서버 성능테스트, 클릭 한 번으로 끝내볼 수 있을까?

김덕수
배민서비스개발팀

이 세션이 누구에게 도움이 될까요?

C 우아콘 WOOWA N 2021

- 현재 운영중인 서비스에 트래픽이 많고, 서버 성능테스트를 자주 하고 계신 분들
- 운영중인 서비스에 트래픽이 점점 늘어나고 있는 분들
- 대규모의 트래픽을 안정적으로 처리하기 위해 서비스 배포 전에 어떤 일을 해야 하는지 알고 싶은 분들

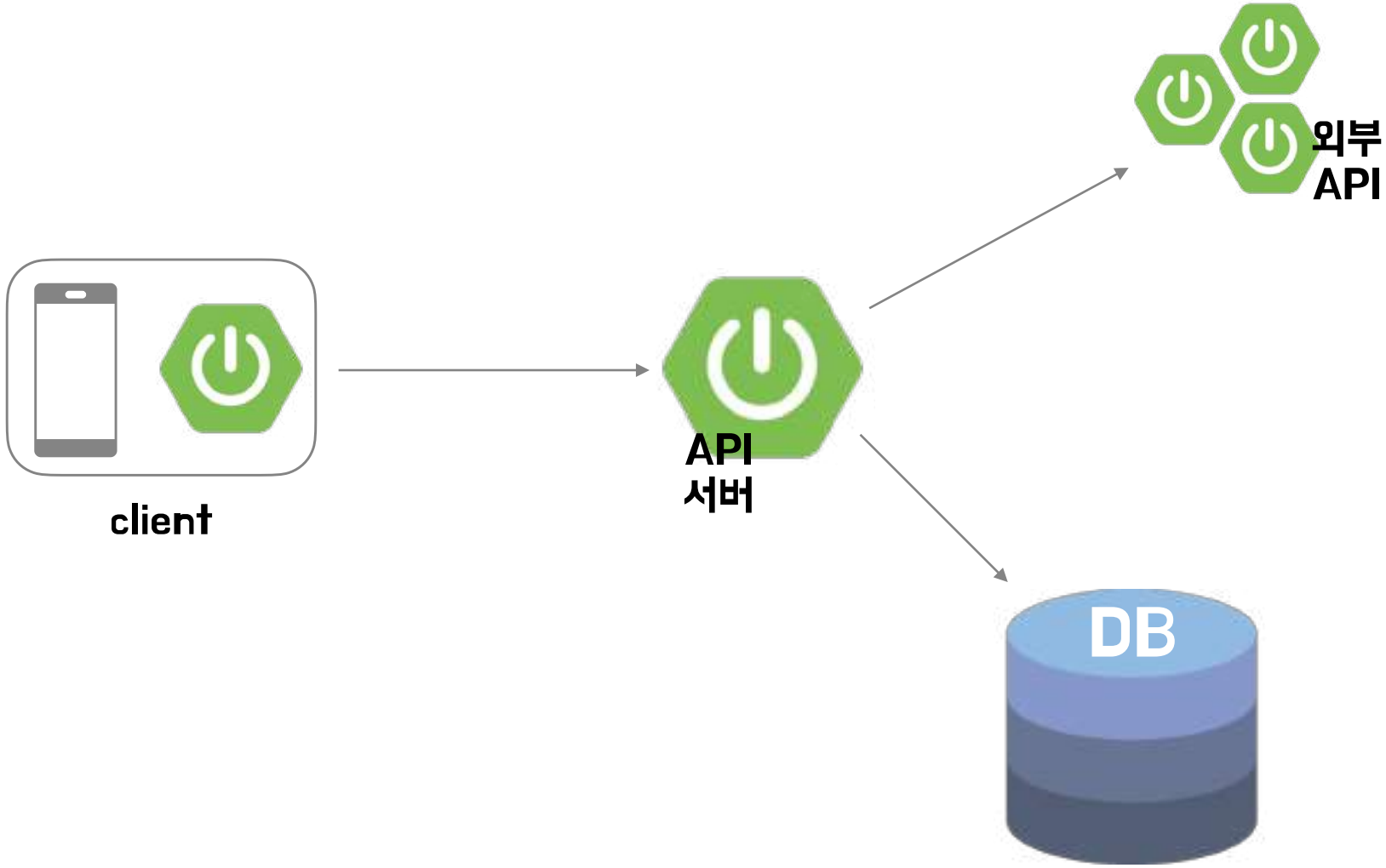
어떤 개발 환경을 갖추고 있나요?

어떤 개발 환경을 갖추고 있나요?

Web
console

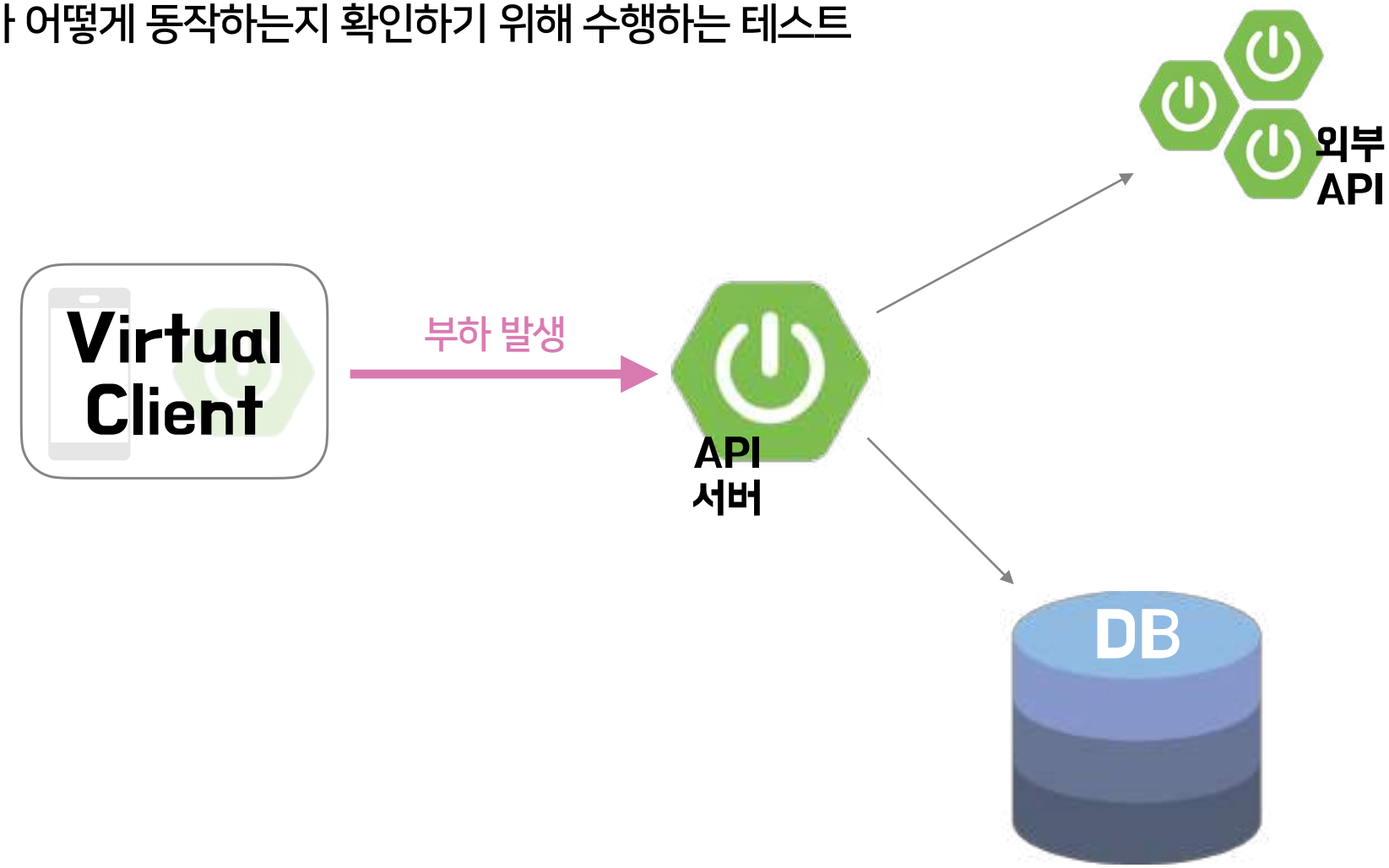
Monitoring
board

서버 성능테스트란?

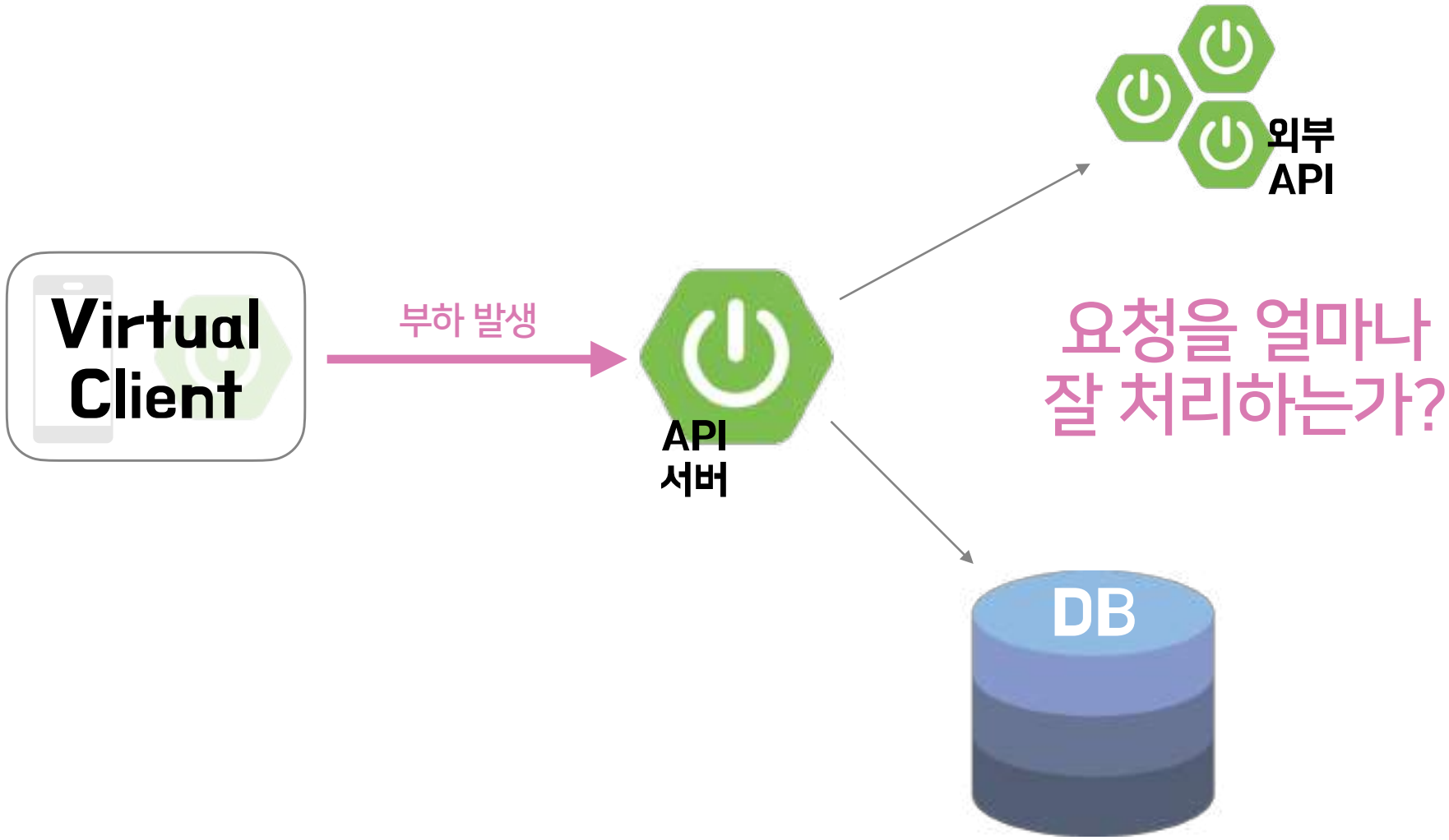


서버 성능테스트란?

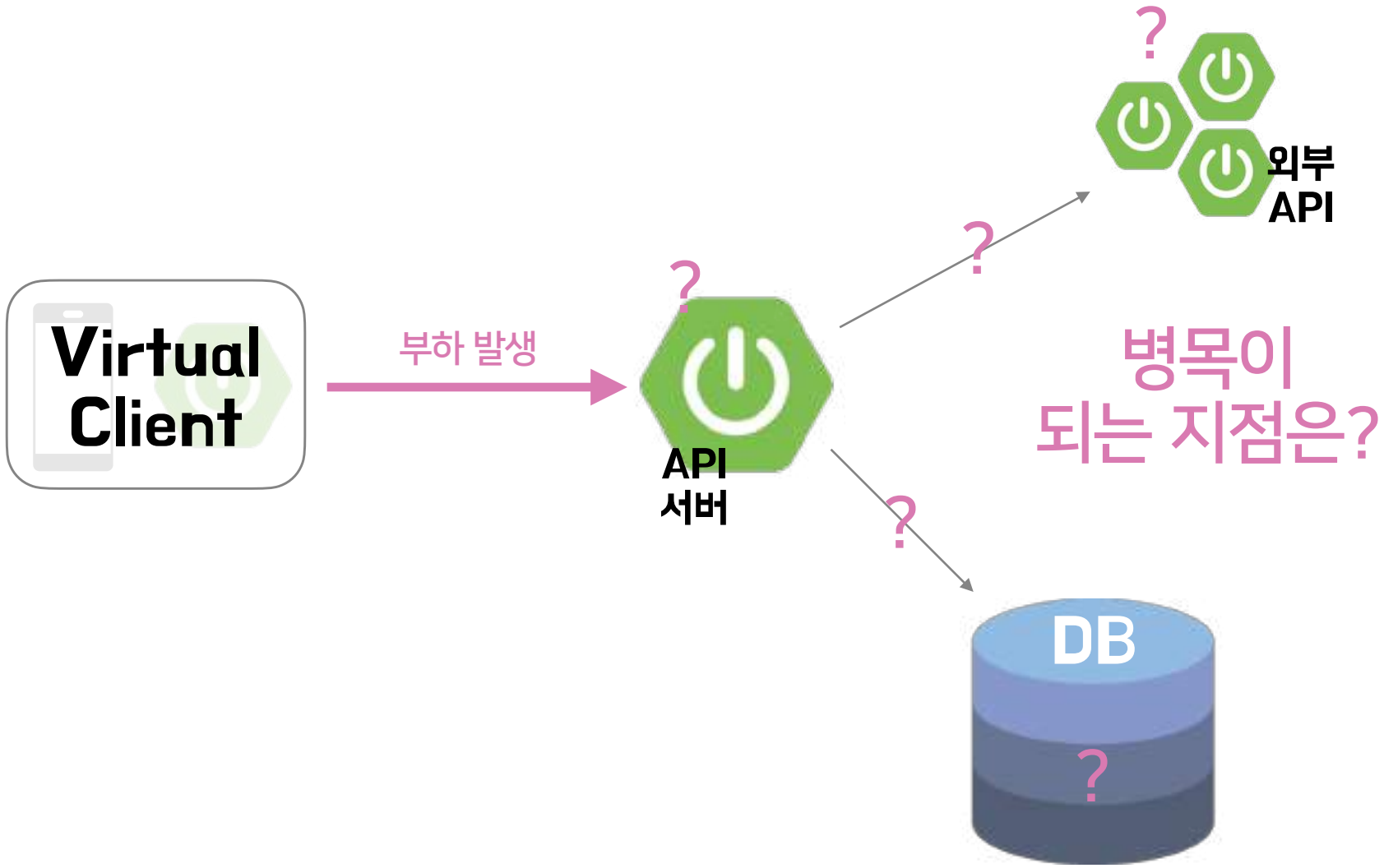
API 요청이 많은 상황에서
서버가 어떻게 동작하는지 확인하기 위해 수행하는 테스트



서버 성능테스트를 하는 이유는?



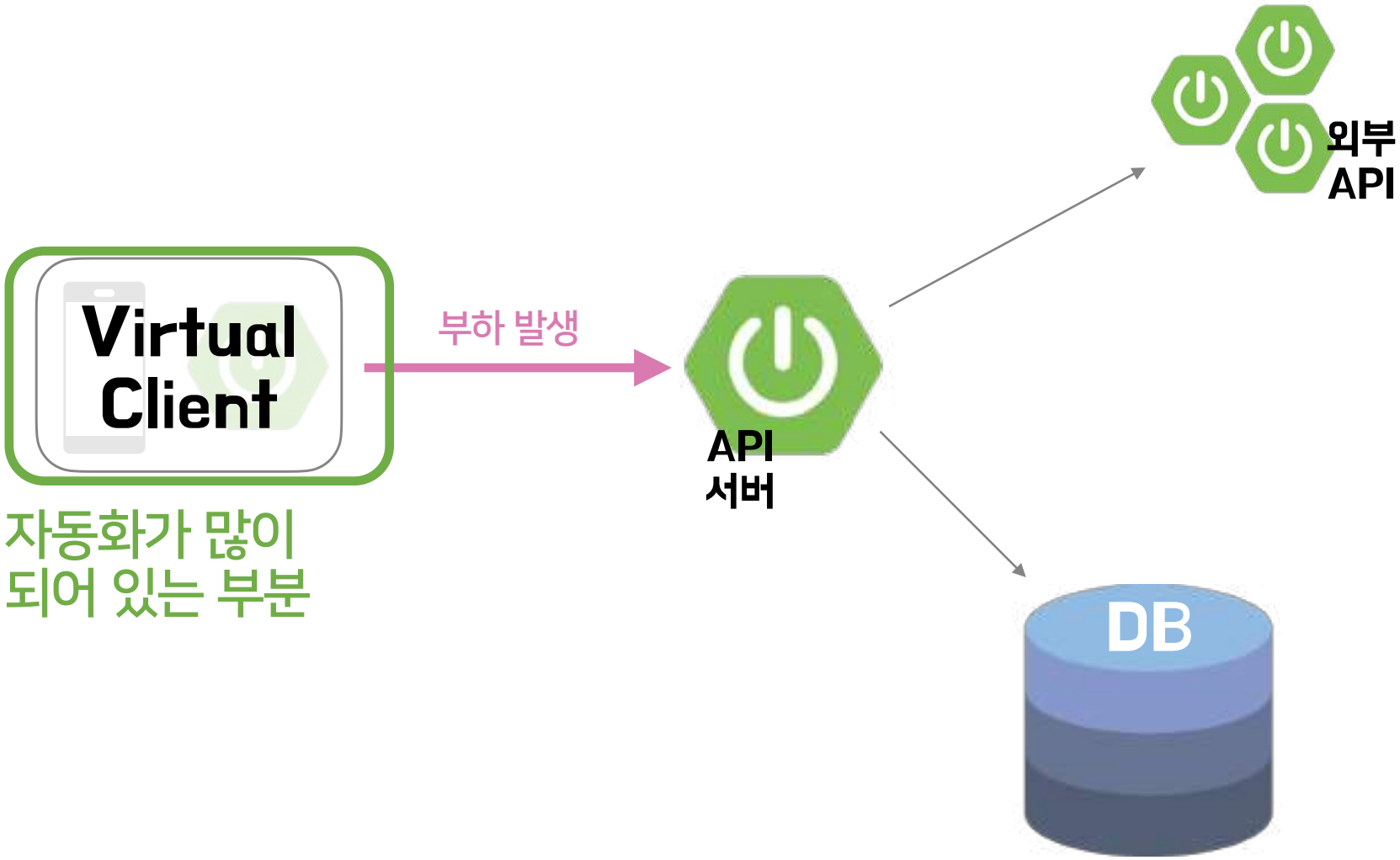
서버 성능테스트를 하는 이유는?



서버 성능테스트, 보통 언제 할까?

- 😰 트래픽 인입이 많을 것으로 예상되는 새로운 서비스를 오픈하는 경우
- 😰 기존 서비스에 병목의 가능성이 있는 변경사항이 생기는 경우
- 😰 평소보다 트래픽을 훨씬 많이 받아야 하는 경우

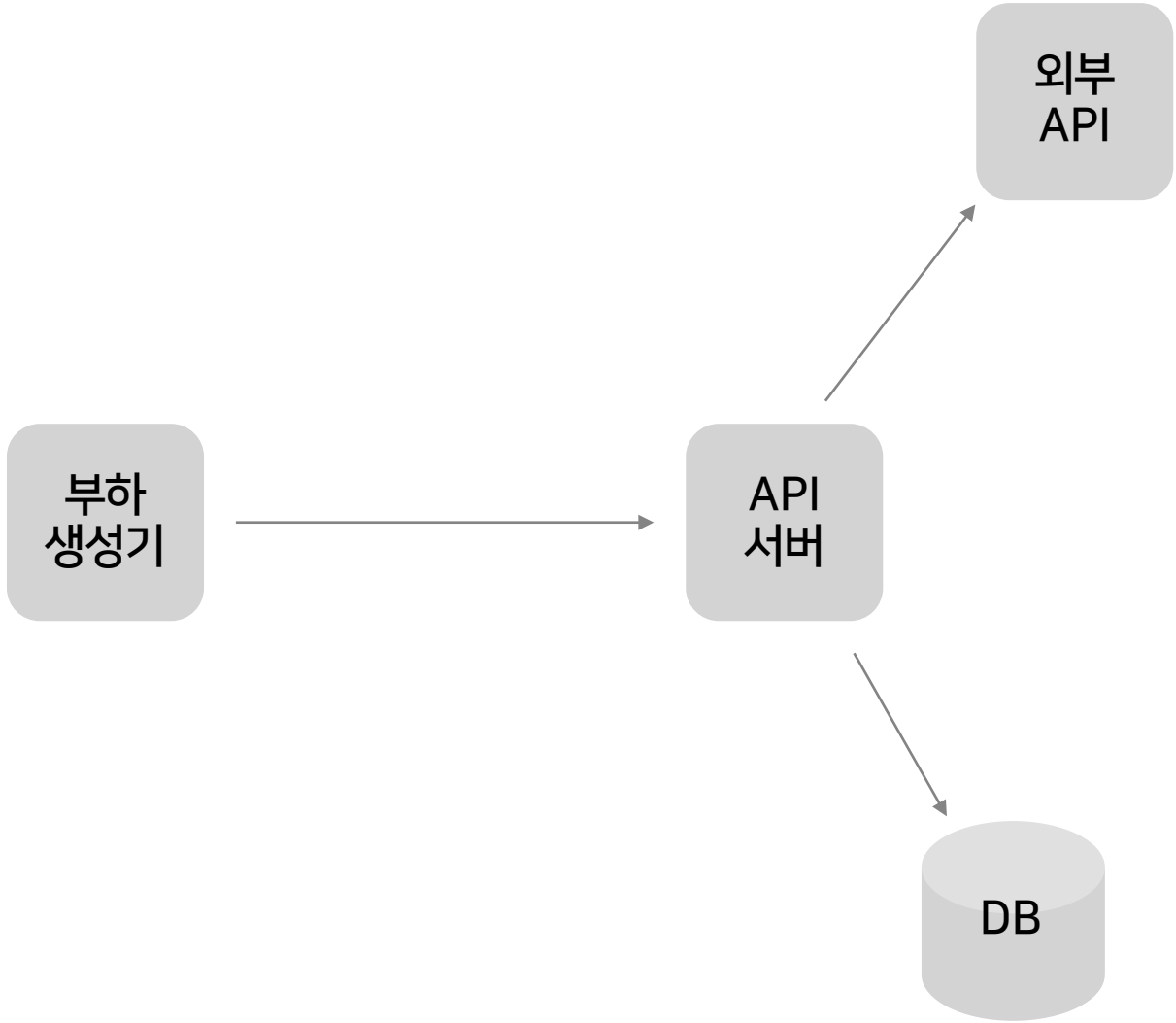
서버 성능테스트를 하려고 보니...



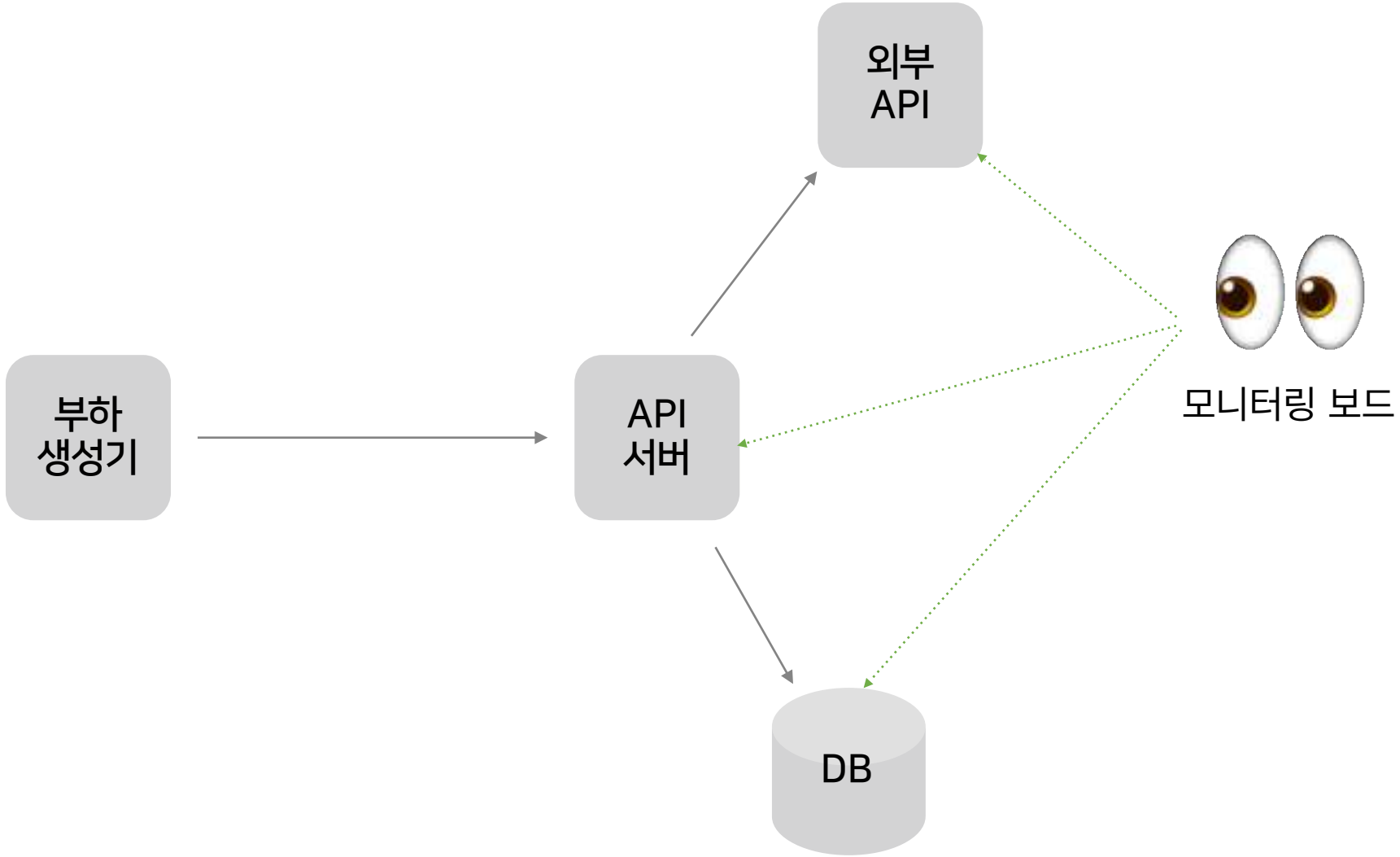
자동화가 많이 되어 있는 부분

이미 자동화가 꽤 많이 되어있지만..
그래도 아직 고되다!

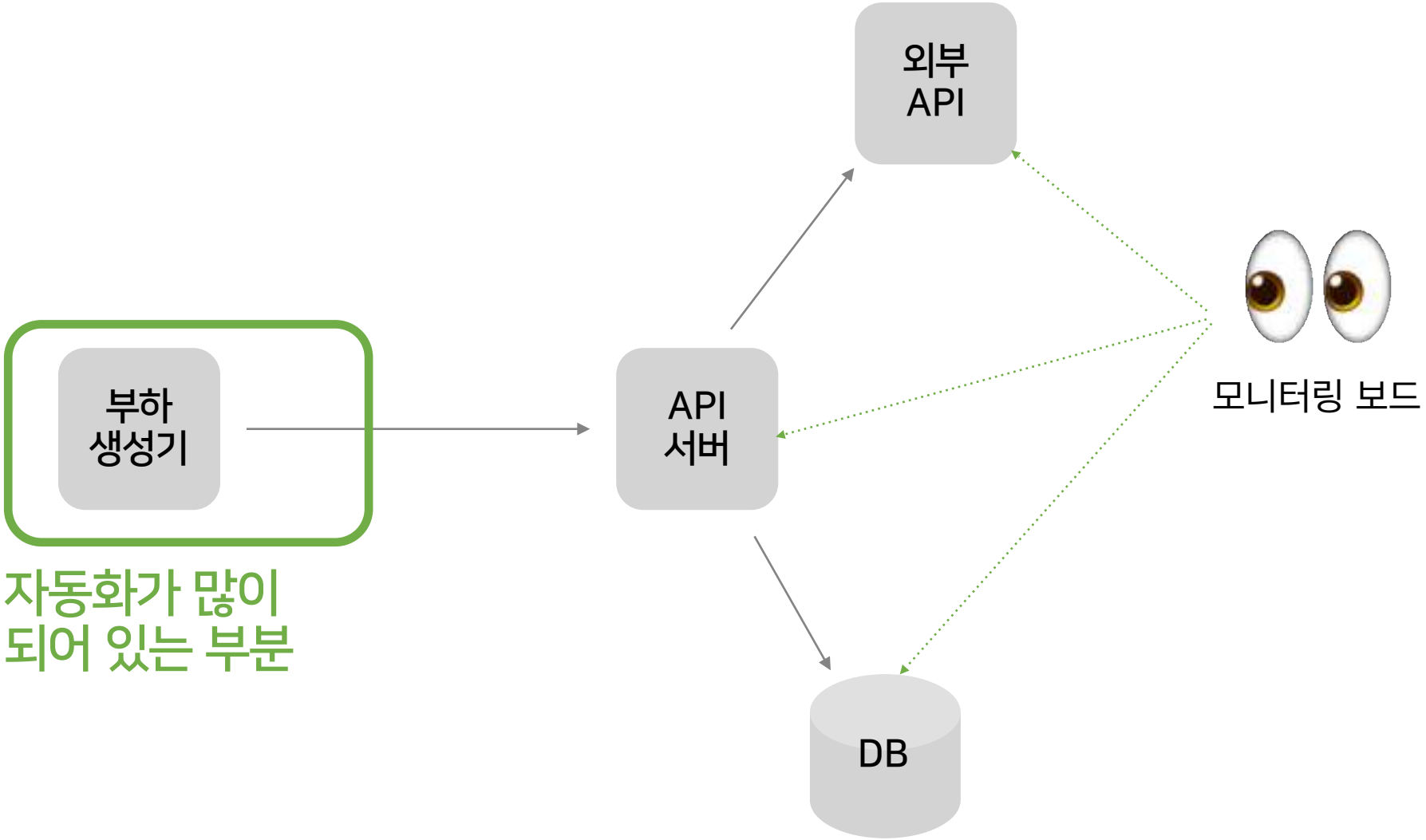
무엇이 고될까?



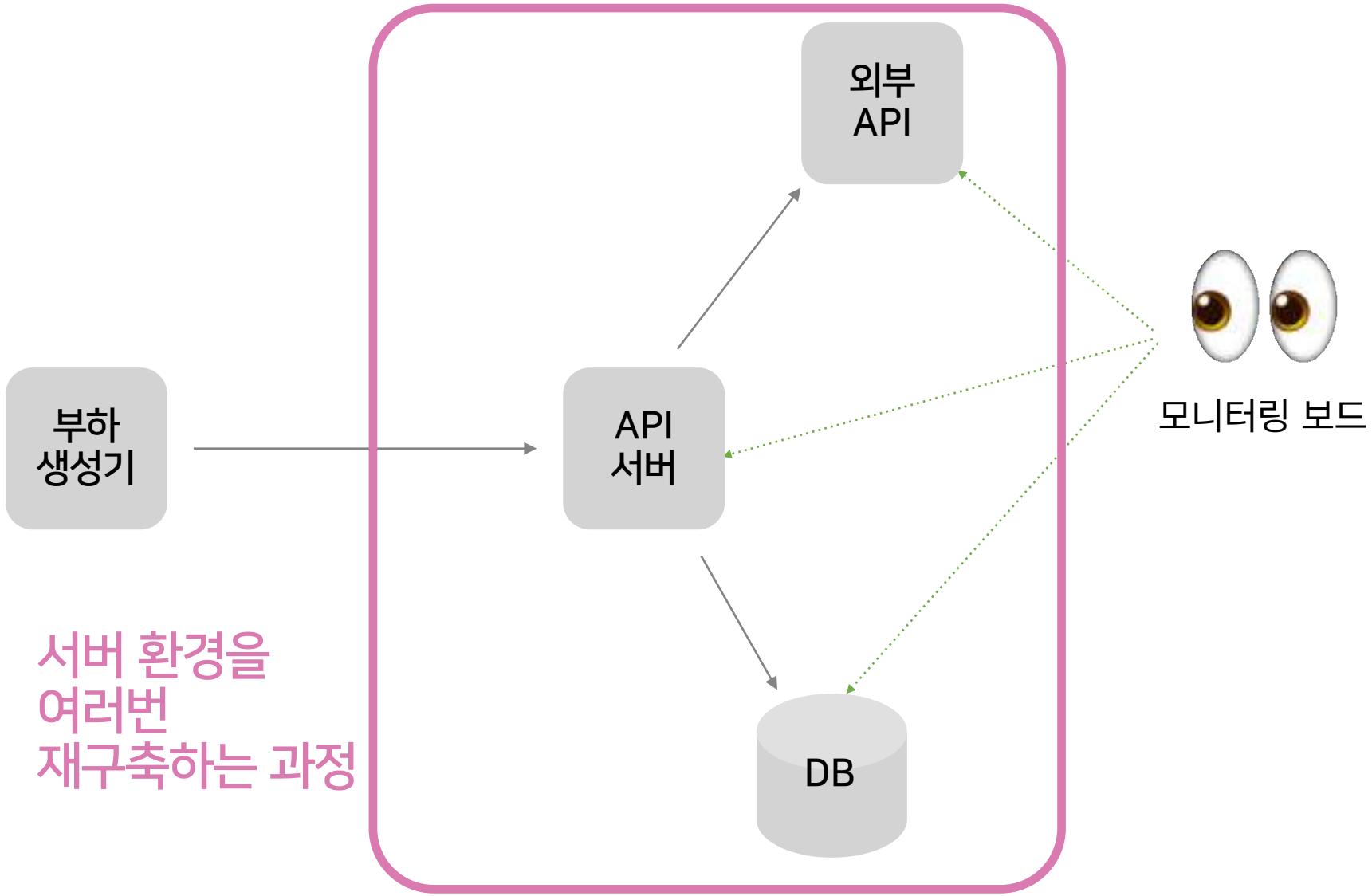
무엇이 고될까?



무엇이 고될까?

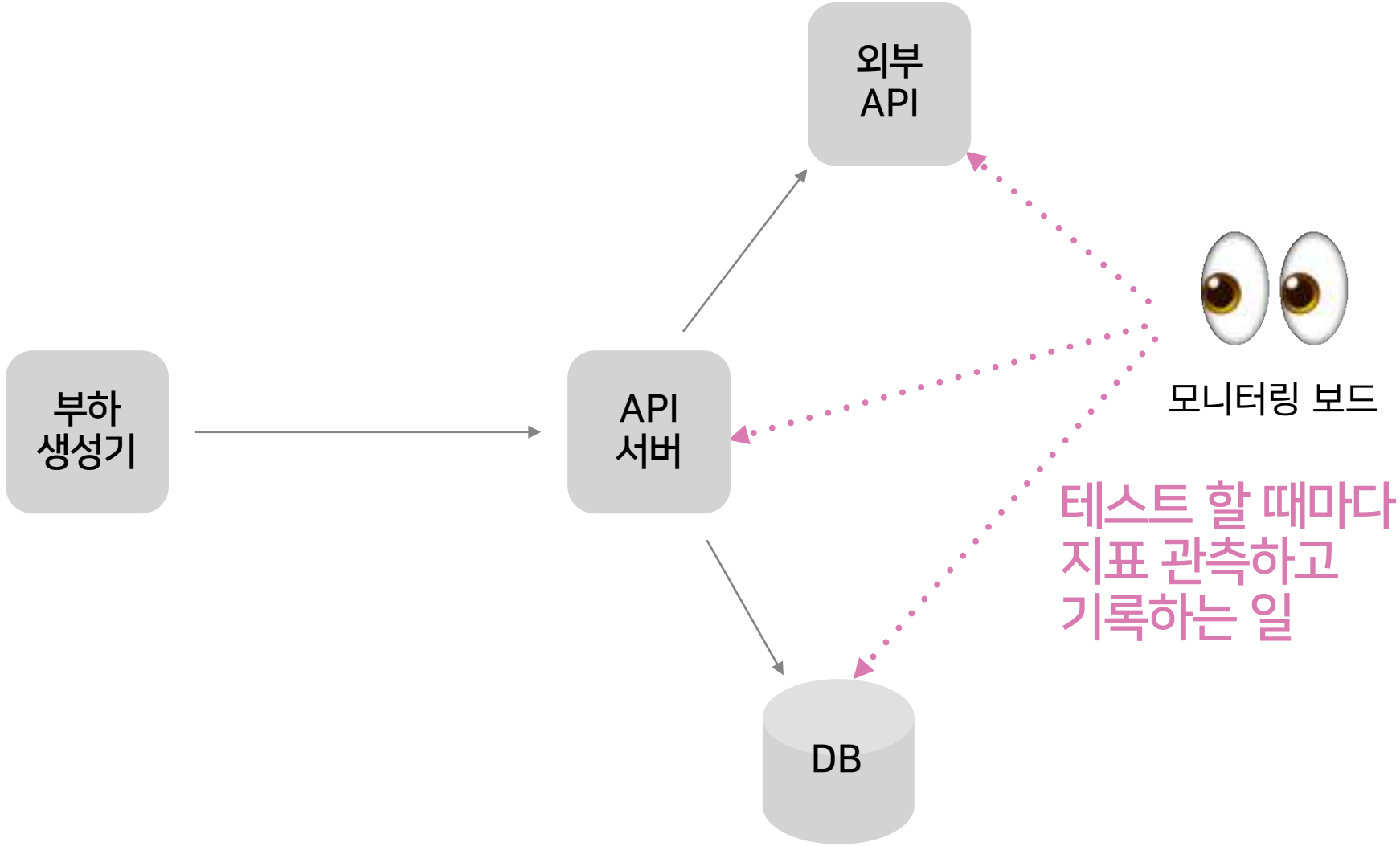


무엇이 고될까?



서버 환경을
여러번
재구축하는 과정

무엇이 고될까?



클릭 한 번으로

클릭 한 번으로
환경도 구축해주고

C 우아콘 WOOWA N 2021

클릭 한 번으로
환경도 구축해주고
성능테스트도 돌려주고

클릭 한 번으로
환경도 구축해주고
성능테스트도 돌려주고
결과 지표도 기록해주면

클릭 한 번으로
환경도 구축해주고
성능테스트도 돌려주고
결과 지표도 기록해주면
얼마나 좋을까?

어떻게 해결할까?

성능테스트 과정 도식화

어떻게 해결할까?

성능테스트 과정 도식화

시스템의 어느 부분에 어떻게 부하를 줄 것인지 결정
부하 상황에서 시스템의 동작 예측
필요한 데이터 준비

어떻게 해결할까?

성능테스트 과정 도식화

시스템의 어느 부분에 어떻게 부하를 줄 것인지 결정
부하 상황에서 시스템의 동작 예측
필요한 데이터 준비

output

input

어떻게 해결할까?

성능테스트 과정 도식화

서버 수, 서버 스펙 결정
scale up & scale out

어떻게 해결할까?

성능테스트 과정 도식화

준비된 시나리오에 따라
실제 요청 발생

어떻게 해결할까?

성능테스트 과정 도식화

부하 상황에서
모니터링 보드 지표 관측 및 기록

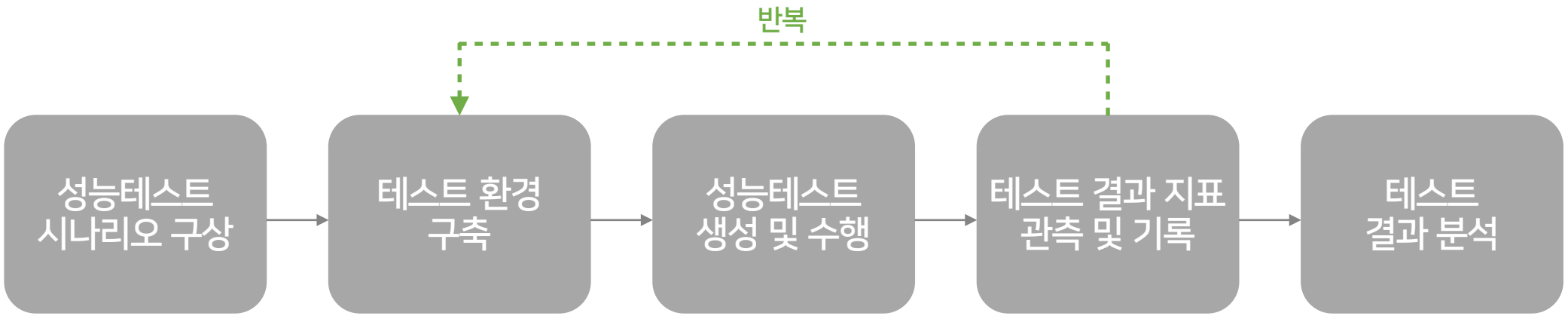
어떻게 해결할까?

성능테스트 과정 도식화

요청을 잘 처리 하는지
시스템이 기대하던 대로 동작 하였는지

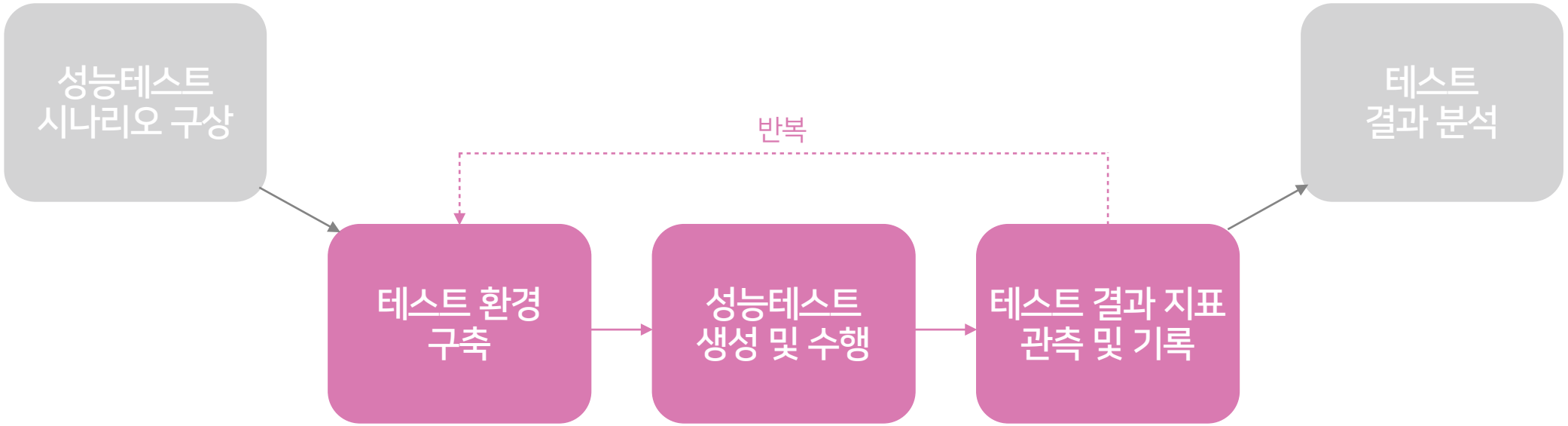
어떻게 해결할까?

성능테스트 과정 도식화



어떻게 해결할까?

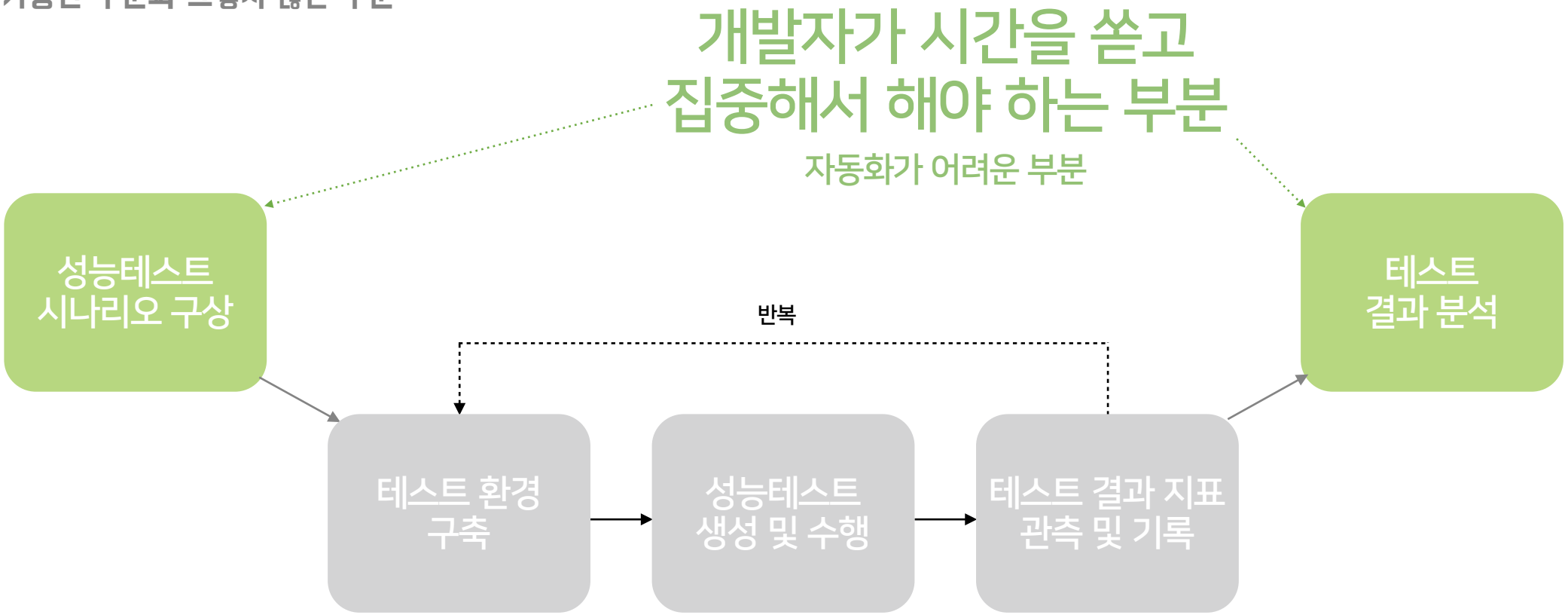
성능테스트 과정 도식화



반복되는 과정으로,
전형적인 파이프라인 구조

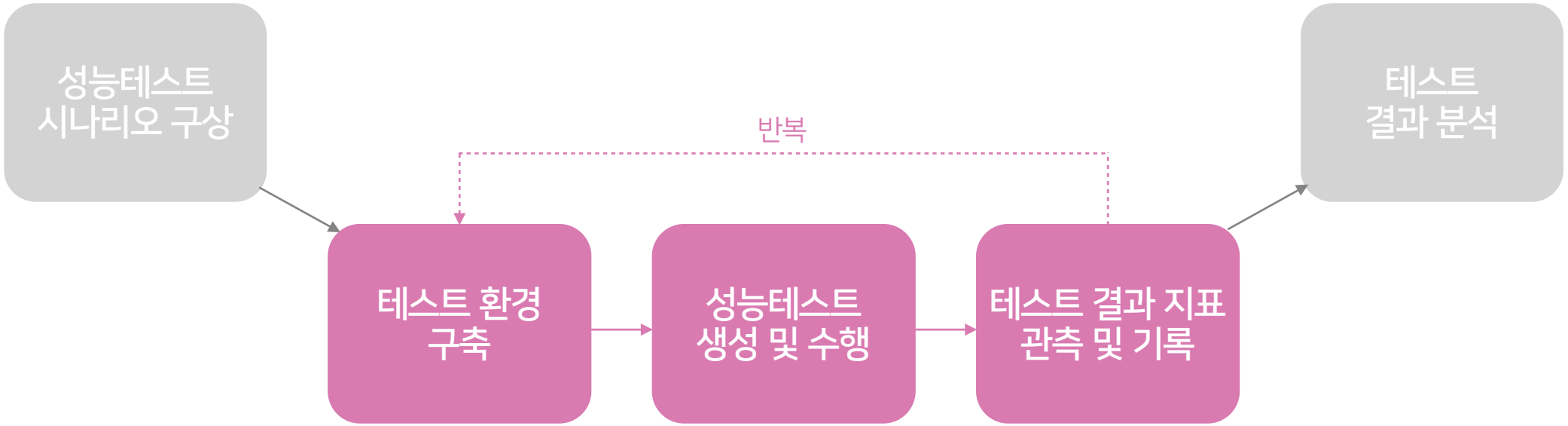
어떻게 해결할까?

자동화 가능한 부분과 그렇지 않은 부분



어떻게 해결할까?

자동화 가능한 부분과 그렇지 않은 부분



반복되는 파이프라인 구조로,

자동화 가능하며
자동화 해야하는 부분

어떻게 해결할까?

어떤 기술을 사용할까?

어떻게 해결할까?

어떤 기술을 사용할까?

+

+

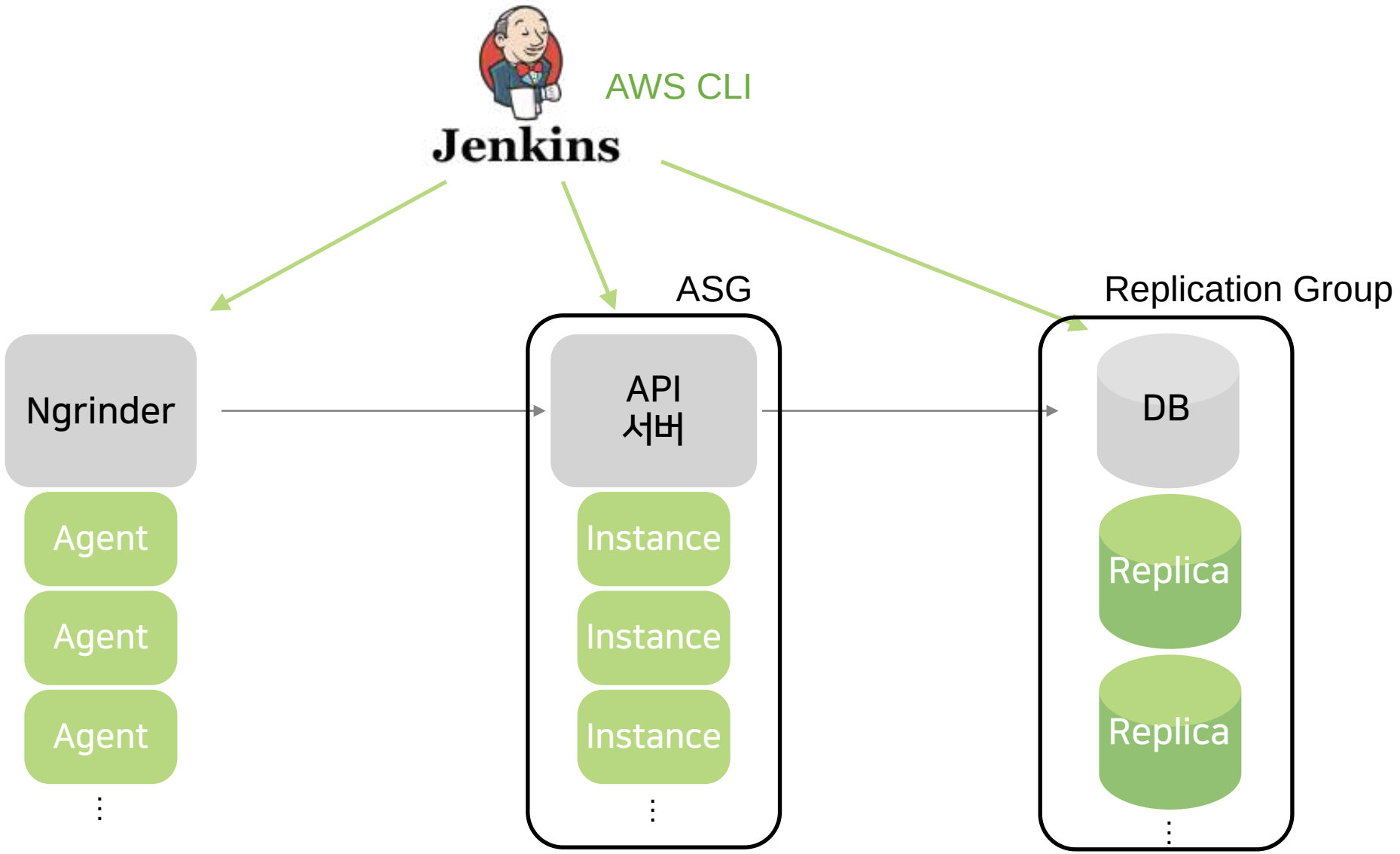
REST API 지원

Groovy script로
pipeline 구축 가능

어떻게 동작할까?

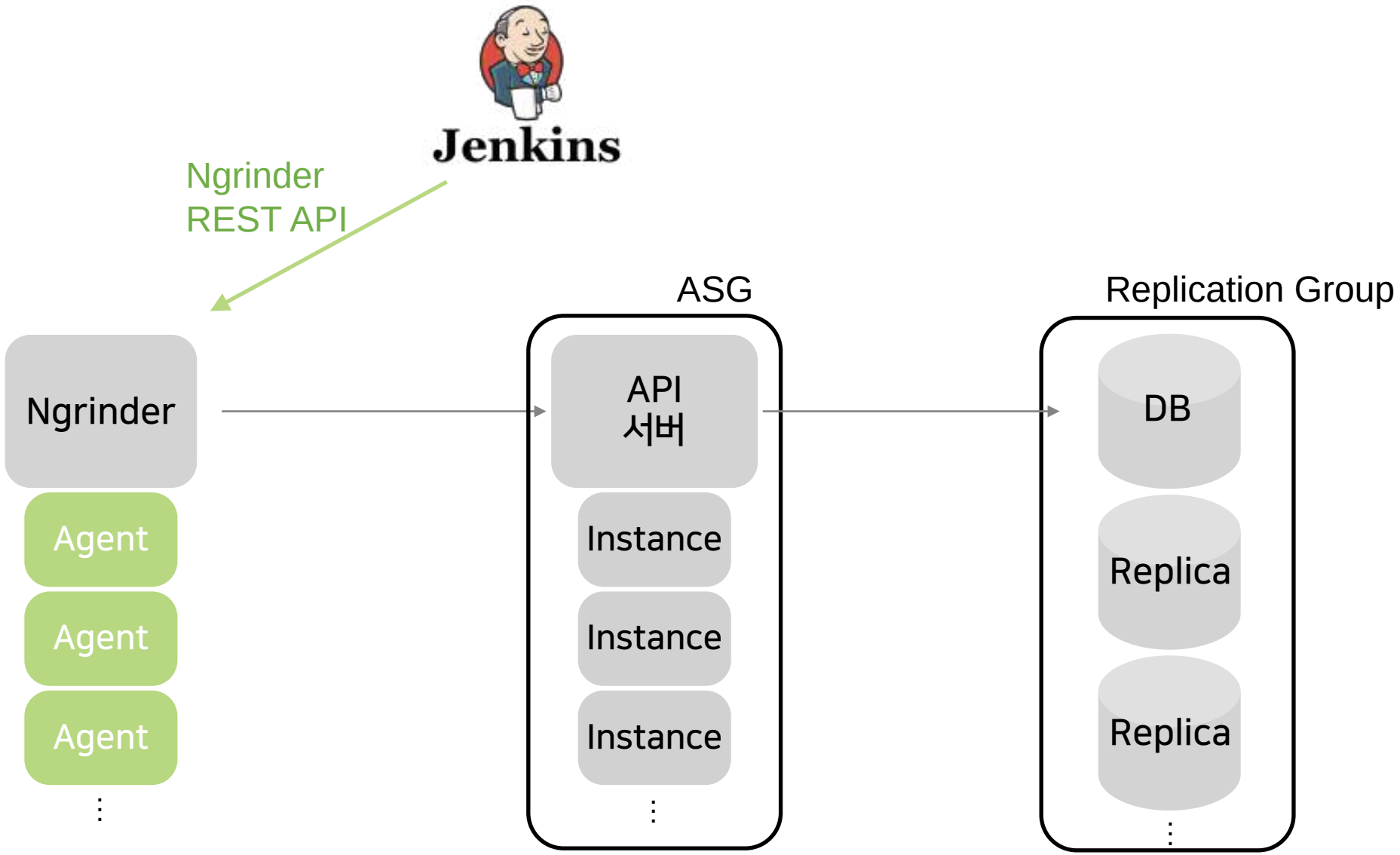
어떻게 동작할까?

테스트 환경 구축 - scale up & scale out



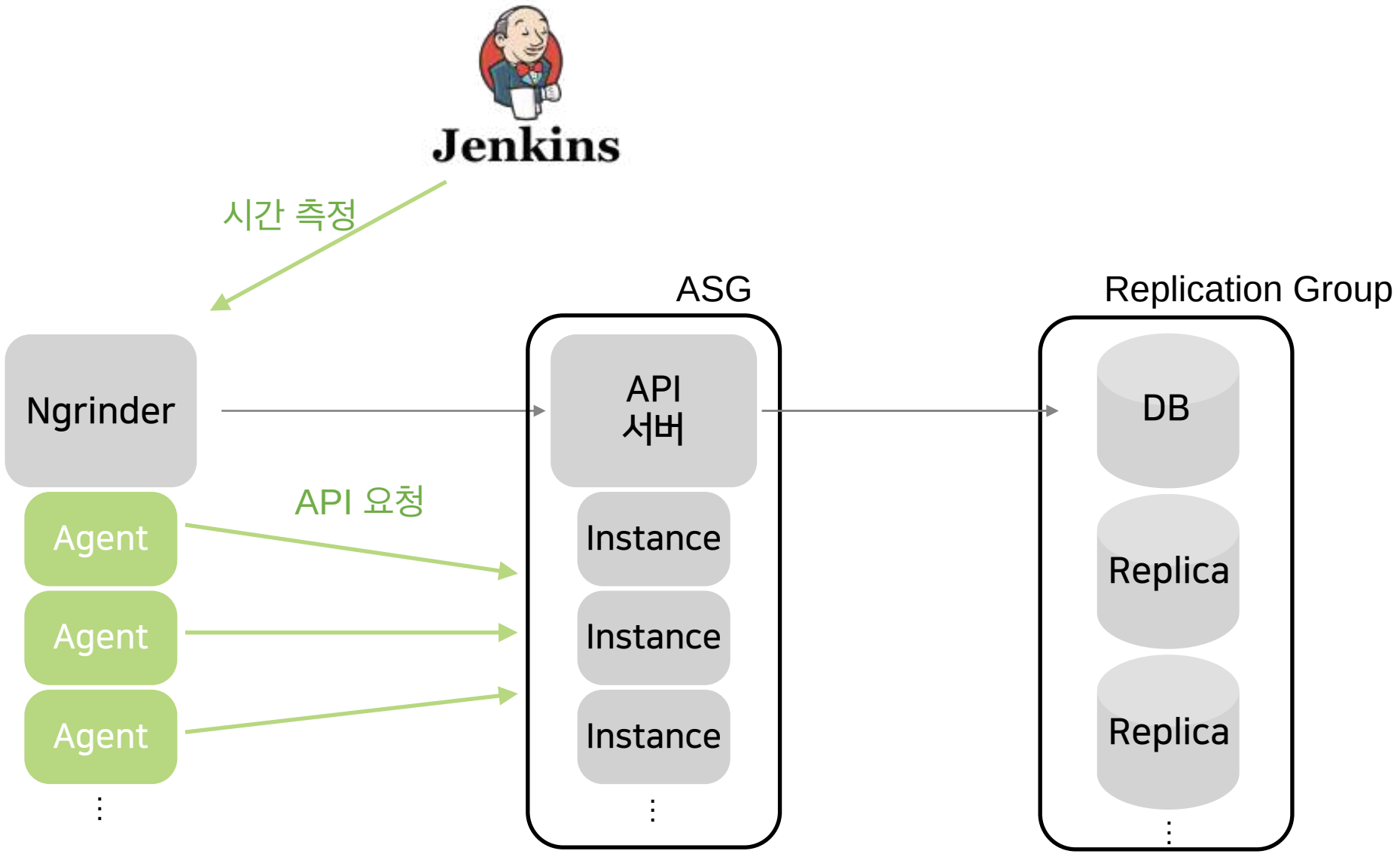
어떻게 동작할까?

신규 테스트 생성 & 수행 시작



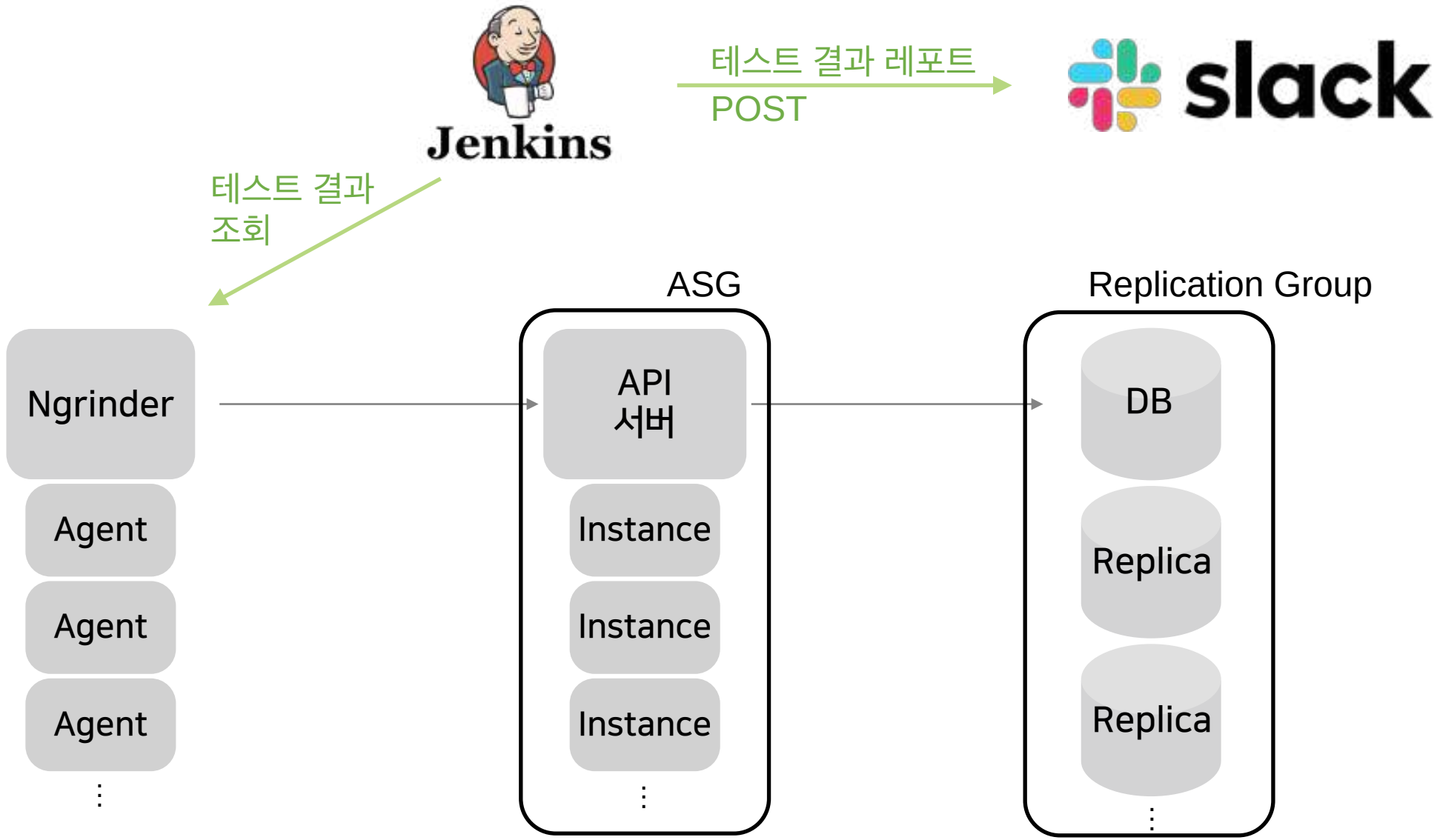
어떻게 동작할까?

테스트 모니터링



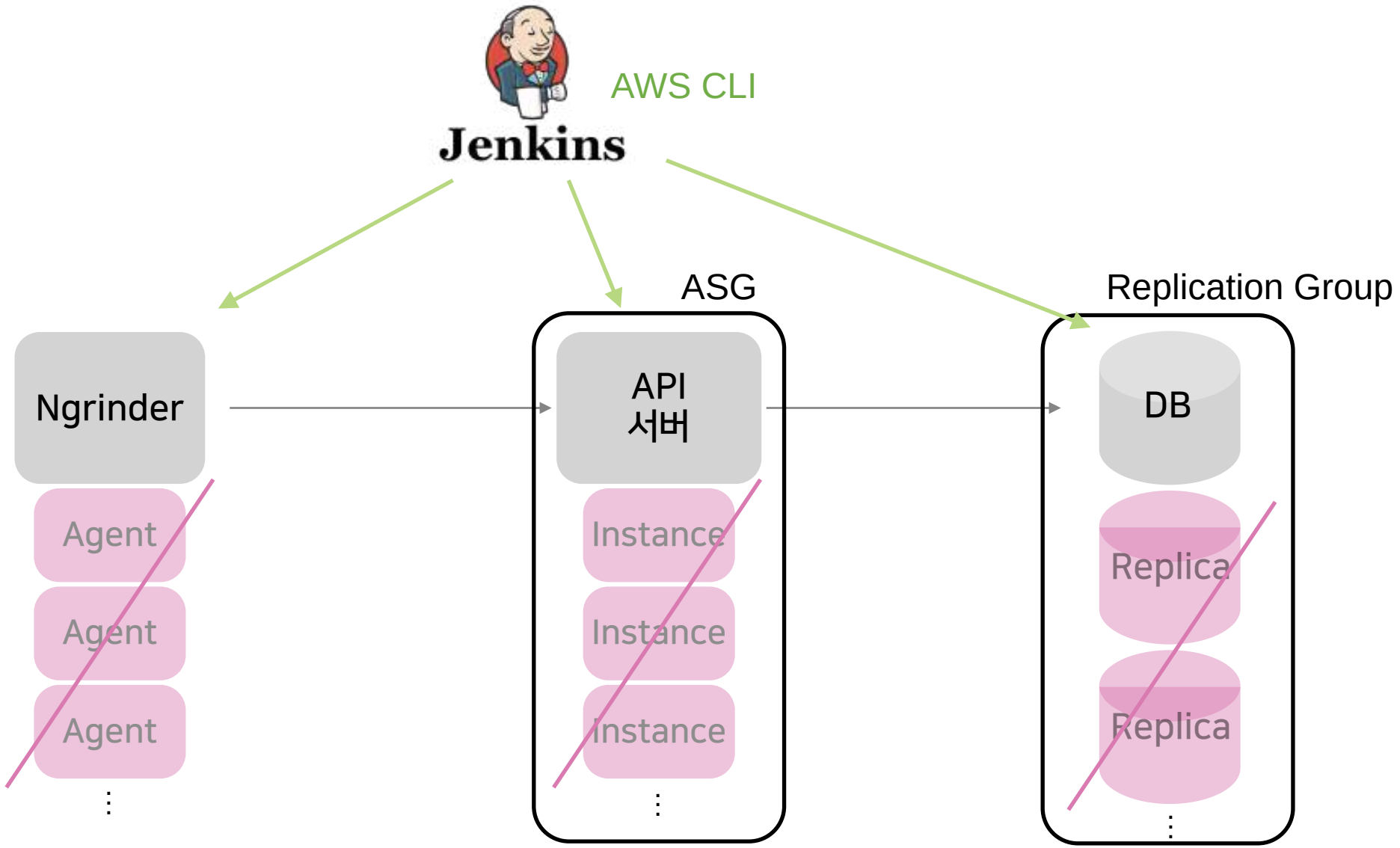
어떻게 동작할까?

테스트 결과 기록

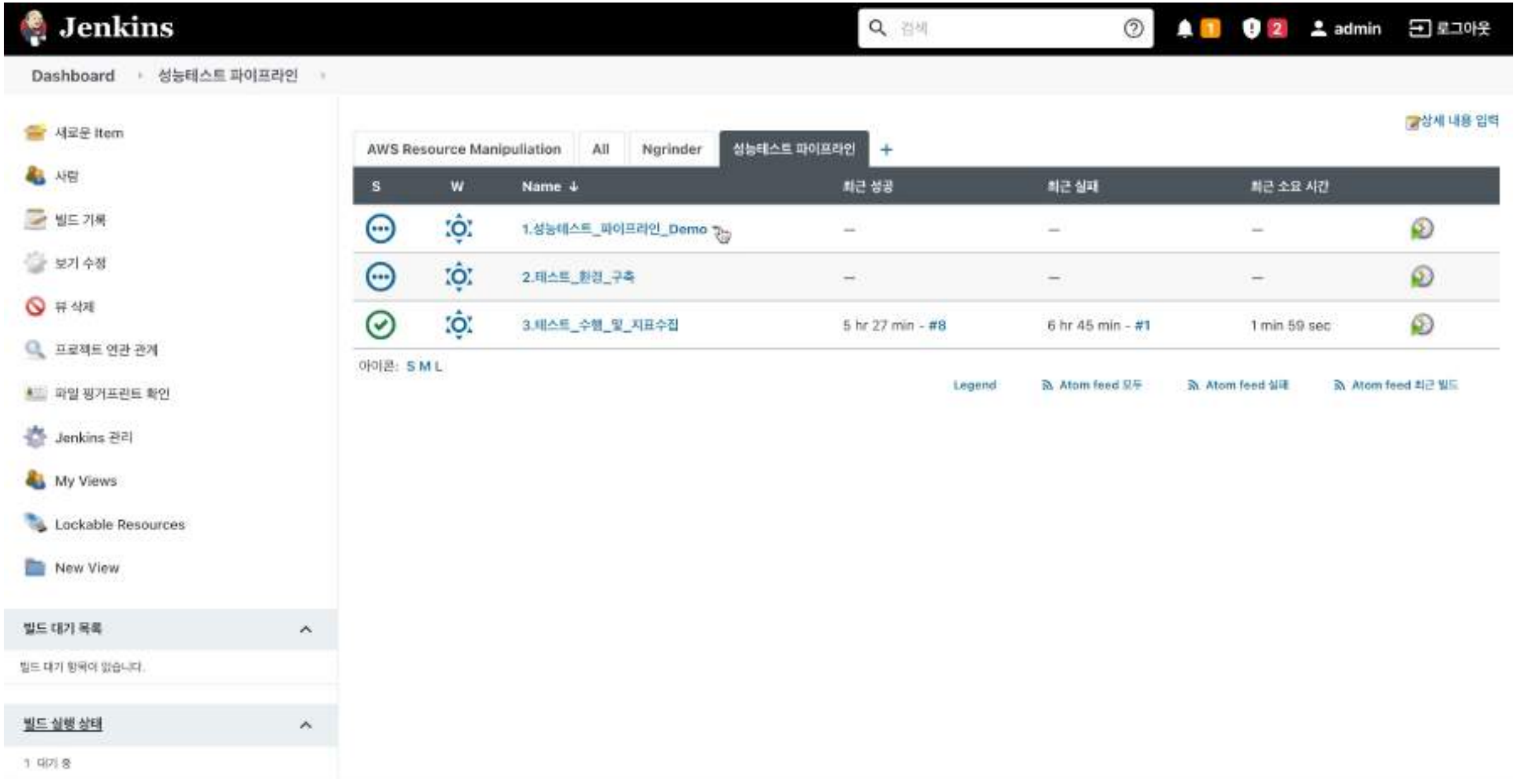


어떻게 동작할까?

자원 정리



데모 영상


Jenkins
admin
로그아웃
Dashboard
성능테스트 파이프라인
새로운 Item
사람
빌드 기록
보기 수정
뷰 삭제
프로젝트 연관 관계
파일 핑거프린트 확인
Jenkins 관리
My Views
Lockable Resources
New View
빌드 대기 목록
빌드 실행 상태
상세 내용 입력
AWS Resource Manipulation
All
Ngrinder
성능테스트 파이프라인
S
W
Name
최근 성공
최근 실패
최근 소요 시간
1.성능테스트_파이프라인_Demo
2.테스트_환경_구축
3.테스트_수행_및_지표수집
5 hr 27 min - #8
6 hr 45 min - #1
1 min 59 sec
아이콘: S M L
Legend
Atom feed 모두
Atom feed 실패
Atom feed 최근 빌드

핵심은?


Build Now


탄광 카나리아 APP 1:50 AM
테스트 환경
API server : c5d.2xlarge 10대
Redis Node : cache.r5.2xlarge 4대
목표 TPS
1800
결과 TPS
1811.31
테스트 결과 Ngrinder 레포트
테스트 결과 Ngrinder 레포트
인프라 모니터링 보드 지표
인프라 모니터링 보드

핵심은?

테스트 Input

테스트 Output

어떻게 구현할까?

구현 요구사항

✅ 사용하는 인프라나 성능테스트 시나리오가 달라지더라도 파이프라인을 쉽게 수정하고 재사용할 수 있어야 한다.

✅ 최소한의 코드 작성으로 자유롭게 파이프라인을 구성할 수 있어야 한다.

어떻게 구현할까?

구현 요구사항

✅ 사용하는 인프라나 성능테스트 시나리오가 달라지더라도 파이프라인을 쉽게 수정하고 재사용할 수 있어야 한다.

✅ 최소한의 코드 작성으로 자유롭게 파이프라인을 구성할 수 있어야 한다.

👉 각각의 실행 가능한 작은 단위의 Jenkins Job으로 나누어 구성

어떻게 구현할까?

AWS Resource Manipuliation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.Modify_AutoScalingGroup
✓	⚙	2.Modify_ElastiCache
✓	☁	3.Modify_DocDB
✓	☁	4.Modify_RDS

AWS Resource Manipuliation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.CloneAndStart_Test
✓	☁	2.CreateNewAndStart_Test

어떻게 구현할까?

우아한 WOOWA N 2021

테스트 환경 구축 Job

AWS Resource Manipuliation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.Modify_AutoScalingGroup
✓	⚙	2.Modify_ElastiCache
✓	☁	3.Modify_DocDB
✓	☁	4.Modify_RDS

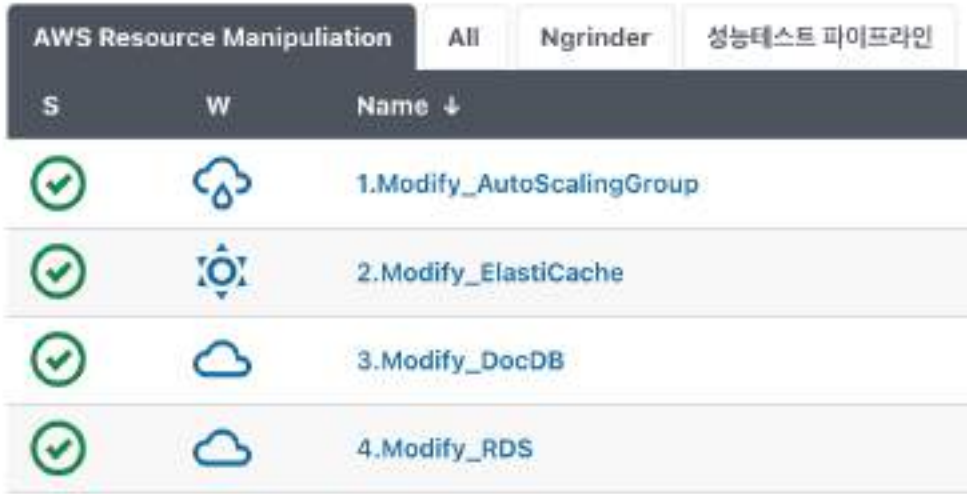
AWS Resource Manipuliation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.CloneAndStart_Test
✓	☁	2.CreateNewAndStart_Test

성능테스트 수행 & 지표 기록 Job

어떻게 구현할까?

테스트 환경 구축 Job



→ AWS Cli와
shell script로 구현

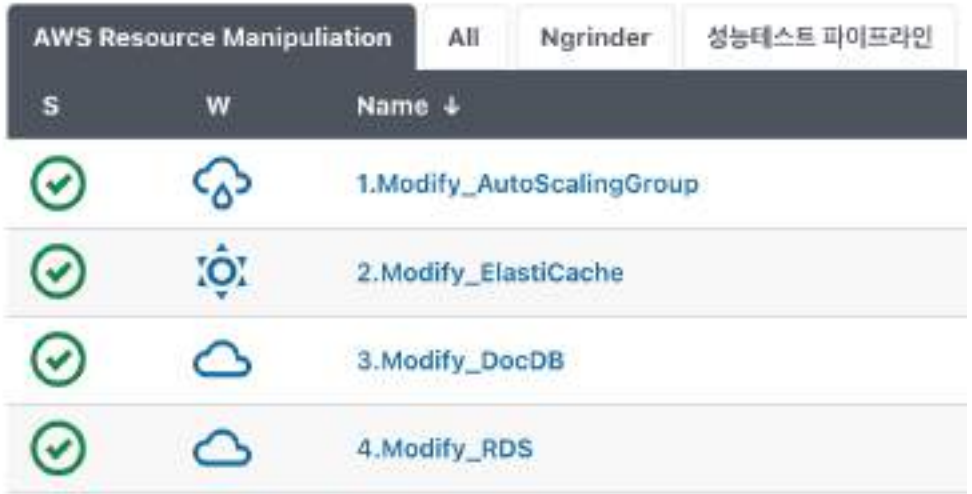
→ Ngrinder REST API와
shell script로 구현

성능테스트 수행 & 지표 기록 Job

어떻게 구현할까?

우아한 WOOWA N 2021

테스트 환경 구축 Job



테스트 환경 구축 Job

적당히
으쌰으쌰
조합해서..

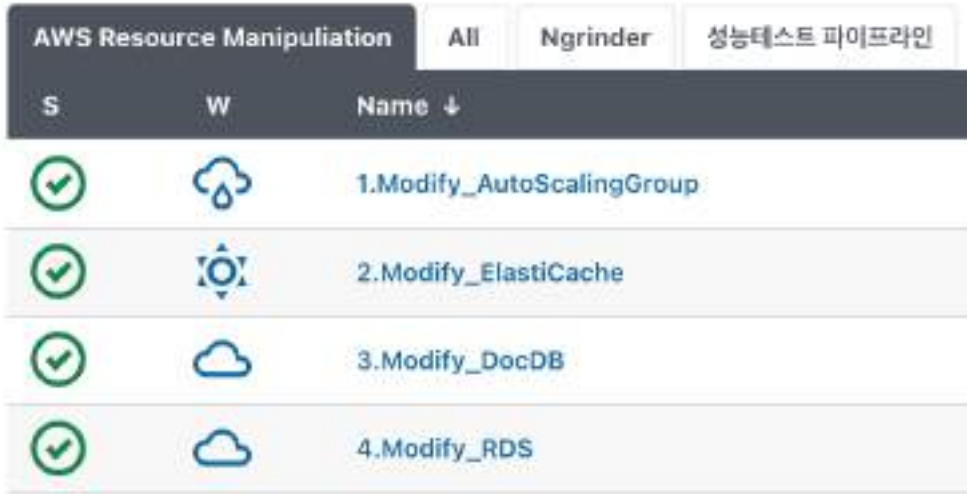
테스트 수행 및 지표 기록 Job

성능테스트 수행 & 지표 기록 Job

어떻게 구현할까?

우아한 WOOWA N 2021

테스트 환경 구축 Job



성능테스트 1차

테스트 환경 구축 Job

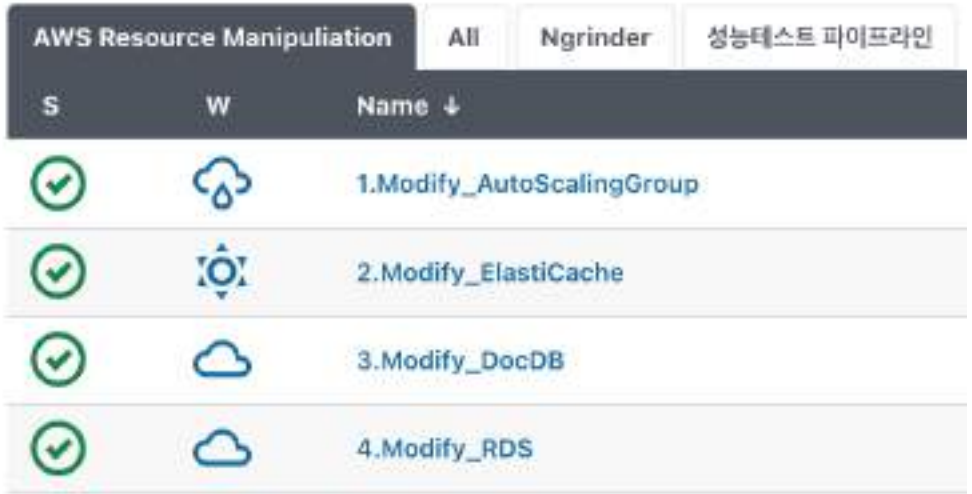
테스트 수행 및 지표 기록 Job

성능테스트 수행 & 지표 기록 Job

어떻게 구현할까?

우아한 WOOWA N 2021

테스트 환경 구축 Job



성능테스트 수행 & 지표 기록 Job

어떻게 구현할까?

우아콘 WOOWA CON 2021

테스트 환경 구축 Job

AWS Resource Manipulation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.Modify_AutoScalingGroup
✓	⚙	2.Modify_ElastiCache
✓	☁	3.Modify_DocDB
✓	☁	4.Modify_RDS

AWS Resource Manipulation | All | Ngrinder | 성능테스트 파이프라인

S	W	Name ↓
✓	☁	1.CloneAndStart_Test
✓	☁	2.CreateNewAndStart_Test

성능테스트 수행 & 지표 기록 Job

API 10대, TPS 1000	API 10대, TPS 1200	API 10대, TPS 1400	API 10대, TPS 1600	API 10대, TPS 1800	자원정리
25min 9s	6min 41s	6min 39s	6min 40s	6min 39s	30min 13s
25min 9s	6min 41s	6min 39s	6min 40s	6min 39s	30min 13s

어떻게 구현할까?

통합 파이프라인 조립

```
pipelines {
    stages {
        stage("API 서버 10대, Redis 2대") {
            environment {
                NUMBER_OF_API_SERVER = 10
                REDIS_NODE_TYPE = 'cache.r5.2xlarge'
                NUMBER_OF_REDIS_NODE = 2
                ...
            steps {
                build job: 'Congifure_TestEnvironments', parameters: [
                    string(
                        name: 'NUMBER_OF_API_SERVER',
                        value: NUMBER_OF_API_SERVER),
                    ...
                ]
                build job: 'Run_PerformanceTest', parameters: [
    ...
```

어떻게 구현할까?

통합 파이프라인 조립

```
pipelines {
    stages {
        stage("API 서버 10대, Redis 2대") {
            environment {
                NUMBER_OF_API_SERVER = 10
                REDIS_NODE_TYPE = 'cache.r5.2xlarge'
                NUMBER_OF_REDIS_NODE = 2
                ...
            steps {
                build job: 'Congifure_TestEnvironments', parameters: [
                    string(
                        name: 'NUMBER_OF_API_SERVER',
                        value: NUMBER_OF_API_SERVER),
                    ...
                ]
                build job: 'Run_PerformanceTest', parameters: [
    ...
```

어떻게 구현할까?

통합 파이프라인 조립

```
pipelines {
    stages {
        stage("API 서버 10대, Redis 2대") {
            environment {
                NUMBER_OF_API_SERVER = 10
                REDIS_NODE_TYPE = 'cache.r5.2xlarge'
                NUMBER_OF_REDIS_NODE = 2
                ...
            steps {
                build job: 'Congifure_TestEnvironments', parameters: [
                    string(
                        name: 'NUMBER_OF_API_SERVER',
                        value: NUMBER_OF_API_SERVER),
                    ...
                ]
                build job: 'Run_PerformanceTest', parameters: [
...
```

AWS Resource Manipulation Job으로 구성

어떻게 구현할까?

통합 파이프라인 조립

```
pipelines {
    stages {
        stage("API 서버 10대, Redis 2대") {
            environment {
                NUMBER_OF_API_SERVER = 10
                REDIS_NODE_TYPE = 'cache.r5.2xlarge'
                NUMBER_OF_REDIS_NODE = 2
                ...
            steps {
                build job: 'Congifure_TestEnvironments', parameters: [
                    string(
                        name: 'NUMBER_OF_API_SERVER',
                        value: NUMBER_OF_API_SERVER),
                    ...
                ]
                build job: 'Run_PerformanceTest', parameters: [
...
```

성능테스트 수행 Job 으로 구성

어떻게 구현할까?

'뭘로 구현 하느냐' 보다는 '어떻게 문제를 푸느냐'

자동화를 통해 얻을 수 있는 이점들

✅ 실수 없이 같은 성능테스트를 여러 번 재현 가능

✅ 옆에서 항상 기록하면서 지켜봐야 하는 사람이 필요 없음

✅ 불필요한 자원 낭비를 줄일 수 있음

성능테스트 자동화, 어떤 시스템에 유용할까?

✅ 트래픽이 많고, 서버 성능 측정을 해야할 일이 잦은 경우

- ✓ 이벤트 시스템과 같이 새로운 기획이 잦고,
  일회성으로 많은 트래픽을 받아야 하는 경우
- ✓ 저장소 변경이 비교적 잦은 경우
- ✓ 피크타임과 그렇지 않은 시간대 사이에 트래픽 변동이 커서
  scale out & scale in이 자주 일어나는 경우

앞으로 우리는...

- 🤔 다양한 시나리오에 대한 테스트를 수행하고 기록하여 자산으로 남기는 일
- 🤔 장애 상황에 대한 기록과 시뮬레이션

서버 성능테스트,
클릭 한 번으로
끝내볼 수 있을까?

우아콘 WOOWA CON 2021

서버 성능테스트, 클릭 한 번으로 끝내볼 수 있을까?

C 우아콘 WOOWA N 2021

Reference

결제시스템 성능, 부하, 스트레스 테스트

🔗 https://techblog.woowahan.com/2572/

서버 사이드 테스트 자동화 여정

🔗 https://engineering.linecorp.com/ko/blog/server-side-test-automation-4/

AWS CLI Command Reference

🔗 https://docs.aws.amazon.com/cli/latest/index.html

Ngrinder Rest API

🔗 https://github.com/naver/ngrinder/wiki/REST-API-PerfTest

들어주셔서
감사합니다

Thank you!

C 우아콘
WOOWA
N 2021