



신경망 배포탑재 기술

인공지능 기술의 대중화
(AI Democratization)를 위한
제2회 탱고 커뮤니티 컨퍼런스

성명 이경희

소속 한국전자통신연구원

후원



주관



주최



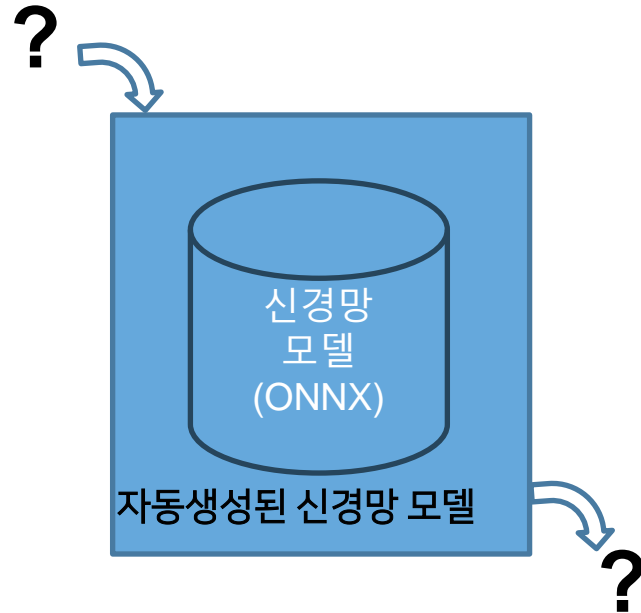


목 차

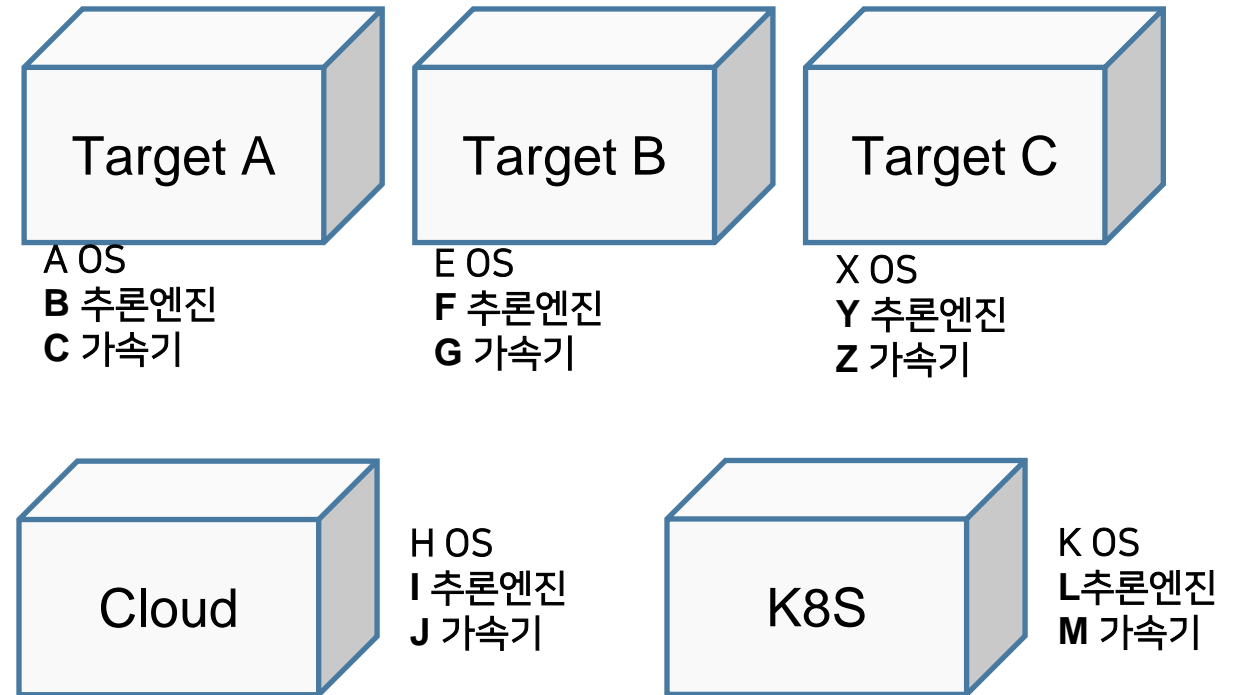
I	배포탑재 기술의 개요	03
II	구현 현황	09
III	향후 계획	23

배포탐재 기술의 필요성

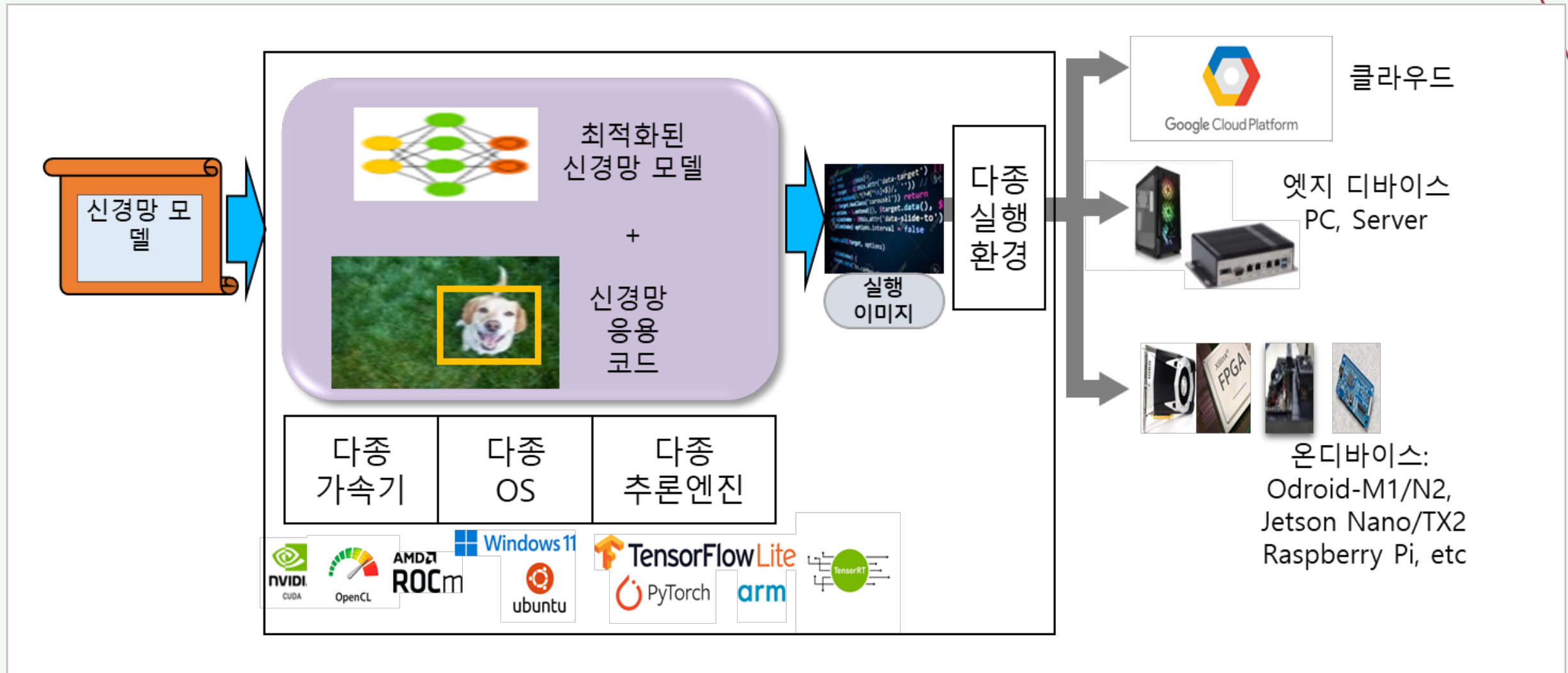
- 입력은 어떻게 주어야 하나?
- 출력은 무엇인가? 어떻게 사용?



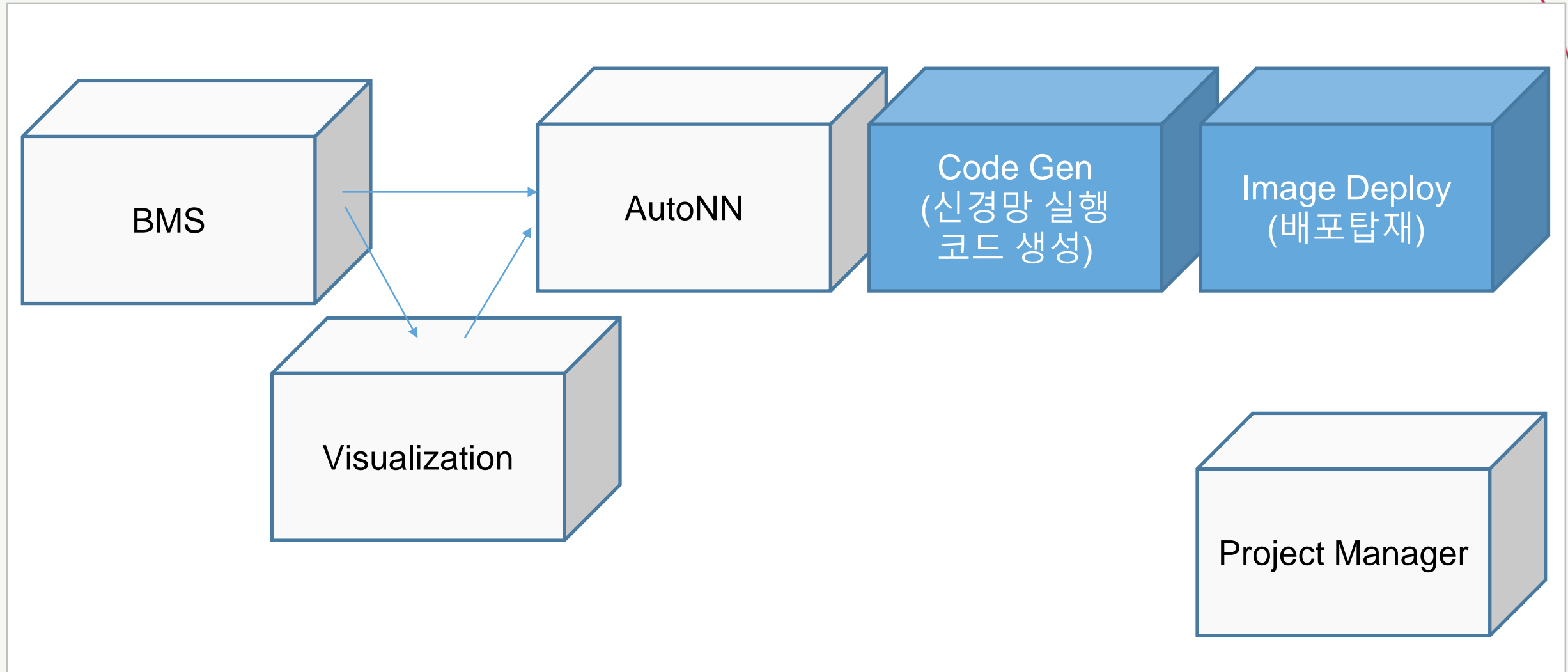
- 타겟마다 실행 환경이 다른데 어떻게?
- 클라우드 등에서 실행시키려면 어떻게?



신경망 통합개발 프레임워크 (2/3)



신경망 통합개발 프레임워크 (3/3)





다양한 배포환경 지원

○ 신경망 가속기의 다양성

- x86, ARM 등의 CPU
- NVIDIA GPU, ARM Mali 등 GPU
- 기타 NPU 등

○ 다양한 추론엔진 지원

- PyTorch, NVIDIA TensorRT
- TensorFlow, TensorFlow Lite
- ARM ACL(Arm Compute Library), Apache 재단의 TVM



타겟 디바이스 인지형 신경망 응용 생성 및 배포

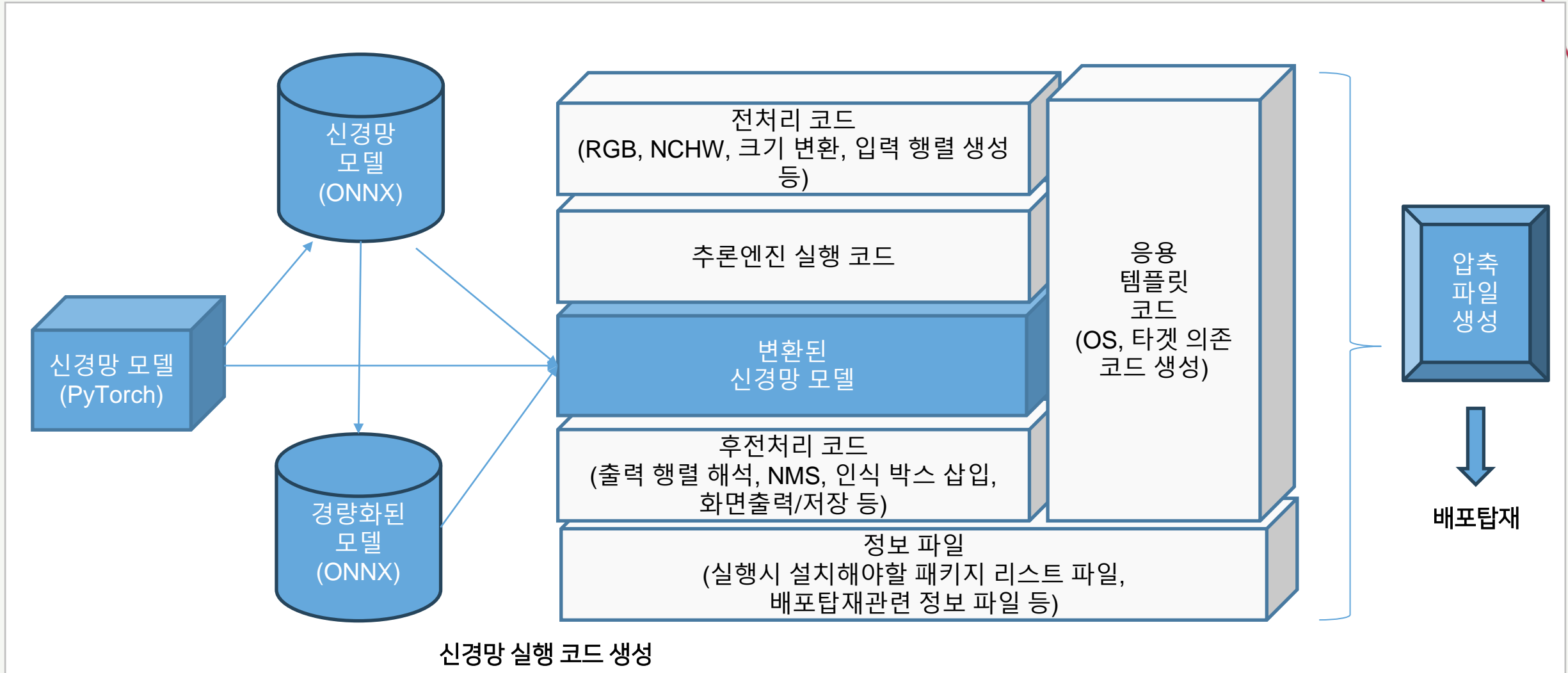
○ 타겟 디바이스 맞춤형 신경망 응용 생성 지원

- 추론엔진 의존적 실행 코드 자동 생성
- 신경망별 맞춤형 신경망 입출력 코드 자동 생성

○ 배포 및 탑재의 편의성 제공

- cloud, K8S상 배포 지원 기능
- docker, 실행 파일 자동 생성 기능

배포탐재의 동작



배포탑재 관련 소스 코드 구조





폴더명			모듈명
/TANGO	deploy_codegen	optimize_codegen	신경망 실행 코드 생성 모듈
	deploy_targets	cloud	클라우드 배포 모듈
		k8s	K8s 배포 모듈
		ondevice	온디바이스 배포 모듈



신경망 실행 코드 생성 모듈 - 기능

- **신경망 생성 모델의 실행을 위한 전처리/후처리 코드 생성**
 - 전처리: 이미지 resize/crop, 평균치 조정, 신경망 입력용 텐서 생성(NCHW)
 - 후처리: 신경망 모델의 출력 해석 (인식 객체명, 확률값)
- **신경망 실행을 위한 입출력 코드 생성**
 - 입력: 파일/동영상/카메라 입력 코드 생성
 - 출력: 동영상, 화면, 텍스트 등으로 결과 출력 코드 생성
- **배포탑재 및 실행을 위한 정보 파일 해석 및 생성**
 - AutoNN과 Project manager에서 생성한 yaml파일 해석
 - 클라우드/엣지클라우드/온디바이스상 신경망 배포 실행을 위한 yaml 파일 생성

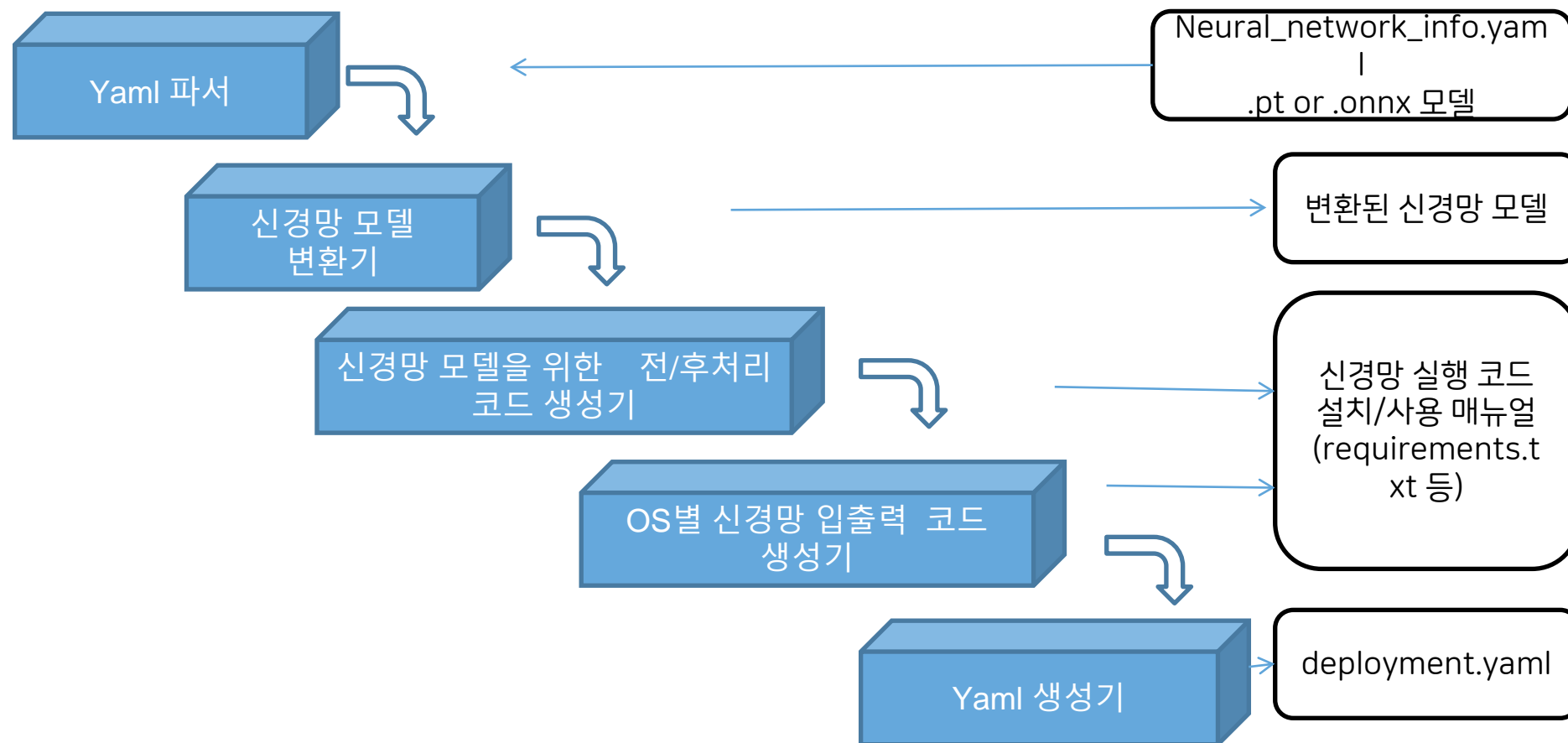
신경망 실행 코드 생성 모듈 - 지원 현황

Target Environment	Device Environment		Runtime engine
Cloud Environment 	GCP(Google Cloud Platform)		PyTorch
Kubernetes Environment 	x86 + NVIDIA GPU	PC	PyTorch
	ARM CPU + NVIDIA GPU	Jetson Nano	TensorRT
	ARM CPU + NVIDIA GPU 	Jetson AGX Orin	TensorRT, PyTorch
		Jetson AGX Xavier	TensorRT, PyTorch
		Jetson Nano	TensorRT, PyTorch
	ARM CPU + Adreno GPU	S22	Tensorflow Lite(Android)
OnDevice Environment 	ARM CPU + Mali GPU	Odroid-N2	TVM, ACL

신경망 실행 코드 생성 모듈 - 구조



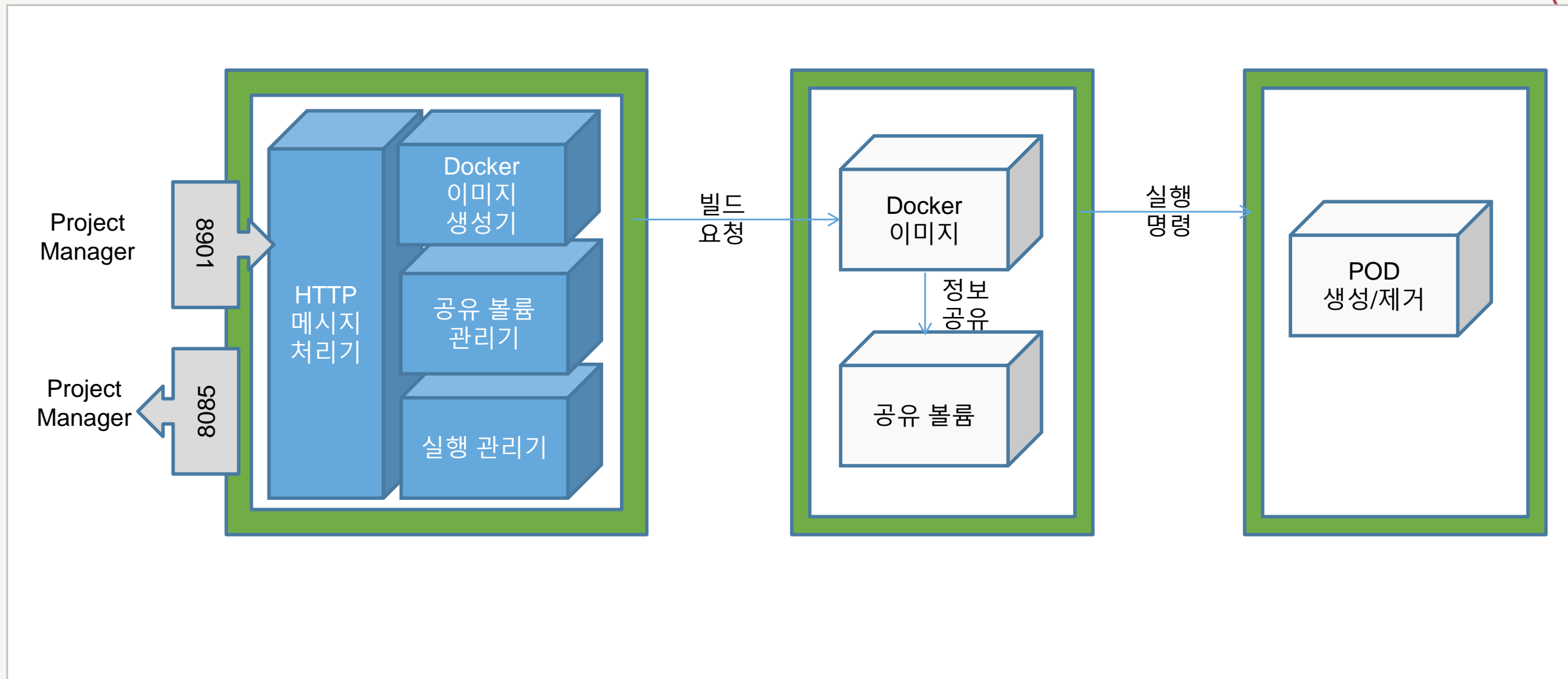
신경망 실행 코드 생성 모듈 - 동작 흐름



신경망 실행 코드 생성 모듈 - deployment.yaml의 예

```
build:
  architecture: x86
  accelerator: cpu
  os: ubuntu
  components:
    engine: pytorch
    libs: [python==3.9, torch>=1.1.0]
    custom_packages:
      apt: # with apt command
        - vim
      pypi: # pip command
        - flask==1.2.3
  deploy:
    type: docker #or native
    work_dir: /test/test
    pre_exec: [['tensorrt-converter.py', param1, param2], ['hello.py']]
    entrypoint: [run.sh, -p, "opt1", "arg"]
    network:
      service_host_ip: 1.2.3.4
      service_host_port: 8088
  k8s:
    nfsip: 192.168.0.189
    nfspath: /var/lib/docker/volumes/tango_shared/_data
```

K8s 배포 모듈 - 구조



온디바이스 배포 모듈 - 기능

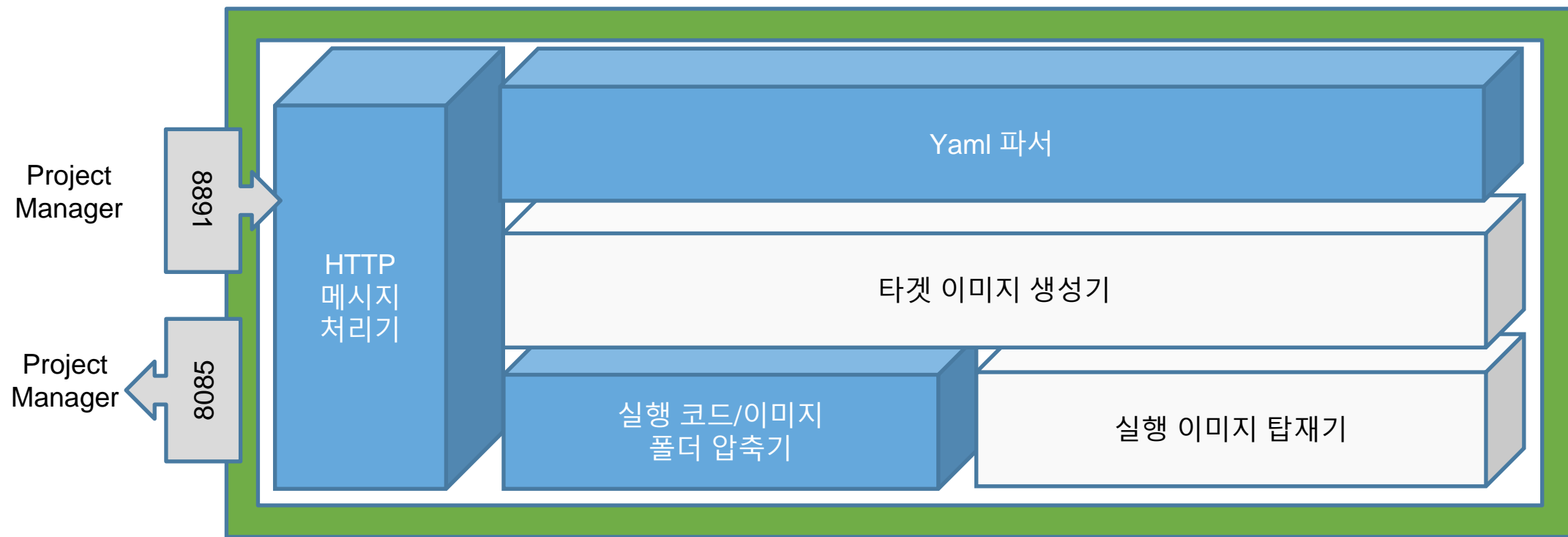
○ 배포탑재 및 실행을 위한 정보 파일 해석

- 신경망 실행 코드 생성 모듈에서 전달받은 deployment.yaml 파일 해석

○ 신경망 실행 코드 전달

- 신경망 실행 코드 압축
- 설치 및 사용 매뉴얼 제공

온디바이스 배포 모듈 - 구조





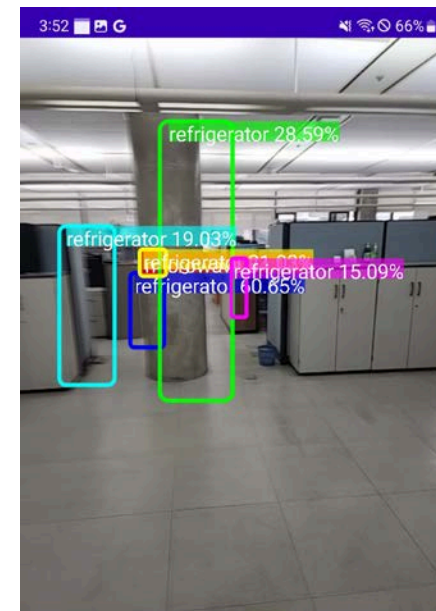
실행 화면 - 프로젝트 매니저에서 실행

The screenshot displays the TANGO Project Manager interface. On the left is a dark sidebar with the TANGO logo and navigation links: Project Management, Target Management, Data Management, and Visualization. The main content area is titled 'classification_on_PC' and includes a 'Deploy Config' section with fields for 'Light Weight Level' (5), 'Precision Level' (5), 'User Editing' (No), 'Input Source' (/images), and 'Output Method' (0). Below this is a 'Progress' section with two buttons: 'Manually Generation of Neural Networks' and 'Automatic Generation of Neural Networks'. A progress bar shows the status 'Progress - VISUALIZATION completed'. The progress bar consists of five chevron-shaped steps: BMS, VISUALIZATION, Auto NN, Code Gen, and Image Deploy. The 'Code Gen' and 'Image Deploy' steps are highlighted with a red rectangular box. Below the progress bar is a link to 'Open VISUALIZATION' and a 'Log' section.



갤럭시 S22용 객체 인식 응용 생성 및 실행

- 신경망 실행 코드 생성 모듈
 - PyTorch -> ONNX 변환
 - OpenVINO 모델로 변환
 - TensorFlow 모델로 변환
 - TensorFlow Lite 모델로 변환
 - 안드로이드용 코드 생성
(이미지 입력, 전처리 코드, 후처리 코드, NMS 코드, 출력 코드 등)
 - 안드로이드용 응용 생성
- ↓
- 온디바이스 배포 모듈
 - 안드로이드 응용 프로그램 다운로드
- ↓
- 갤럭시 S22 스마트폰: 응용 프로그램 설치 후 실행



BaseModel: yoloe, RunMode: NONE_FP32, InputSize: 640
FPS: 6.17
inference: 87ms postprocess: 3ms
Confidence Threshold: 0.25
IoU Threshold: 0.45



NVIDIA AGX Orin용 객체 인식 응용 생성 및 실행

- 신경망 실행 코드 생성 모듈
 - PyTorch -> ONNX 변환
 - TensorRT용 실행 코드 생성
(ONNX2TensorRT 모델 변환 코드,
이미지 입력, 전처리 코드, 후처리 코드, NMS 코드, 출력 코드 등)

↓
- 온디바이스 배포 모듈
 - TensorRT용 응용 코드 다운로드

↓
- NVIDIA AGX Orin
 - 실행시 필요한 패키지 설치
 - TensorRT용 응용 실행
(ONNX2TensorRT 모델 변환, 객체 인식 기능 실행)



PC용 Classification 응용 생성 및 실행

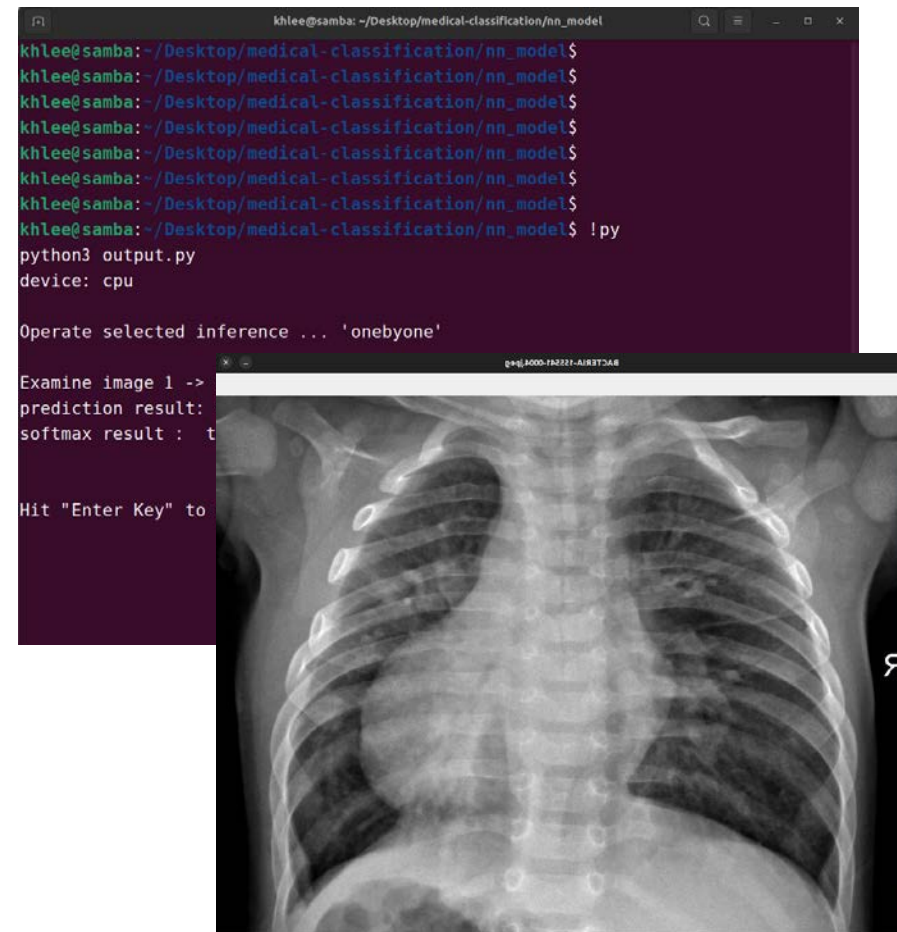
- 신경망 실행 코드 생성 모듈
 - PyTorch 모델 입력
 - PyTorch용 실행 코드 생성
(이미지 입력, 전처리 코드, 후처리 코드, 출력 코드 등)



- 온디바이스 배포 모듈
 - PyTorch용 응용 코드 다운로드



- PC
 - 실행시 필요한 패키지 설치
 - PyTorch 응용 실행





K8S를 통한 응용 배포 및 실행

○ 신경망 실행 코드 생성 모듈

- PyTorch 기반 신경망 실행 코드 생성
- 타겟 디바이스의 IP 정보 등



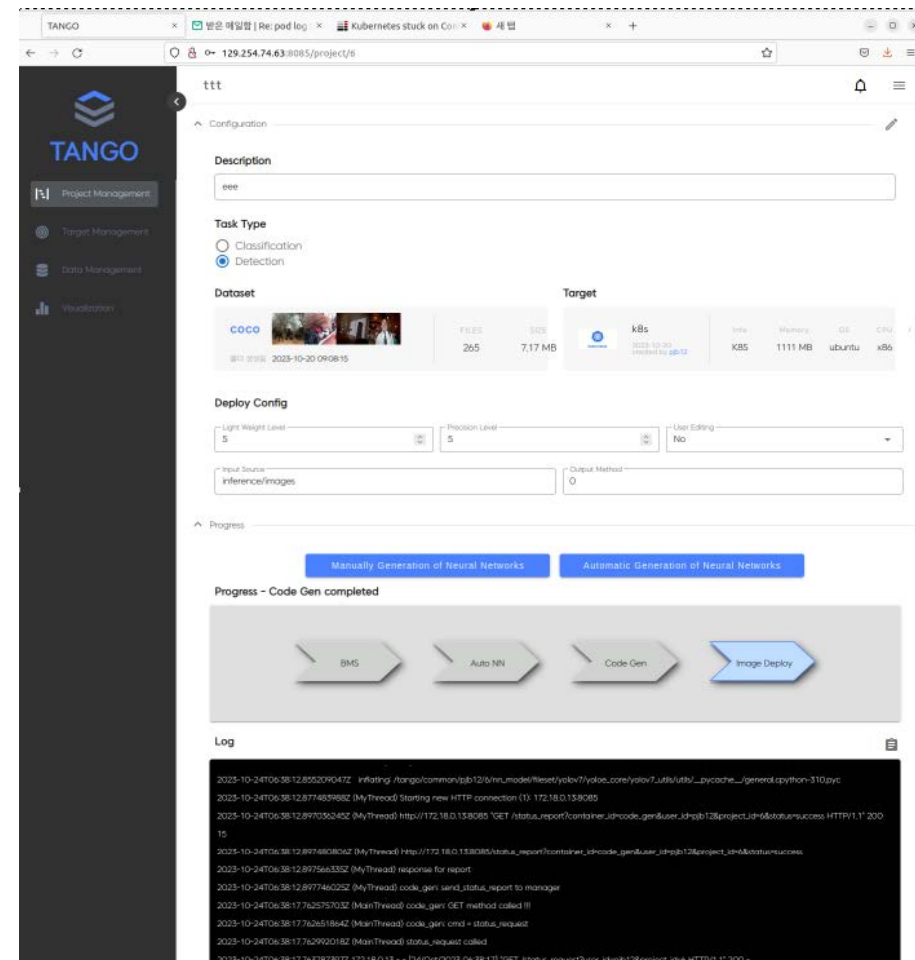
○ K8S 배포 모듈

- 원격 타겟디바이스에 쿠버네티스 환경 구축
- 원격 타겟디바이스에 신경망 구동 환경 구축
- 원격에서 신경망 실행 명령 전달



○ 타겟 시스템

- 신경망 실행
- 원격시스템에서 신경망 구동 결과 확인 가능



배포탑재 기능 확대

- 지원 타겟 환경 확대
 - 가속기 지원 확대
 - 추론엔진 지원 확대
- 분산 실행 환경 지원
 - 신경망 학습
 - 신경망 추론
- 코드 생성 범위 확대 및 외부 IDE 연동 지원
 - 사용자가 원하는 수준의 코드 자동생성 및 배포/탑재 방식 지원

감사합니다.

T

A

N

감사합니다.

