

인공지능 기술의 대중화(AI Democratization)를 위한

TANGO 커뮤니티 제1회 컨퍼런스



신경망 실행 이미지 생성 및 자동 배포 기술

성명 박종현

소속 래블업 주식회사



주관



주최



후원



목차

1. Overview
2. 실행 이미지 생성 모듈
3. 배포/실행 모듈

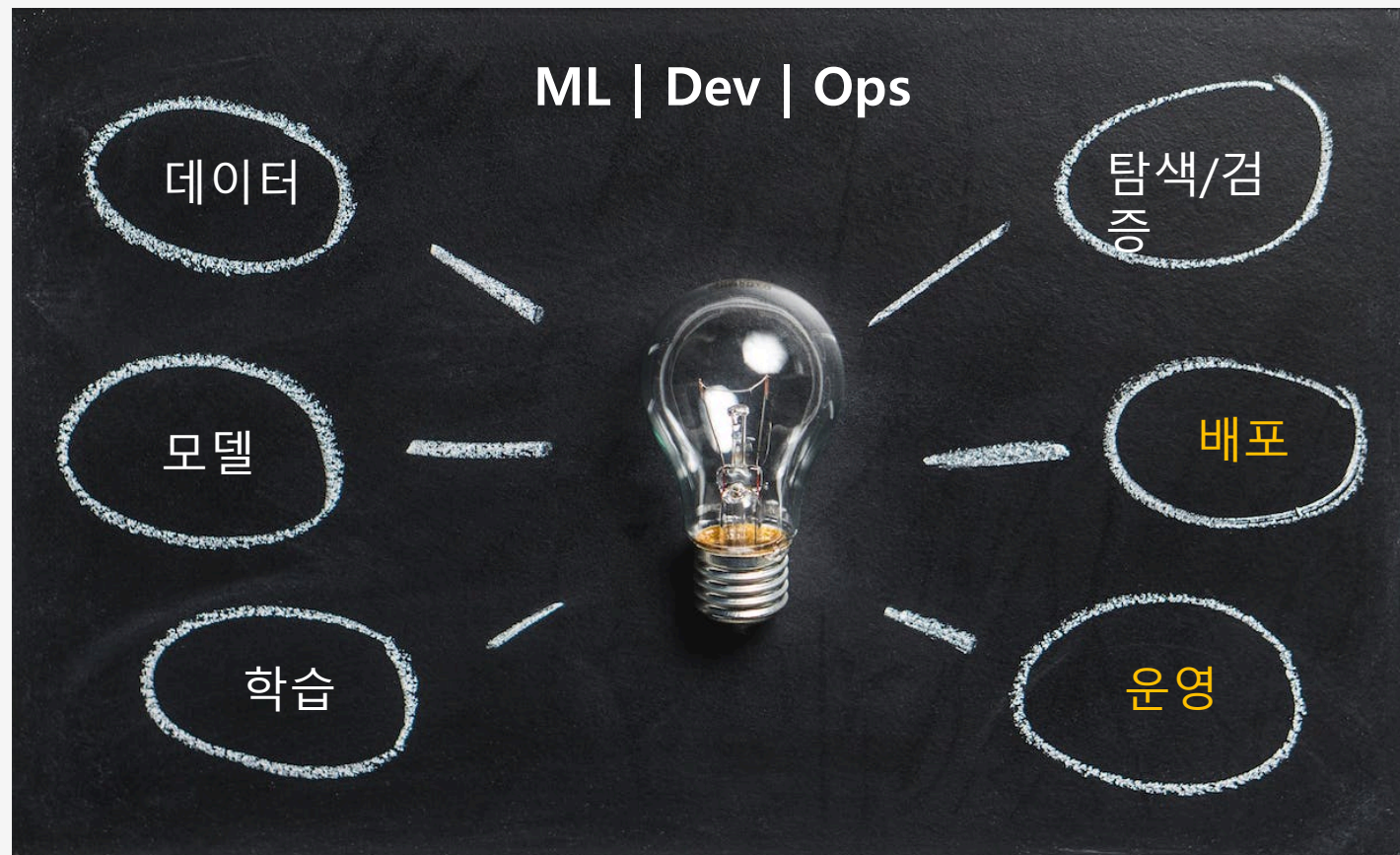
인공지능 기술의 대중화(AI Democratization)를 위한
TANGO 커뮤니티 제1회 컨퍼런스

1. Overview – 인공지능 모델의 학습과 배포 엔터프라이즈 동향

- 관심의 확장
 - 모델의 학습을 어떻게 하면 잘 할까
 - → 배포를 포함하는 파이프라이닝을 어떻게 구축할까
- 2016년 알파고 → 2018~19년 학습을 위한 GPU 인프라 및 운영 S/W 플랫폼 구축 시장이 열림(국내 기준)
- 2021~22년 사내 파이프라인 구축 프로젝트 다수 시작
- TANGO도 이런 요구에 대응하는 한 프로젝트

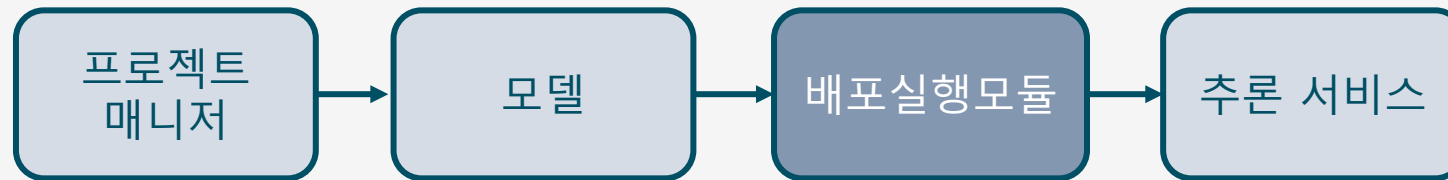


1. Overview – 인공지능 모델 배포의 어려움



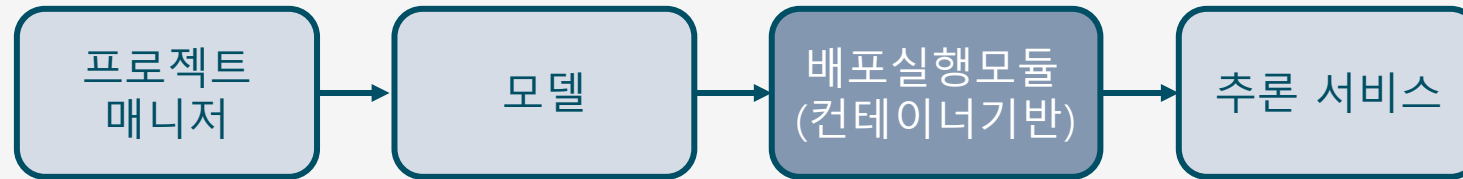
- 많은 머신러닝 모델이 실제 서비스로 이어지지 못하고 좌초
- 모델 개발, 학습, 배포 체계가 복잡해지면서 연구자/개발자의 어려움이 증가
- 이질적인 가속 환경: CPU, GPU, TPU, IPU, ASIC / Cloud, Edge, On-Premise / 5G, 6G
- 모델을 성공적으로 배포하고 운영하기 위해서는 연구 역량뿐만 아니라 DevOps 역량까지 필요 → MLOps

1. Overview – 개발 목표



앞선 모듈에서 생성된 모델을 어떻게 자동으로
이질적 목표 장치에 배포/실행하고 추론 서비스를
제공할 것인가?

2. 실행 이미지 생성 모듈 - 컨테이너 기반 배포

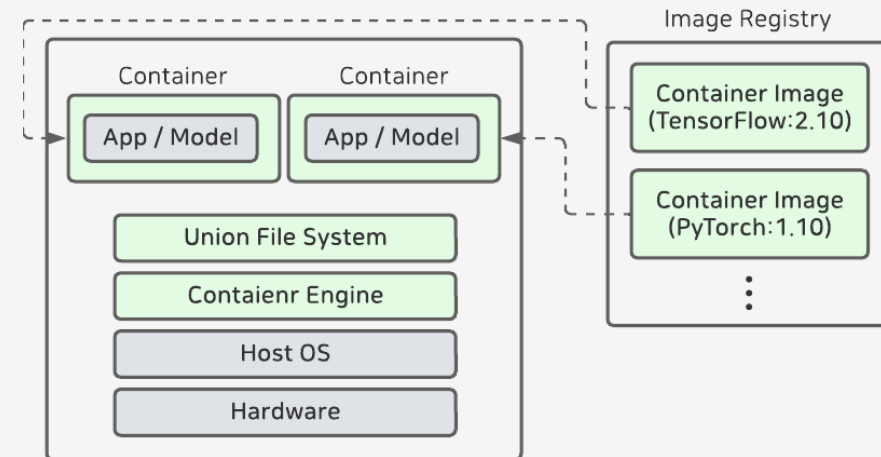


다양한 OS, 플랫폼 아키텍처, 패키지 환경에서
다양한 방식으로 모델 배포를 고려

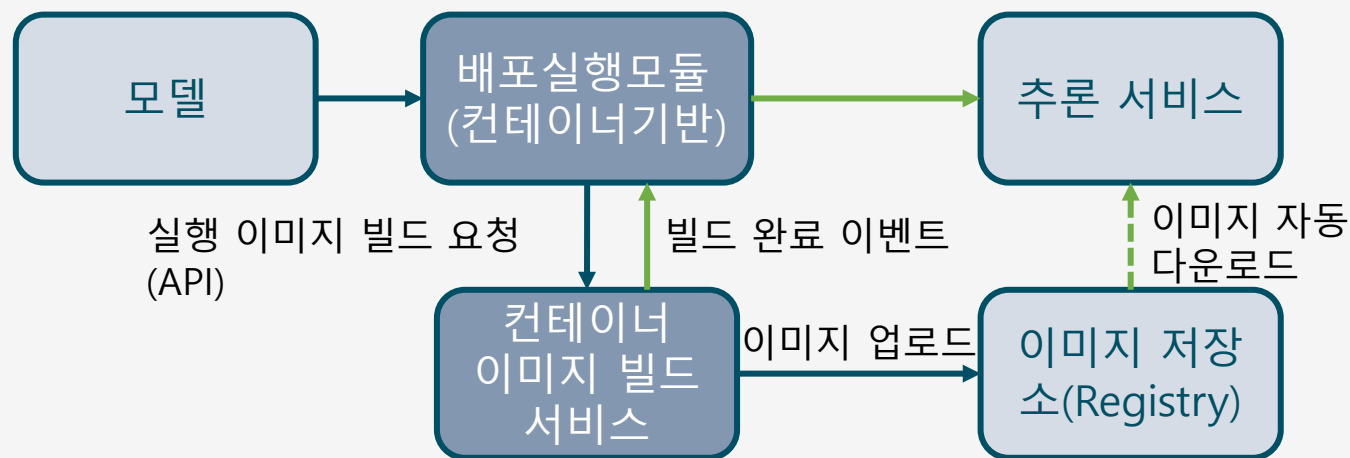
- 예) 동일 장치에 A 사용자는 TensorFlow를, B 사용자는 PyTorch 모델을 실행

컨테이너 기반의 실행 환경 배포

- 실행에 필요한 라이브러리, 설정 등 환경을 하나의 환경 이미지로 패키징하여 실행 환경 격리(isolation)



2. 실행 이미지 생성 모듈 - 컨테이너 이미지 자동 빌드 시스템



컨테이너 이미지 빌드 서비스

- 요청에 기반한 동적 추론 환경 이미지 빌드
- REST API로 서비스 하여 독립성을 유지하면서도 전체 모듈의 구성 요소가 될 수 있도록 설계
- 다중 사용자를 고려한 인증 기능

2. 실행 이미지 생성 모듈 - 표준 API 기반의 이미지 빌드 서버

- 빌드 과정 정의에 필요한 API 파라미터 내역
 - 소스/타겟 이미지 이름
 - 설치할 패키지(APT, PyPI, Conda 등)
 - 기반 아키텍처
 - 커스텀 레이블
 - 이미지 내 root 권한 허용 여부
 - 빌드 후 저장소 자동 업로드(push) 여부
 - ...
- 전달된 API 파라미터 기반으로 동적으로 빌드 정의 파일(Dockerfile) 및 빌드 환경 구성
- 선행 모듈과 공통 인터페이스로 통신 가능

```
RUN apt-get update && \
    apt-get install -y \
        ca-certificates {% if allow_root %} sudo {% endif %} \
        wget curl git-core \
        vim-tiny zip unzip \
        python3 python3-pip \
        libssl-dev \
        proj-bin libproj-dev \
        libgeos-dev libgeos++-dev \
        mime-support ncurses-term \
    {%- if packages['apt']|length == 0 %}
    gcc g++ && \
    {%- else %}
    gcc g++ \
    {%- endif %}
    {%- for custom in packages['apt'] %}
    {%- if loop.index0 == (loop.length)-1 %}
    {{ custom }} && \
    {%- else %}
    {{ custom }} \
    {%- endif %}
    {%- endfor %}
    apt-get clean && \
    rm -rf /var/lib/apt/lists/ && \
```


2. 실행 이미지 생성 모듈 - 표준 API 기반의 이미지 빌드 서버

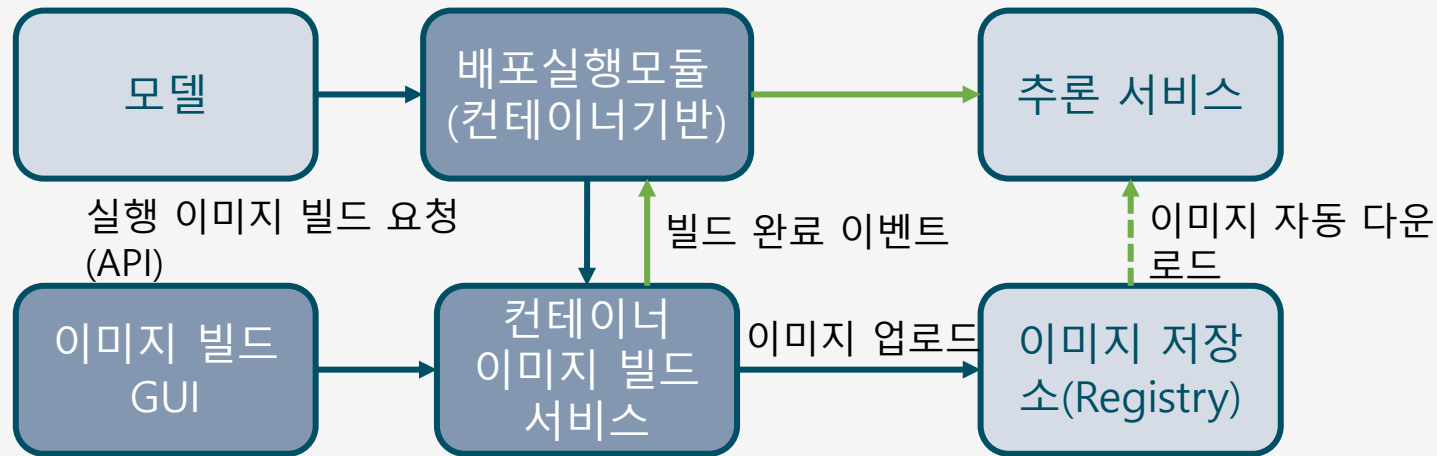
빌드 작업 모니터링을 위한 실시간 이미지 빌드 로그 구독

```

2022-10-26 17:17:44.631 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: pickleshare in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (0.7.5)\n'}
2022-10-26 17:17:44.631 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: traitlets==5 in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (5.5.0)\n'}
2022-10-26 17:17:44.632 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: stack-data in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (0.5.1)\nRequirement already satisfied: decorator in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (5.1.1)\n'}
2022-10-26 17:17:44.632 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: matplotlib-inline in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (0.1.6)\n'}
2022-10-26 17:17:44.632 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: pygments>=2.4.0 in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (2.13.0)\n'}
2022-10-26 17:17:44.632 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: backcall in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (0.2.0)\n'}
2022-10-26 17:17:44.633 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: pexpect>4.3 in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (4.8.0)\n'}
2022-10-26 17:17:44.633 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: jedi>=0.16 in /usr/local/lib/python3.10/dist-packages (from ipython->r requirements.txt (line 41)) (0.18.1)\n'}
2022-10-26 17:17:44.737 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting pydantic!=1.7.1,!<1.7.2,!<1.7.3,!<1.8,!<1.8.1,<2.0.0,>=1.6.2\n'}
2022-10-26 17:17:44.743 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading pydantic-1.10.2-py3-none-any.whl (154 kB)\n'}
2022-10-26 17:17:44.770 DEBUG image.builder.bg_svc [88099] {'stream': ' _____ 154.6/154.6 kB 5.9 MB/s eta 0:00:00'}
2022-10-26 17:17:44.770 DEBUG image.builder.bg_svc [88099] {'stream': '\n'}
2022-10-26 17:17:44.793 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting starlette==0.20.4\n'}
2022-10-26 17:17:44.799 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading starlette-0.20.4-py3-none-any.whl (63 kB)\n'}
2022-10-26 17:17:44.839 DEBUG image.builder.bg_svc [88099] {'stream': ' _____ 63.6/63.6 kB 1.4 MB/s eta 0:00:00'}
2022-10-26 17:17:44.839 DEBUG image.builder.bg_svc [88099] {'stream': '\n'}
2022-10-26 17:17:44.844 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: anyio<5,>=3.4.0 in /usr/local/lib/python3.10/dist-packages (from starlette==0.20.4->fastapi->r requirements.txt (line 46)) (3.6.2)\n'}
2022-10-26 17:17:44.846 DEBUG image.builder.bg_svc [88099] {'stream': 'Requirement already satisfied: six>=1.4.0 in /usr/local/lib/python3.10/dist-packages (from python-multipart->r requirements.txt (line 47)) (1.16.0)\n'}
2022-10-26 17:17:44.867 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting h11>=0.8\n'}
2022-10-26 17:17:44.871 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading h11-0.14.0-py3-none-any.whl (58 kB)\n'}
2022-10-26 17:17:44.884 DEBUG image.builder.bg_svc [88099] {'stream': ' _____ 58.3/58.3 kB 5.3 MB/s eta 0:00:00'}
2022-10-26 17:17:44.887 DEBUG image.builder.bg_svc [88099] {'stream': '\n'}
2022-10-26 17:17:44.901 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting click>=7.0\n'}
2022-10-26 17:17:44.918 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading click-8.1.3-py3-none-any.whl (96 kB)\n'}
2022-10-26 17:17:44.938 DEBUG image.builder.bg_svc [88099] {'stream': ' _____ 96.6/96.6 kB 5.0 MB/s eta 0:00:00'}
2022-10-26 17:17:44.938 DEBUG image.builder.bg_svc [88099] {'stream': '\n'}
2022-10-26 17:17:44.994 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting cachetools<6.0,>=2.0.0\n'}
2022-10-26 17:17:45.090 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading cachetools-5.2.0-py3-none-any.whl (9.3 kB)\n'}
2022-10-26 17:17:45.018 DEBUG image.builder.bg_svc [88099] {'stream': 'Collecting rsa<5,>=3.1.4\n'}
2022-10-26 17:17:45.023 DEBUG image.builder.bg_svc [88099] {'stream': ' Downloading rsa-4.9-py3-none-any.whl (34 kB)\n'}

```

2. 실행 이미지 생성 모듈 - 단독 사용을 위한 GUI 환경



컨테이너 이미지 빌드 서비스 전용 GUI

- 목표: CLI에서 docker build 명령으로 빌드하는 것보다 편리한 환경을 제공하는 것
- 학습용 GPU 서버팜 운영하는 고객의 가장 많은 문의 중 하나가 커스텀 이미지 빌드하는 방법

2. 실행 이미지 생성 모듈 - 단독 사용을 위한 GUI 환경

The image displays two side-by-side screenshots of the Backend.AI Forklift web interface. The left screenshot shows the 'CUSTOM' tab for 'Custom Image Build'. The right screenshot shows the 'LOGS' tab for the same build process.

Left Screenshot: Custom Image Build Interface

Header: Backend.AI Forklift, Welcome, te***@gmail.com, User Name: te***@gmail..., Logout

Left Sidebar: Summary, Builder (selected), Tasks, Environments

Form Fields:

- Source docker image*: [Empty]
- Target docker image*: [Empty]
- Minimum required CPU core(s)*: 1
- Minimum required memory size (minimum 64m)*: 64m
- Runtime type of the image*: python
- Path of the runtime*: python3
- Base LINUX distribution*: ubuntu
- Advanced Options: [Dropdown arrow]

Buttons: DOWNLOAD DOCKERFILE, PREVIEW DOCKERFILE, BUILD

Right Screenshot: Logs Interface

Header: Backend.AI Forklift, Welcome, te***@gmail.com, User Name: te***@gmail..., Logout

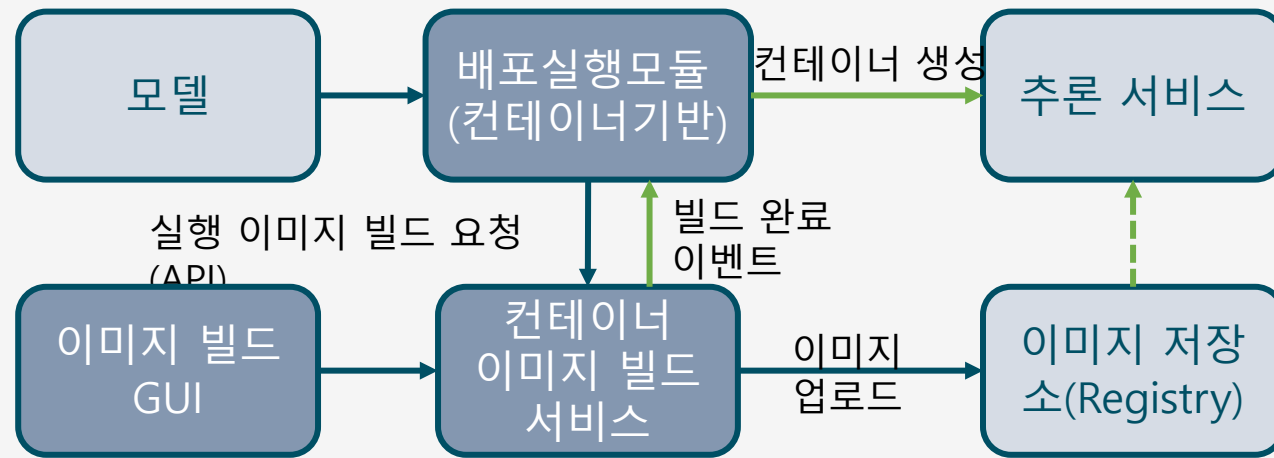
Left Sidebar: Summary, Builder, Tasks (selected), Environments

Buttons: REFRESH, CLEAR LOGS

Table: Logs of images being built

#	Message
1	Step 1/10 : FROM kmkwon94/torch:1.0v
2	Get:3 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
3	Get:4 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [83.3 kB]
4	Get:5 http://security.ubuntu.com/ubuntu bionic-security/universe amd64 Packages [1554 kB]
5	Get:6 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages [3035 kB]
6	Get:7 http://archive.ubuntu.com/ubuntu bionic/main amd64 Packages [1344 kB]
7	Get:8 http://security.ubuntu.com/ubuntu bionic-security/restricted amd64 Packages [1210 kB]
8	Get:9 http://security.ubuntu.com/ubuntu bionic-security/multiverse amd64 Packages [22.8 kB]
9	Get:10 http://archive.ubuntu.com/ubuntu bionic/restricted amd64 Packages [13.5 kB]
10	Get:11 http://archive.ubuntu.com/ubuntu bionic/universe amd64 Packages [11.3 MB]
11	Get:12 http://archive.ubuntu.com/ubuntu bionic/multiverse amd64 Packages [186 kB]
12	Get:13 http://archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [2332 kB]
13	Get:14 http://archive.ubuntu.com/ubuntu bionic-updates/multiverse amd64 Packages [37.1 kB]
14	Get:15 http://archive.ubuntu.com/ubuntu bionic-updates/restricted amd64 Packages [1256 kB]
15	Get:16 http://archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [3467 kB]
16	Get:17 http://archive.ubuntu.com/ubuntu bionic-backports/universe amd64 Packages [20.6 kB]
17	Get:18 http://archive.ubuntu.com/ubuntu bionic-backports/main amd64 Packages [64.0 kB]
18	Fetchd 26.4 MB in 13s (2106 kB/s) Reading package lists...
20	

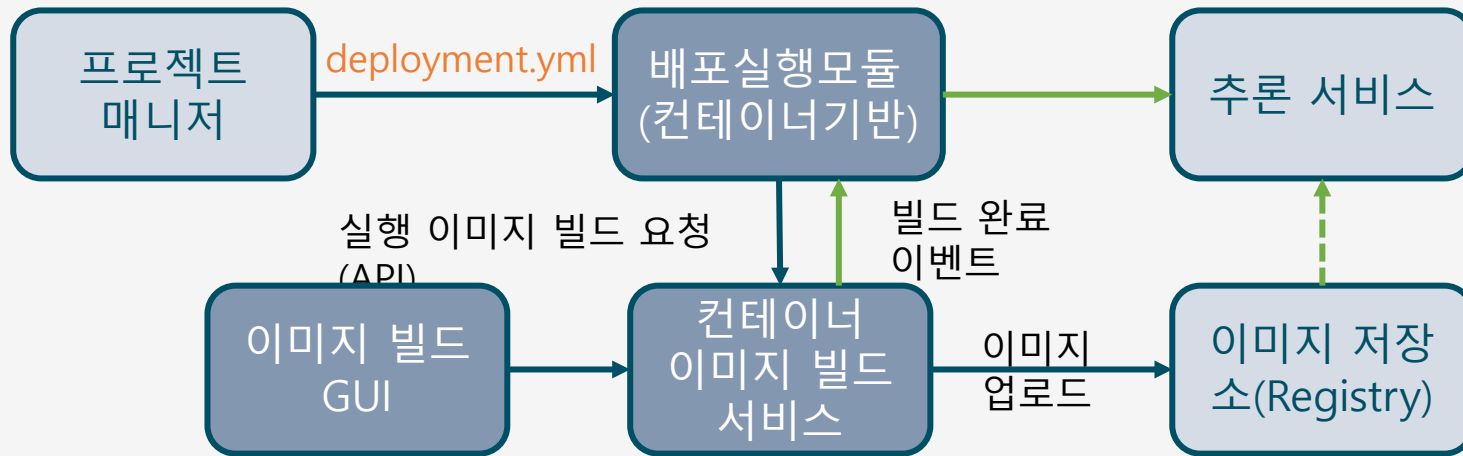
3. 배포/실행 모듈



Python 기반의 경량 서버 모듈

- 실행 이미지 생성 요청 (→ 이미지 빌드 서비스)
- 이미지 생성 완료 확인 후 이미지를 사용하여 추론 배포 컨테이너 실행 (→ aiodocker → docker)
- 추론 배포 컨테이너 실행 후 이벤트를 Project Manager로 전송
- Project Manager에서 사용자, 프로젝트 별 추론 컨테이너 상태를 조회할 수 있는 API 제공

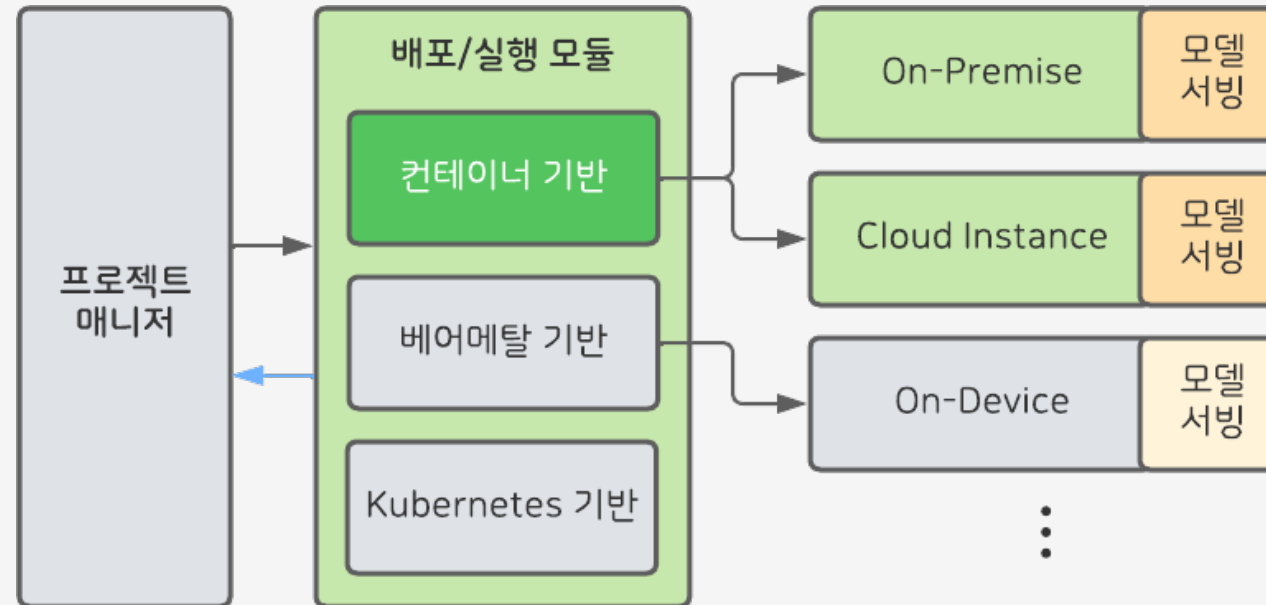
3. 배포/실행 모듈 - 선행 모듈과의 통신 인터페이스 정의



클라이언트(프로젝트 매니저)로부터
배포/실행 관련 정보를 받을 때 그 형식을
deployment.yml 파일로 단일화

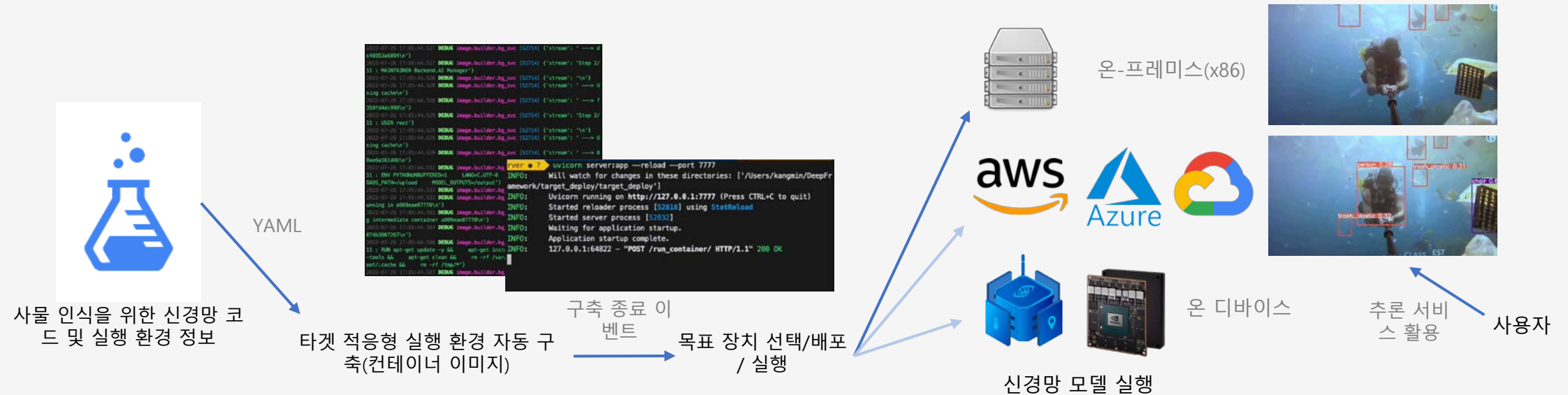
```
build:
  ...
  architecture: linux/amd64
  components:
    custom_packages:
      apt:
        - vim
        - python3.9
      pypi:
        - flask==1.2.3
        - torch>=1.1.0
  ...
  deploy:
    ...
    entrypoint: ["python", "/app/serve.py"]
    mount:
      - src: /host/path/myapp
        dst: /app
    network:
      ...
```

3. 배포/실행 모듈 - 비-컨테이너 배포 모듈 확장 가능 구조



프로젝트 매니저와의 통신 인터페이스
통일을 통해 베어메탈 등 비-컨테이너
배포 모듈 확장

3. 배포/실행 모듈 – 컨테이너 기반 실행 환경 자동 배포



데모 영상: <https://d.pr/v/1udkU5>

감사합니다.