

Kickoff Meeting

이종 HW 적응적인 모빌리티-SDI 분업형 SW 개발을 위한 지속적 통합/검증/배포 기술 개발 계획

In-Young Ko
iko@kaist.ac.kr

Web Engineering and Service Computing Lab,
Korea Advanced Institute of Science and Technology (KAIST)

2025. 04. 15

OUTLINE

1. Introduction

2. 1차년도 연구

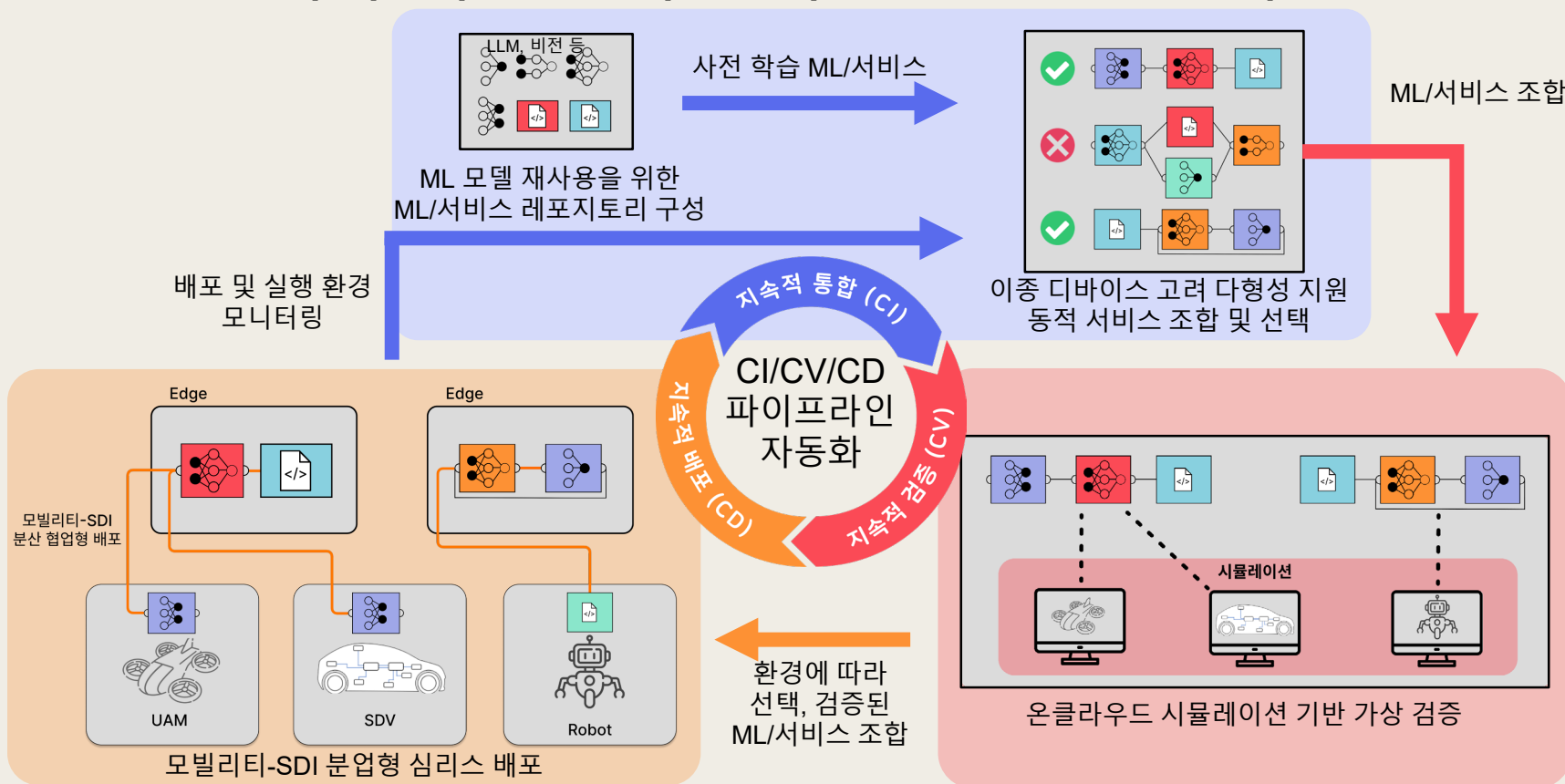
- CI, CV, CD

3. 2차년도 연구 목표

- CI, CV, CD
- 정량적 목표

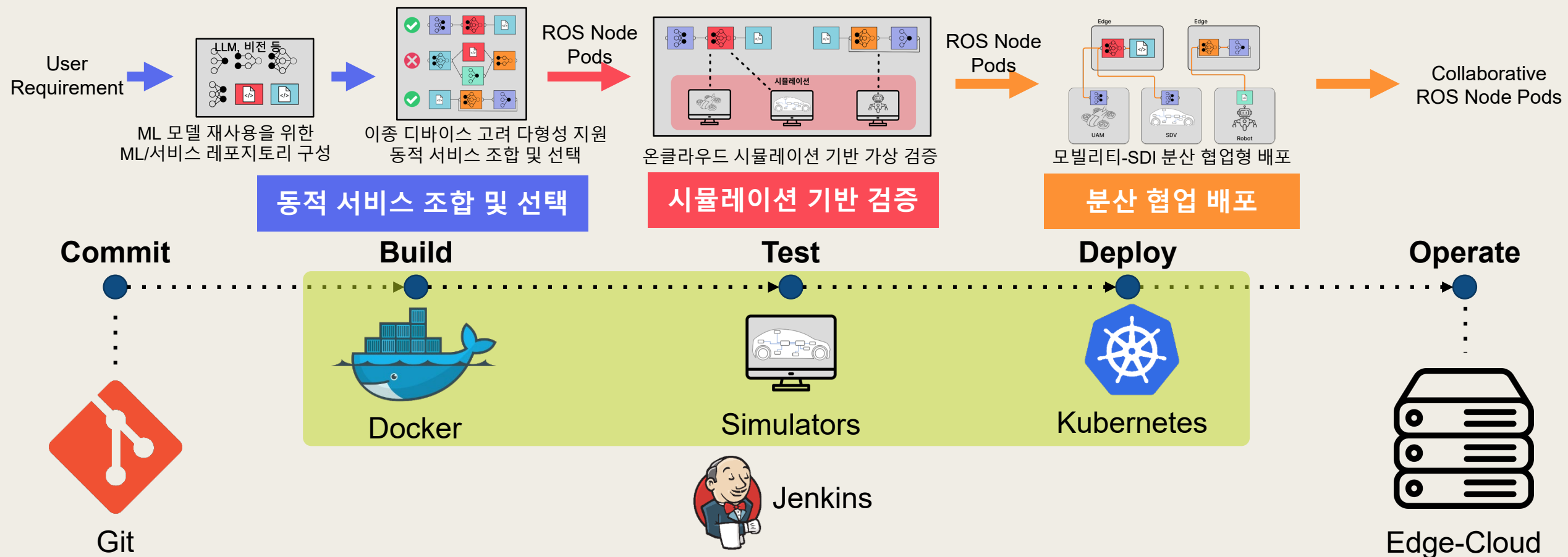
이종 HW 적응적인 모빌리티-SDI 분업형 SW 개발을 위한 지속적 통합/검증/배포 기술


- 이종 디바이스 적응적인 다형적 모빌리티 SW 개발을 위한 동적 서비스 조합 및 선택 기술
 - 모빌리티를 위한 AI 기반 SW 대상 기술 연구
- 소프트웨어 정의형 모빌리티 품질 보증을 위한 온클라우드 가상 검증 기술
- 모빌리티-SDI 분업형 SW의 지속적 배포를 위한 심리스 배포 및 업데이트 기술



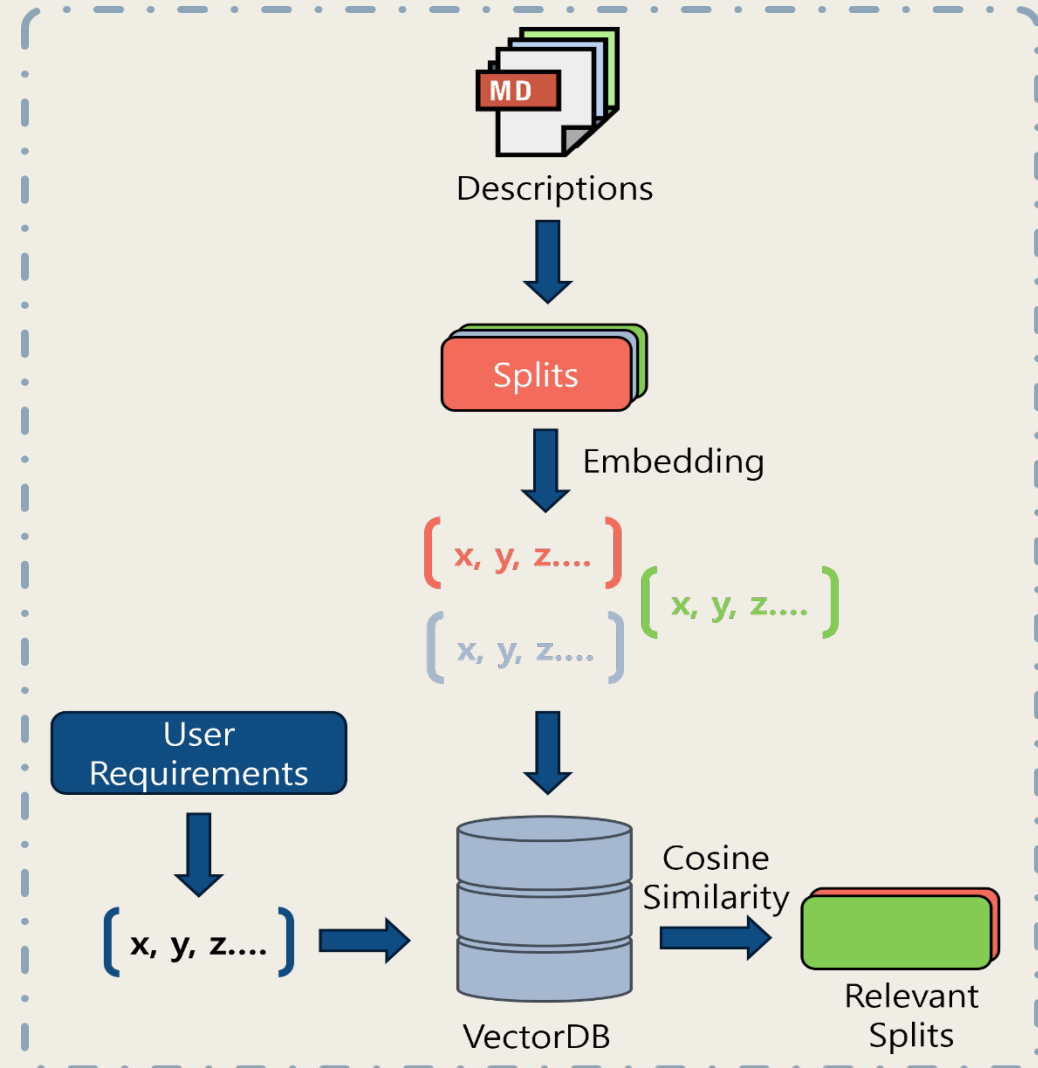
- 지속적 통합(CI), 검증(CV), 배포(CD)

- 시스템의 개발, 테스트, 배포를 통합해 배포 주기를 단축



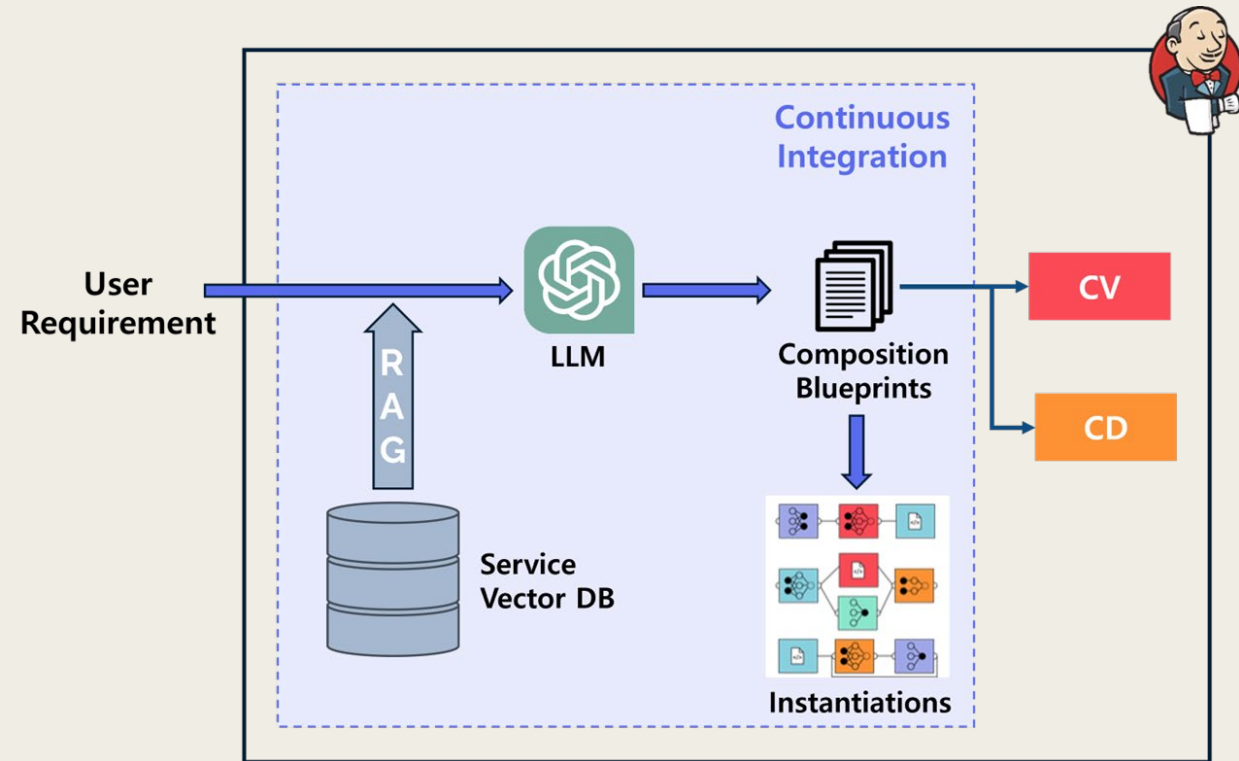
년차	CI 목표	CV 목표	CD 목표	Delievable
1년차 ('24)	ML/서비스 프로토타입 <u>레포지토리</u> 구축	시뮬레이션 기반 검증 기술 연구 & <u>Jenkins 플러그인</u> 개발	분산 협업 배포 <u>아키텍처</u> 조사 및 연구	프로토타입 도구  최종 도구
2년차 ('25)	모니터링 데이터 기반 ML/서비스 조합 기술 연구	SDX의 시뮬레이션 기반 검증 기술 효율성 고도화	SDX-SDI 협업형 배포 기술 연구	
3년차 ('26)	이종 디바이스 고려 ML/서비스 선택 기술 연구			
4년차 ('27)	재사용성 고려 ML/서비스 레포지토리 구축	SDA, SDR 등 다양한 모빌리티로 시뮬레이션 기반 검증 확대	다양한 환경에서의 협업형 배포 안정성 증대 연구	

- Hugging Face 및 기타 서비스 레포지토리에서 Markdown 형식으로 제공되는 서비스 **description**을 수집
 - Embedding 모델을 활용하여 Markdown 내용을 고차원 벡터로 변환
- 벡터로 변환된 서비스 **description**들을 저장하는 ML/서비스 레포지토리 구축
 - 벡터 검색 최적화를 위해 적절한 인덱싱 기법 적용



md 파일을 레포지토리에 저장하는과정

- 사용자 요구사항 문서 및 레포지토리에서 서비스 정보를 LLM으로 전달
 - Retrieval Augmented Generation (RAG)를 이용하여 요구사항과 맞춤 서비스 정보 전달
- Prompt engineering을 활용하여 요구사항을 composition blueprint로 전환[2]
 - Blueprint는 CI, CV, CD 모든 단계에서 사용
- 위 모든 과정은 CI-CV-CD Jenkins 파이프라인을 활용하여 자동화



CI 단계에 중점을 둔 CI-CV-CD 파이프라인의 개요

지속적 시뮬레이션 기반 검증

● 다양한 서비스 조합 검증 필요

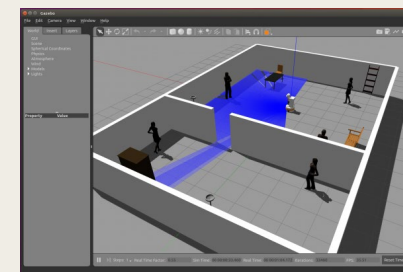
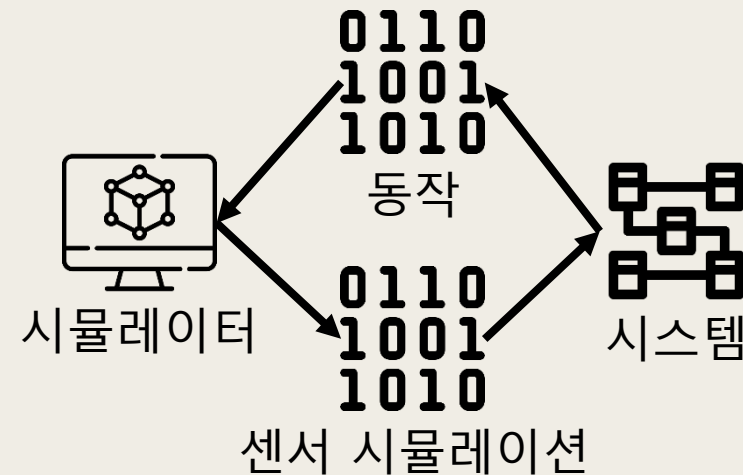
- 효율적으로 다양한 서비스 조합을 검증해야 함
- ➔ 필드 테스트 대체 시뮬레이션 기반 테스트 활용

● 시뮬레이션 기반 테스트

- 단순 입력 대신, 시나리오를 통해 장시간의 동작 확인
(시나리오: 환경과 시스템의 초기 상태 및 시스템 목표)
- 시뮬레이터와 대상 시스템 간 loop에서 발생하는 상호작용 확인

● Jenkins Plugin 형태로 CV를 위한 시뮬레이션 기반 검증 도구 개발

- Input: 서비스 조합, Scenarios
- Output: Simulation Results



Gazebo Simulator

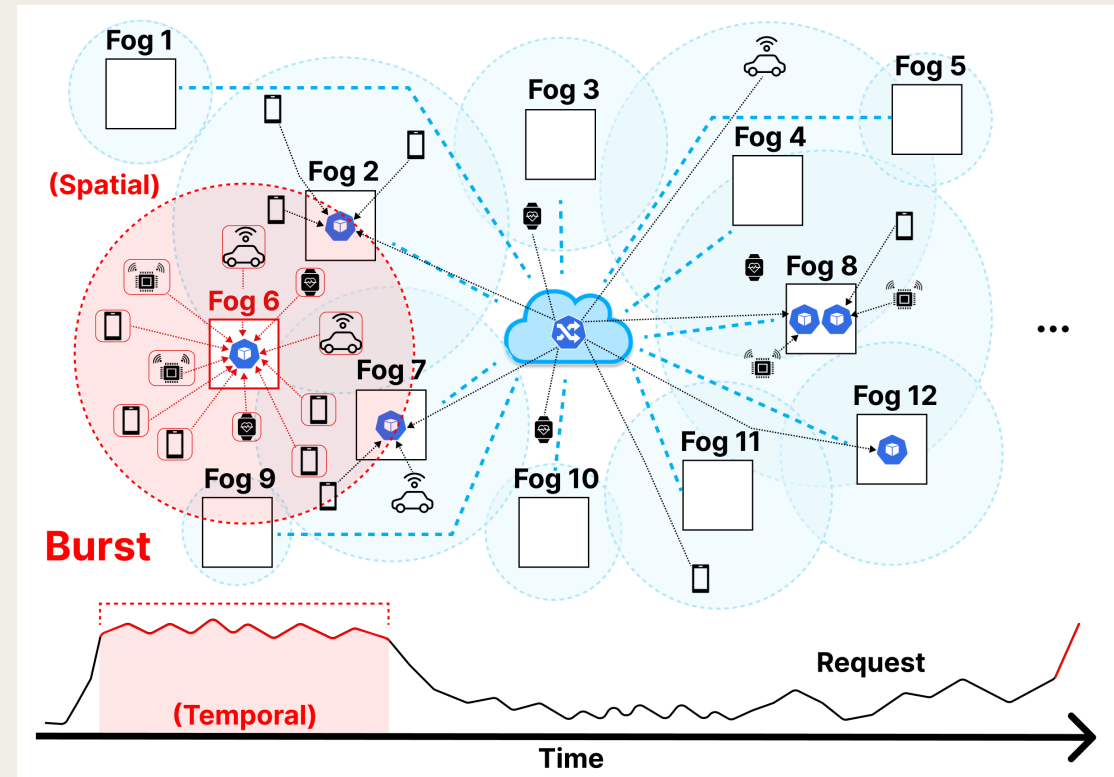


Carla (SDV Simulator)

● SDI → 엣지-포그-클라우드 환경

- 많은 수의 분산된 포그 노드 존재
- 각 디바이스의 이종성으로 인한 문제
- 시공간적 요청 패턴 발생

➔ 디바이스와 시공간적 특성을 모두 고려한 효율적인 배포 아키텍처 및 전략 필요



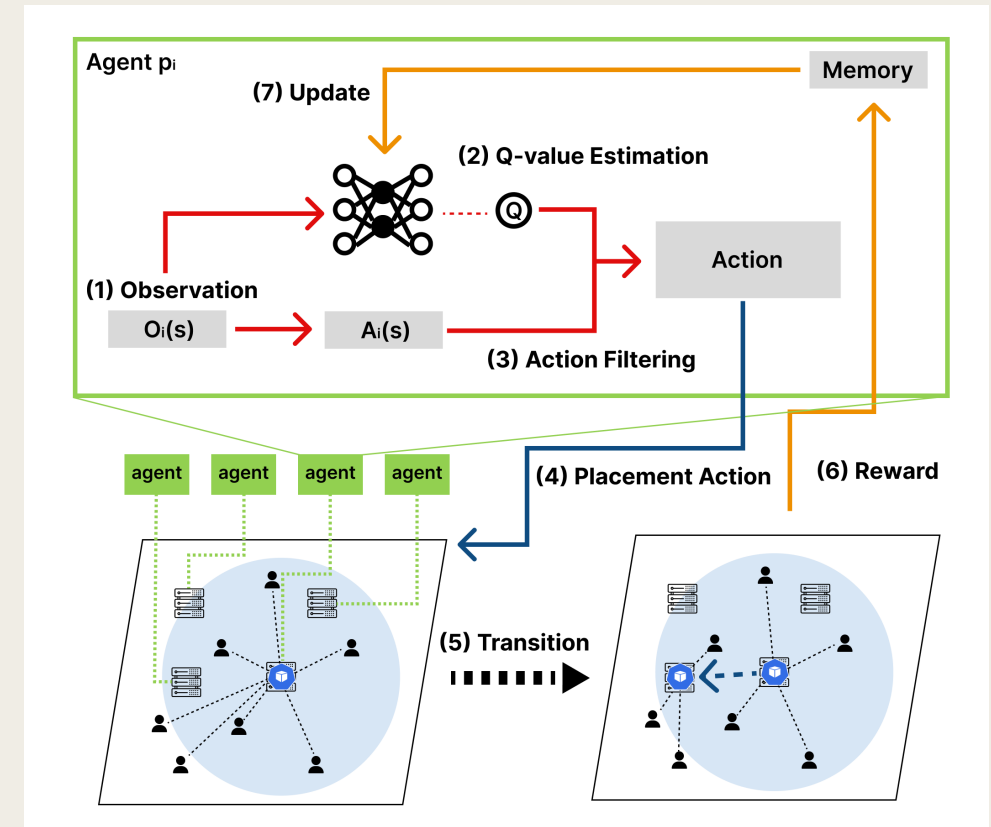
시공간적 Traffic Burst에 따른 서비스 배포 문제 개요

- 멀티 에이전트 강화학습 기반 분산 협업 배포

- 각 디바이스의 이종성으로 인한 특성 학습
- 환경에서 발생하는 시공간적 특성 학습
- ICWS 2024에서 관련 논문 발표 [1]

- 효율적인 분산 배포 전략

- 중앙 서버 없이 지역별 에이전트끼리 서로 협력
- 분산-협력을 통한 지역별 최적의 배포 전략 도출



멀티 에이전트 강화학습 기반 협업적 서비스 배포 아키텍처

[1] Eunchan Park, Kyeongdeok Baek, and In-Young Ko, "HiDRA: A Hierarchical Decentralized Reactive Autoscaler for Spatio-temporal Bursts of Load", In *Proceedings of the 2024 IEEE International Conference on Web Services (ICWS 2024)*, July 7-13, 2024, Shenzhen, China

년차	CI 목표	CV 목표	CD 목표	Delievable
1년차 ('24)	ML/서비스 프로토타입 <u>레포지토리</u> 구축	시뮬레이션 기반 검증 기술 연구 & <u>Jenkins 플러그인</u> 개발	분산 협업 배포 <u>아키텍처</u> 조사 및 연구	<div>프로토타입 도구</div> <div>↓</div> <div>최종 도구</div>
2년차 ('25)	모니터링 데이터 기반 ML/서비스 조합 기술 연구	SDX의 시뮬레이션 기반 검증 기술 효율성 고도화	SDX-SDI 협업형 배포 기술 연구	
3년차 ('26)	이종 디바이스 고려 ML/서비스 선택 기술 연구			
4년차 ('27)	재사용성 고려 ML/서비스 레포지토리 구축	SDA, SDR 등 다양한 모빌리티로 시뮬레이션 기반 검증 확대	다양한 환경에서의 협업형 배포 안정성 증대 연구	



동적 서비스 조합 및 선택

시뮬레이션 기반 검증

분산 협업 배포

Commit

Build

Test

Deploy

Operate



Git



Docker



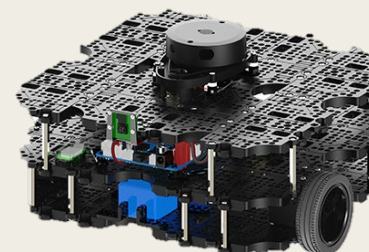
Gazebo Sim



K8s



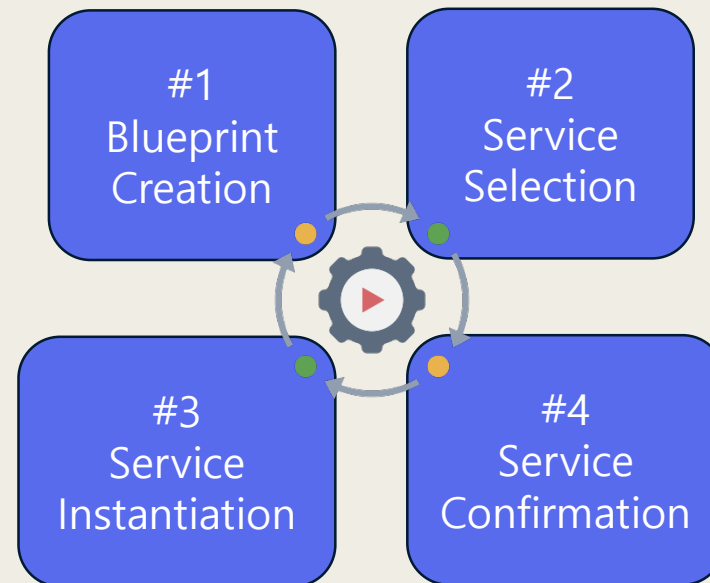
Jenkins



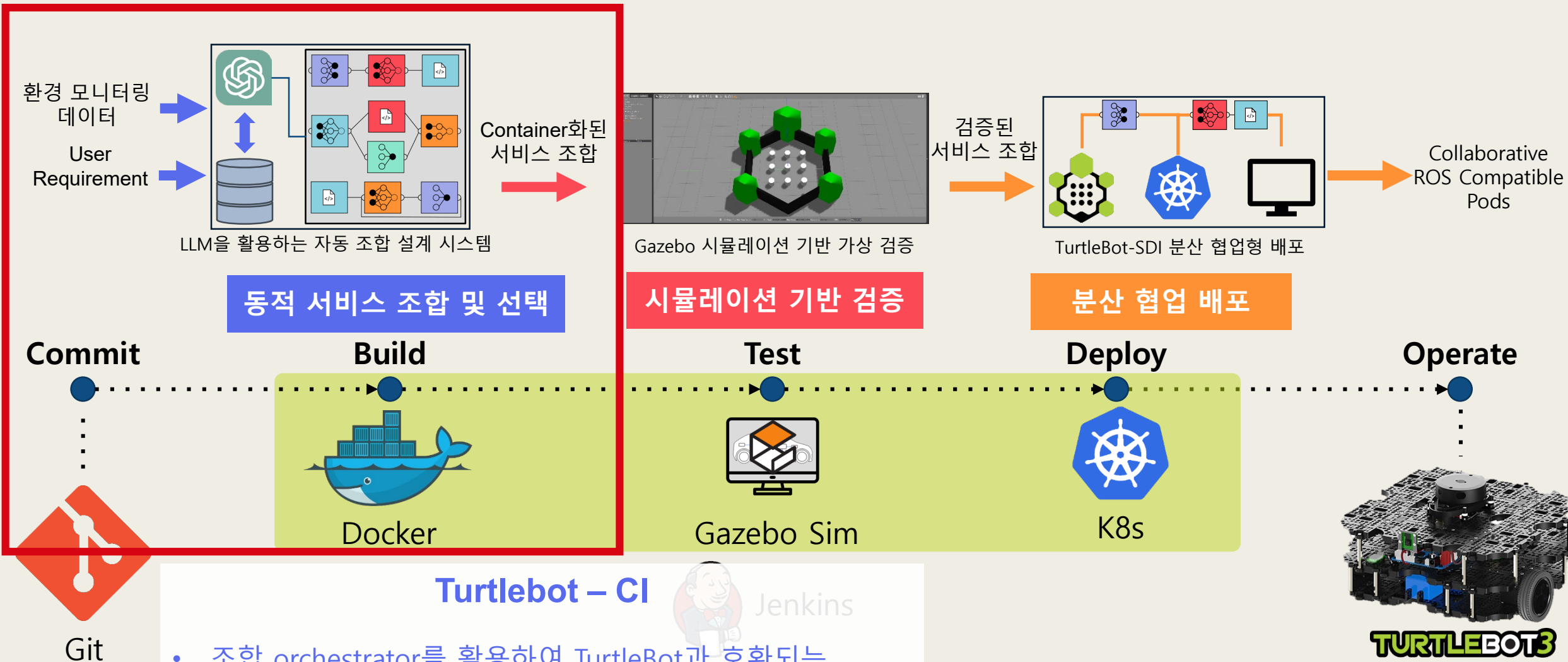
TURTLEBOT3

배송 로봇

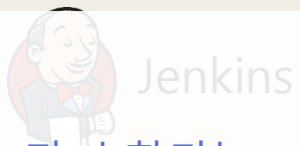
- 조합 생성 과정을 관리하는 조합 오케스트레이터 개발
 - 조합 과정의 모든 단계를 조정하여 각 서비스 출력들을 확인
- HuggingGPT^[1]의 LLM 기반 조합 오케스트레이터 연구를 활용하여 SDI환경에서 적용
 - TurtleBot에서 성능 및 기타 환경 데이터를 수집
 - 환경 데이터를 오케스트레이터와 함께 활용하여 조합 품질을 더욱 향상할 것으로 예상



조합 오케스트레이터 및 조합 절차 단계



Turtlebot – CI



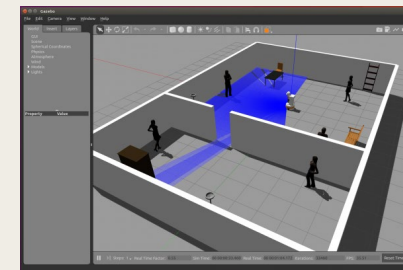
- 조합 orchestrator를 활용하여 TurtleBot과 호환되는 서비스 조합들을 인스턴스화
- 서비스 조합의 각 구성 서비스의 자동 컨테이너화

- **Scenario간 Repair Incompatible(RI)을 고려한 시뮬레이션 검증 기법 개발**

- RI Pair: 동시에 Safety를 만족하기 어려운 시나리오 Pair
- RI Pair를 측정할 수 있는 점수 계량화 기법 개발
- 계량화 결과 기반 RI Scenario Space를 효율적으로 검색하는 기법 개발
- RI Scenario Space를 활용한 효율적 시나리오 검색 기법 개발

- **CV를 위한 시뮬레이션 기반 검증 도구 개발**

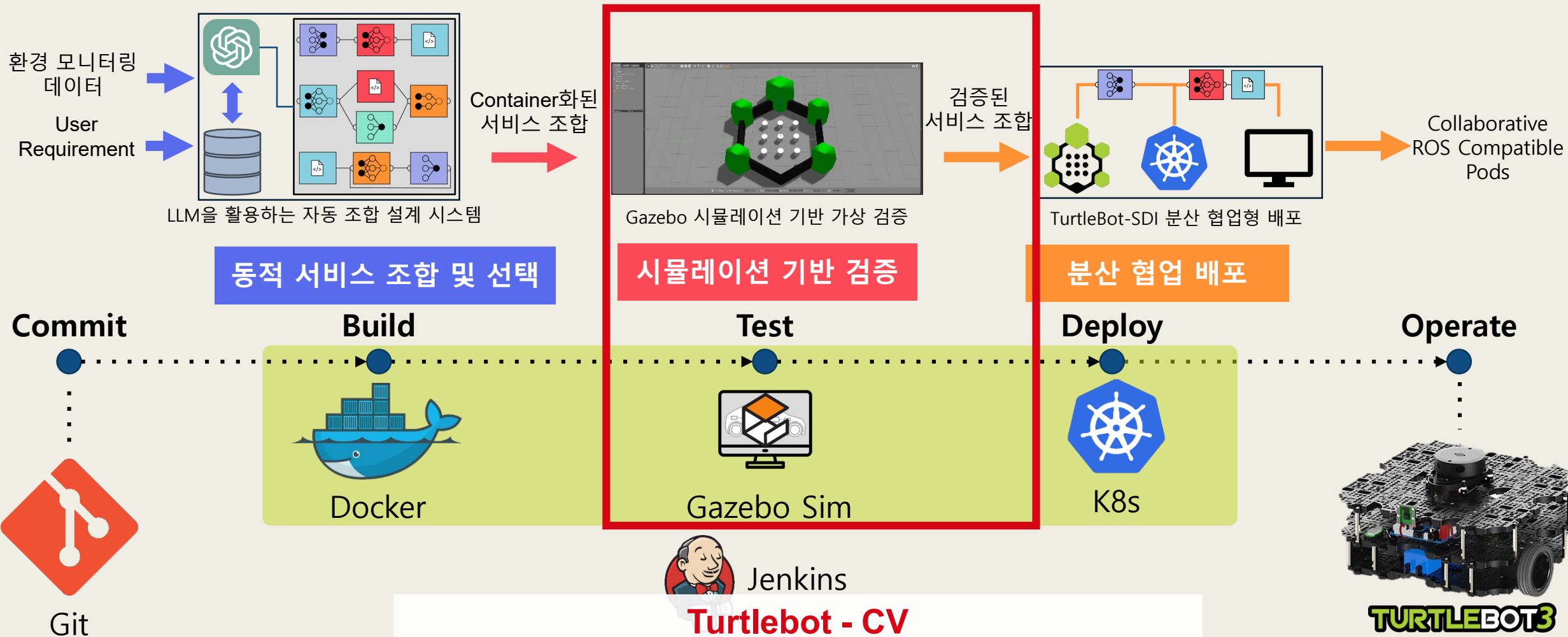
- Gazebo, Carla 등 SDx를 위한 시뮬레이터를 활용하여 요구사항에 맞게 시나리오를 생성하는 검증 도구 개발
- Turtlebot의 시나리오 영역 설정
 - 도로의 형태, 목표 지점, 주행 상황 등



Gazebo Simulator



Carla (SDV Simulator)



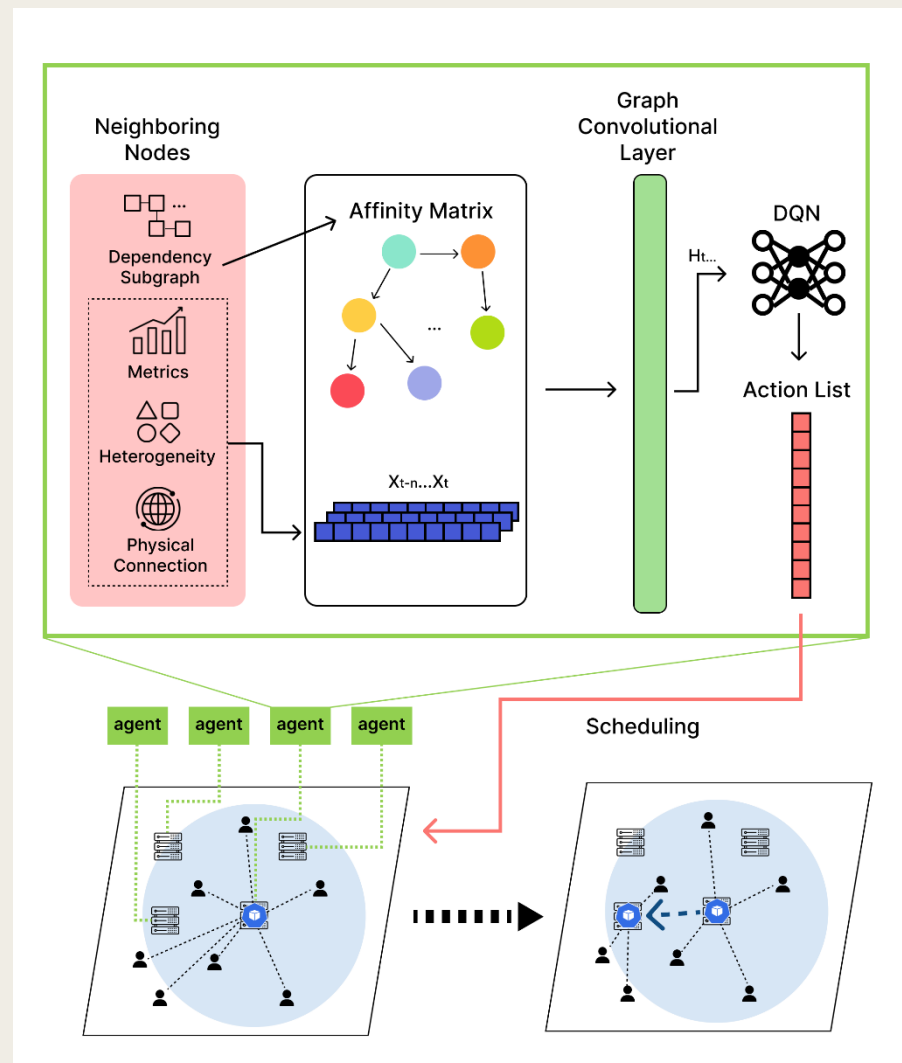
- Turtlebot의 시나리오 영역 설정
- Turtlebot과 Gazebo를 활용한 시뮬레이션 기반 테스트 시행

● 엣지-클라우드에서의 멀티-에이전트 분산-협업 배포 전략

- 서비스 종속성, 물리적 종속성, 노드의 이종성 등을 GNN을 활용하여 학습
- 환경에 분산되어 있는 에이전트들이 협업하여 최적화된 배포 전략 도출

● CD를 위한 엣지-클라우드 테스트베드 구축

- VM과 가상 네트워크 환경 (ns3, mininet 등) 으로 엣지-클라우드 테스트베드 구축



GNN기반 분산 스케줄링 에이전트 구조



동적 서비스 조합 및 선택

시뮬레이션 기반 검증

분산 협업 배포

Commit

Build

Test

Deploy

Operate



Git



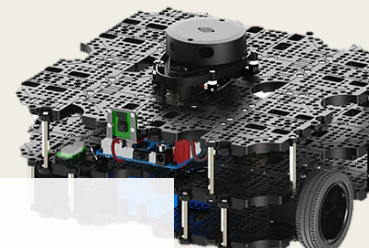
Docker



Gazebo Sim



K8s



TURTLEBOT3



Jenkins

Turtlebot - CD

- 검증된 서비스 컨테이너를 ROS 호환 K8s 파드로 전환
- SDI 환경 종속성을 고려하여 서비스들 배포

- 국제 2건, 국내 1건 작성 목표

- 국제 1건 달성 완료, 1건 제출 예정

Song, Xiangchi, et al. "Personalized User Models in a Real-world Edge Computing Environment: A Peer-to-peer Federated Learning Framework." Journal of Web Engineering 23.8 (2024): 1057-1083. (기여율 50%)

Lim, Sujeong, et al. "WYSIMWYG: Simulation-based Decision Support for Data Distribution of Hyperlocal Services in Edge Cloud Environments." Proceedings of the 2025 ACM Conference on Human Factors in Computing Systems. (CHI'25), 2025 [Late Breaking Work] (기여율 50%, Accepted)

Cho, Eunho, et al. "Adaptive ML-Enabled Edge-Cloud System Framework for Safe and Efficient Autonomous Systems", In Proceedings of 5th International Workshop on Big data driven Edge Cloud Services (BECS 2025) (기여율 50%, 제출 예정)

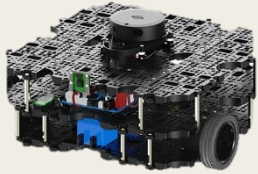
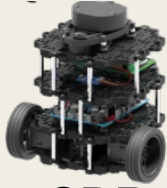




Yang, Juyoung, et al. "Troubleshooting Microservices with Heterogeneous Graph Neural Network", In Proceedings of 25th International Conference on Web Engineering (ICWE 2025) (기여율 50%, 제출 예정)

- 국내 0.5건 달성 완료

Arogya Kharel, and In-Young Ko, "Analyzing Model Hubs for Effective Composition of Pre-trained Machine Learning Models", Journal of Korean Institute of Information Scientists and Engineers (KIISE), Vol. 52, No. 1, pp. 42-51, Jan 2025

정량적 목표 수행 계획

- 정량 목표 - 모빌리티 분업형 SW 개발 기술의 지원 도메인/디바이스/환경 조합
 - 모빌리티 분업형 서비스 개발 기술의 ad-hoc polymorphism order 개수
 - 1차년도 달성: 2개 → ARM / x86 시스템 동시 지원
 - 2차년도 목표: 4개 → 두 가지 도메인 동시 지원 (SDV, SDR)

구현 시스템 \ 디바이스 종류	 SDV (TurtleBot Waffle)	 SDR (TurtleBot Burger)
ARM System		
x86 System		

Thank you

*이중 HW 적응적인 모빌리티-SDI 분업형 SW 개발을 위한
지속적 통합/검증/배포 기술 개발 계획*

In-Young Ko
iko@kaist.ac.kr

Web Engineering and Service Computing Lab,
Korea Advanced Institute of Science and Technology (KAIST)

2025. 04. 15

EOF