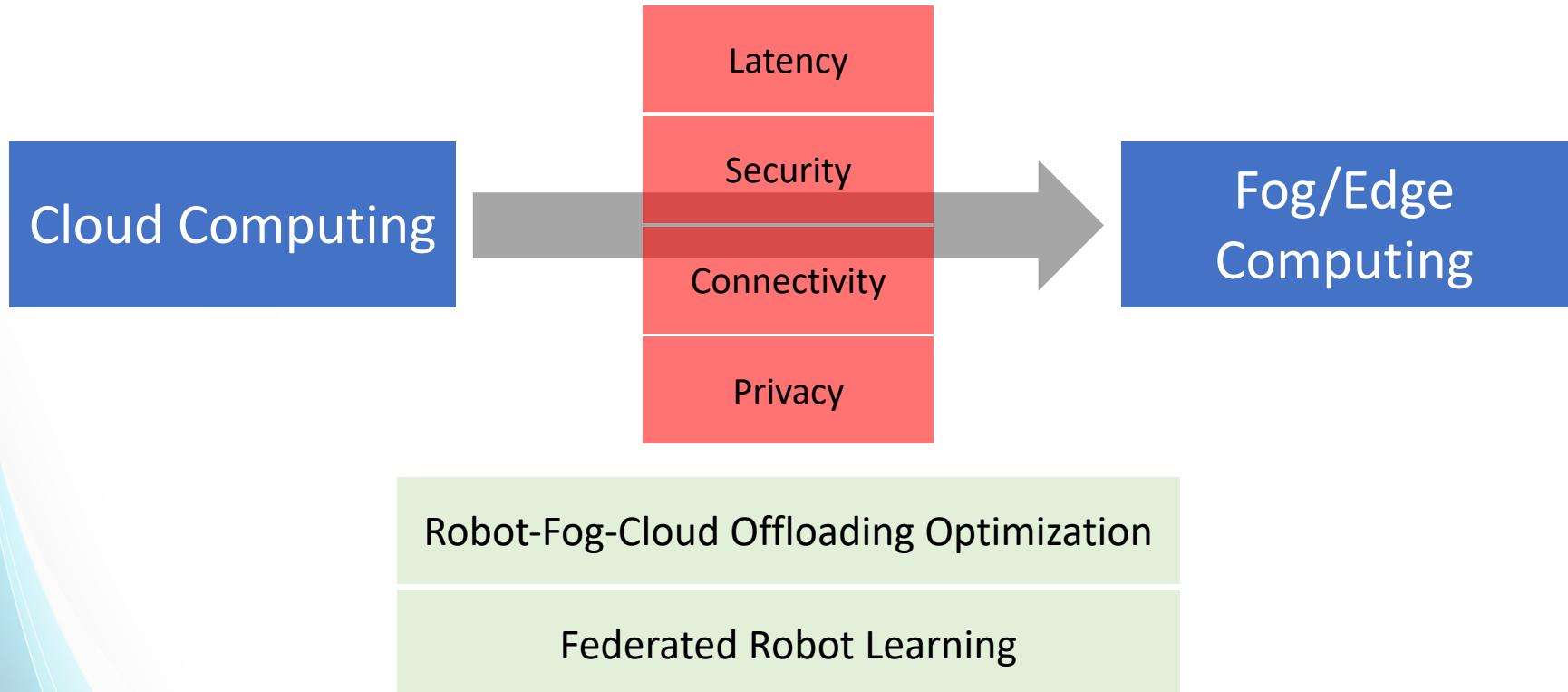


High processing power
separation to reduce
energy consumption



High performance and
high precision control
regardless of the robot's size

Summary

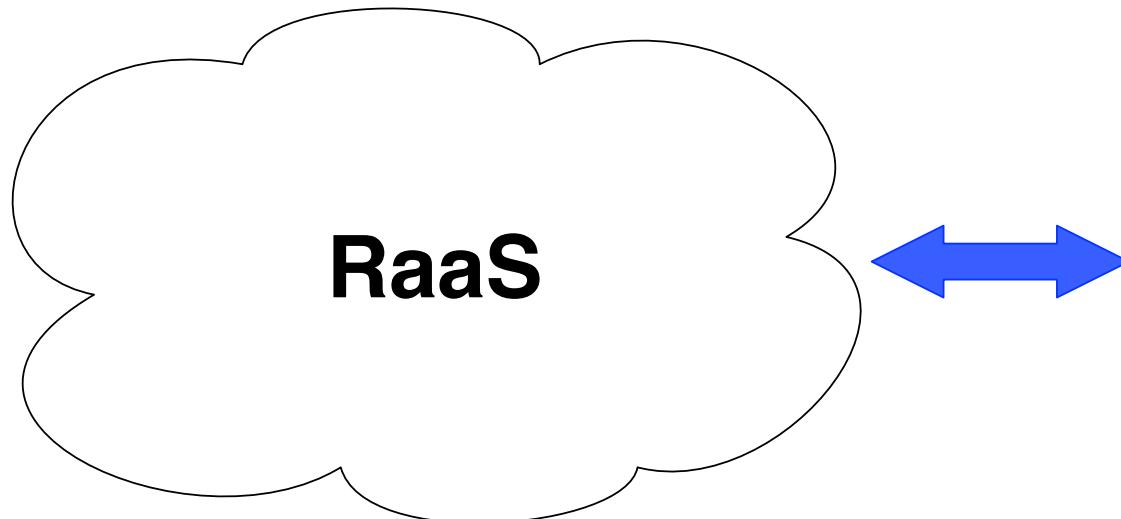


Robots-as-a-Service(RaaS) 소개

슬라이드 출처: <https://www.slideserve.com/adem/raas-robotics-as-a-service>

RaaS: Robotics as a Service

- A Service Model for Cloud Robotics

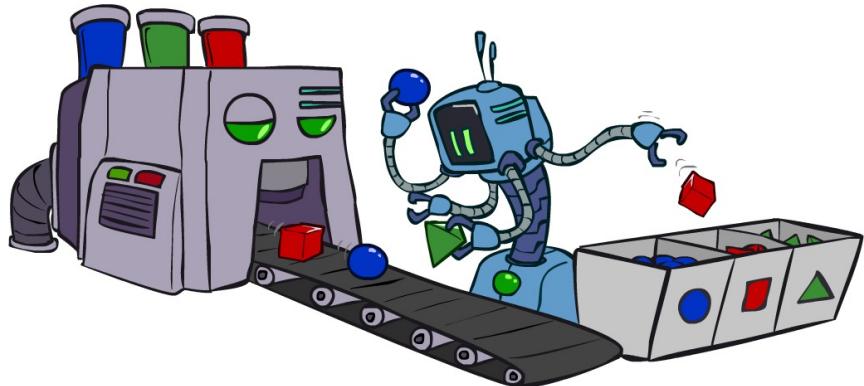


Arjun Singh, Ben Kehoe, Ken Goldberg, Pieter Abbeel

University of California, Berkeley

Example Use Case

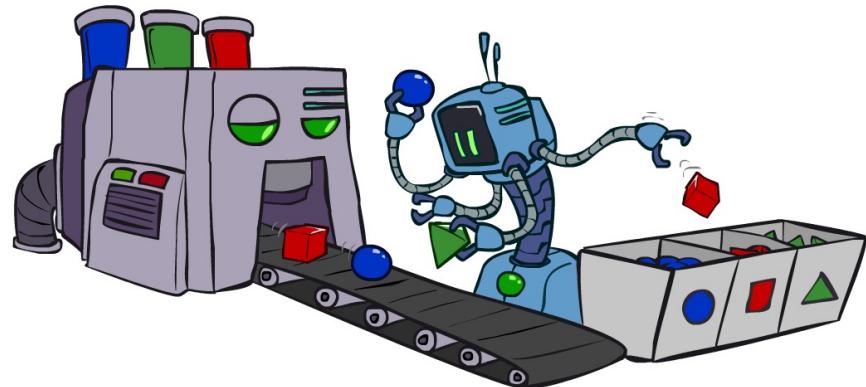
- Objects coming down conveyer belt
- Identify each object, pick it up, and place into correct bin
 - Low-level control
 - Object recognition
 - Grasp planning
 - Motion planning



Drawing: Ketrina Yim

How to Solve It with ROS

- Acquire computing hardware
- Install ROS and relevant packages
- Execute and maintain software and computing hardware



Drawing: Ketrina Yim

Pros and Cons of ROS

- Huge positive impact on robotics (especially sharing)
 - Core great for messaging, low-level control – components that make a robot *operating system*
 - Asking it to do too much
 - All code for everything robotics related

Packages required for barebones ROS installation

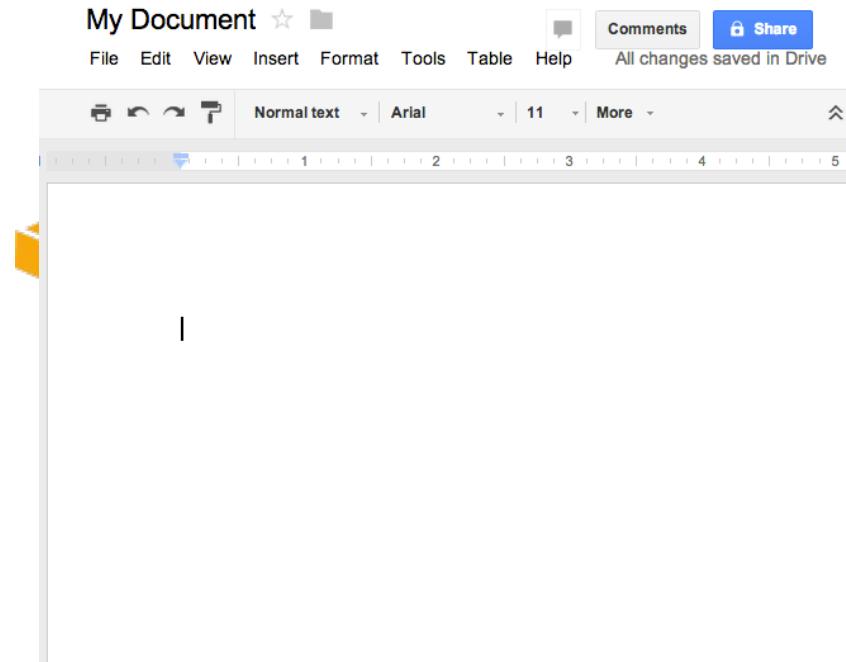
Packages required for full ROS installation

Pros and Cons of ROS

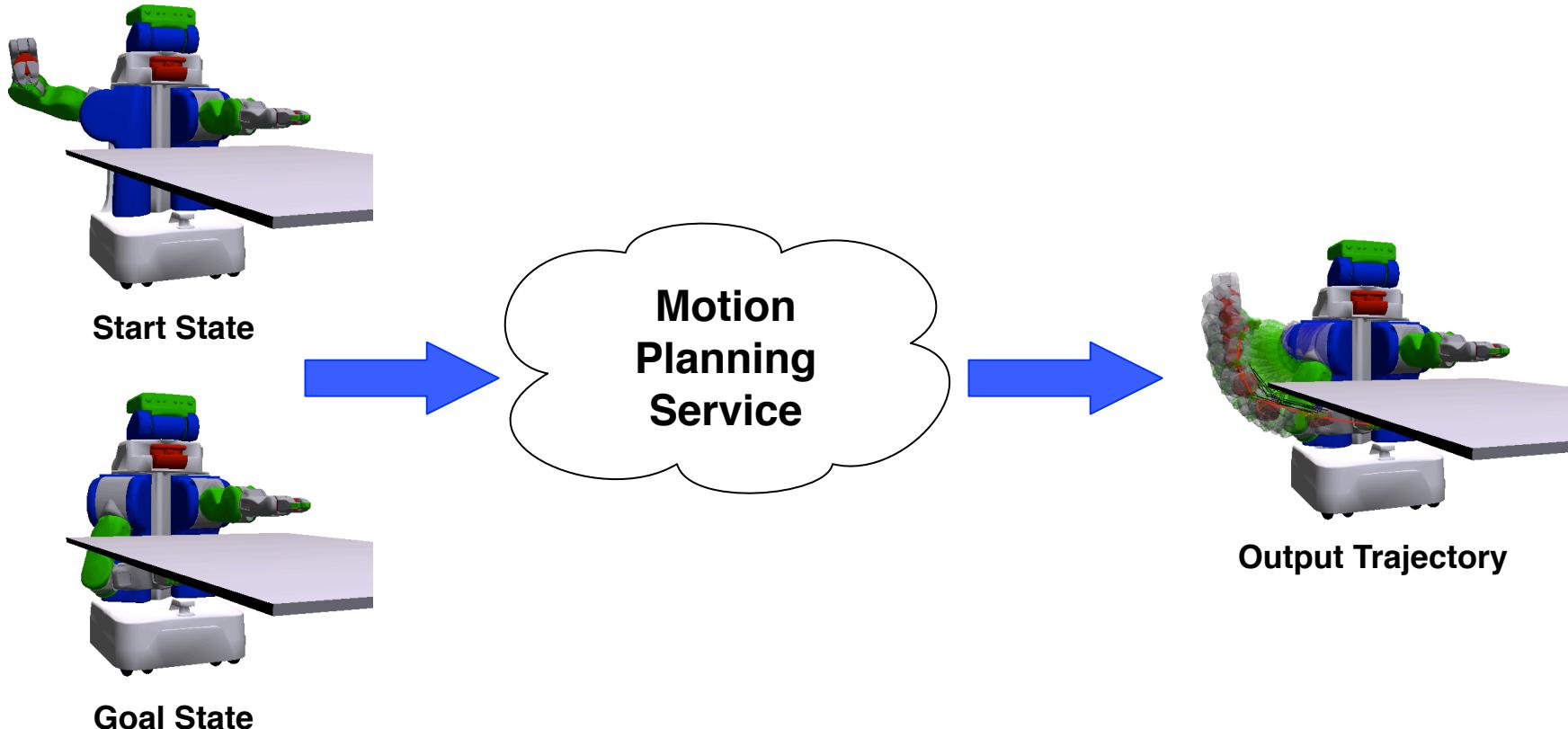
- Intimidating for non-roboticists
- Nontrivial to set up secure distributed networking with ROS -- need to understand VPNs, have control over network environment, etc.
- Dependencies can turn into a nightmare (especially with multiple ROS versions)

Computing Environment has Changed

- ROS design started in 2006 – a lot has changed!
- Cloud computing: Easy access to vast numbers of machines
- Software engineering: Service-oriented architectures and Software as a Service
 - E.g. Google Docs vs. Microsoft Office



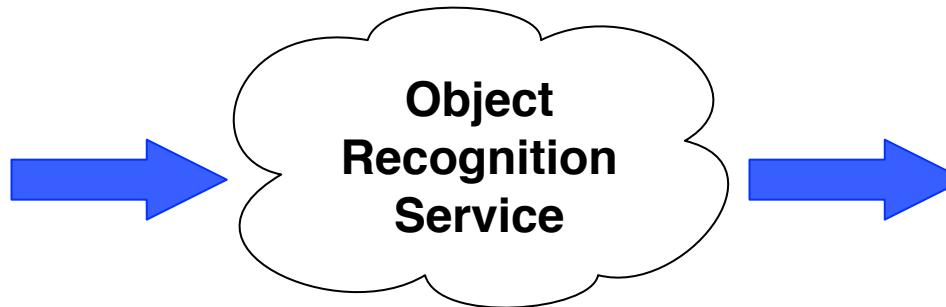
Example Services – Motion Planning



Example Services – Object Recognition



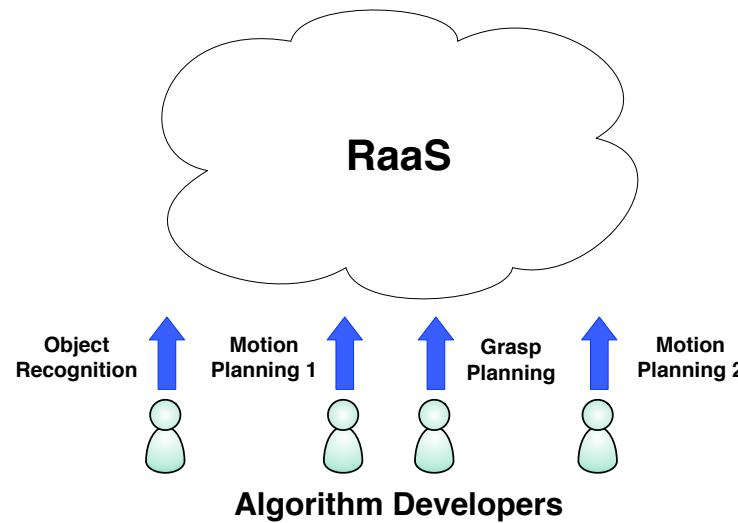
Input Scene



Detected Objects
and Poses

Our Idea: Robotics as a Service (RaaS)

- Publish algorithms as services
- Consume services on your robot
- Collaborative data collection



m Users

Related Work

Cloud Robotics

Rapyuta (www.rapyuta.org/)	Robotics Platform as a Service
Rocon (www.robotconcert.org/)	Higher-level orchestration
Goldberg et al. 1995	Networked robotics
Arumugam et al. 2010	Cloud computing for robots
Ciocarlie et al. 2010	Big data for robotics
Kuffner 2010	Cloud-enabled robots
Remy and Blake 2011	Service-oriented robotics
Blake et al. 2011	Service-oriented robotics
Waibel et al. 2011	Shared knowledge

Open-Source Tools

Movelt! (Sucan and Chitta)	Motion planning
GraspIt! (Ciocarlie and Miller)	Grasp planning
OpenCV (Bradski)	Computer vision
OpenRAVE (Diankov)	Motion planning
Ladon (www.ladonize.org)	Web service framework

Commercial Products

Mashape	Index of service APIs
PiCloud	Move computation into cloud

Why RaaS?

Limited Resources

Code Sharing

Encapsulation

Parallelism

Common Interfaces

Why RaaS?

Limited Resources

Code Sharing

Encapsulation

Parallelism

Common Interfaces

- Some robotic platforms have limited onboard computation (e.g. Baxter)
 - Can't run sophisticated robotics algorithms
 - Need offboard computation – use the cloud or buy computers
- Services give a straightforward way to move computation to the cloud



Photo: David Yellen for *IEEE Spectrum*

Why RaaS?

Limited Resources

Code Sharing

Encapsulation

Parallelism

Common Interfaces

- Spend months working on your algorithm, finally finish it, and you want to share it
 - Need to figure out what all the dependencies are
 - Need to document how to install everything
 - Need to document how to use your code
 - Don't have time for any of these
- Packaged service: only worry about API users interact with

Why RaaS?

Limited Resources

Code Sharing

Encapsulation

Parallelism

Common Interfaces

- Robotics requires integrating many different components

- Vision, NLP, control, messaging, planning, grasping, etc. – each alone is complex
- Huge number of dependencies – something is bound to conflict

- Services force encapsulation of components

- Use object detection and planning systems without worrying about conflicts

Object Detection:
Dependency A 1.0
Dependency B 2.1
Dependency C 2.5

Motion Planning:
Dependency B 2.1
Dependency C 3.2



Object Detection:
Dependency A 1.0
Dependency B 2.1
Dependency C 2.5

Motion Planning:
Dependency B 2.1
Dependency C 3.2

Why RaaS?

Limited Resources

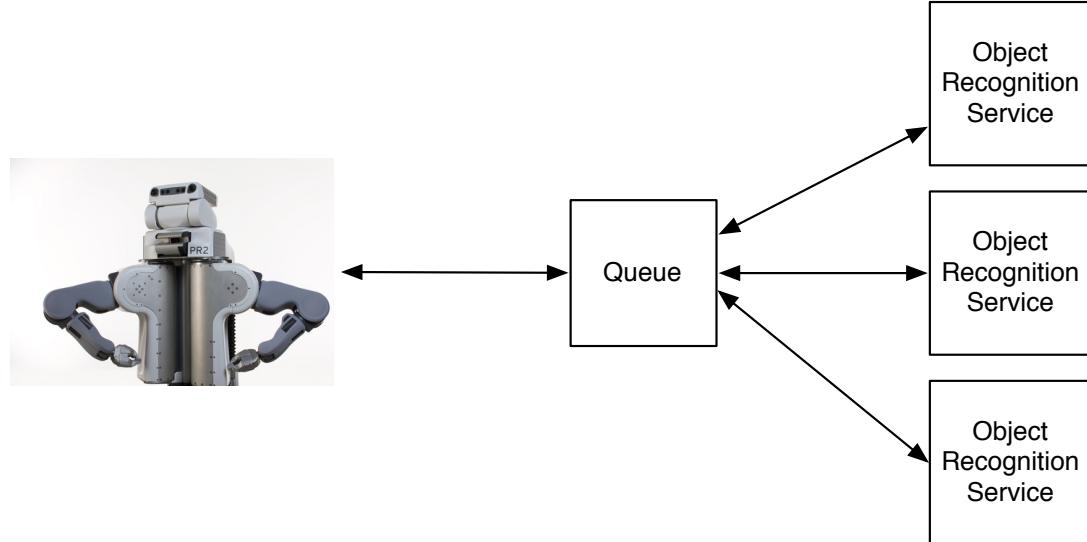
Code Sharing

Encapsulation

Parallelism

Common Interfaces

- Service: natural interface for parallelizing computation
- Insulates user from managing parallelism
- Automatically run multiple instances of a service on multiple machines



Why RaaS?

Limited Resources

Code Sharing

Encapsulation

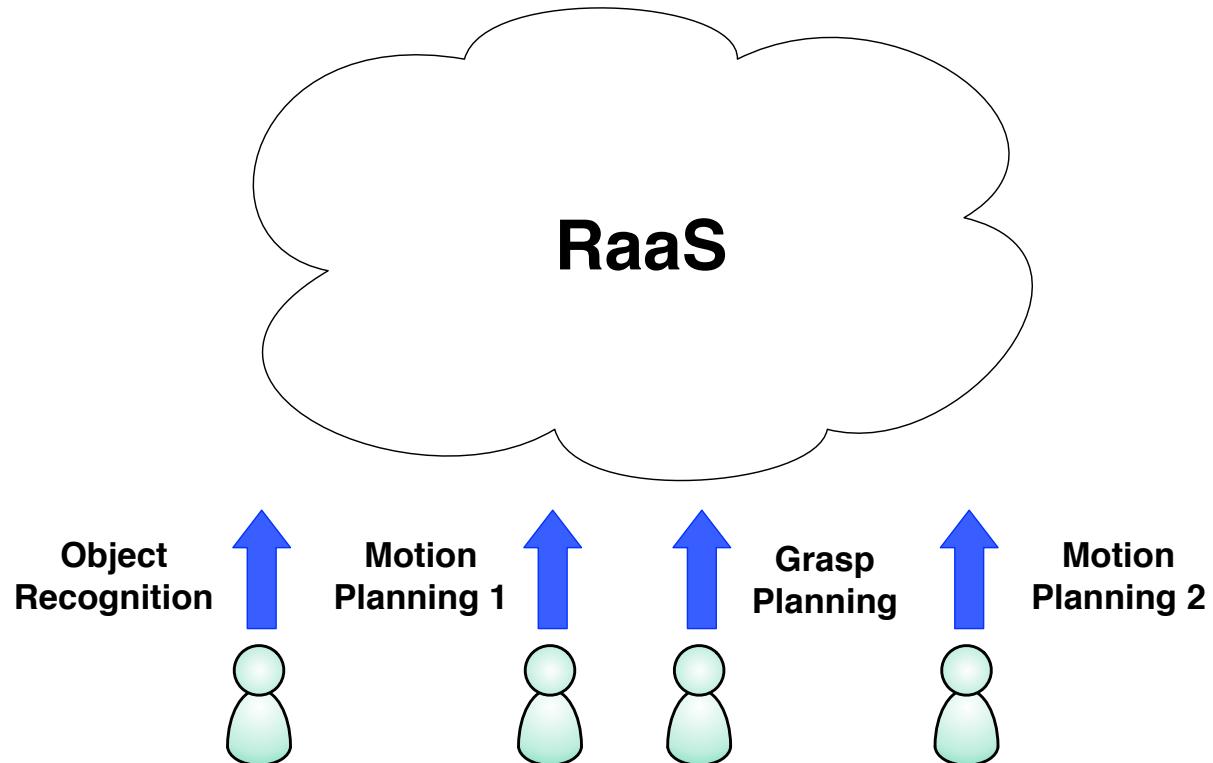
Parallelism

Common Interfaces

- Easy benchmarking and comparison
- Use common interfaces when defining services
 - E.g. object recognition systems
 - Input: Image
 - Output: List of object identities
 - Can't expect researchers to implement interfaces at a library level
 - Swap out services

RaaS Workflow

RaaS Workflow – Algorithm Developers



RaaS Workflow – Algorithm Developers

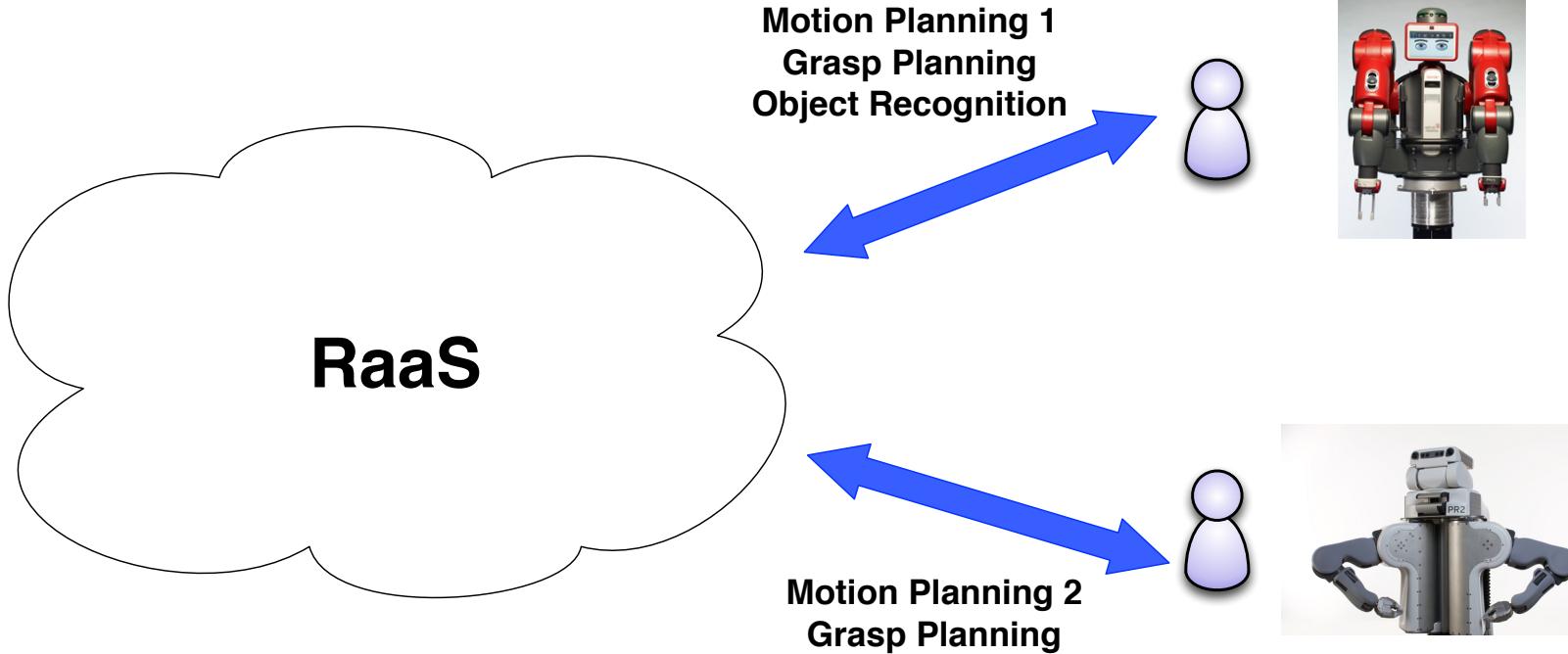
- Write your usual code
- Wrap with service code
- Create Amazon Machine Image (AMI)
- Publish your service



RaaS Workflow – Algorithm Developers

- Make creating a web service as simple as possible
- Ignore HTTP, serialization, encoding, webservers, etc.
- Rich set of types available for service methods
 - Strings, floats, integers, binary blobs, raw files, timestamps, durations, poses, transformations, vectors, matrices, images, point clouds
 - Create your own
- Can serve ROS nodes as web services

RaaS Workflow – Algorithm Users



RaaS Workflow – Algorithm Users

- Choose services
- Launch machines in the cloud
- Connect to machine and use services
- Detect objects in images, plan motions, etc.

Service Definition:

Name	Author	AMI ID
<code>class ObjectDetectionService(Service):</code>		0e8
<code>@method_input(Image, 'image')</code>		18f
<code>@method_output([String], 'detected_objects')</code>		3a
<code>def detect_objects(self, **inputs):</code>		Id4
<code> image = inputs['image']</code>		
<code> detected_objects = my_detector.detect_objects(image)</code>		
<code> return detected_objects</code>		

Client Code:

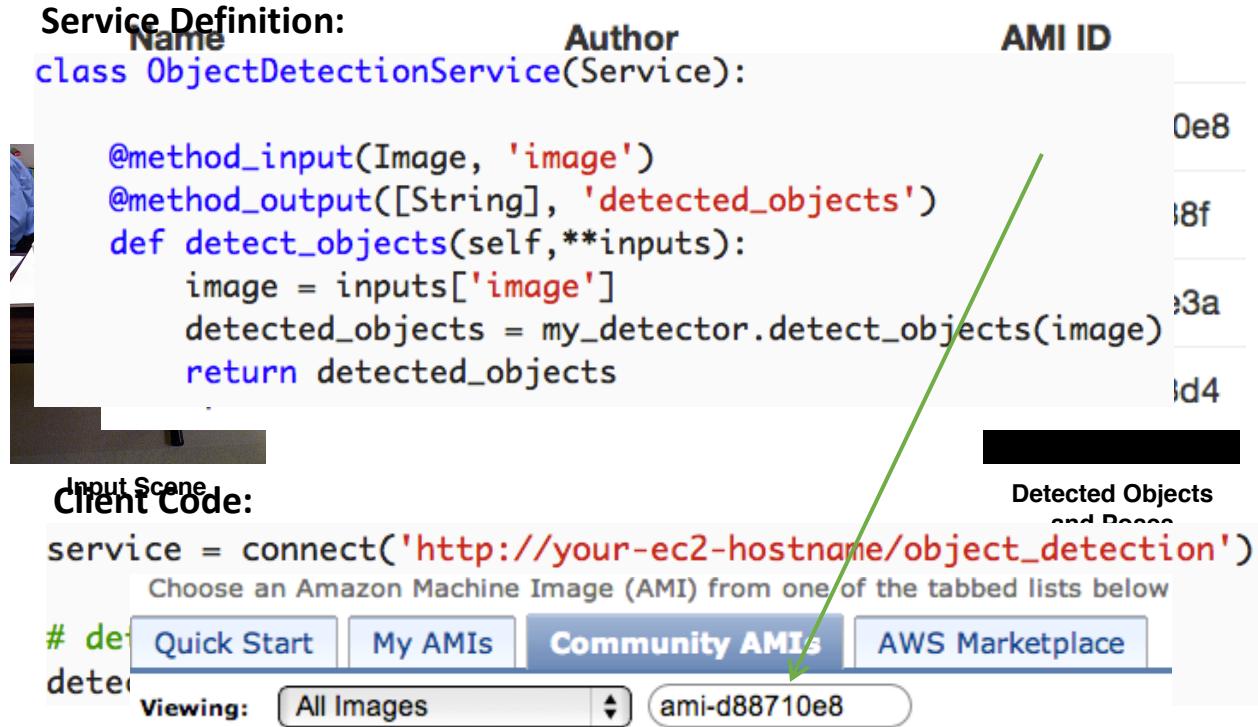
```
service = connect('http://your-ec2-hostname/object_detection')
```

Choose an Amazon Machine Image (AMI) from one of the tabbed lists below

de Quick Start My AMIs **Community AMIs** AWS Marketplace

Viewing: All Images ami-d88710e8

Detected Objects and Pages

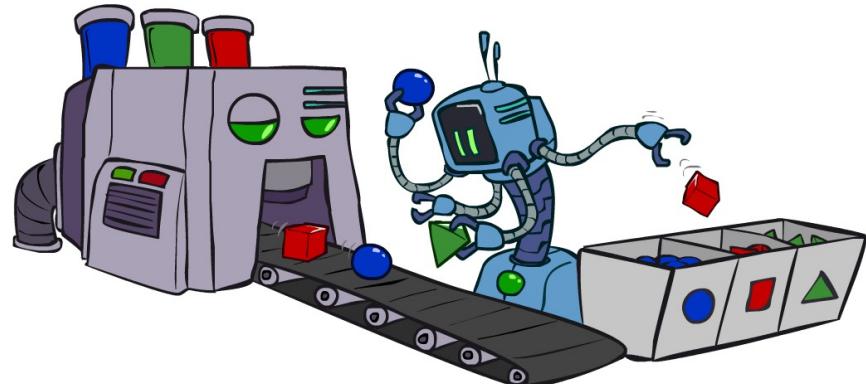


RaaS Workflow – Algorithm Users

- Compare different algorithms implementing the same interface by changing a single URL
- Again, ignore serialization, encoding/decoding, HTTP, etc. – use like an ordinary function call
- Use web-service-based ROS Nodes as if they were local ROS Nodes

How to Solve It with RaaS

- No need to worry about computing hardware
- Install barebones ROS – (i.e. only messaging, low-level control, etc.)
- Use object recognition, motion planning, grasp planning services from RaaS
- RaaS complements ROS



Drawing: Ketrina Yim

Collaborative Data Collection – Example

- Ten camera rig with controllable turntable
- High quality 3d models from about 600 images in 5 minutes of human time
- Ship us an object, we upload model to cloud, you can recognize it in images



Roadmap

(Almost) Complete (pre-alpha)

- Service framework implemented for Python
- Service directory implemented for EC2
- Need to polish up rough edges and automate tedious steps

Future Work

- Linux containers rather than Amazon Machine Images
 - Can then run services anywhere – in your lab or in the cloud
- Infrastructure for other programming languages
- Data collection APIs

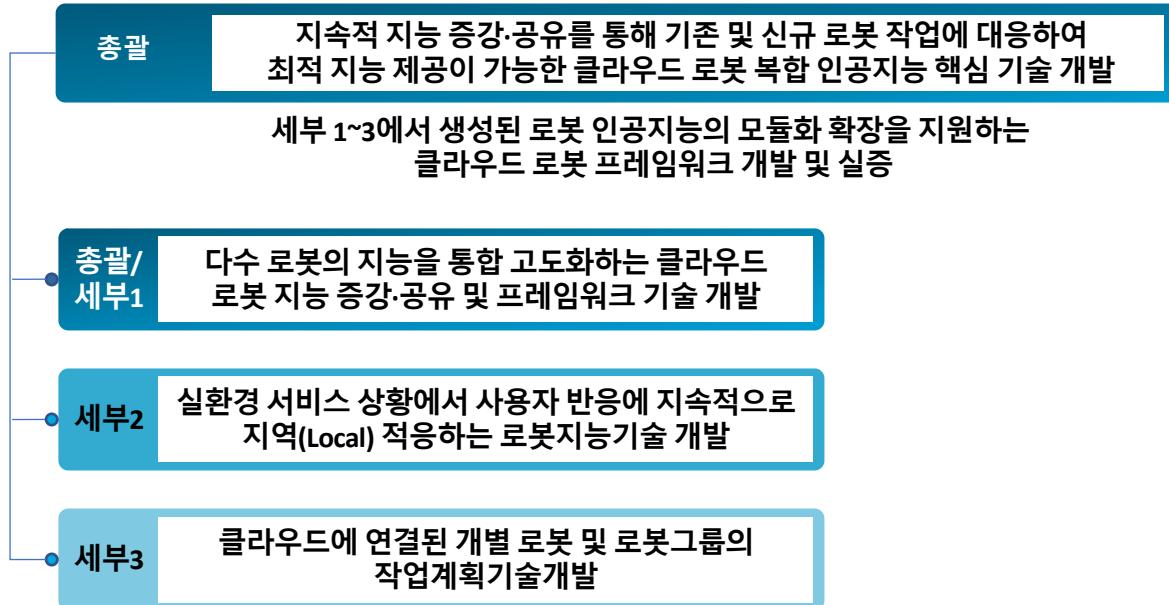
Interested? Email us! arjun@eecs.berkeley.edu

클라우드 로봇 지능 과제 연구 소개

ETRI, LG전자 ('20~'23)

과제 수행 체계

1개의 총괄과제와 3개의 세부과제로 구성되어 있으며, LG전자는 총괄/세부1과제를 주관함



연구개발 필요성: 서비스 로봇의 현실

IT·통합 ▾

현장 투입된 감정인식 로봇들 해고 위기 '재계약 안해'

페퍼 로봇 해고 위기, 기업용 서비스 계약 연장 의향
15% 불과

주간기술동향 최신ICT이슈/로봇 | 2018. 10. 24. 10:34 | Posted by 보라개구리 Purple Frog

뉴스 2019년 03월 06일 17시 59분 KST | 업데이트됨 2019년 03월 07일 10시 12분 KST

'사회적인 로봇의 외로운 죽음' - 최초의 소셜 로봇 지보가 주인들에게 직접 자신의 죽음을 전했다(영상)

폐기되는 '인공 바보들(Artificial Dumbs)'

조선비즈 | 새너제이, 샌프란시스코=황민규 기자

입력 2017.08.14 06:00

로봇 개발 과정에서 인간과의 소통에 방점이 찍히는 이유는 최근 수년간 세계 각지에서 폐기되는 로봇의 수가 늘고 있다는 점과 무관하지 않다. 각종 로봇이 미디어의 스포트라이트를 받으며 등장했지만, 인간과의 소통 문제로 '스위치 오프(전원 장치 끔)' 운명에 처하고 있는 것이다.



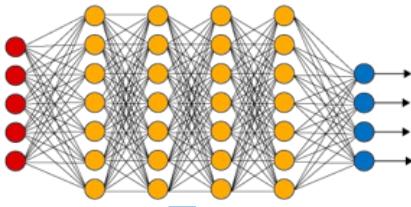
연구개발 필요성

- 현재 서비스 로봇 기술은 변화하는 환경에서도 같은 수준의 상호작용을 제공
 - 상업적 용도의 서비스 로봇에서 활용이 제한
 - **환경과 사용자 변화에 적응하는 맞춤형 서비스를 제공함으로써** 다양한 서비스 분야에서 신시장 창출 가능
※ 데이터·AI 경제 활성화 계획('19.1.), AI R&D 전략('18.5.) 및 혁신성장 실현을 위한 5G+ 전략('19.4) 등에서 **AI·로봇·클라우드가 융합된 新산업 육성 계획 발표**
- 상업용·개인용 서비스 로봇은 높은 수준의 상호작용, 맥락이해, 개인화 기술 필요
 - 현재 서비스 로봇의 멀티모달 상호작용과 맥락 이해 기술은 다수 사용자 환경에 공통 적용을 전제로 한 범용 지능에 의존
 - **맥락 기반 개인화 및 적응화 기술**은 다양한 상황에서 효과적으로 로봇 서비스를 제공 가능
- 맥락이해와 지역적 적응 학습 기술로 기존 서비스 로봇의 상업적 경쟁력 제고 가능
 - **클라우드 로봇 지능을 통해 자체 기술력이 부족한 중소기업과 스타트업의 로봇 서비스 수준 향상**

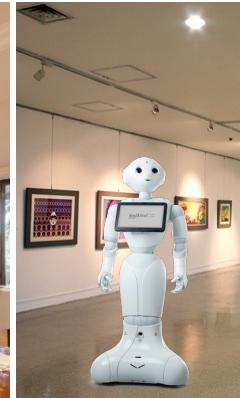
연구 개발 필요성: 현재 서비스 로봇의 맥락 이해 지능

현재 서비스 로봇의 맥락 이해 지능은.....

정적, 폐쇄적



훈련



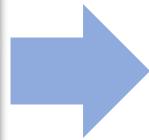
대용량 데이터셋 구축
Labeled Dataset

배포 활용

연구 개발 필요성: 현재 서비스 로봇의 맥락 이해 지능

현재 서비스 로봇의 맥락 이해 지능은.....

정적, 폐쇄적



서비스 환경 변화

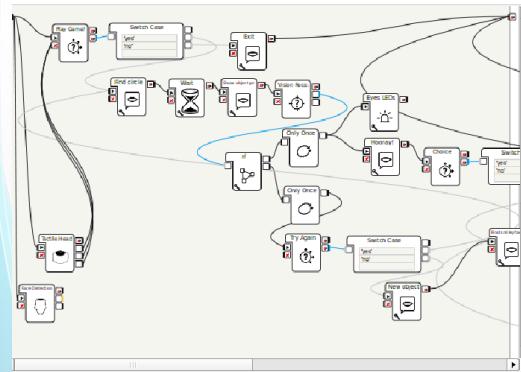
검출/인식 성능 저하

태스크 변화

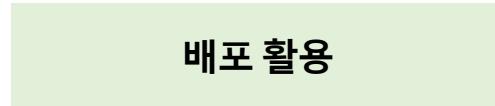
검출/인식 불가능

연구 개발 필요성: 현재 서비스 로봇의 서비스 지능

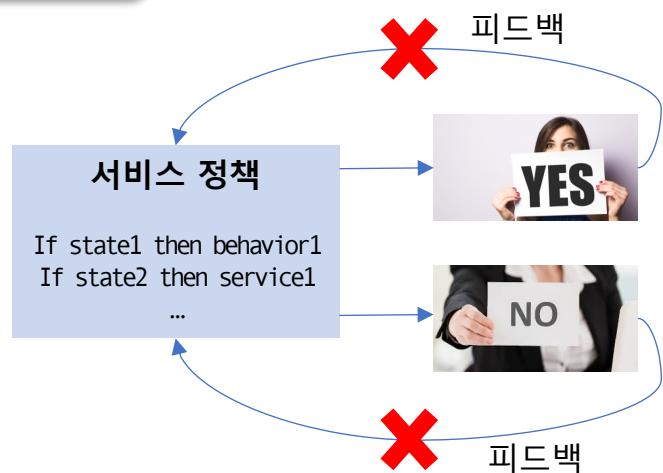
현재 서비스 로봇의 서비스 지능은.....



서비스 사전 저작

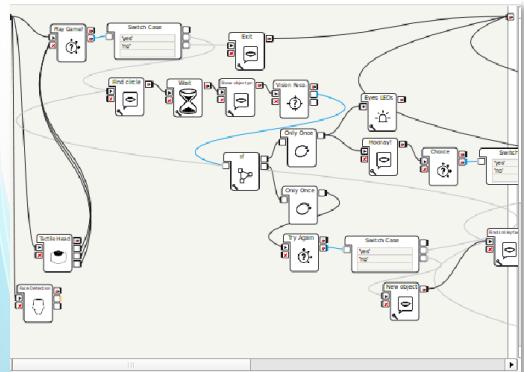


일방적



연구 개발 필요성: 현재 서비스 로봇의 서비스 지능

현재 서비스 로봇의 서비스 지능은.....

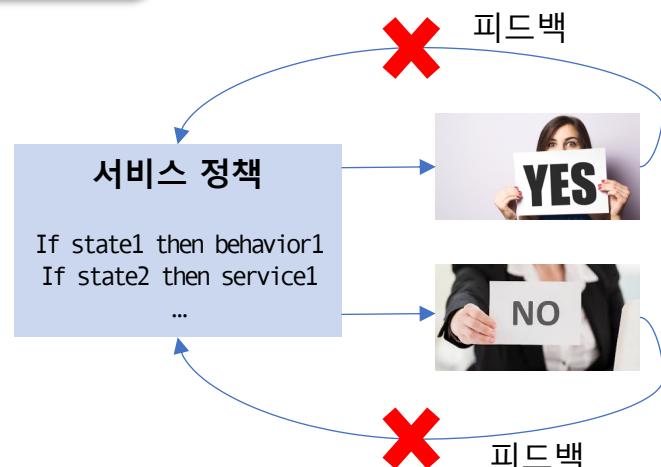


전시관 안내



카페 고객 응대

일방적



사용자 반응에 상관없는
행동과 서비스

사용자 만족도와
관심 저하

연구개발 목표

비전

실환경 서비스 상황에서 사용자 반응에 지속적으로 지역 적응하는 로봇 지능 기술 개발

최종
목표

동적 환경과 맥락 이해를 통해 사용자 반응에 적응하고 개인화 하는 지역적 적응 학습 기술과 클라우드 협업 기반 전역 최적지능 적용 학습 기술 개발 및 다중 도메인 대상 실환경 검증

연구
내용

R1

환경과 맥락을 이해하는
멀티모달 맥락이해 기술

- 개인/군집 특성 기반 개인화 맥락 이해
- 포괄적 상황인지 기반 서비스 맥락 이해
- 멀티모달 데이터 융합 맥락 이해

R3

변화하는 환경/서비스 대응을 위한
지역적 적응 학습

- 맥락 이해 지능의 지역 적응 학습
- 서비스 개인화/특화 지능의 지역 적응 학습
- 지역적 지속 적응 학습 최적화

클라우드 로봇 복합인공지능
통합시스템

학습 모델·데이터 공유



R2

사용자 반응에 적응하는
서비스 개인화와 특화

- 서비스 개인화/특화 정책 심층강화학습
- 멀티모달 단서 기반 사용자 반응 해석
- 사용자 반응의 원인 요소 추정

R4

클라우드 협업 기반
전역 최적지능 획득 및 적용

- 다중/분산 통합 전역 지능 획득과 적용
- 지역 공유/특화 지능 분리 학습
- 전역 지능 적용 학습 최적화

R5 다중 도메인 서비스 로봇 활용 및 실환경 검증



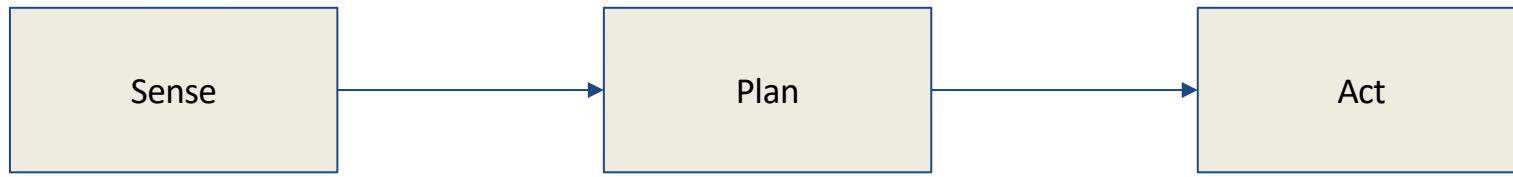
이기종 복수 로봇 적용



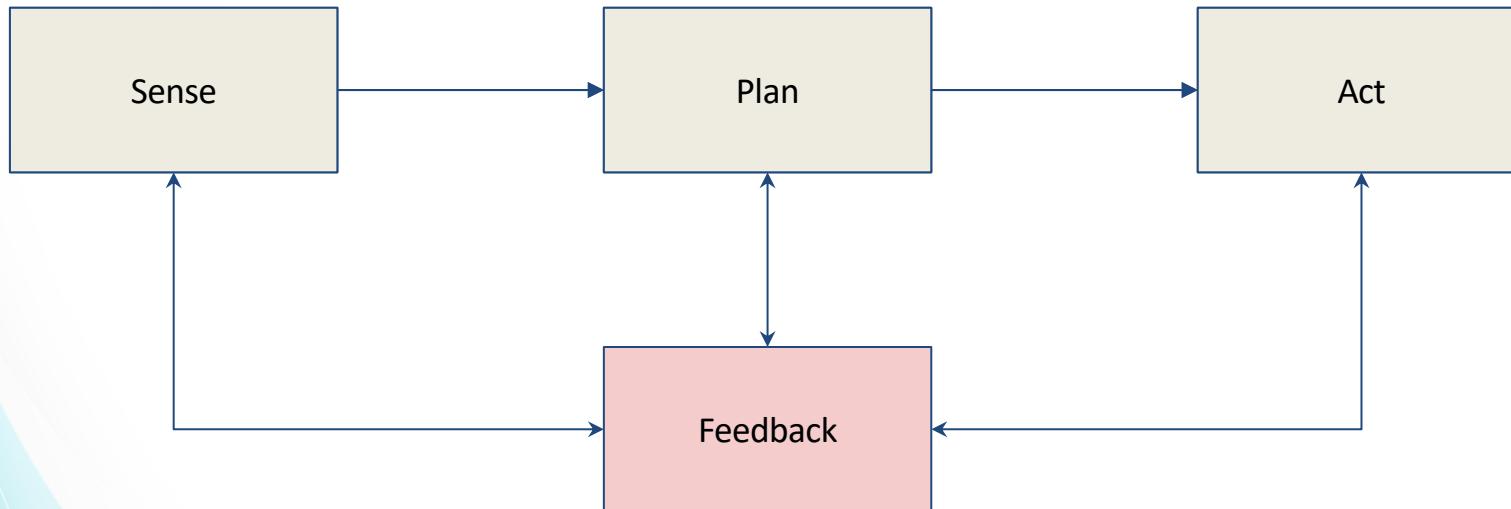
다중 서비스 도메인 테스트베드/실환경 운영



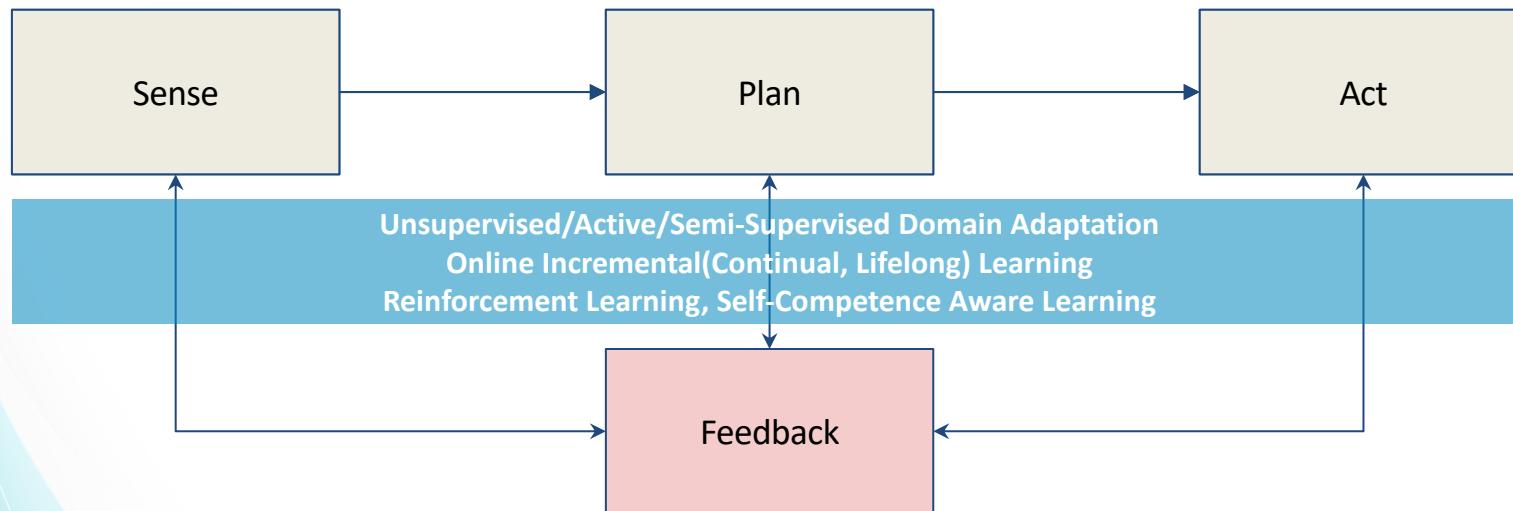
기존 로봇



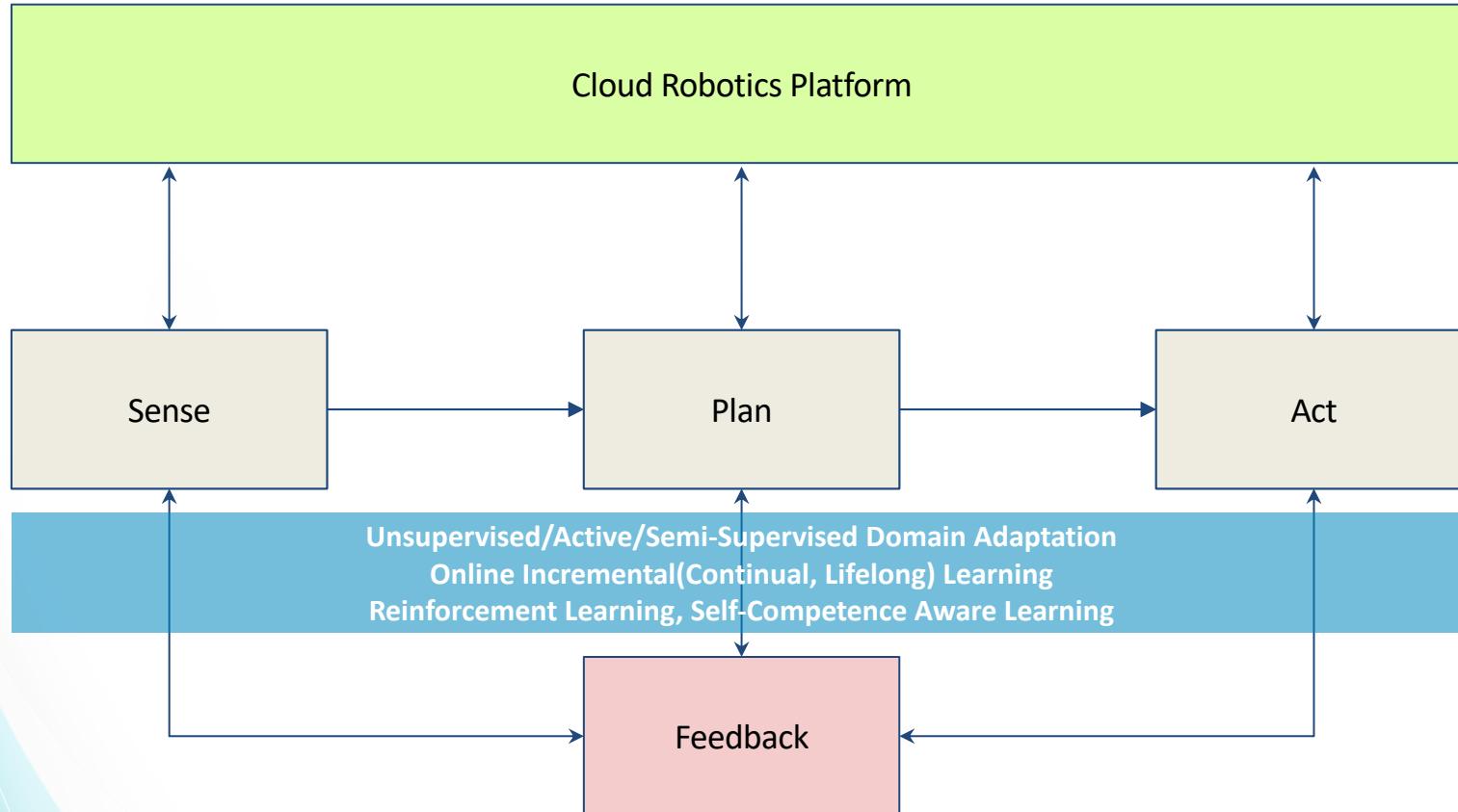
지역 적응 학습하는 로봇



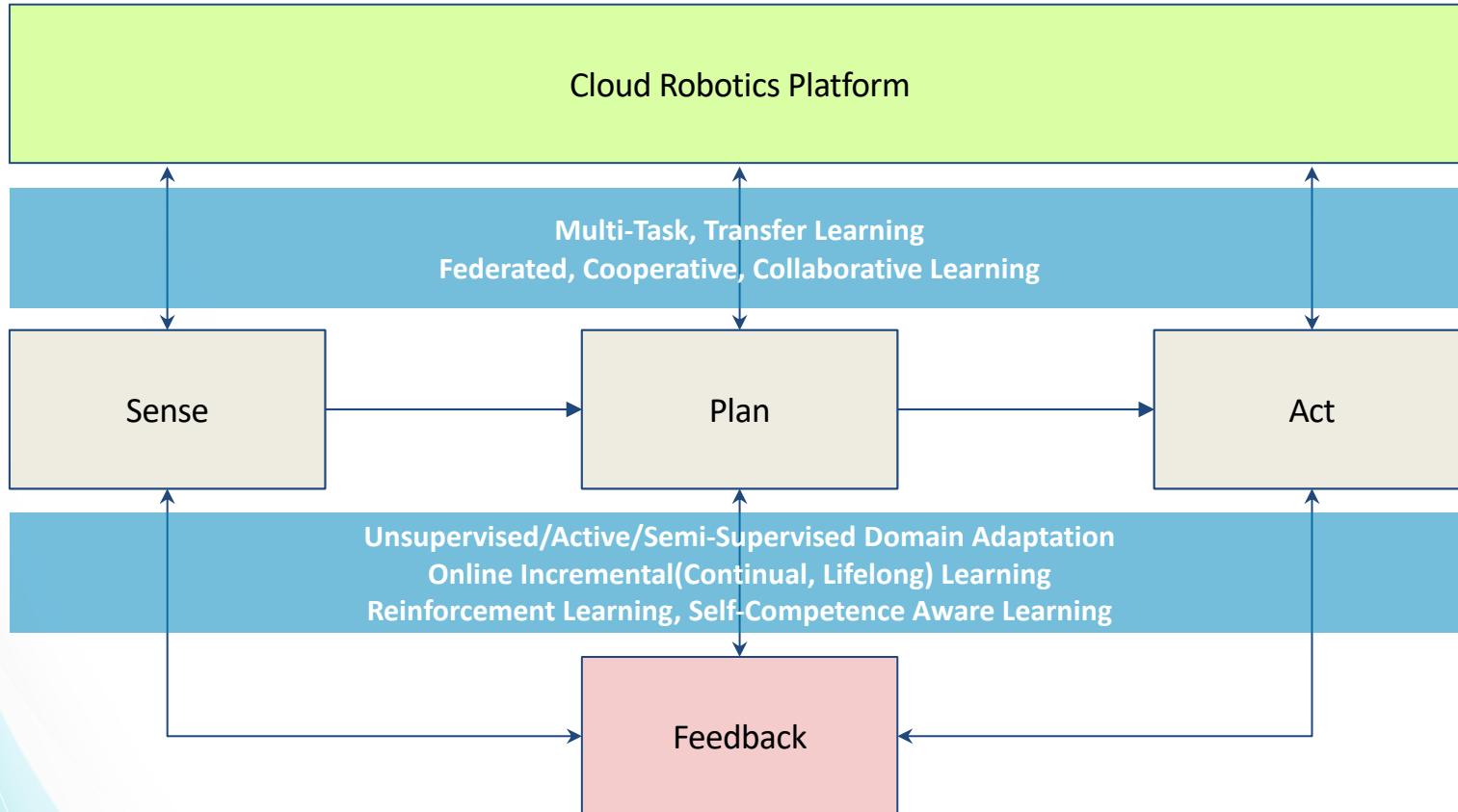
지역 적응학습하는 로봇



클라우드 기반 지능 증강 로봇



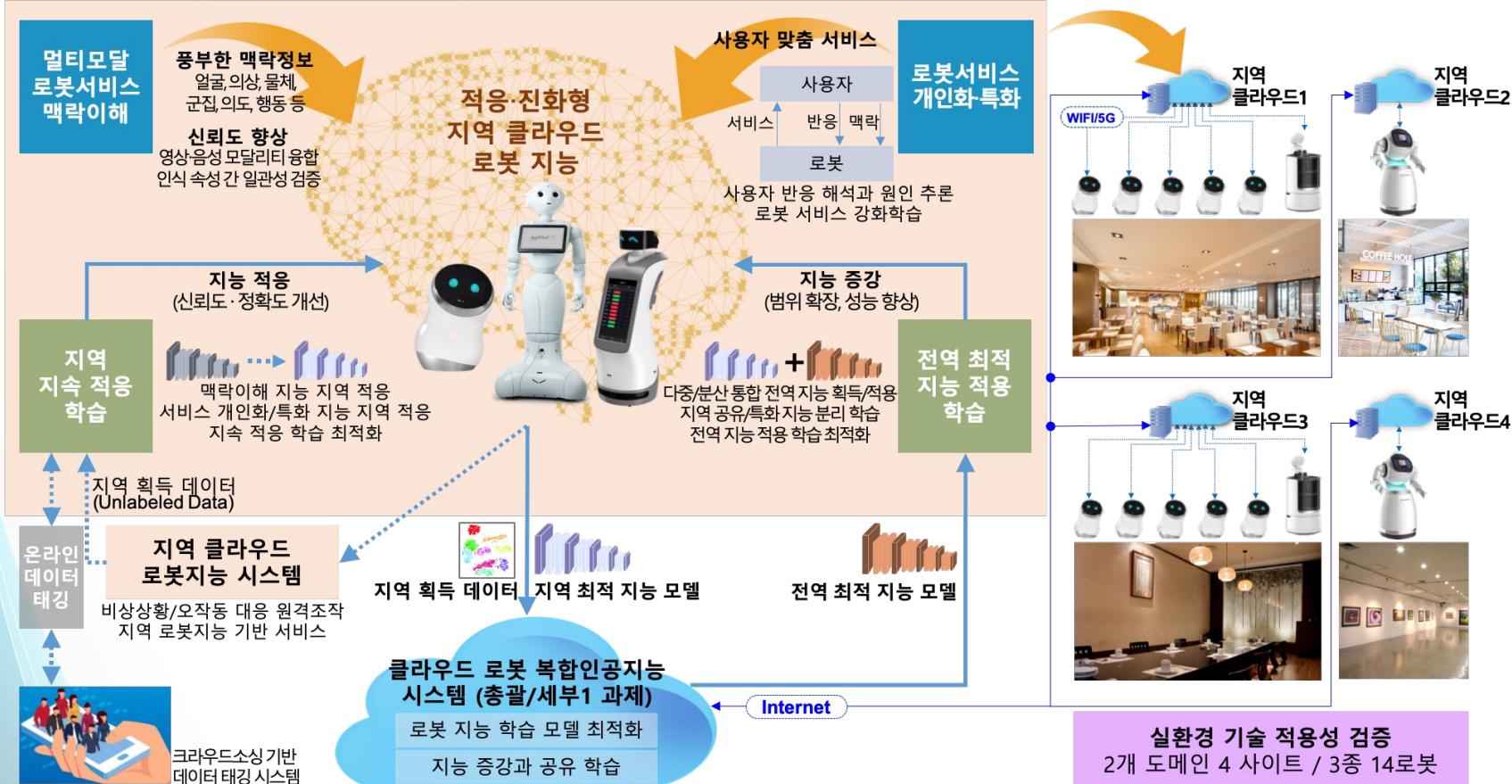
클라우드 기반 지능 증강 로봇



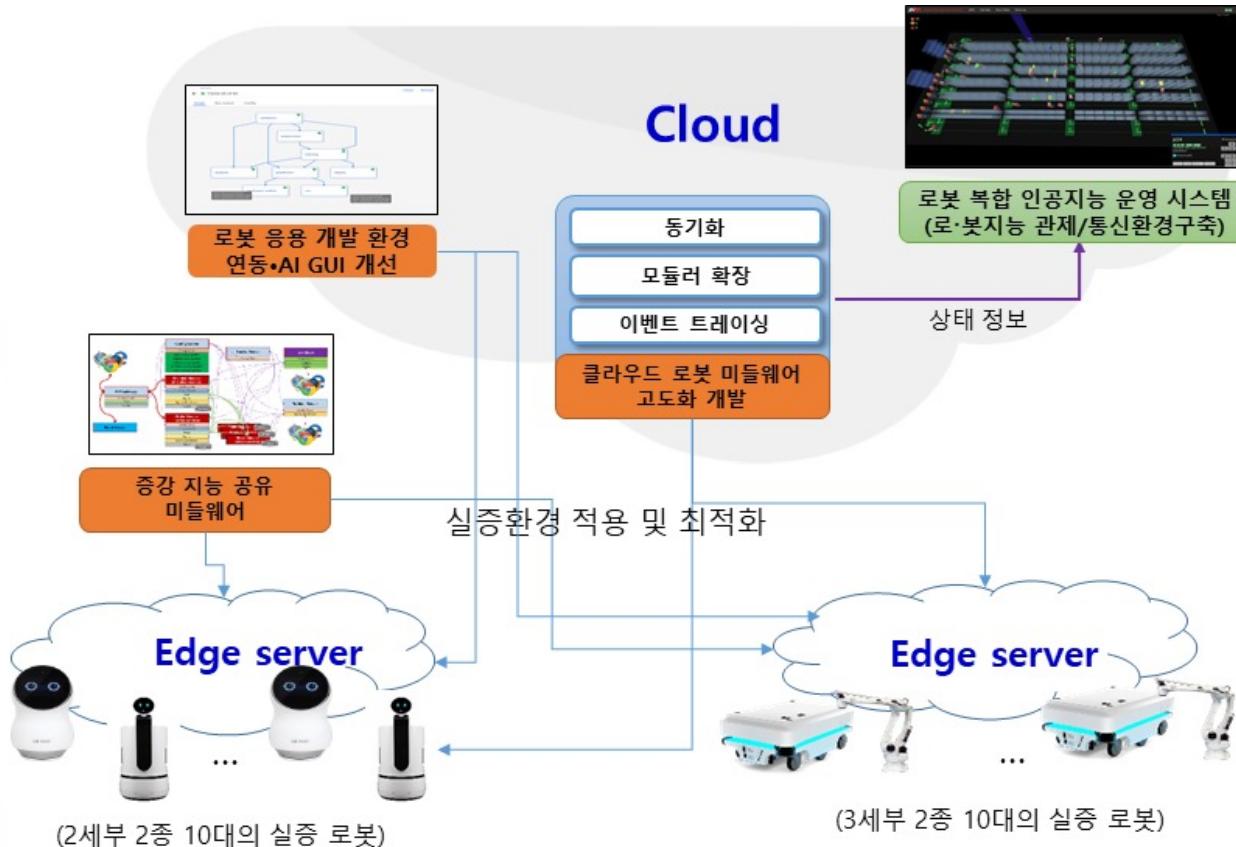
기술 활용 시나리오: 음식점 도메인 예시



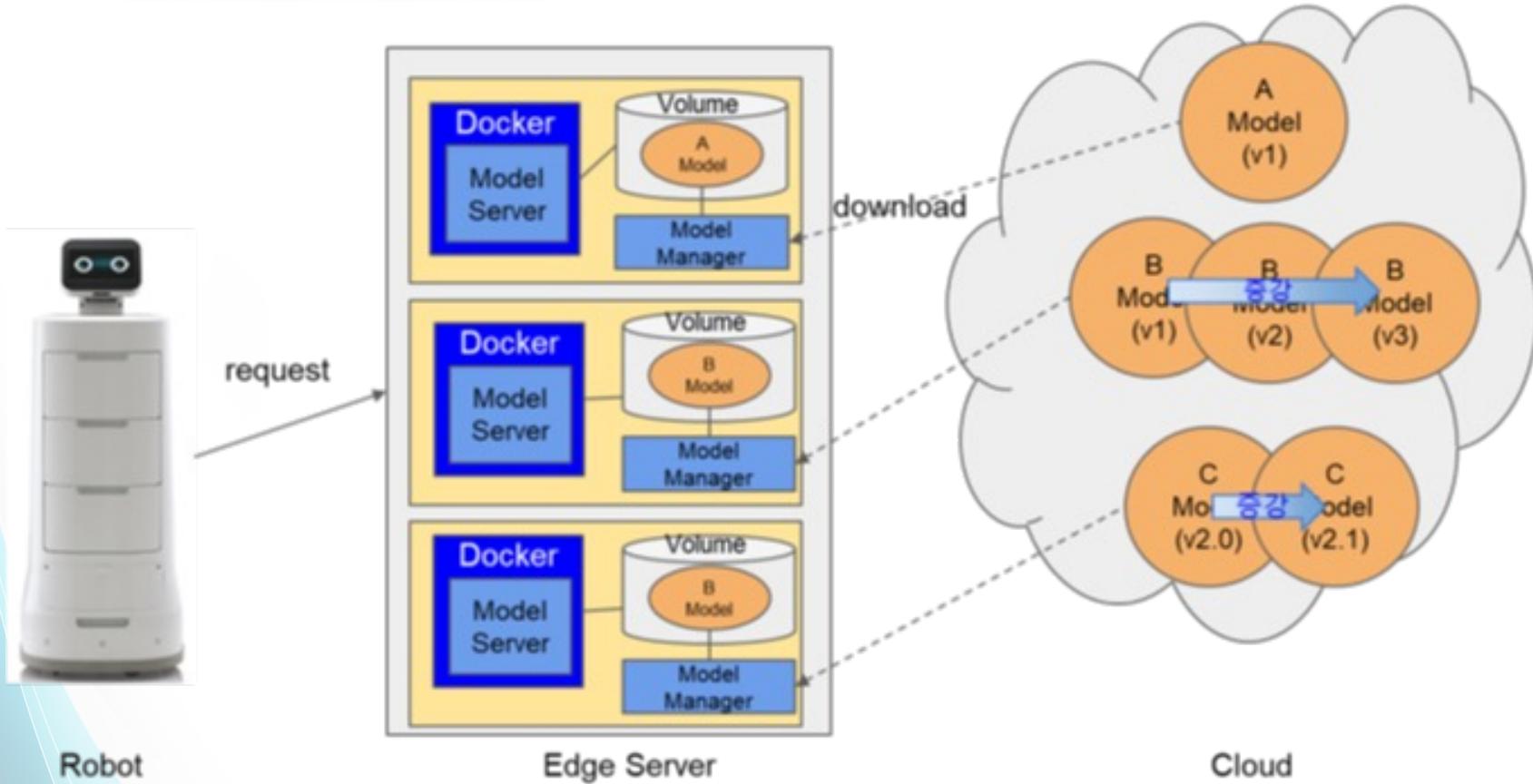
연구개발 기술과 실증 체계



클라우드 로봇 복합인공지능 플랫폼



클라우드 로봇 복합인공지능 플랫폼

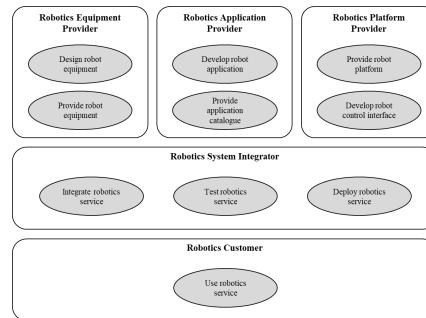


클라우드 로봇 표준화 – RaaS (1)

□ 클라우드 로보틱스 서비스 프레임워크 및 기능 요구사항(RaaS) 표준 개발

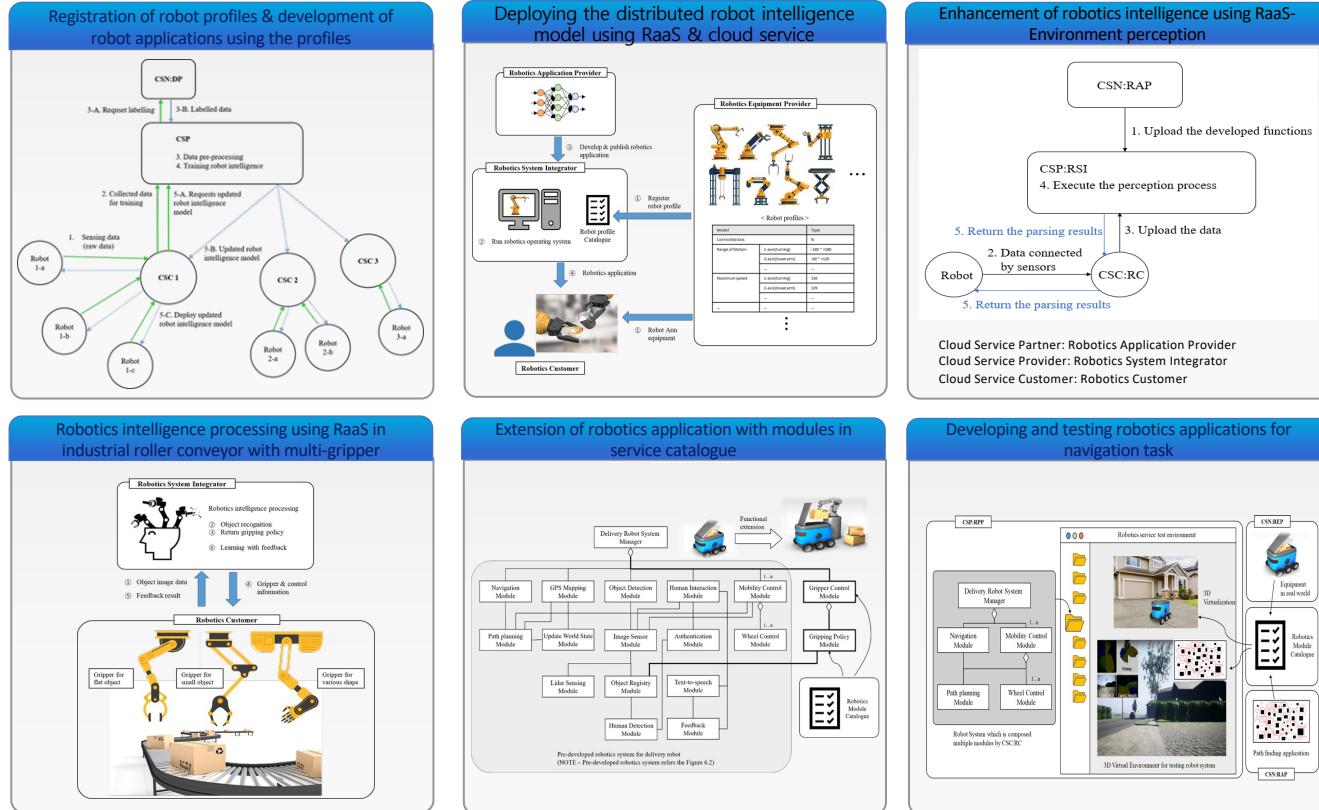
- ITU-T SG13 NP 승인(2020) 및 표준 기고서 17건
- NP: Y.RaaS-reqts(Cloud computing – Functional requirements of robotics as a service)의 에디터쉽 확보 유지하고 주 에디터로 표준 문서 개발 주도 (https://www.itu.int/itu-t/workprog/wp_item.aspx?isn=16731)
- ISO/IEC JTC1 SC42(Artificial Intelligence) 및 ISO TC299(Robotics) 등 유관 기관과 표준안 개발 내용 공유, 상호 검토
- 년 3회(3/7/12월) ITU-T SG13(미래 네트워크 클라우드 컴퓨팅 분야) 온라인 회의 참석
- 로봇 산업 생태계 및 이해당사자를 정의하고 표준 문서에 반영
- 클라우드 로봇을 위한 지능 증강 서비스, 로봇 응용 개발 등 6종의 유즈케이스 기고하여 표준안에 반영

ITU TELECOMMUNICATION STANDARDIZATION SECTOR STUDY GROUP 13		SG13-TD612/WP2 STUDY GROUP 13		SG13-TD764/WP2 STUDY GROUP 13	
Question(s):	17/13	TD	Virtual, 20-31, July 2020	TD	Virtual, 16 July 2021
Source:	Editor				
Title:	Initial draft Recommendation ITU-T Y.RaaS-reqts: "Cloud computing –Functional requirements for Robotics as a Service"				
Purpose:	Proposal				
Contact:	Sanggil Shin ETRI Korea (Republic of)	Tel +82 42 860 1379 Fax +82 42 861 3404 E-mail: ssh@etri.re.kr		Tel +82 42 860 1379 Fax +82 42 861 3404 E-mail: ssh@etri.re.kr	
Contact:	Wen Wu China Telecom China	Tel +86 20 38639103 Fax +86 20 38636215 E-mail: wyiner@chinatelecom.cn		Tel +86 20 38639103 Fax +86 20 38636215 E-mail: wyiner@chinatelecom.cn	
Keywords:	Robotics, cloud computing, robotics as a service, RaaS				
Abstract:	This document contains the initial draft Recommendation ITU-T Y.RaaS-reqts: "Cloud computing – Functional requirements of robotics as a service". This document includes the results of discussions on the Q17/13 meeting which was held via virtual meeting in 20-31 July 2020.				
The following table shows discussion results for contributions.					
No	Source	Contribution title and Proposals	Agreements		
C1018	ETRI	New: initiating a new item on cloud computing – overview and requirements for robotics as a service (Y.RaaS-reqts)	Accepted for generating living list with C1018		
C1019	ETRI	New: initiating a new item on cloud computing – overview and requirements for robotics as a service (Y.RaaS-reqts)	Accepted with modification		
During this meeting, it is agreed as follows:					
<ul style="list-style-type: none">• Initiate new draft Recommendation for ITU-T Y.RaaS-reqts: "Cloud computing – Functional requirements for Robotics as a Service" with modification of title.• Generate the new living lists for this work item with the contents of C1018.					

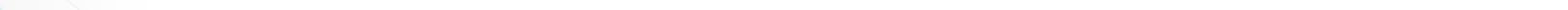


[Robotics Ecosystem and Stakeholders]

클라우드 로봇 표준화 – RaaS (2)



RaaS 정리



RaaS란 무엇인가?

THE BRAVE, NEW WORLD OF ROBOTS-AS-A-SERVICE



로봇 플랫폼과 로봇 운영 솔루션을 모두 임대 활용하고 사용한 만큼 비용을 지불

RaaS의 장점

- 비용 절감과 위험 분산
 - 초기 구입과 설치 비용
 - 초기 도입에 뒤따르는 AS 비용
 - 즉, 비용 대비 효과에 대한 크 위험 존재
- 유연성과 확장성
 - 수요가 계절 별로 변동이 있는 등 자원 요구가 변화하는 경우 대응이 가능
 - 추가 서비스의 설치와 활용이 용이
 - RaaS의 유연성은 낮은 통합 비용과 신속한 시운전이 가능한 경우에만 해당됨 --> 통합과 시운전 비용은 고정 비용이므로 최소화할 수 있어야 함

RaaS의 적용 분야와 장점

- 산업 로봇에 부적절
 - 프로그래밍 하기가 번거로운 산업용 로봇 애플리케이션은 고정 비용이 높아, 단기간 시험적으로 설치하는 것이 무의미
 - 로봇 시스템이 지속적으로 필요하고, 유연성은 거의 필요하지 않으며, 적용 사례 성공에 대한 위험을 스스로 부담하는 경우 RaaS는 부적합
- 서비스 로봇 투자가 활발하지 않은 업계가 주요 대상
 - 몇몇 공급자는 코봇을 이용하여 제조업에 접근한 반면, **RaaS의 주요 대상은 청소, 보안, 배달 서비스 및 물류 부문** [link](#)
 - 위험을 외주화하고 대체적으로 안정적인 시장에서 활동하는 경우, 'Pay per Use' 비즈니스 모델을 고려할 수 있음
 - 로봇을 구매하는 대신 로봇을 사용한 만큼만 비용을 지불
 - RaaS 제공 업체는 **KPI (핵심 성과 지표)** 등의 형태로 성능 보장을 약속하므로 로봇 활용의 위험이 로봇 사용자에게서 RaaS 공급자에게로 이전됨

RaaS 비즈니스 사례: 미국 6 River

- 창고 관리용 소형 로봇 서비스
- 로봇 플랫폼은 Chuck으로 상품 진열대까지 자율주행을 통해 찾아가 재고 관리 가능
- 비용: 로봇 8대 임대료 25만 달러와 연간 유지비 5만 달러



RaaS 비즈니스 사례: 미국 Fetch Robotics

- 창고 진열대에서 물건을 집어서 컨베이어 벨트까지 옮기거나 사용자에게 전달
- 라이다 센서, 3D 카메라, 지도 구축과 자율주행, 사람과 사물 구분
- 비용: 500kg용 물류 로봇의 월 이용료는 3천 ~ 5천 달러



RaaS 비즈니스 사례: 미국 InVia Robotics

- 물류창고용 로봇 서비스 제공
- 비용: "작업량 한 단위"를 기준으로 가격을 설정 (예: 상품 픽업 1개 당 10센트)
- 적용 기업: Rakuten Super Logistics



미국의 RaaS 기업들

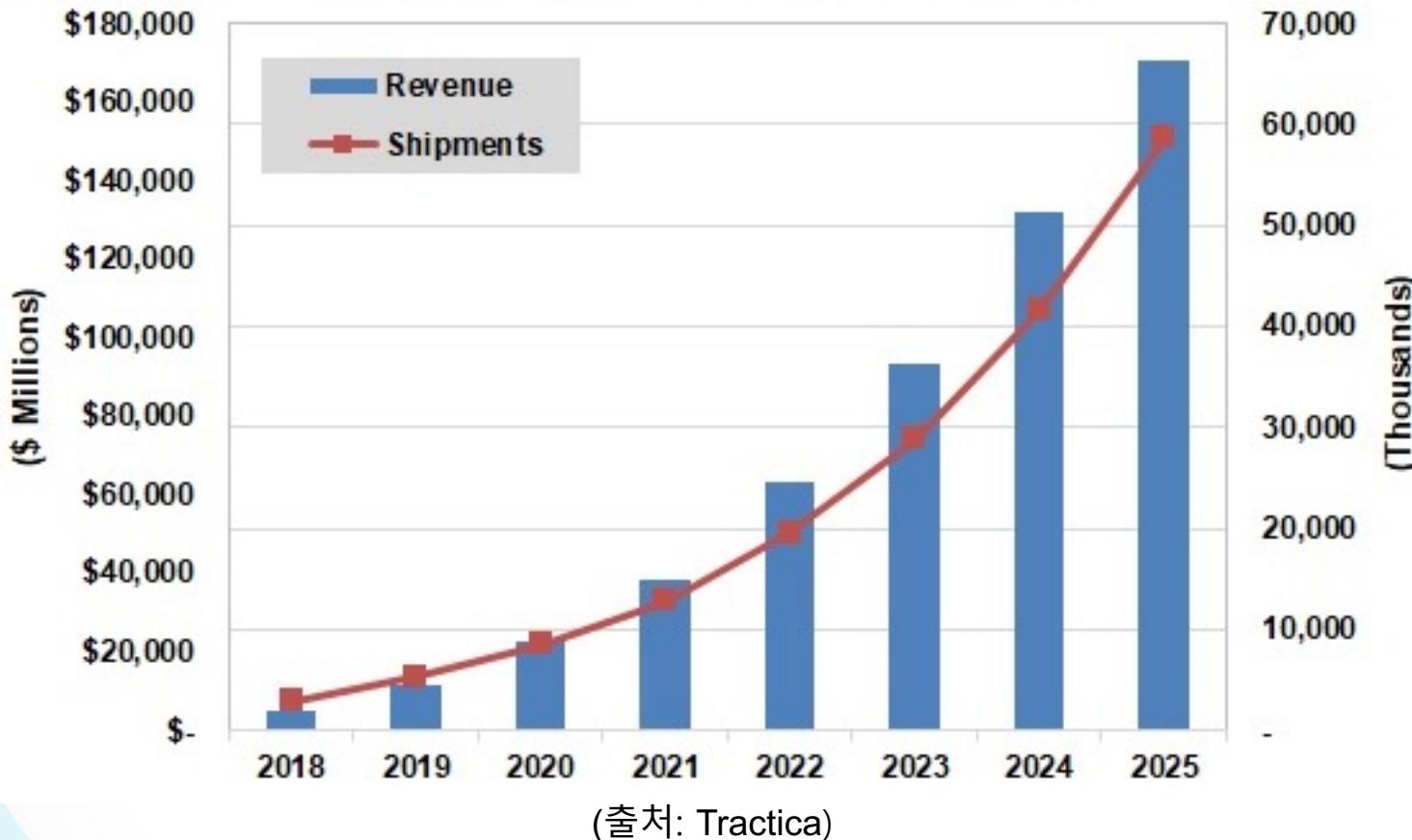
기업	서비스	과금	홈페이지 주소
Aethon	배송	로봇 1대당 한 달 사용료 US\$ 1500~2000	https://aethon.com
Kiwibot	배송	1회 배달에 US\$ 3.80, 월 사용료 US\$ 14.99	https://kiwibot.com
Marble	배송	배달 건마다 비용 적용	https://www.marble.io
Nuro	배송	1회 배달에 US\$ 5.95	https://nuro.ai
Robomart	배송	가게가 직접 소비자에게 '배달'되는 개념, 적재 단위에 따라 비용 다양	https://robomart.co
Savioke	배송	로봇 1대당 한 달 사용료 US\$ 2000	https://www.savioke.com
Starship	배송	1회 배달에 US\$ 1.99	https://www.starship.xyz
Peanut Robotics	청소	시간당 비용 지불	https://peanutrobotics.com
Maidbot	청소	로봇 1대당 한 달 사용료 US\$ 419	https://maidbot.com
Cobalt Robotics	보안	로봇 1대당 한 달 사용료 US\$ 6000	https://cobaltrobotics.com
Knightscope	보안	로봇 1대당 한 달 사용료 US\$ 4500~8000	https://www.knightscope.com
SMP Robotics	보안	로봇 1대당 한 달 사용료 US\$ 1500	https://smprobotics.com
Hirerobotics	공장	일주일 최소 40시간, 한 달 30일 사용 기준 1시간당 사용료 US\$ 33	https://www.hirebotics.com
Kindred	공장	지능성 기능별로 지불	https://www.kindred.ai
Ready Robotics	공장	로봇 1대당 한 달 사용료 US\$ 2000~4000	https://ready-robotics.com
RobotWorx	공장	계약조건에 따라 다름.	https://www.robots.com
6 River	물류	사용 첫 해 로봇 8대당 US\$ 250,000, 이후 1년에 US\$ 50,000	https://6river.com
Fetch Robotics	물류	계약조건에 따라 다름.	https://fetchrobotics.com
InVia Robotics	물류	Picking 동작마다 10센트	https://www.inviarobotics.com
Locus Robotics	물류	알려진 바 없음.	https://locusrobotics.com

(출처: https://dream.kotra.or.kr/kotranews/cms/news/actionKotraBoardDetail.do?SITE_NO=3&MENU_ID=180&CONTENTS_NO=1&bbsGbn=243&bbsSn=243&pNttSn=183363)

RaaS 시장 전망

- 'ABI Research' 연구소는 2026년 RaaS로 달성 가능한 매출이 340억 US 달러(원화로 37조 원)에 이른다고 언급

클라우드 로봇 시장 전망



정리

- 클라우드 로봇 기술과 클라우드 플랫폼의 발전 성숙을 기반으로 로봇 응용 개발, 배포, 관리를 원격 시스템으로 처리 가능
- 이를 기반으로 로봇 플랫폼, SW 솔루션, 관제와 과금을 서비스로 일괄 제공하는 RaaS 비즈니스 모델이 가능
- RaaS는 로봇 활용 기업의 초기 투자 비용 절감과 로봇 적용 위험을 분산함으로써 로봇 시장 확장에 기여
- 솔루션 초기 개발 비용이 적은 대형 창고 물류를 중심으로 RaaS 초기 시장 형성
- 클라우드 플랫폼을 기반으로 한 로봇 지능 생성과 배포, 효과적 데이터 수집과 지능 증강, 로봇 지능의 지역 적응과 성장을 포함하는 "클라우드 로봇 지능" 기술을 통해 RaaS 적용 범위의 확장 기대

감사합니다.

실환경에서 잘 동작하는 기술을 넘어,
실환경에서 잘 학습하는 기술을!

An important challenge for machine learning is not necessarily finding solutions that work in the real world, but rather finding stable algorithms that can learn in real world.

(Timothée Lesort, 2020)