

Git 브랜치 전략

Git Branch Strategy

류창선



Git Flow

Github Flow



Gitlab Flow



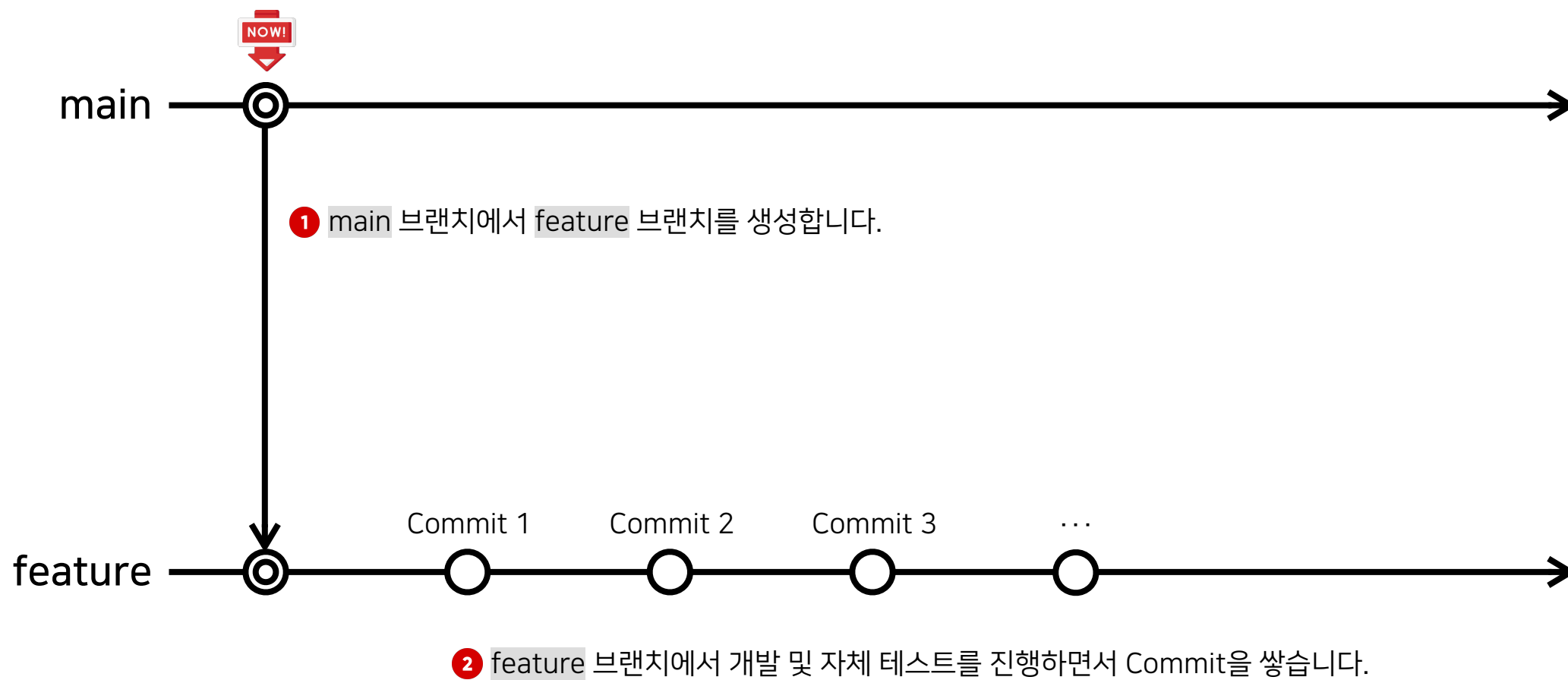
TBD

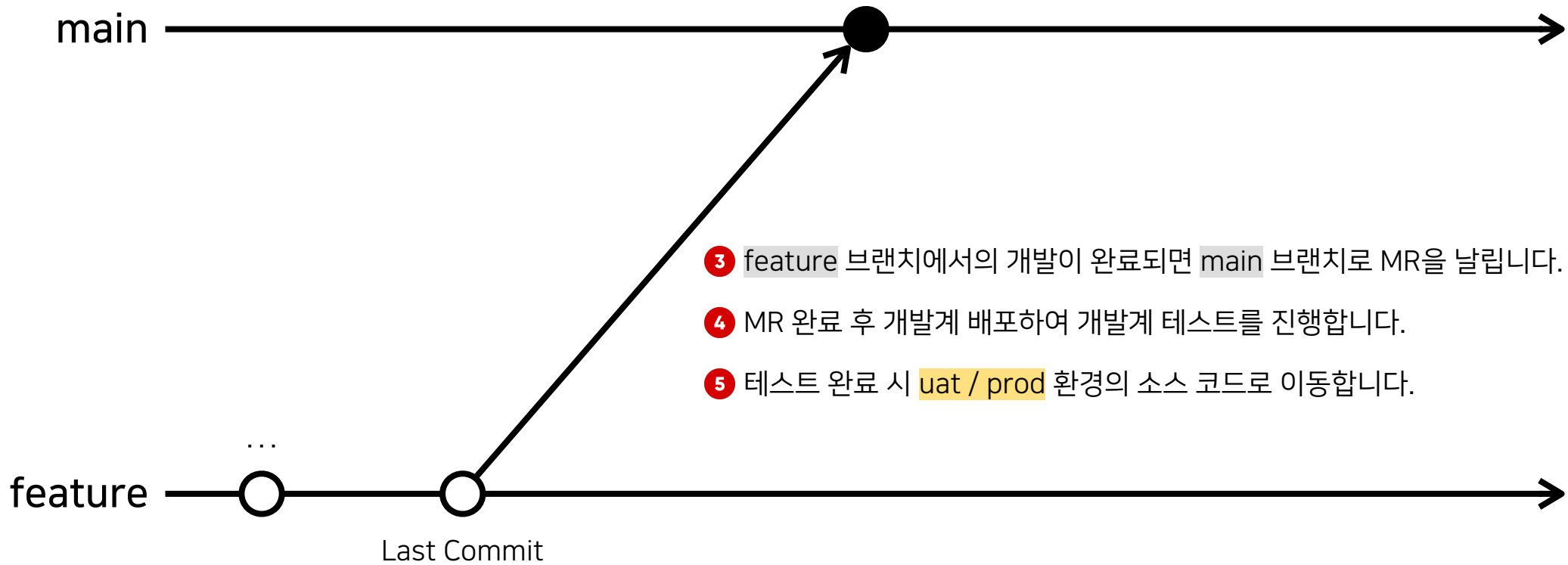


Git 브랜치 전략

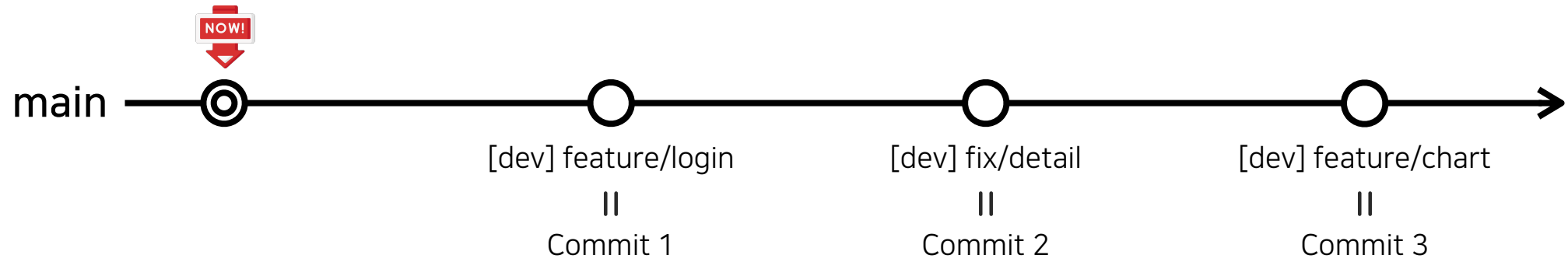
현재 기준

dev



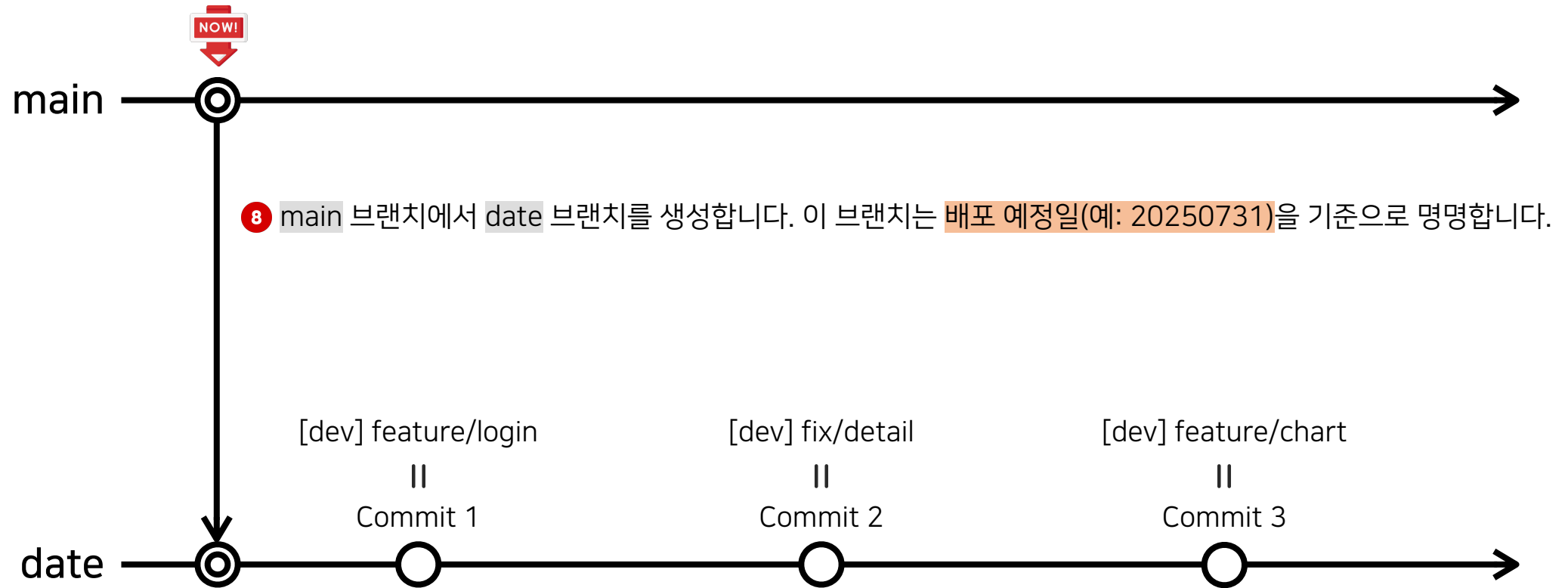


uat



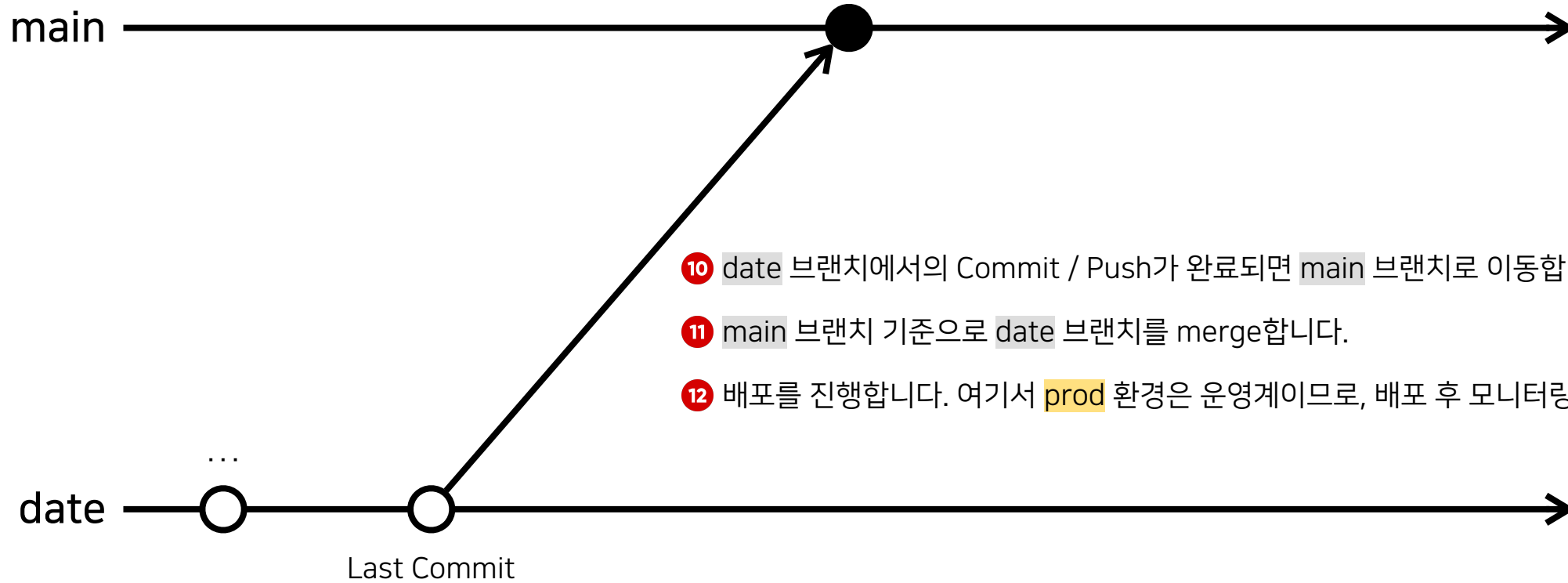
- 6 main 브랜치에 dev 환경에서 작업한 feature 브랜치의 최종 작업본을 하나씩 Commit / Push 합니다.
- 7 배포 후 uat 환경에서의 테스트를 진행합니다.

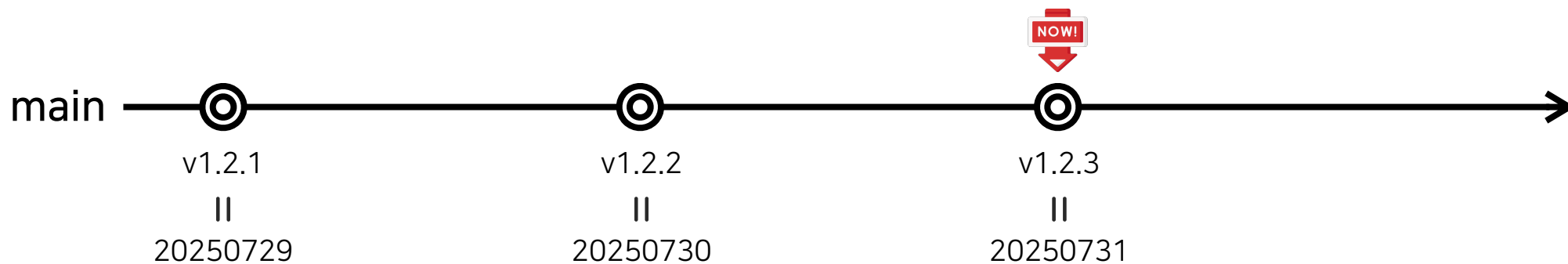
prod



8 main 브랜치에서 date 브랜치를 생성합니다. 이 브랜치는 배포 예정일(예: 20250731)을 기준으로 명명합니다.

9 date 브랜치에 dev 환경에서 작업한 feature 브랜치의 최종 작업본을 하나씩 Commit / Push 합니다.





- 13 방금 배포한 서비스 버전을 v1.2.3이라고 가정합니다.
- 14 운영 배포 후 장애가 발생하여 Rollback을 진행해야 하는 상태입니다.
- 15 운영 배포 후 안정화 모니터링 단계를 거쳤던 v1.2.2를 Rollback 시점으로 삼습니다.
- 16 v1.2.3의 hash를 확인합니다.
- 17 git revert hash 명령어를 사용해 Rollback한 후 재배포하여 장애 발생 여부를 확인합니다.

Gitlab 통합 이후 기준



Git 브랜치 전략

Gitlab CI/CD



TBD