

docker



Jenkins

목차

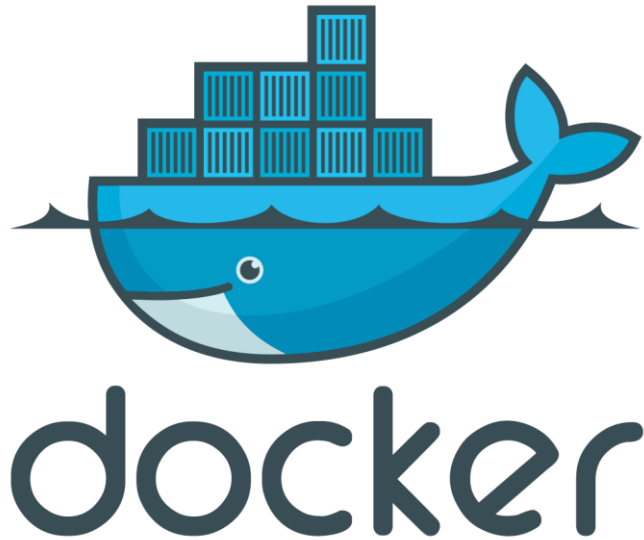
1. Docker

- 동작 방식
- 컨테이너
- 이미지

2. Jenkins

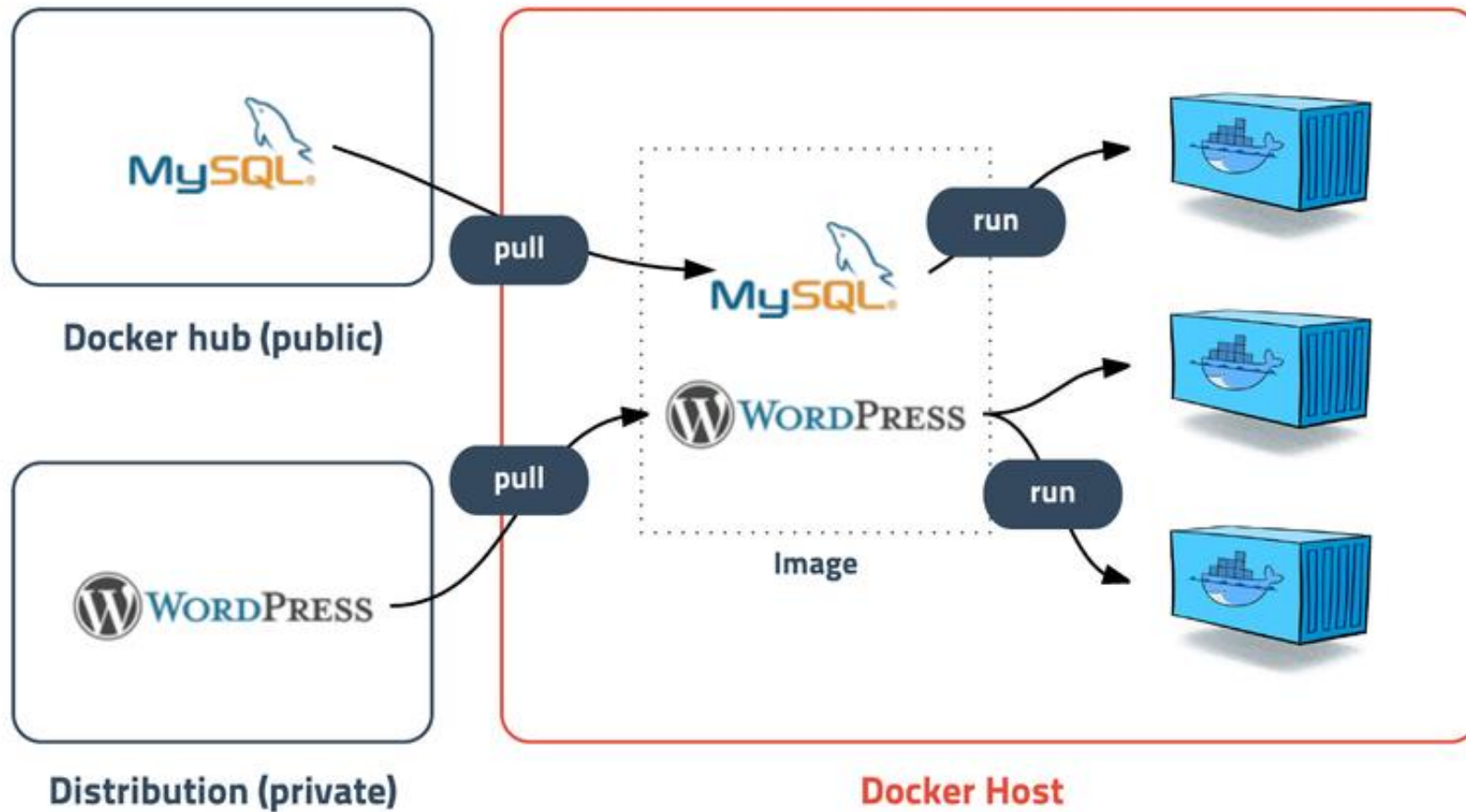
- 동작 방식

1. Docker

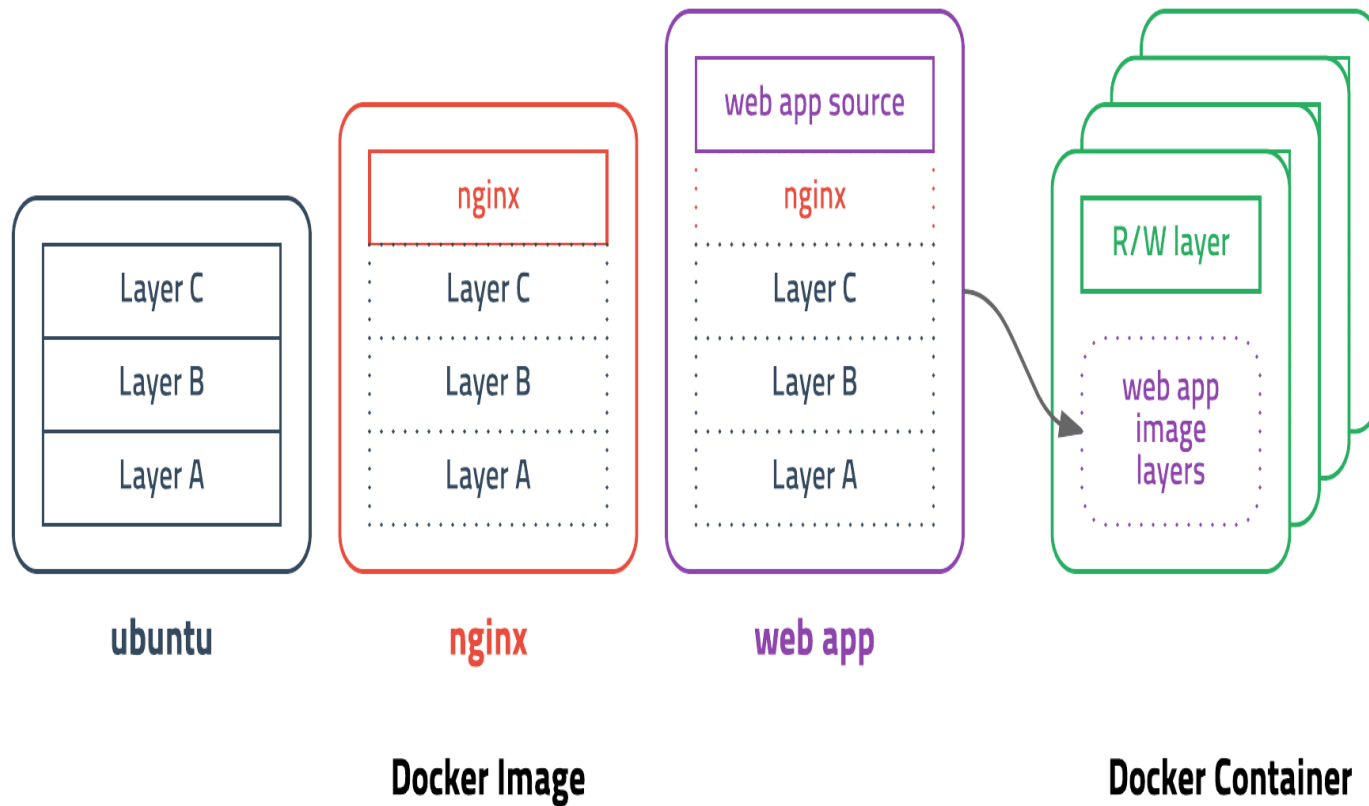


- 애플리케이션을 신속하게 구축,
테스트 및 배포할 수 있는 소프트웨어 플랫폼
- 개별 소프트웨어에 독립된 실행환경 제공
- 이미지를 통해 컨테이너 생성
- 컨테이너 단위로 소프트웨어 관리

1. Docker - 동작방식

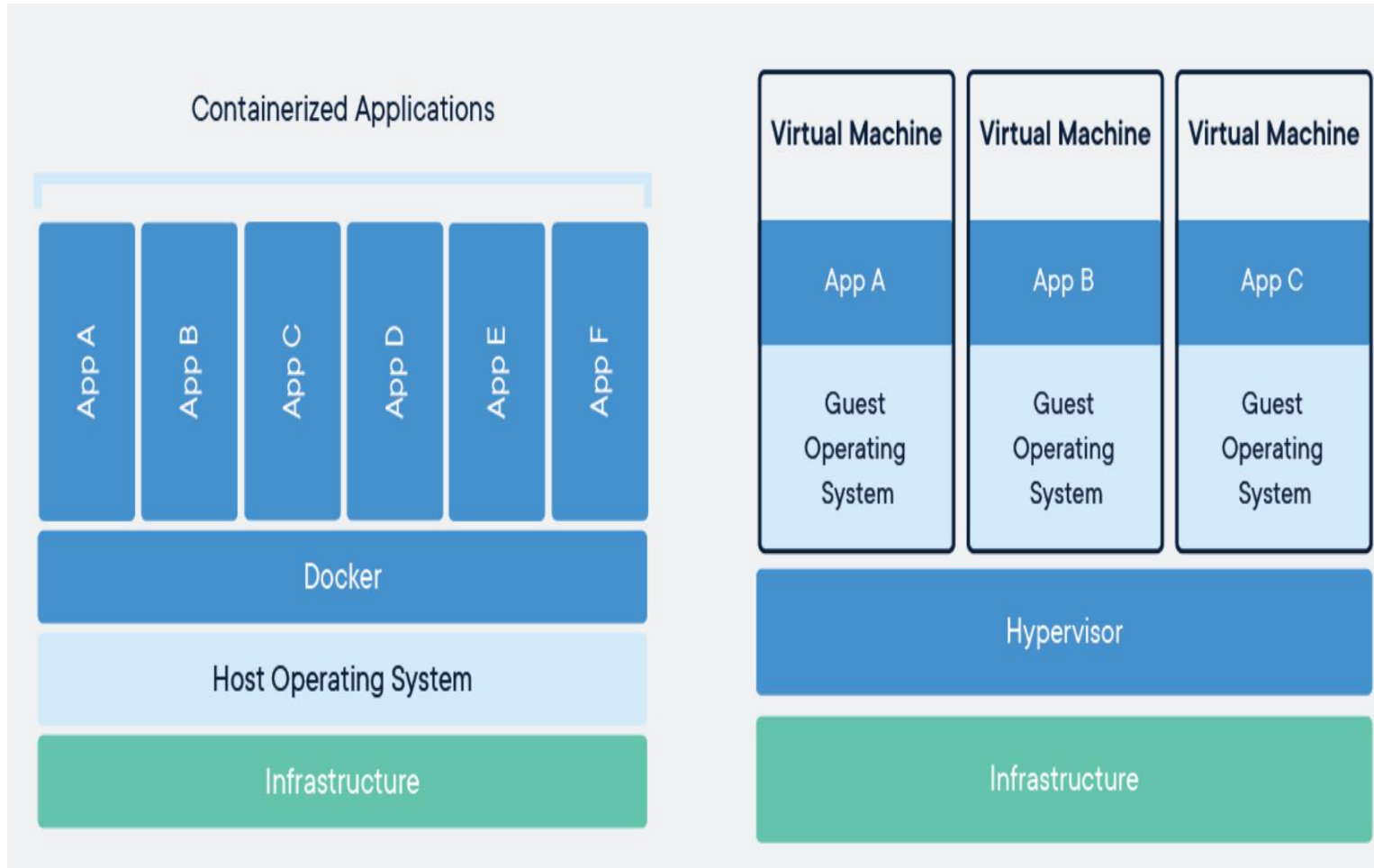


1. Docker - 이미지



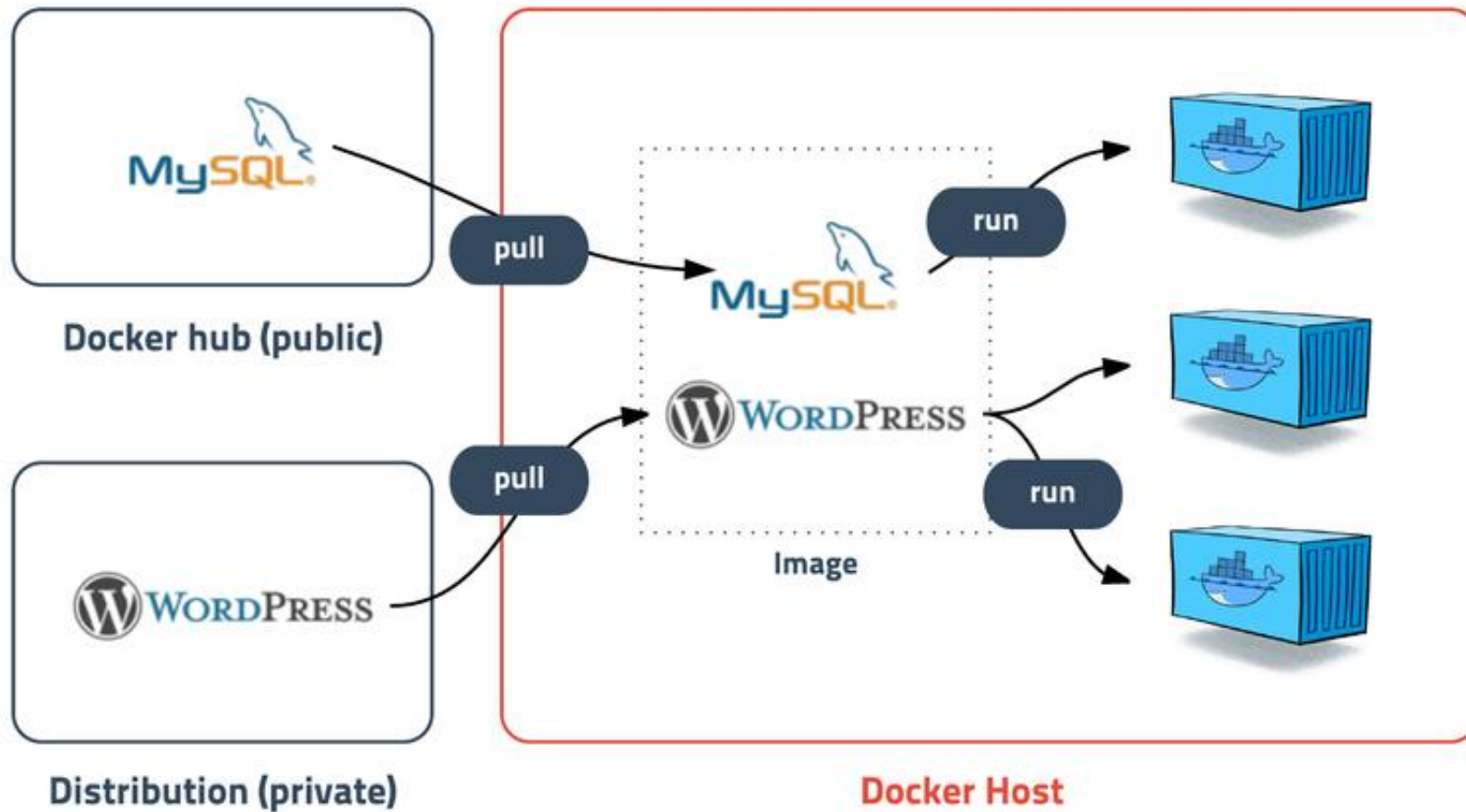
- 컨테이너 실행에 필요한 파일과 모든 설정값 등을 포함하고 있음
- 불변하는 값으로, 하나의 이미지로 여러 컨테이너 생성 가능
- layer 저장방식으로, 효율적으로 용량 관리가 가능

1. Docker - 컨테이너



- 컨테이너는 Docker에서 사용하는 소프트웨어 관리 단위
- 일종의 소프트웨어 실행에 필요한 모든 것을 포함하는 완전한 파일 시스템 패키징
- VM에 비해 효율적이고, 확장성이 좋다

1. Docker - 동작방식



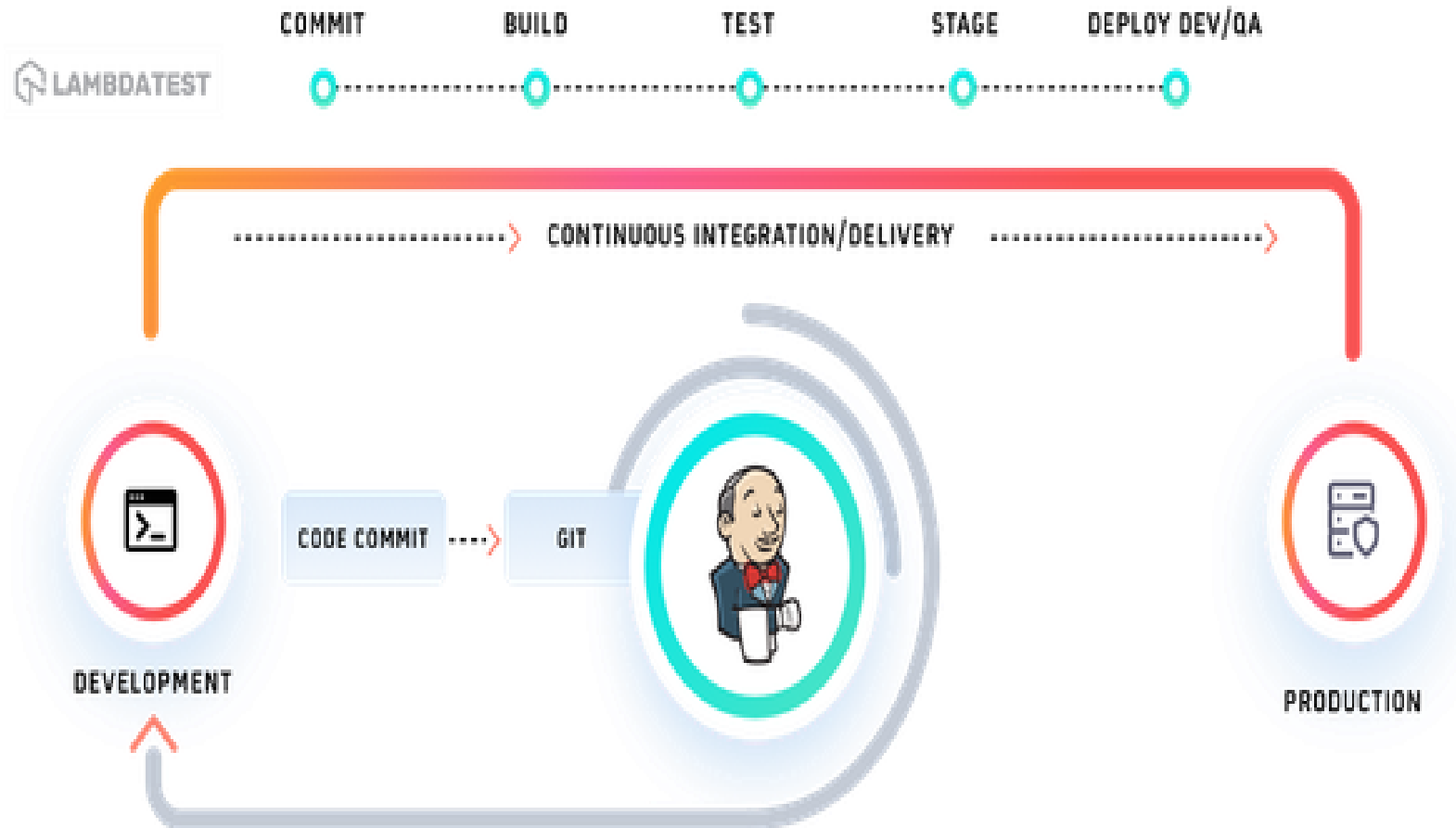
2. Jenkins



Jenkins

- 여러 개발자가 작업한 내용을 공유 영역의 저장소에 지속적으로 업로드하여 CI(지속적 통합)이 가능하도록 하는 도구
- 표준 컴파일 환경에서 컴파일 오류 검출
- 자동화 테스트 수행 : 코딩 규약 준수 여부 체크
- 결함 테스트 환경에 대한 배포 작업
- 애플리케이션 서버로의 배포 간략화

2. Jenkins - 동작 방식



1. 로컬에서 개발 후 깃허브로 커밋 및 푸시
2. 젠킨스가 이를 인지하여 빌드 및 테스트
3. 빌드가 성공할 경우 배포 서버로 전송
4. 빌드가 실패할 경우 로그를 남기고 알림