

Search Reliability Engineering (지진에도 흔들리지 않는 네이버 검색시스템)

김재현, 손주식

System&Solution

NAVER

CONTENS

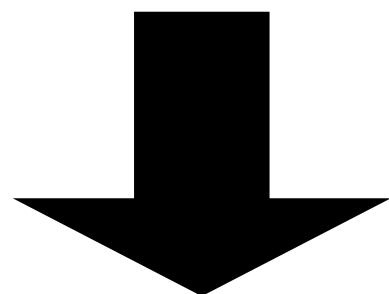
1. 들어가며
2. 네이버 검색시스템의 SRE
3. 실제 사례 소개
4. Search Reliability Engineer

1. 들어가며

네이버 검색시스템의 목표

“많은 사용자가 동시에 검색어를 입력해도 서비스에 문제가 없어야 한다.”
대용량 처리 (High Throughput)

“네이버 검색창에 검색어를 입력하면 1초 안에는 결과가 나와야 한다.”
짧은 대기시간 (Low Latency)



결국 장애가 일어나지 않는 것이 가장 중요
장애가 발생하면 대용량 처리와 짧은 대기시간을 보장할 수 없음

소방관 이야기 (1)

우리 기억 속의 소방관



소방관 이야기 (1)

우리 기억 속의 소방관



평소에 잘 대비하는 것이 훨씬 비용 효율적



소방관 이야기 (2)

두 가지 난제

얼마나 비용 효율적인지 증명할 수 있는가?

→ 정확한 비용 측정 / 예측의 중요성

화재가 나지 않았을 때, 예방 활동 덕분이었다는 것을 어떻게 증명할 수 있는가?

→ 정확하고 구체적인 경보 체계 확립 필요

→ 정확한 사후분석 (post-mortem) 필요

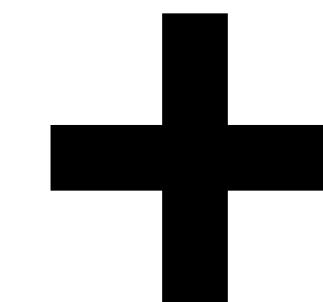
네이버 검색시스템의 현재 상황

수백 개의 검색 서비스

수만 대의 서버 장비

하루 수십억 건의 검색 요청

수백억 건의 컨텐츠

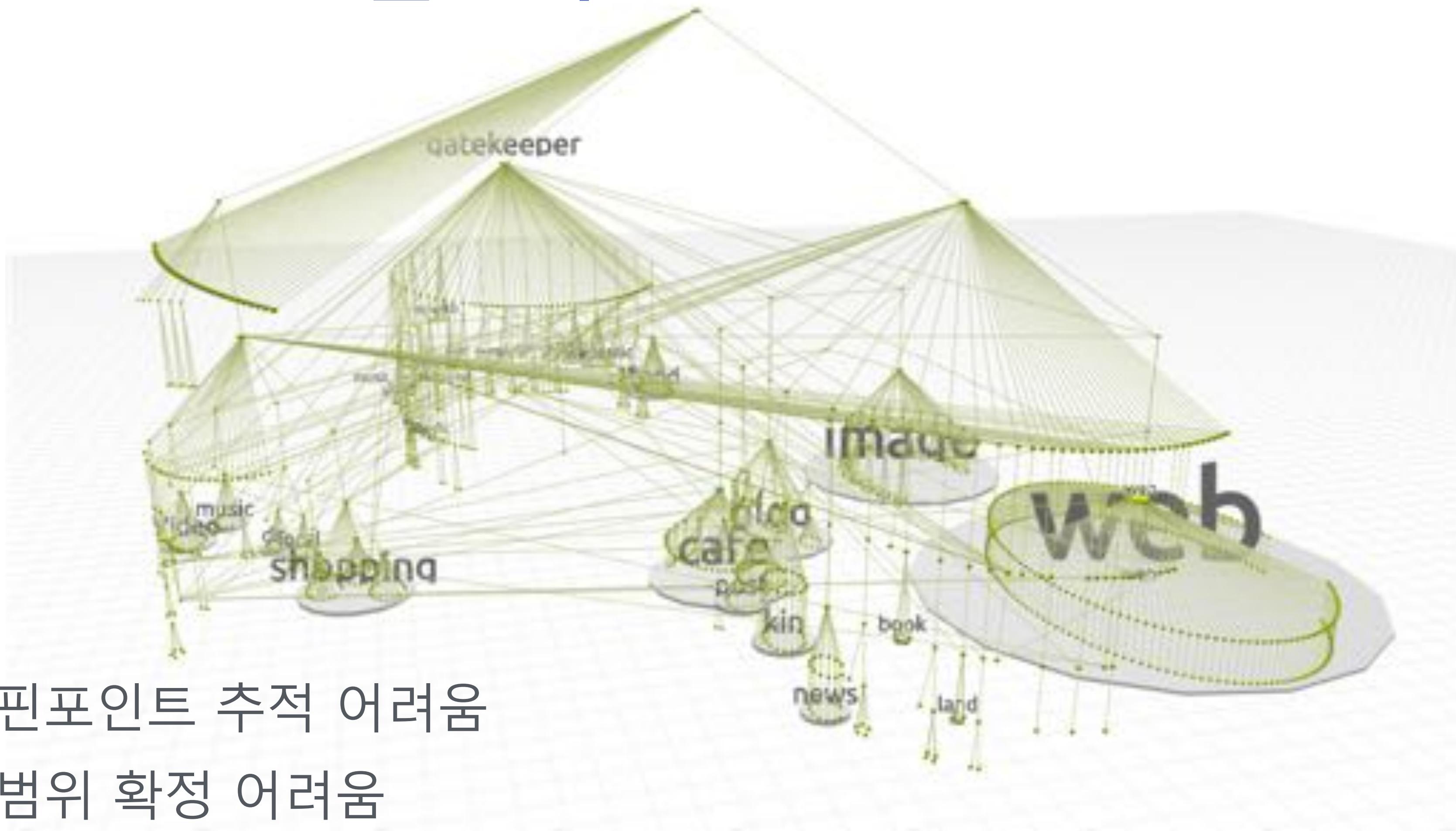


수십 개의 조직

수백 명의 구성원

다양한 엔진, 시스템, 도구들

시스템이 거대해질수록 어려워지는 부분



문제 원인 핀포인트 추적 어려움

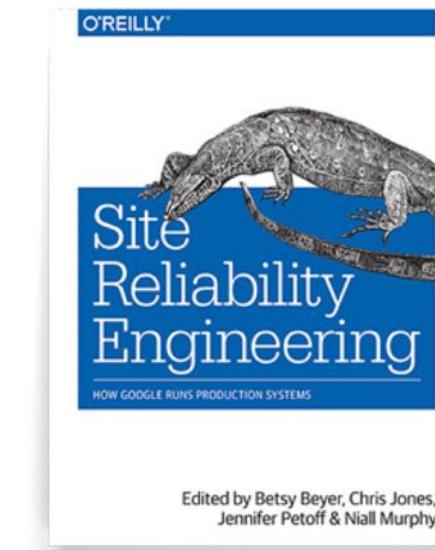
문제 영향 범위 확정 어려움

장애 복구 완료 후에도 모든 구성요소 정상화 확인 어려움

SRE : Site Reliability Engineering

SRE란?

글로벌 스케일의 서비스를 제공하면서 어떻게 하면 시스템의 신뢰성을 보장할지 고민하는 기술 분야이자 방법론, 문화



SRE : Search Reliability Engineering

네이버 검색시스템에서 바라보는 SRE

모든 검색서비스 정상 동작

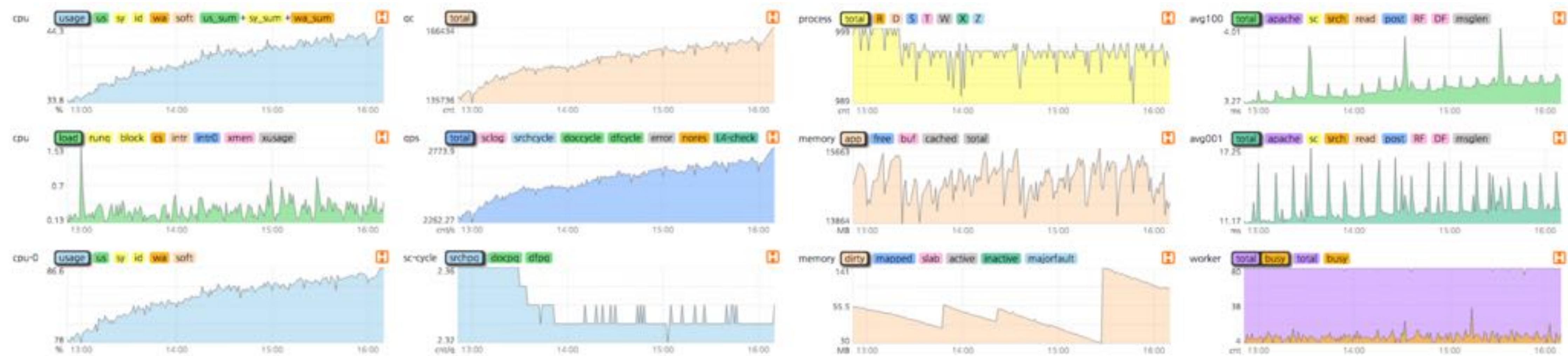
1년 10분 이하 다운타임

고비용 사후처리보다 저비용 사전예방

이 목표 달성을 위한 모든 활동

2. 네이버 검색시스템의 SRE

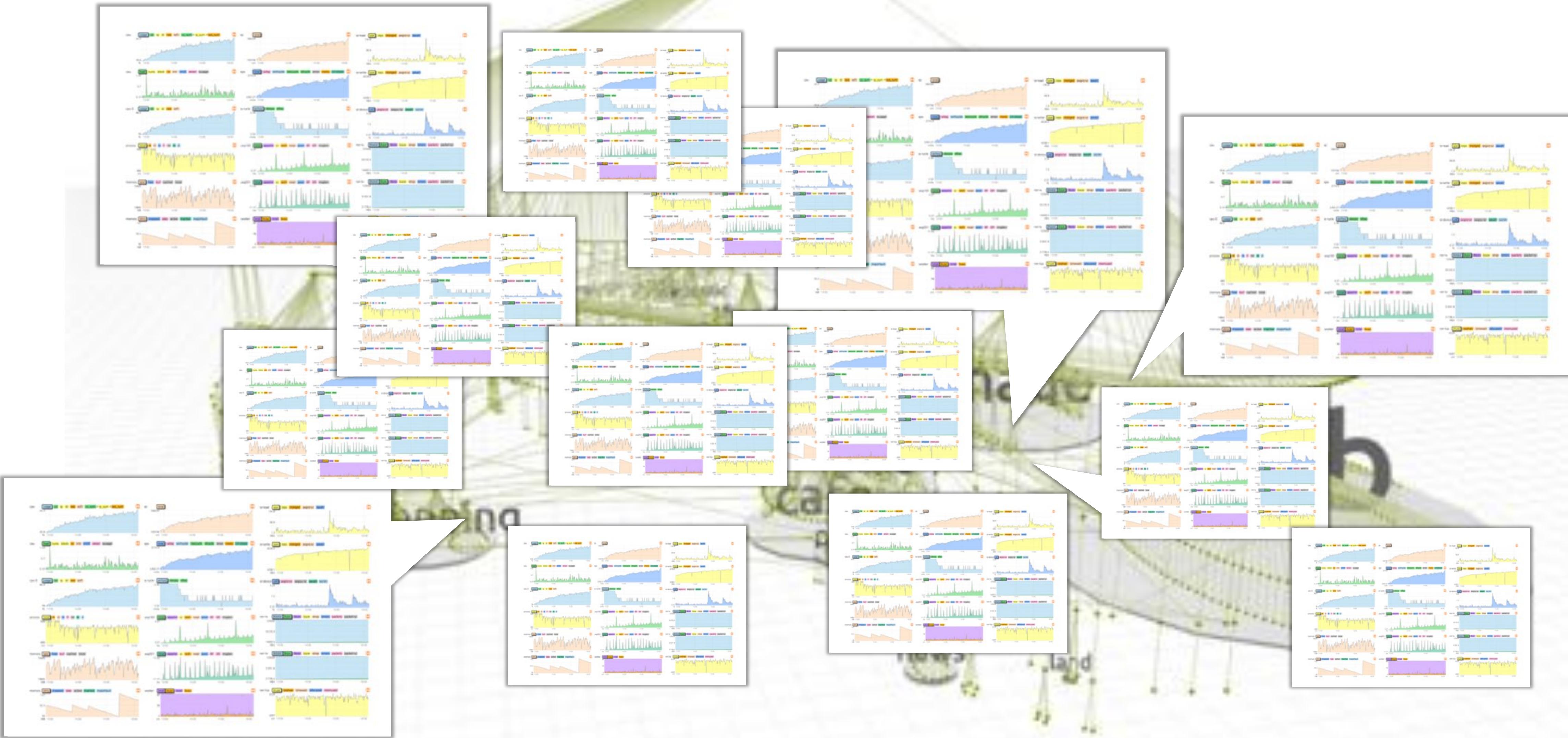
SRE 도입 전 상황



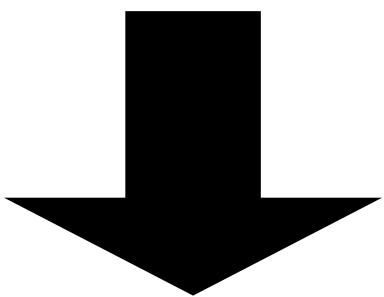
시스템 운영 지표 관측

검색 요청 트래픽, 서버 응답시간, CPU사용량, 디스크I/O, 네트워크 사용량 등등 ...

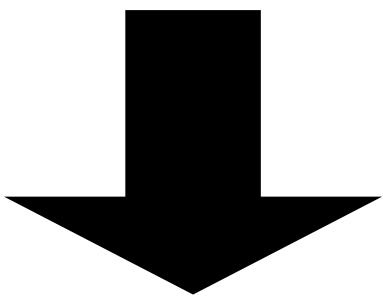
SRE 도입 전 상황



단일 Host 또는 단일 서버 대상 지표 수집



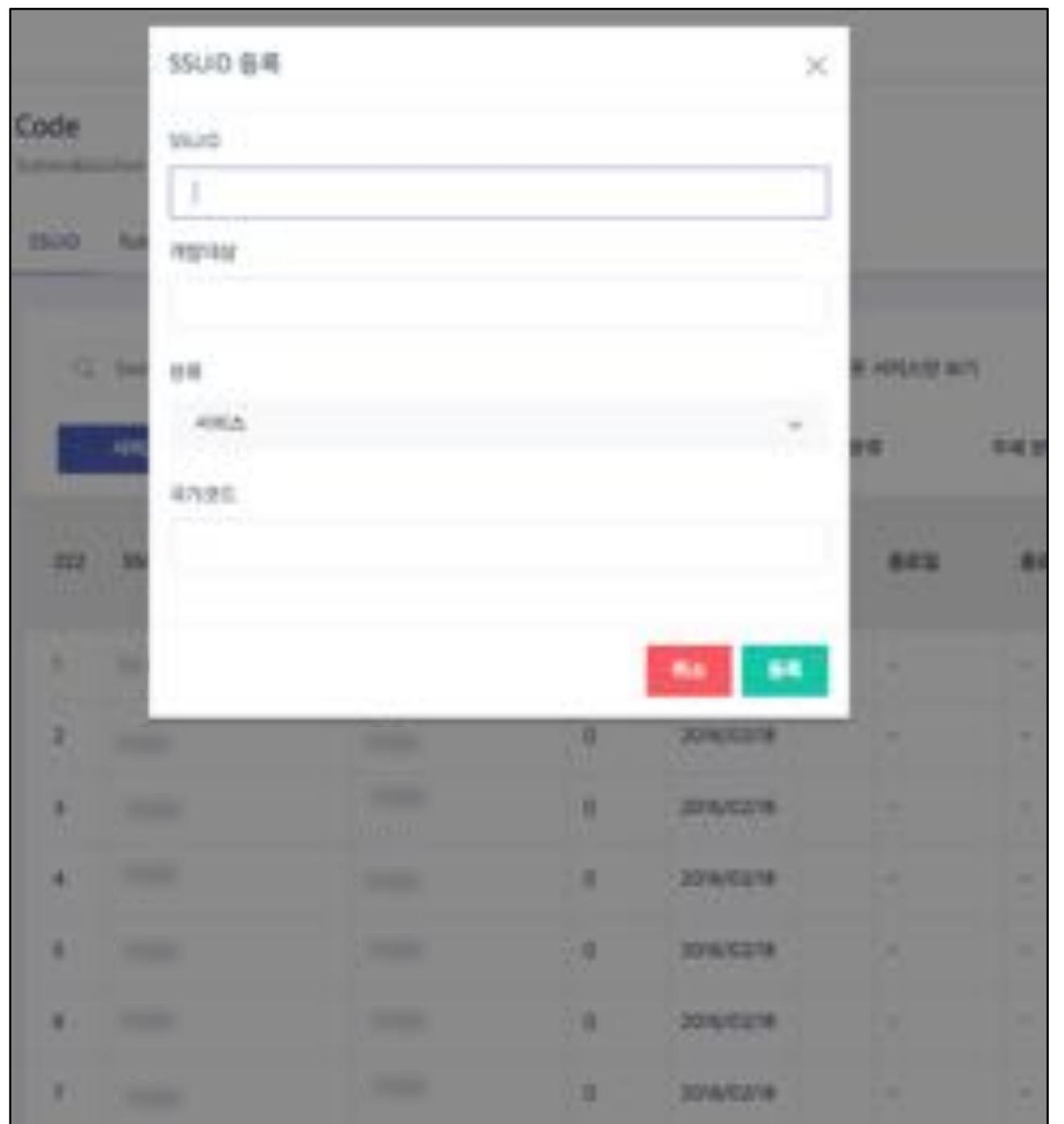
특정 서비스의 전체 레이어 및 서비스 군,
전체 시스템을 볼 수 있는 방법이 없음



효율 정보 체계 도입

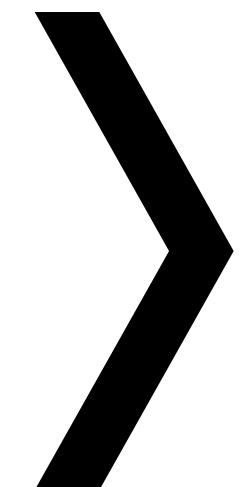
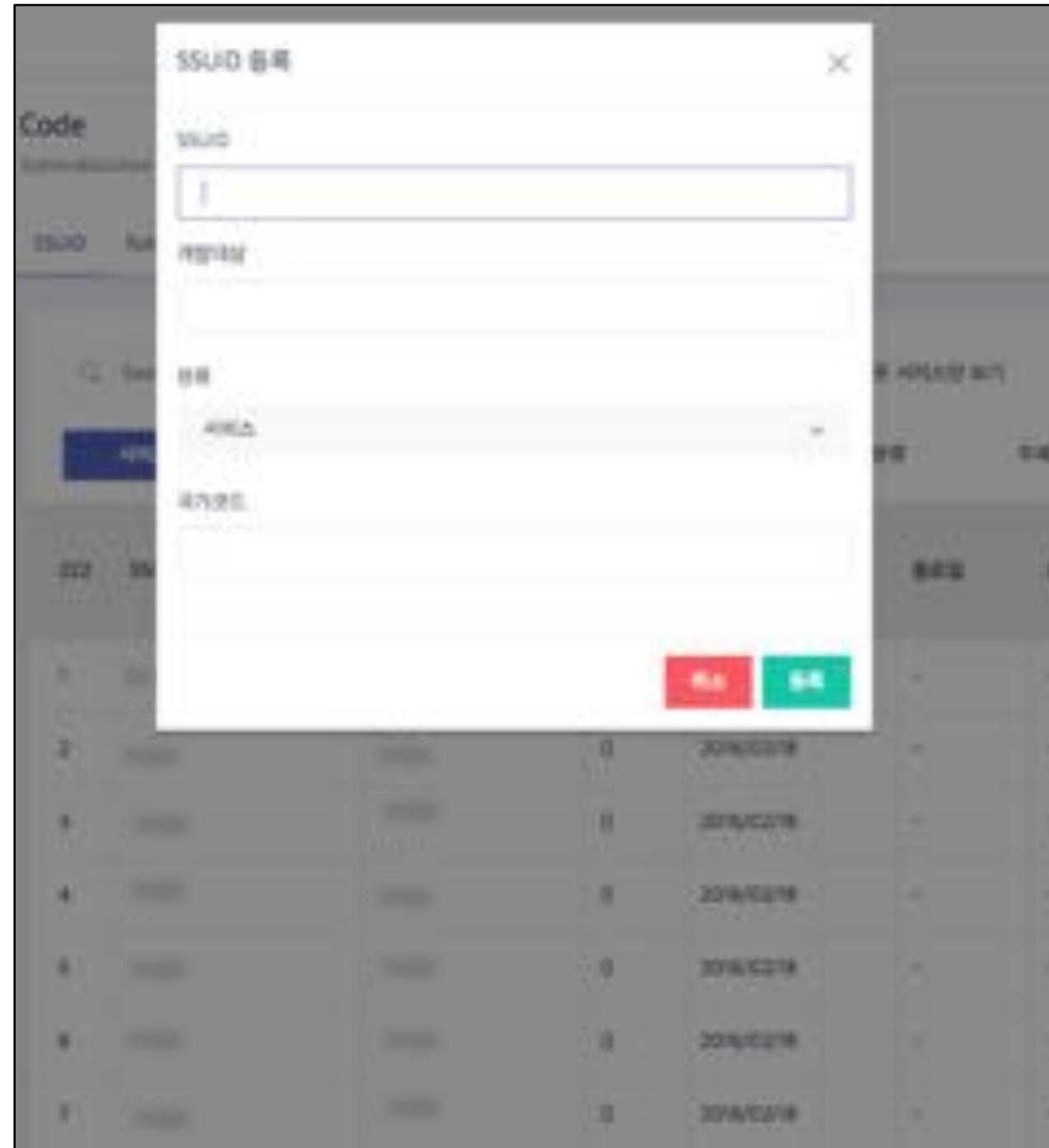
효율적인 정보 체계 도입

서비스 ID 발급

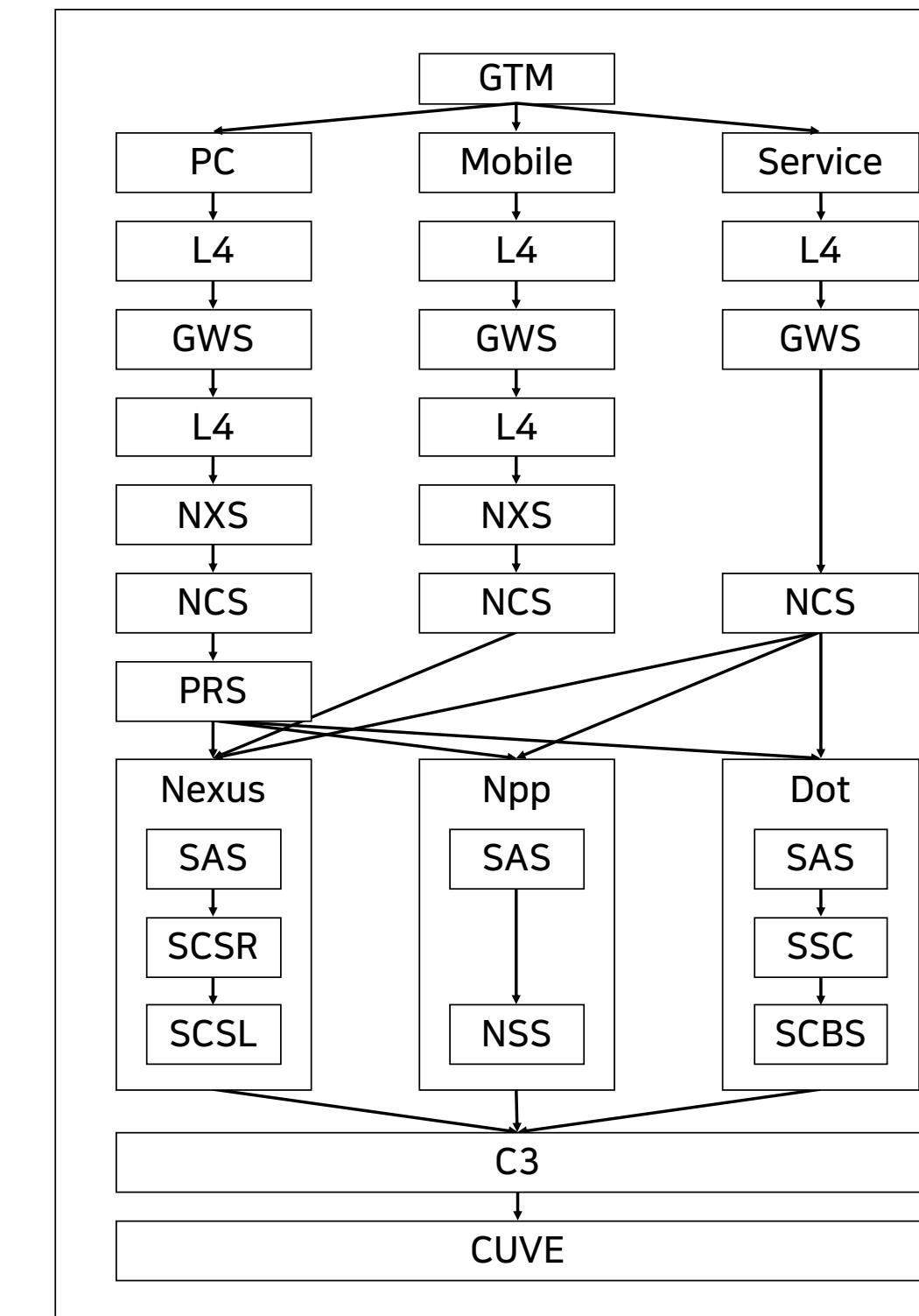


효율적인 정보 체계 도입

서비스 ID 발급



구조 가시화

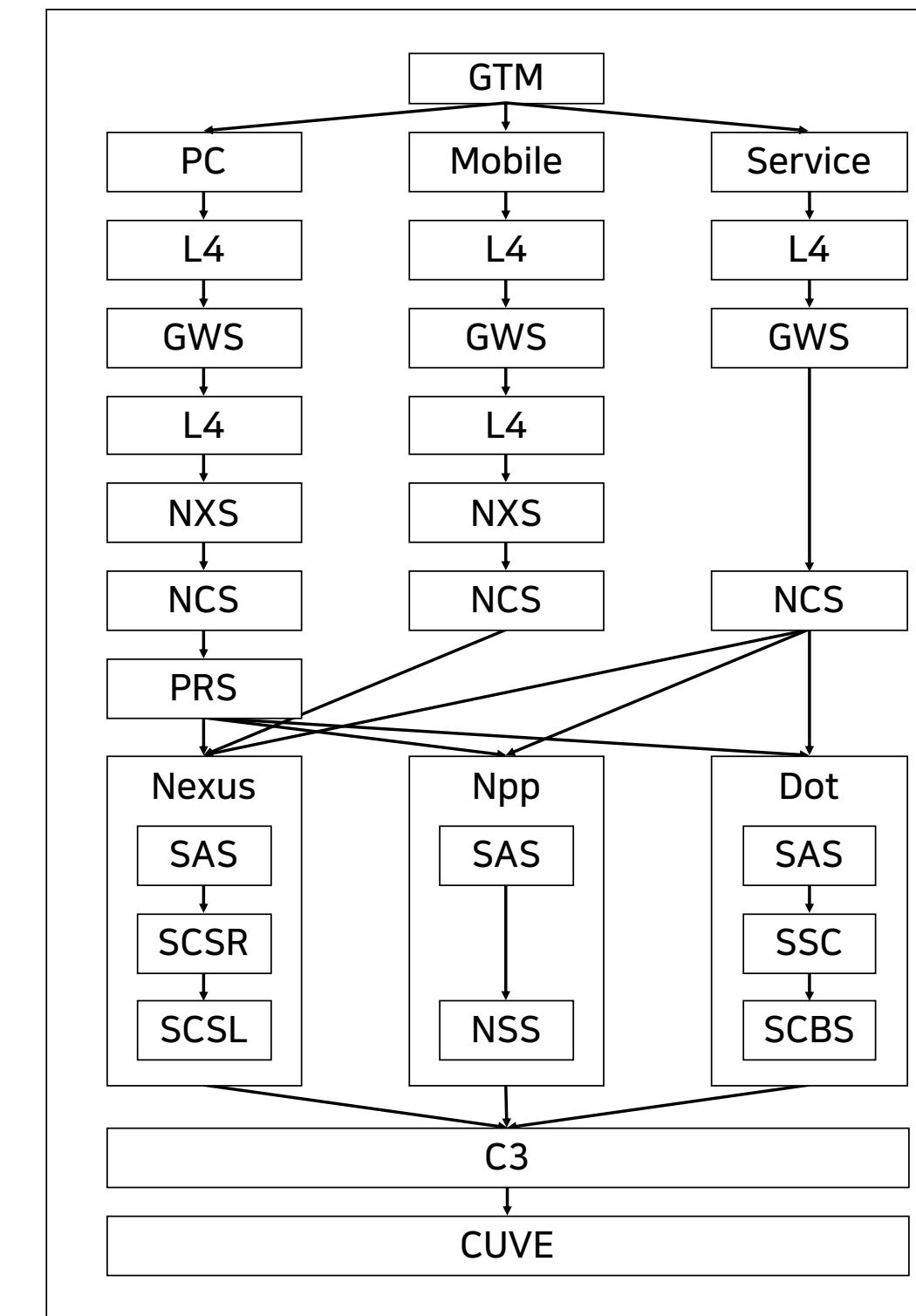


효율적인 정보 체계 도입

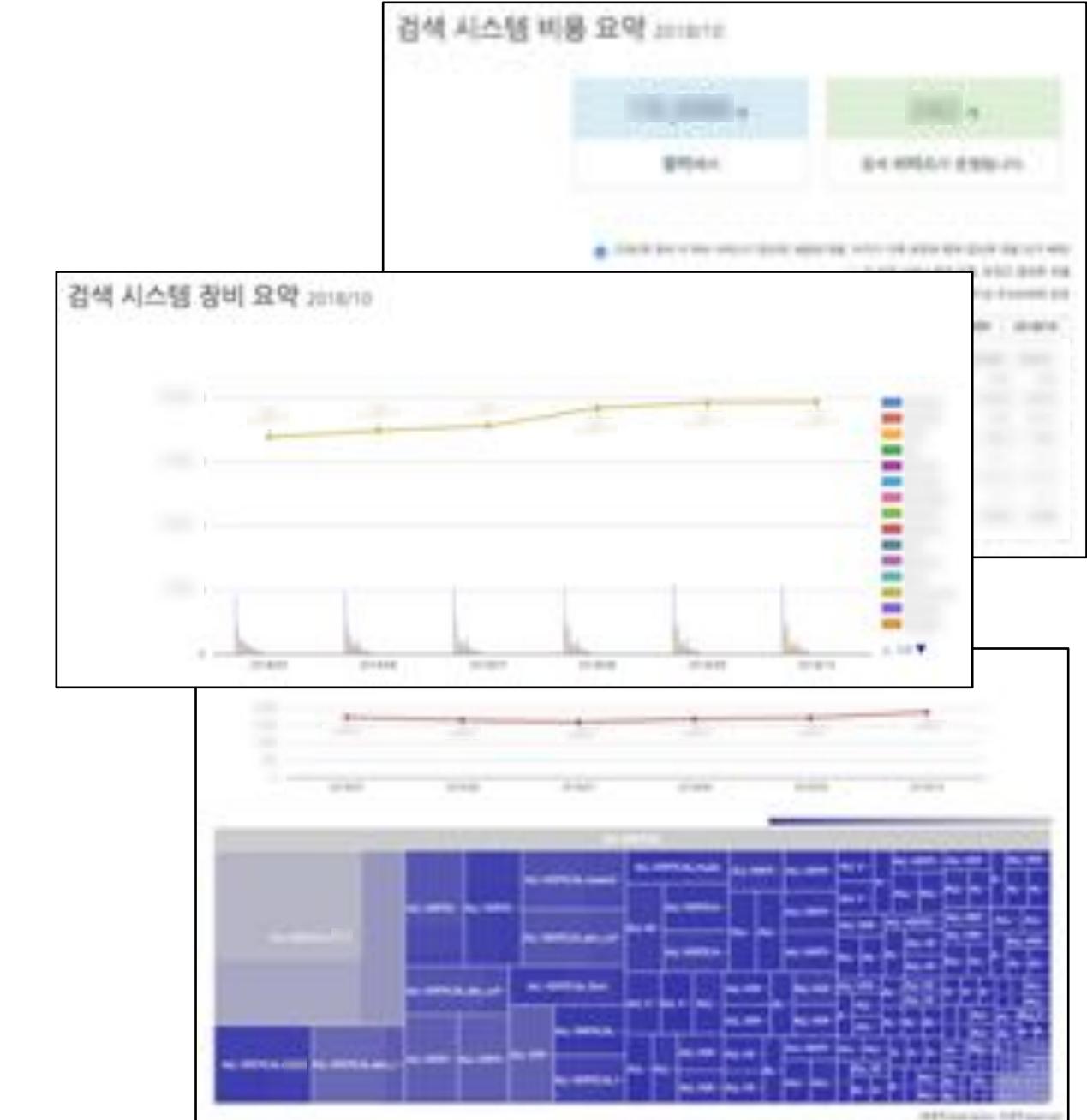
서비스 ID 발급



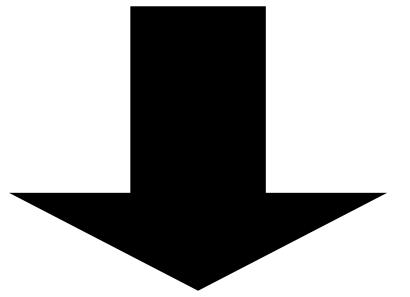
구조 가시화



성능 / 비용 측정



정보가 모이니 다양한 응용 방법 고민 시작



네이버 검색시스템만의 가용량 지표 개발

가용량 지표 개발 (1)

기존 방식

$$\text{Availability} = \frac{MTBF}{MTTR + MTBF}$$

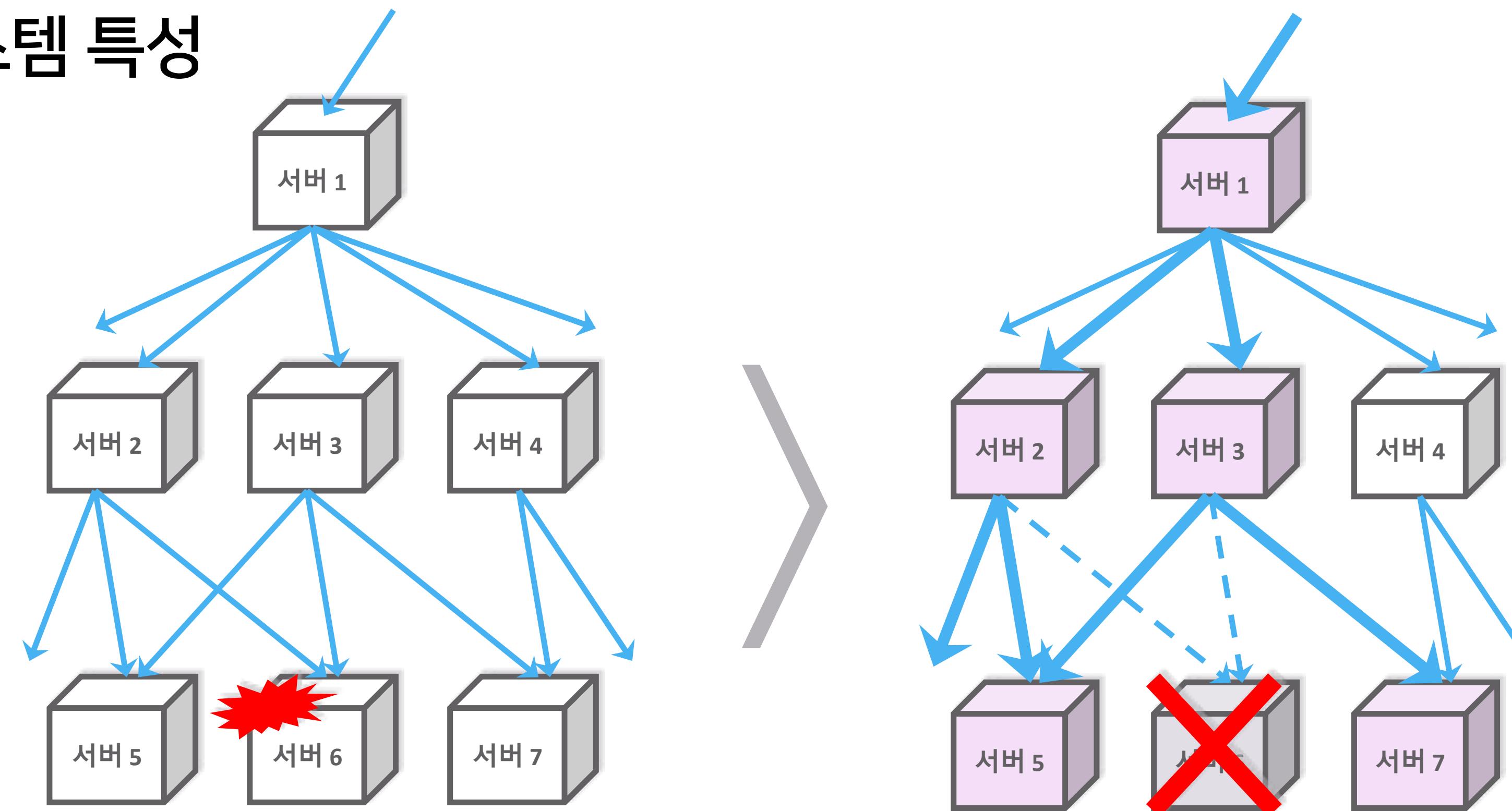
MTTR (Mean Time To Repair, 평균 복구 시간)

MTBF (Mean Time Between Failures, 평균 무고장 시간)

99.998%인데, 좋은 숫자인데... → 1년 동안 10분 장애
네이버 통합검색 10분 장애는 대재난!!!

가용량 지표 개발 (2)

대용량 분산 시스템 특성



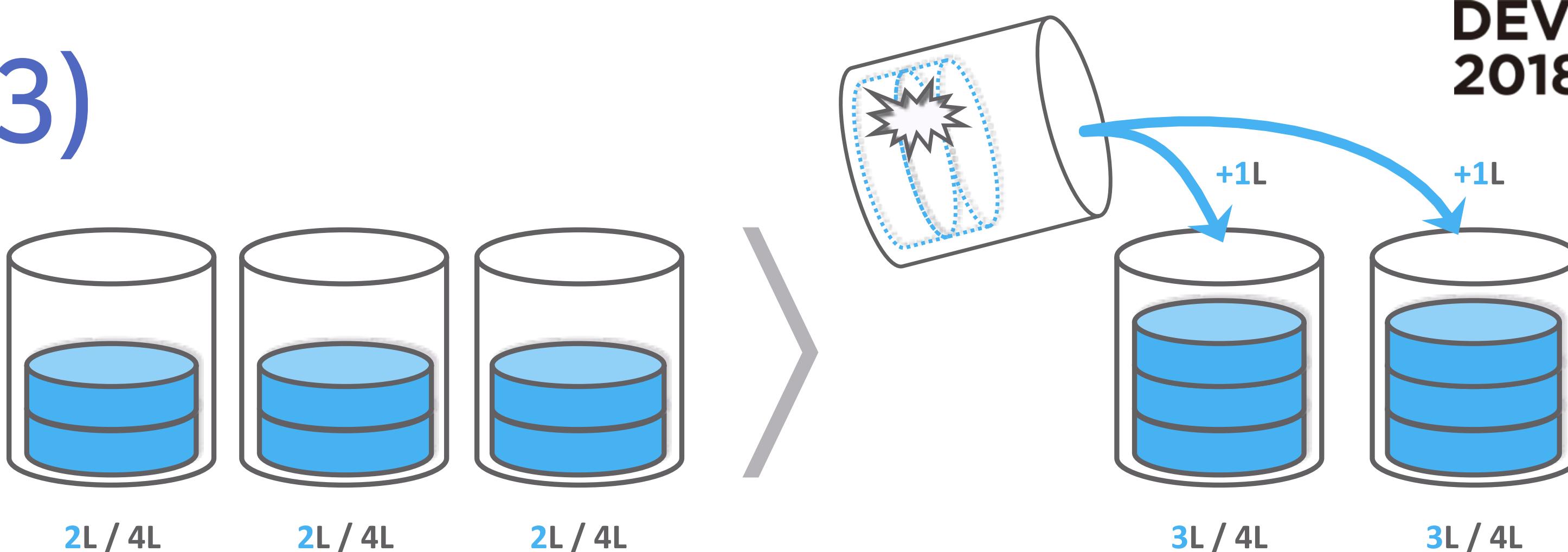
특정 서버에 문제가 생기면 다른 서버들이 많은 영향을 받는다!

가용량 지표 개발 (3)

새로운 방식

부하증가배수

한 친구가 죽으면 나머지 친구들은
몇 배를 받나?

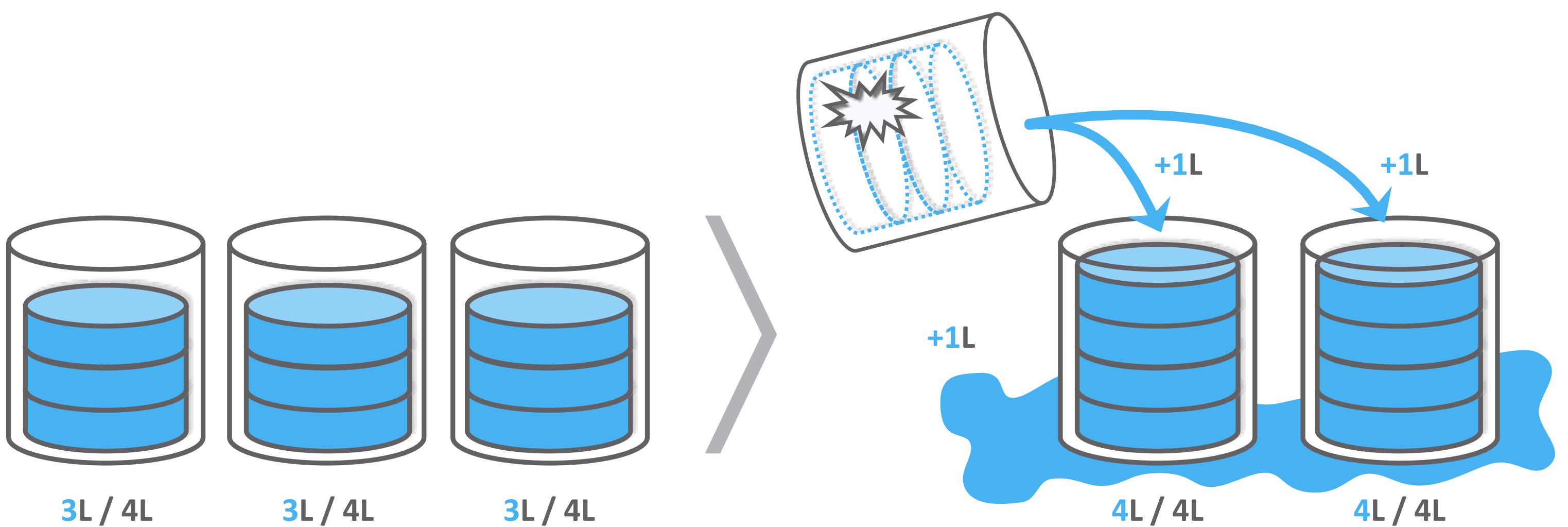


최대가용배수

한 명이 현재 몇 배까지 받을 수 있나?

“임계 상황” 판단

부하증가배수 > 최대가용배수



가용량 경보 시스템 운영

새로운 가용량 지표의 활용

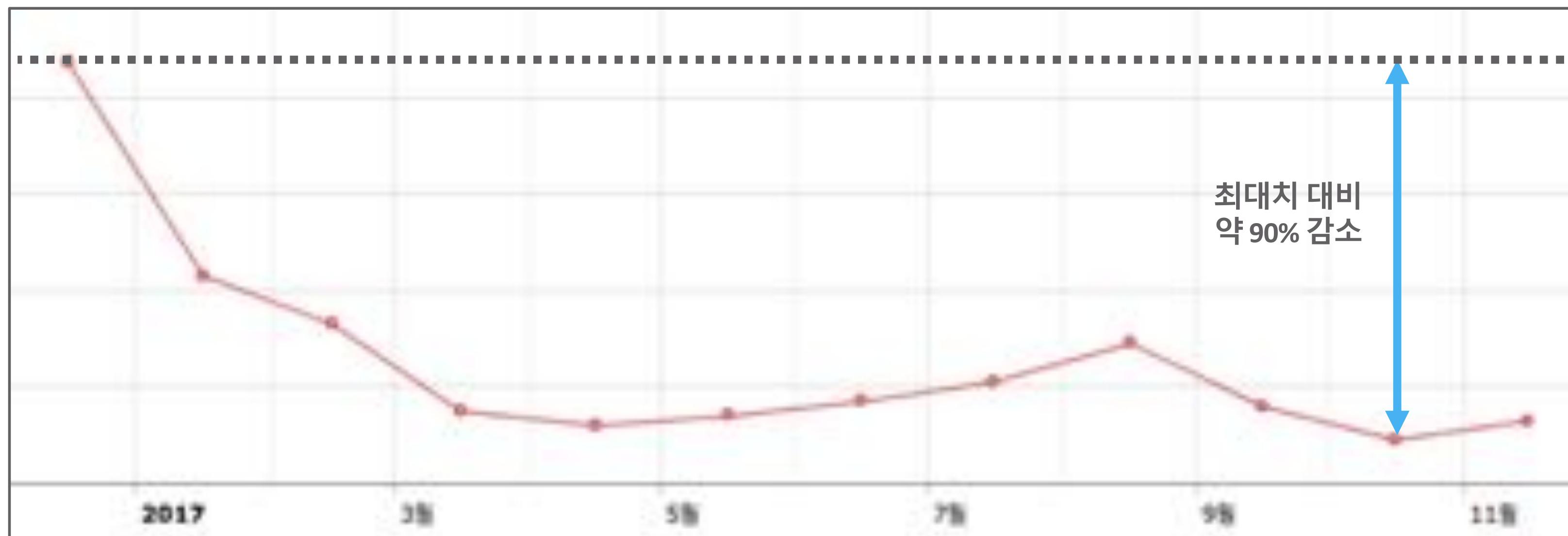
가용량 경보 발생 시 미리 경보

각 서비스 담당자들에게 성능 및 컨설팅 정보 제공

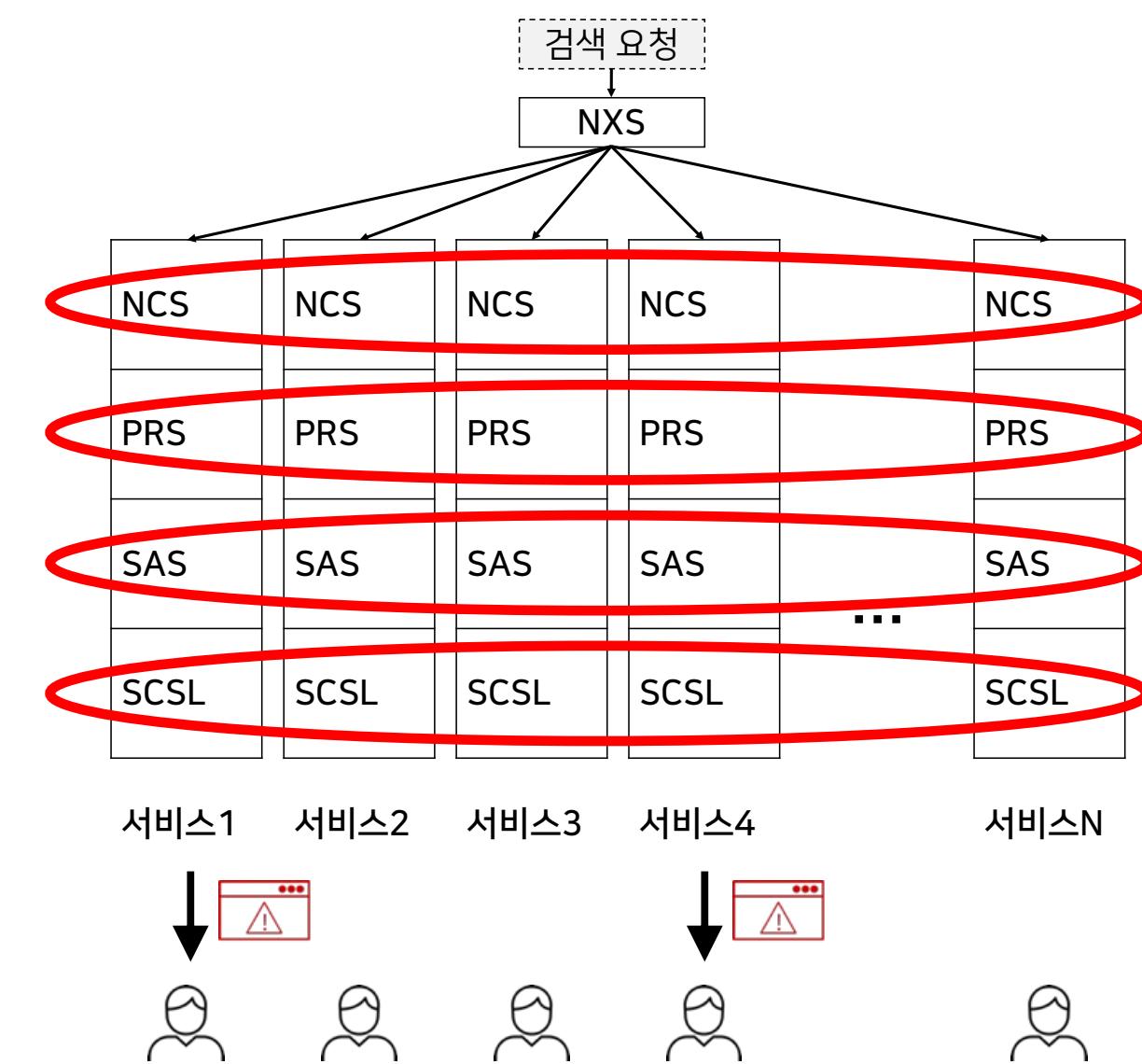
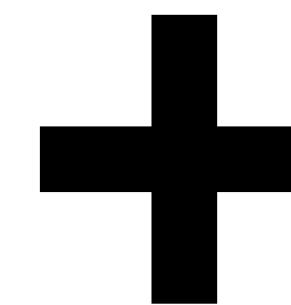
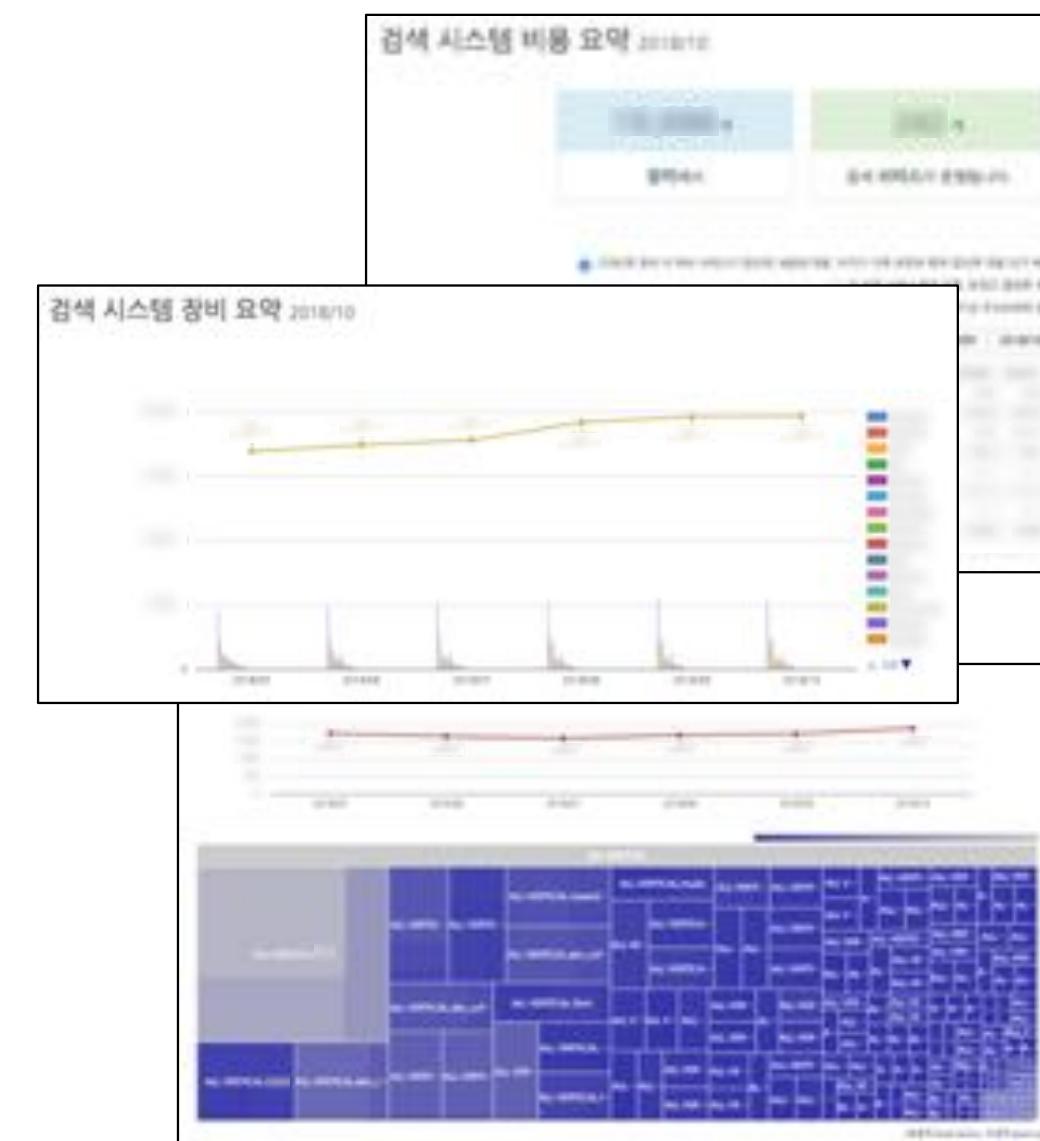


가용량 경보 시스템 운영 효과

임계 상황 발생 횟수 대폭 감소



검색시스템 통합 대시보드 개발



성능, 비용 정보

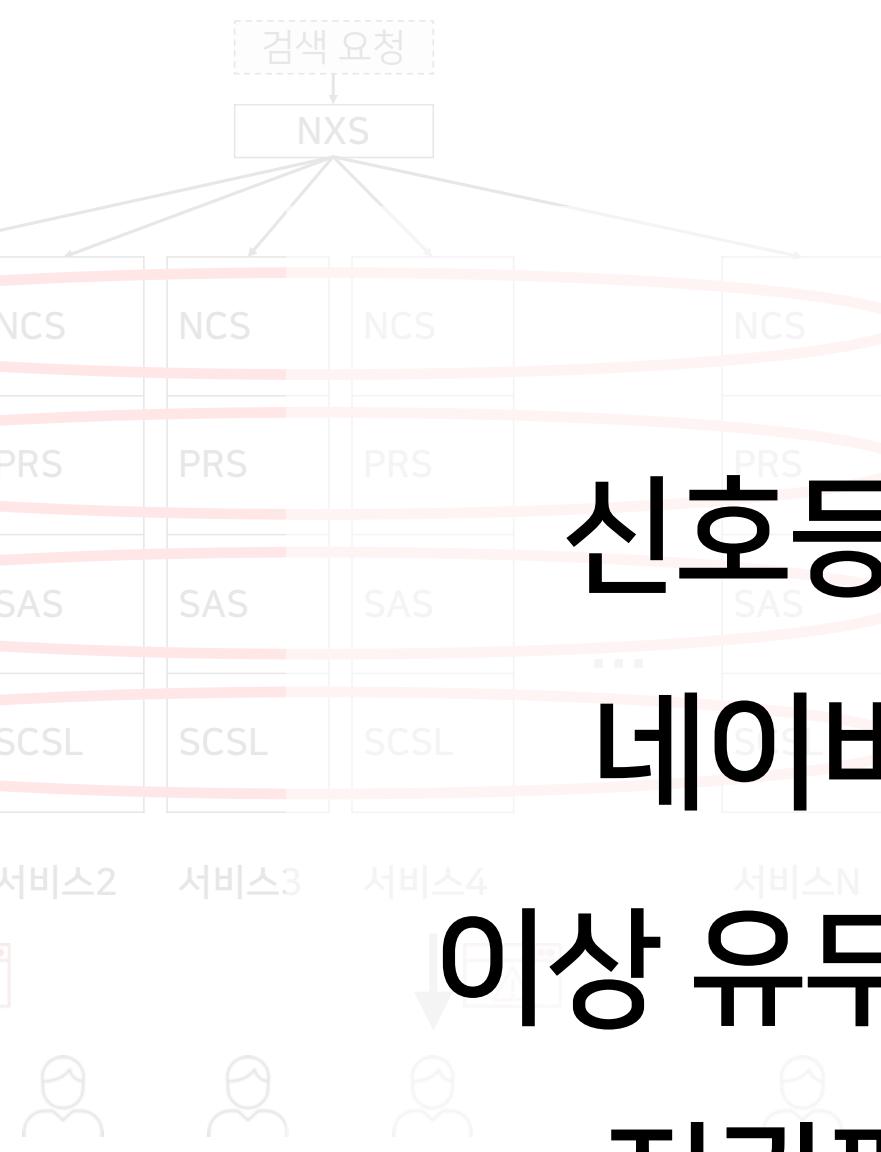
Tagging, Mapping

서비스 단위 가용량 정보

계층별 취합

검색시스템 통합 대시보드

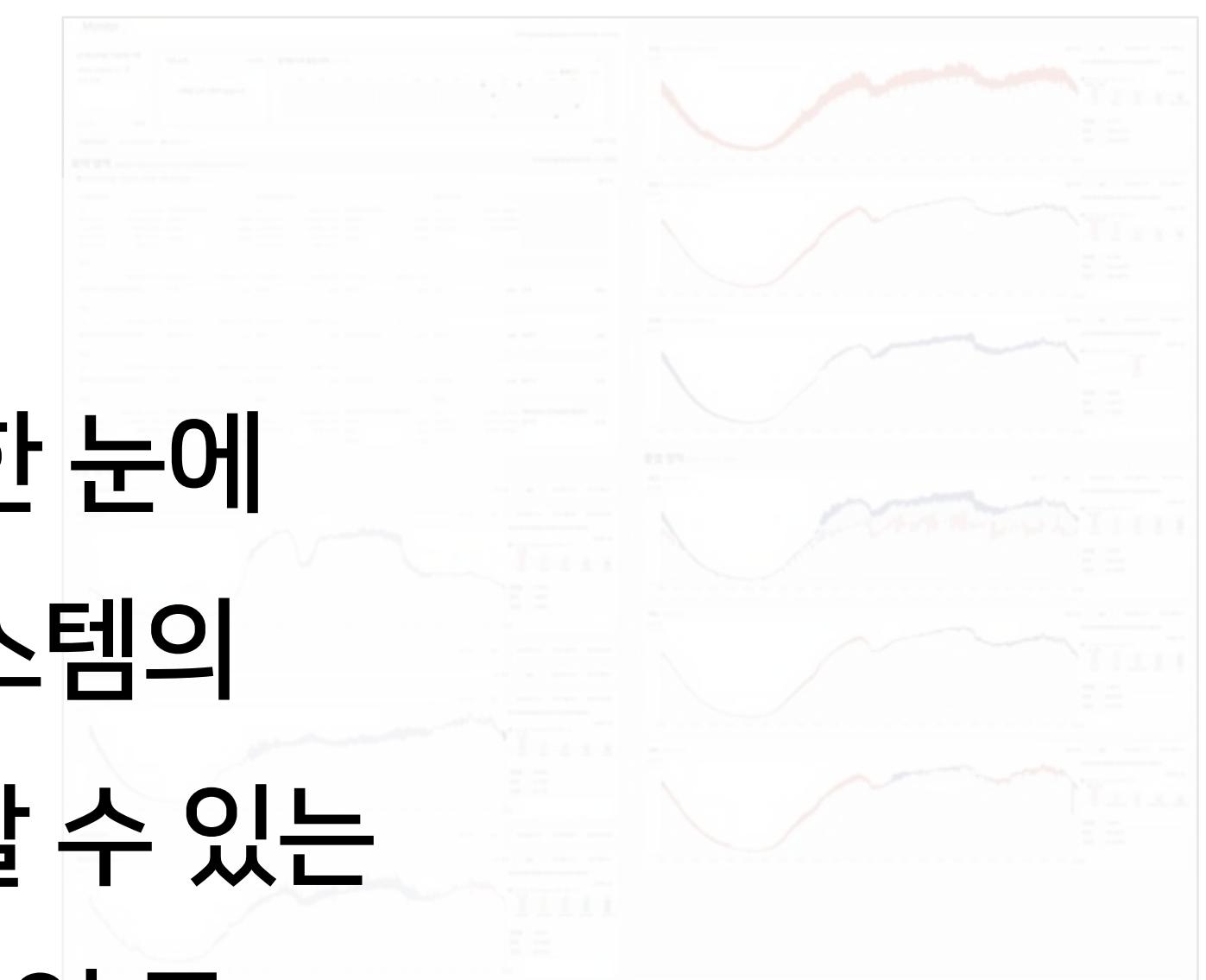
검색시스템 통합 대시보드 개발

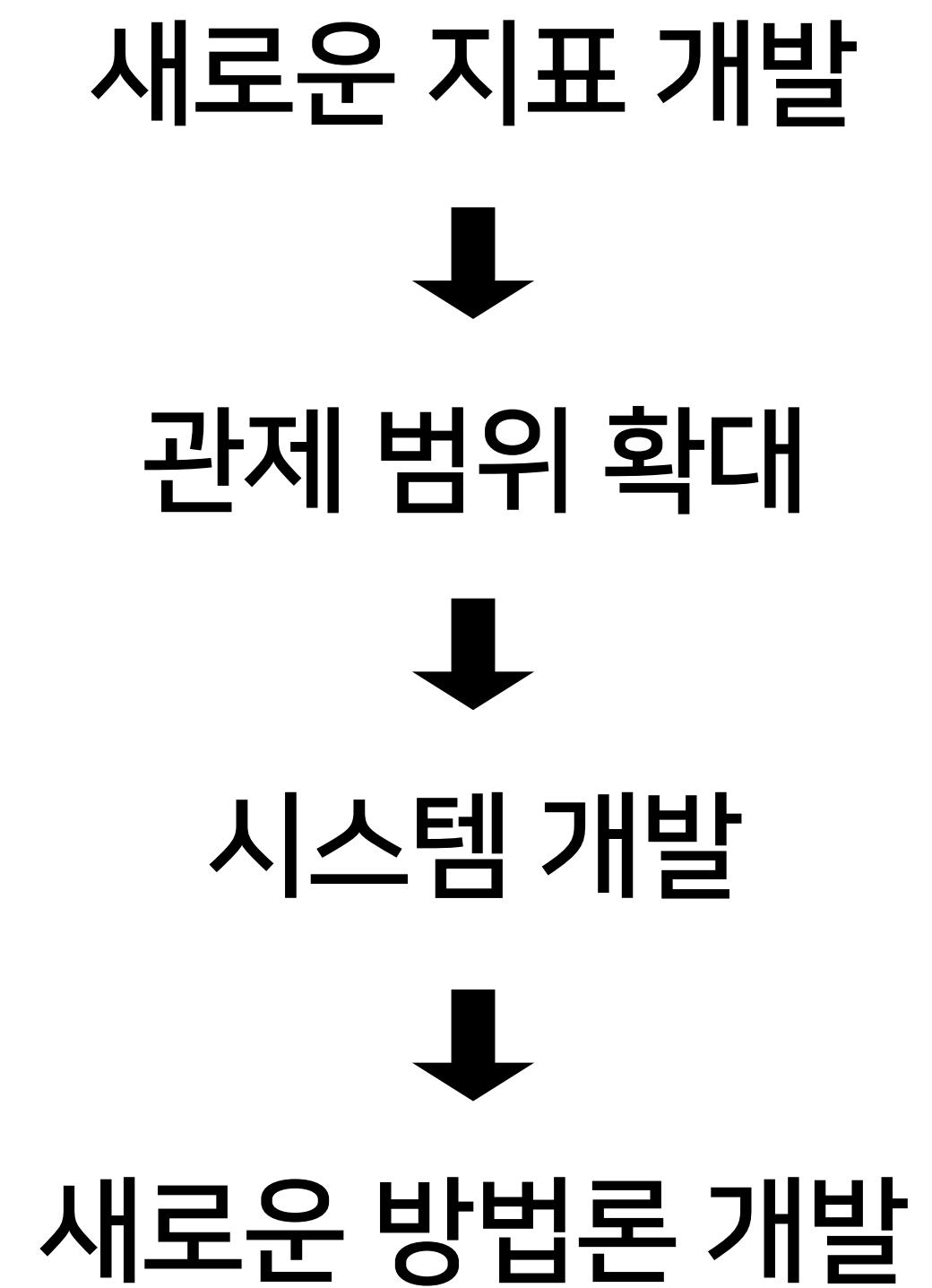


성능, 비
스 단위 가용량 정보
계층별 취합

검색시스템 통합 대시보드
Tagging)

신호등 형태로 한 눈에
네이버 검색시스템의
이상 유무를 확인할 수 있는
전광판 제작, 운영 중



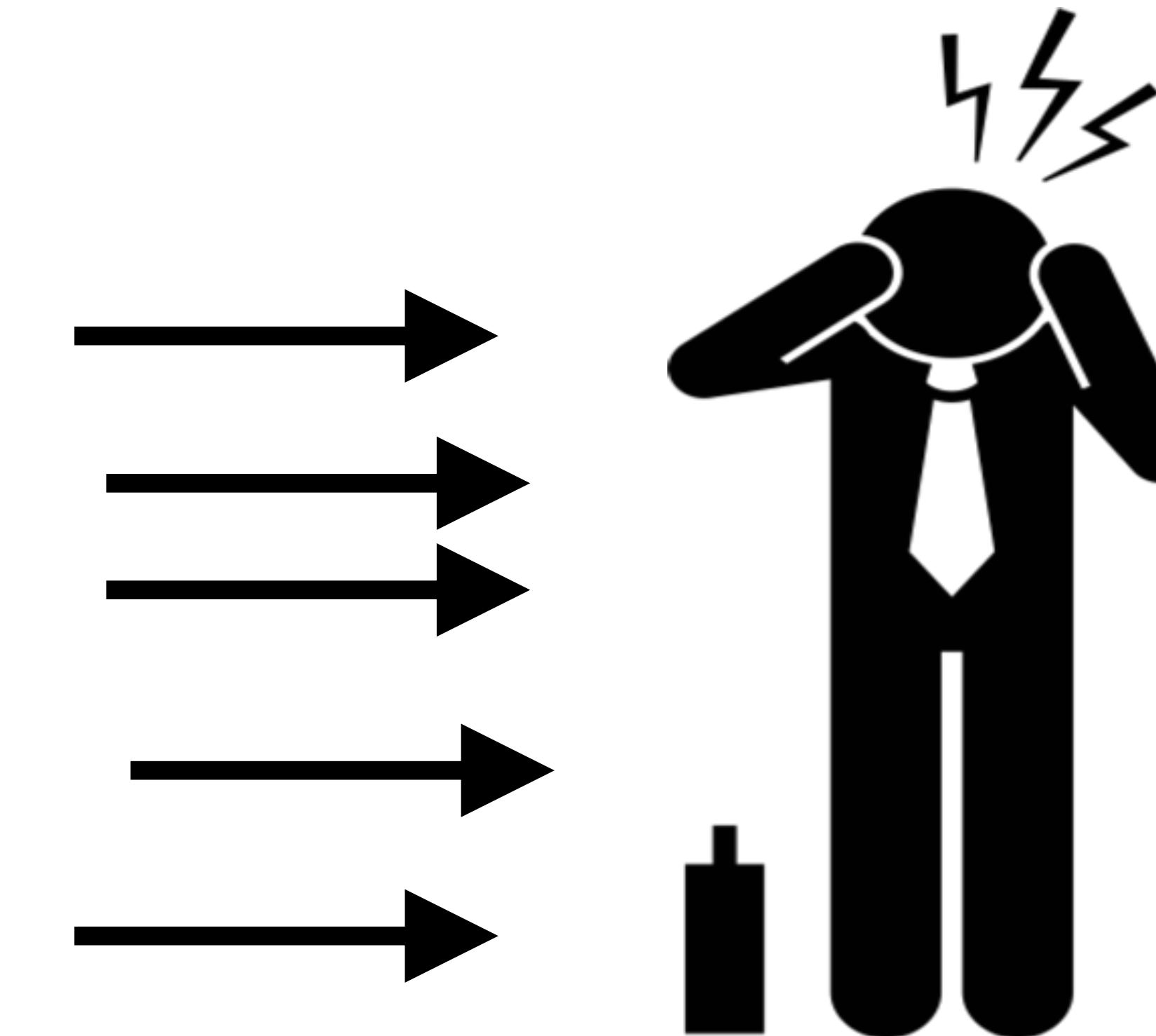


엄청난 양의 경보 폭탄 발생

사례 : 9시간 동안 약 330 건의 경보 발생
(평균 2분마다 1 건 수준)

[감색서버팜(태국어) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(한국) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(영어) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(인도네시아어) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(간체) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(일본어) SAS] 가용성 일개상황 해제	2017-12-06 11:08
[감색서버팜(태국어) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(한국어) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(한국) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(영어) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(인도네시아어) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(일본어) SAS] 일개상황 1.3<=>1.3 farm c3a057	2017-12-06 11:07
[감색서버팜(태국어) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(일본어) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(한국) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(영어) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(인도네시아어) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(한국어) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(간체) SAS] 가용성 일개상황 해제	2017-12-06 11:04
[감색서버팜(인도네시아어) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(일본어) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(한국) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(간체) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(한국어) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(태국어) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(영어) SAS] 일개상황 1.2<=>1.3 farm c3a057	2017-12-06 11:03
[감색서버팜(한국) SAS] 일개상황 해제	2017-12-06 11:00
[감색서버팜(간체) SAS] 가용성 일개상황 해제	2017-12-06 11:00

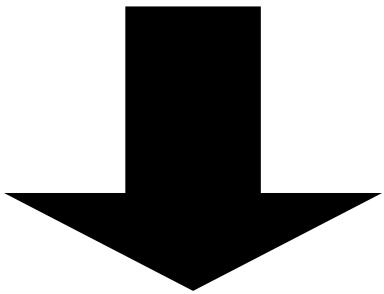
DEVIEW 2018



혹독한 "경보 피로"를 겪는 SRE

[감색서버풀(태국어) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(한국) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(영어) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(인도네시아어) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(간체) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(일본어) SAS] 가용성 경계상황 해제	2017-12-06 11:08
[감색서버풀(태국어) SAS] 등계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(한국어) SAS] 등계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(한국) SAS] 경계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(영어) SAS] 경계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(인도네시아어) SAS] 등계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(일본어) SAS] 등계상황 1.3<=1.3 farm c3a057	2017-12-06 11:07
[감색서버풀(태국어) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(한국) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(영어) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(인도네시아어) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(한국어) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(간체) SAS] 가용성 경계상황 해제	2017-12-06 11:04
[감색서버풀(인도네시아어) SAS] 등계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(한국) SAS] 등계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(영어) SAS] 경계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(간체) SAS] 경계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(한국어) SAS] 경계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(태국어) SAS] 등계상황 1.2<=1.3 farm c3a057	2017-12-06 11:03
[감색서버풀(영어) SAS] 경계상황 1.2<=1.3 farm c3a057	2017-12-06 11:00
[감색서버풀(간체) SAS] 가용성 경계상황 해제	2017-12-06 11:00

필요해서 도입한 지표인데,
너무 많은 경보로 우리를 괴롭힘



“경보 피로”를 줄일 방법을 찾아보자!

거짓 경보

	긴급 대응 필요 상황	대응 불필요 상황
경보 발생	True Positive (실제 장애 상황)	False Positive (색인 업데이트, 캐시 갱신 등 시간이 흐르면 정상화 되는 상황)
경보 미발생	False Negative (장애가 발생했으나 경보가 울리지 않는 경우)	True Negative (정상 상황)

거짓 경보

		거짓 경보
		긴급 대응 필요 상황
경보 발생	대응 불필요 상황	<p>False Positive (색인 업데이트, 캐시 갱신 등 시간이 흐르면 정상화 되는 상황)</p>
경보 미발생	정상 상황	<p>False Negative (장애가 발생했으나 경보가 울리지 않는 경우)</p>

자동 경보 분석

거짓 경보를 줄이기 위해 많이 사용하는 방법 → 경험칙, 휴리스틱

장점 : 그래프만 보고도 금방 장애 유무 판단 가능

단점 : 개인차 존재

더 이상적인 방법 → 경보 분석 자동화

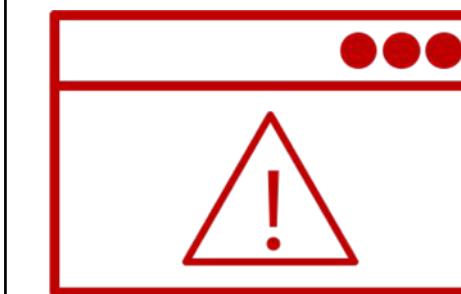
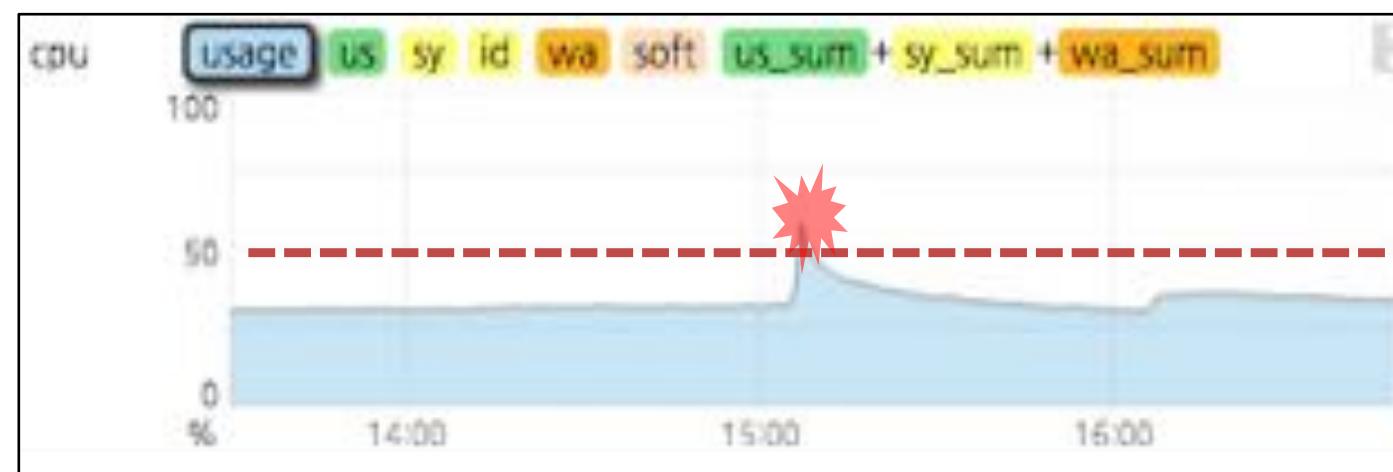
다양한 장애 케이스에 대해서 모든 경우에 대한 모든 대응 자동화 (X)

빠른 대응을 위해 필요한 데이터들을 미리 모아주기 (O)

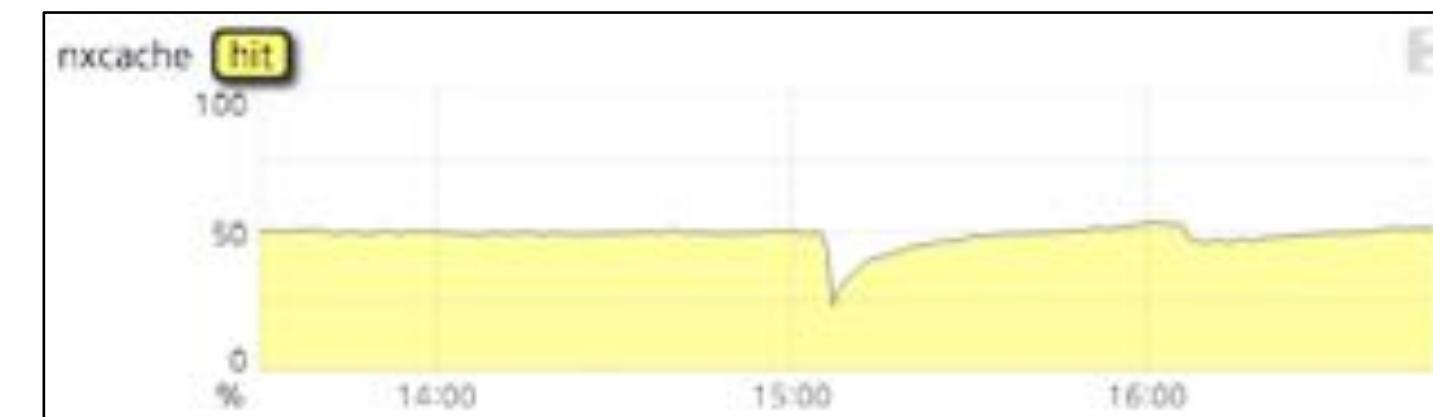
빠른 의사 결정을 위해 필요한 기본적인 상황 판단 자동화 (O)

자동 경보 분석 예제 (1)

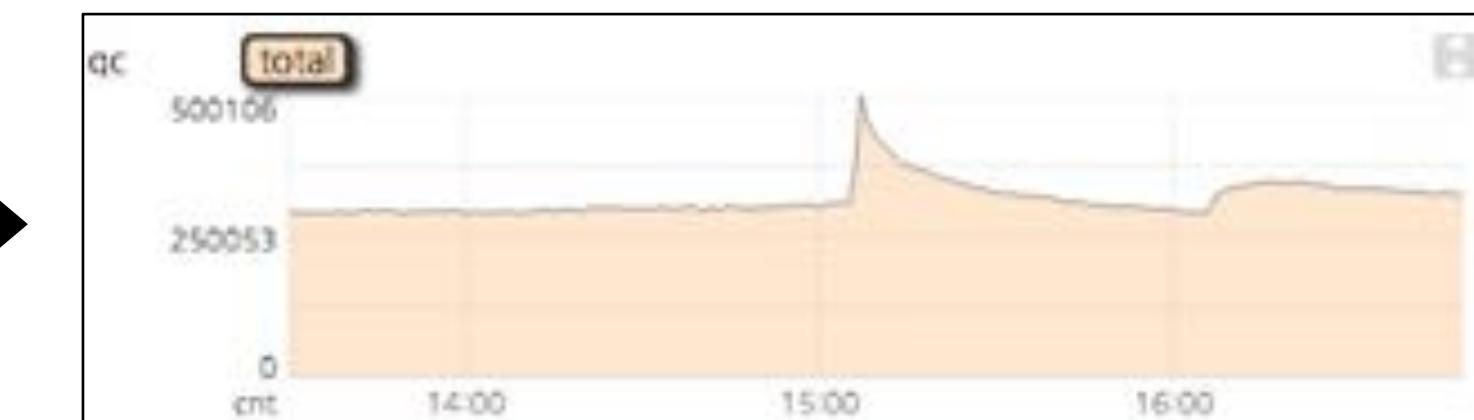
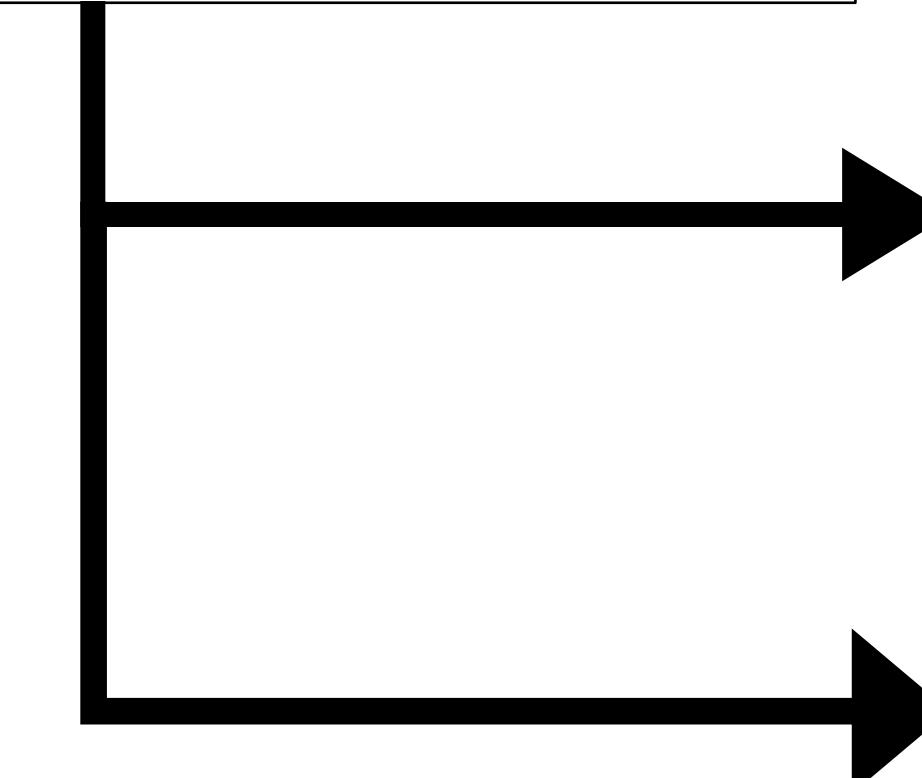
그래프 모양을 보고 판단할 수 있는 경우



CPU 경보 발생



캐시 히트율 감소 확인



트래픽 증가 확인

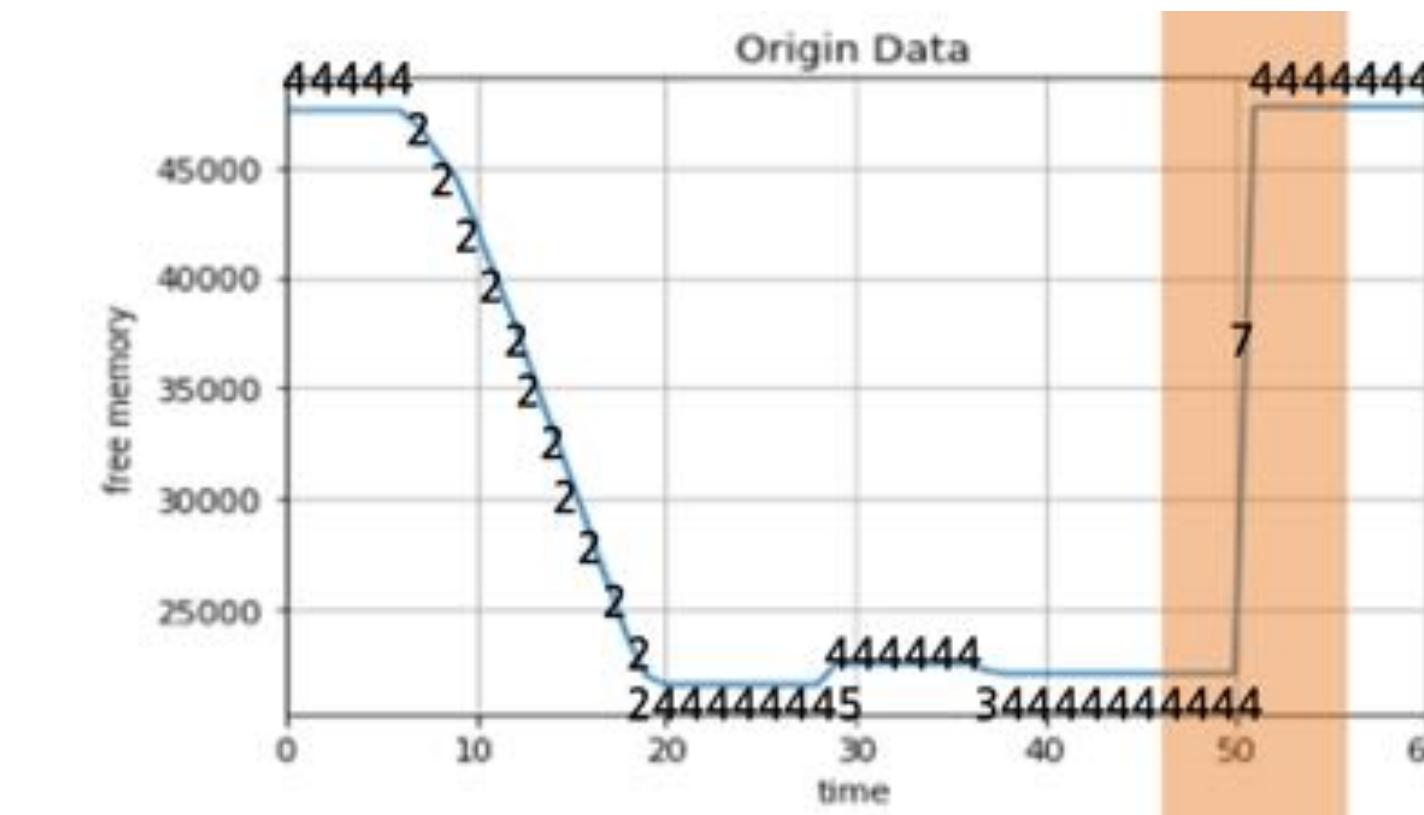
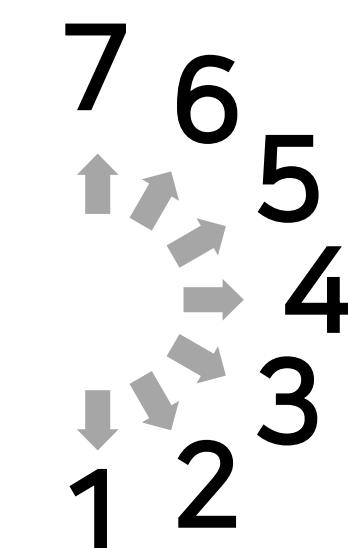
장애 아님! 캐시 리프레시 상황!

자동 경보 분석 예제 (1)

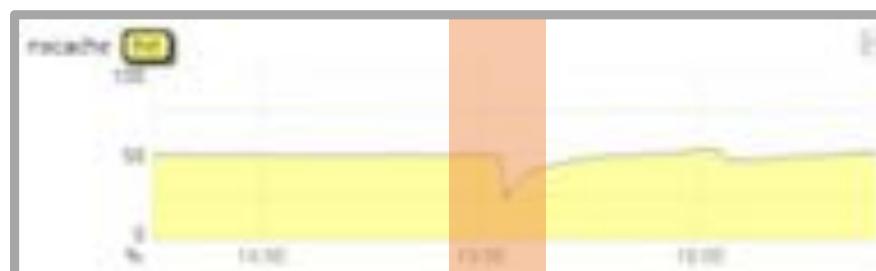
그래프 모양을 보고 판단할 수 있는 경우

지표의 방향을 부호로 인코딩 후

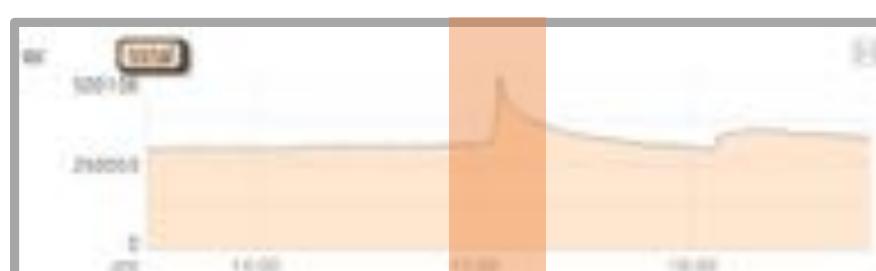
패턴 매칭으로 자동 상황 판단



~~44444222222222222444444454444443444444~~**444474444444**



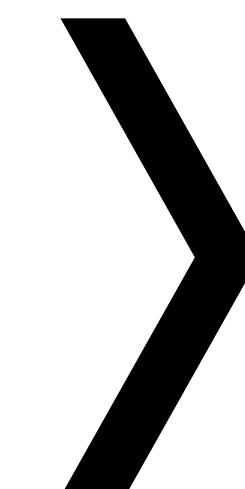
335625355117664533



353265553771223344



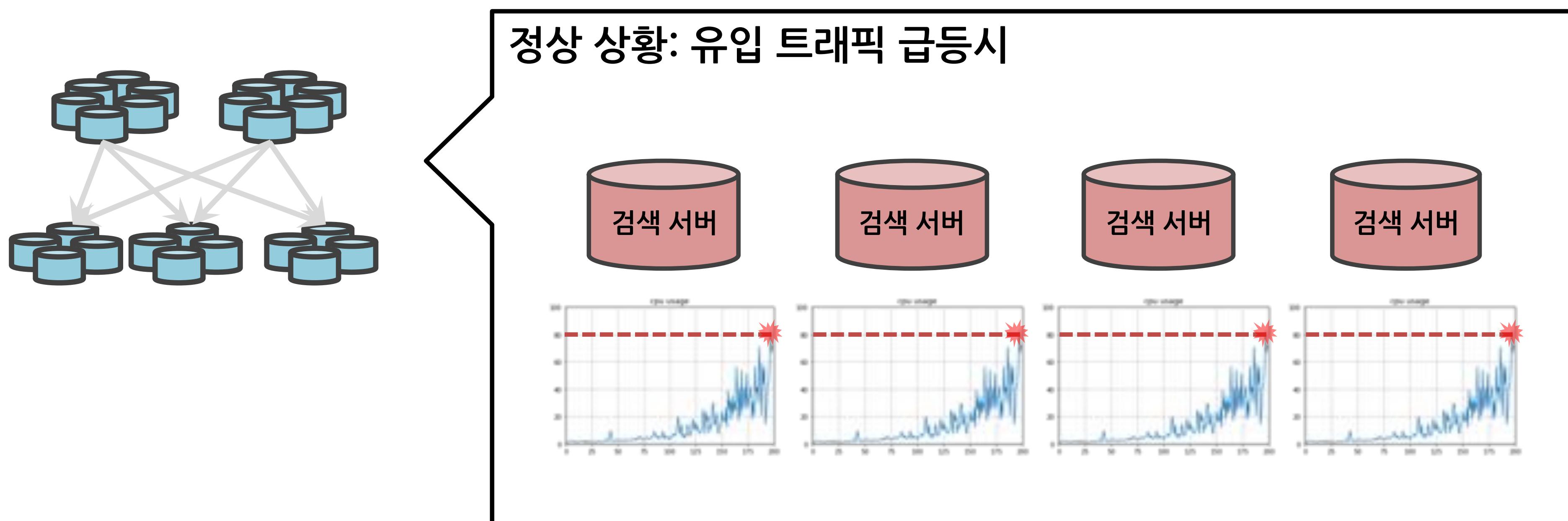
353365344771223344



장애 아님! 캐시 리프레시 상황!

자동 경보 분석 예제 (2)

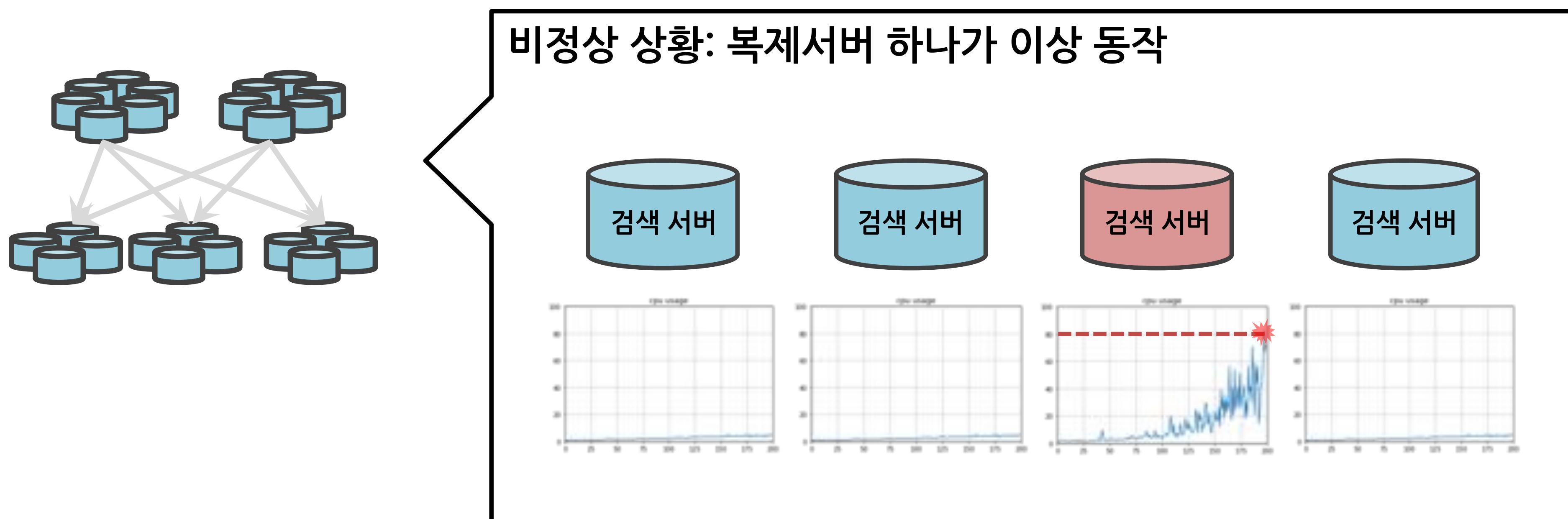
지표 데이터의 유사도를 보고 판단할 수 있는 경우



복제 서버 그룹의 트래픽 지표 → 유사한 형태로 나타나는 것이 정상

자동 경보 분석 예제 (2)

지표 데이터의 유사도를 보고 판단할 수 있는 경우



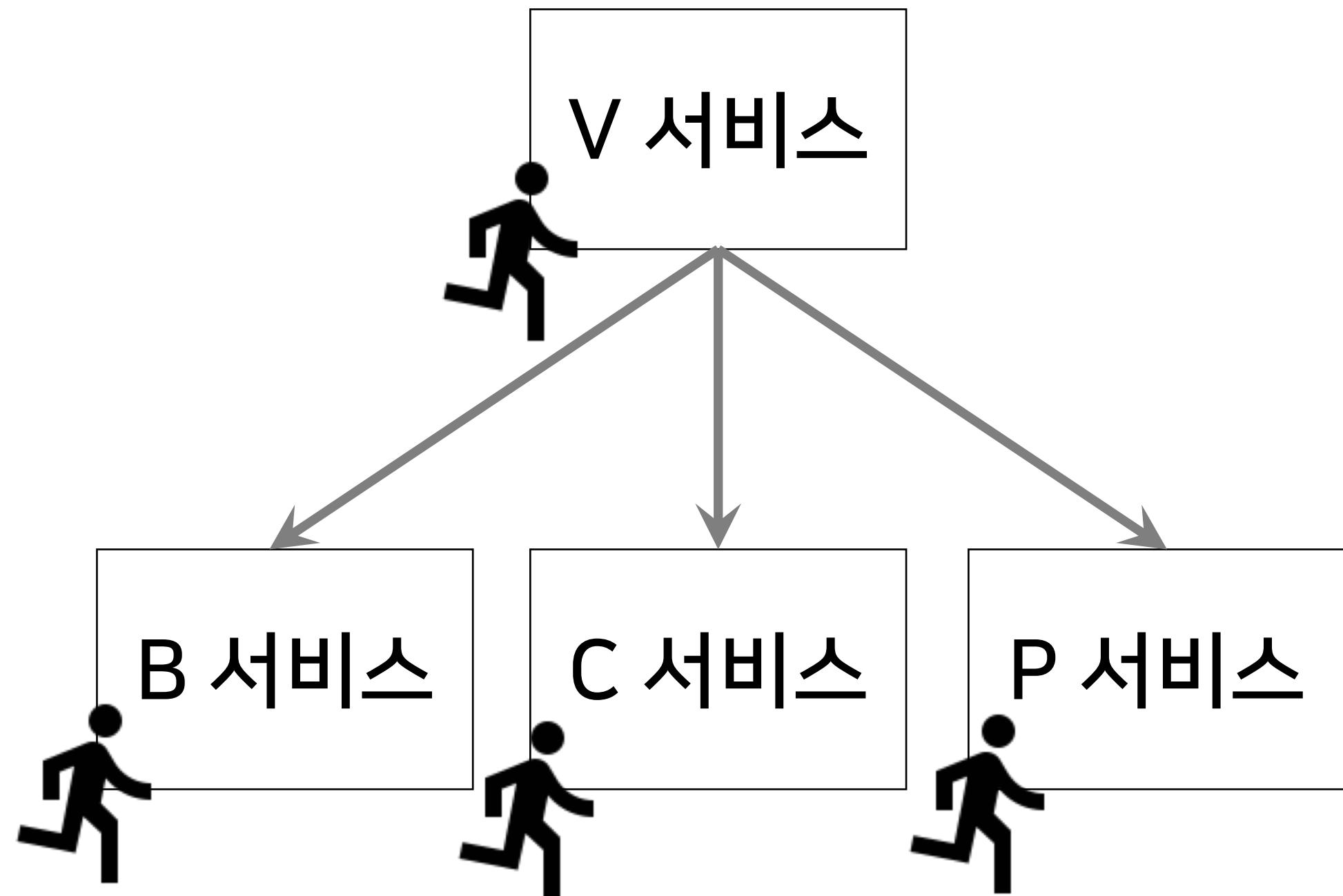
코사인 유사도 $< 0.975 \rightarrow$ 가용량 문제 발생 전에 조치!!

“경험을 압축할 수 있는 알고리즘은 없다”

앤디 제시, AWS CEO

그리고,
저희만의 SRE 핵심 철학 2가지가 있습니다.

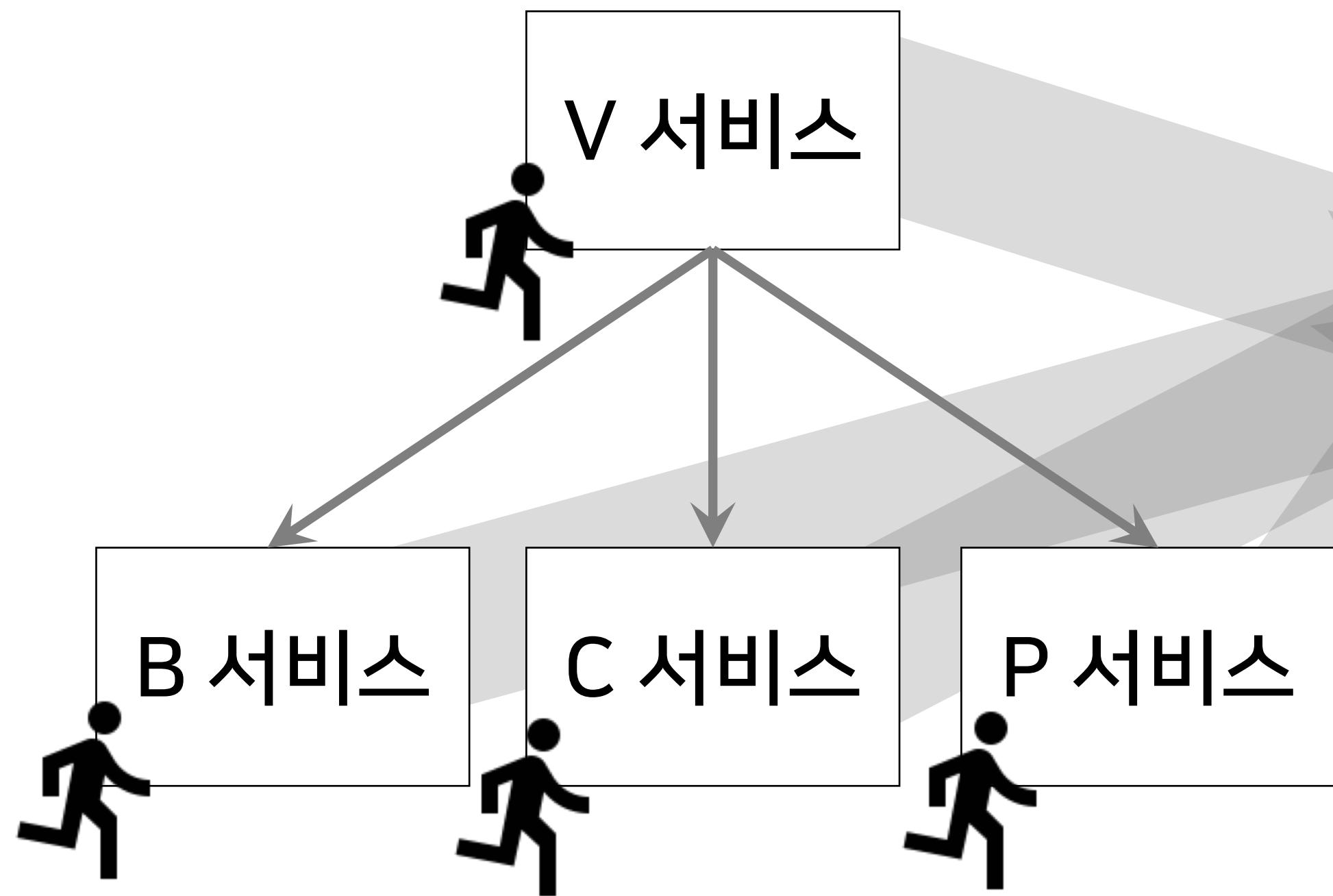
문제 분석 관점 : Macro Analysis



각 서비스 담당자의 분석 영역

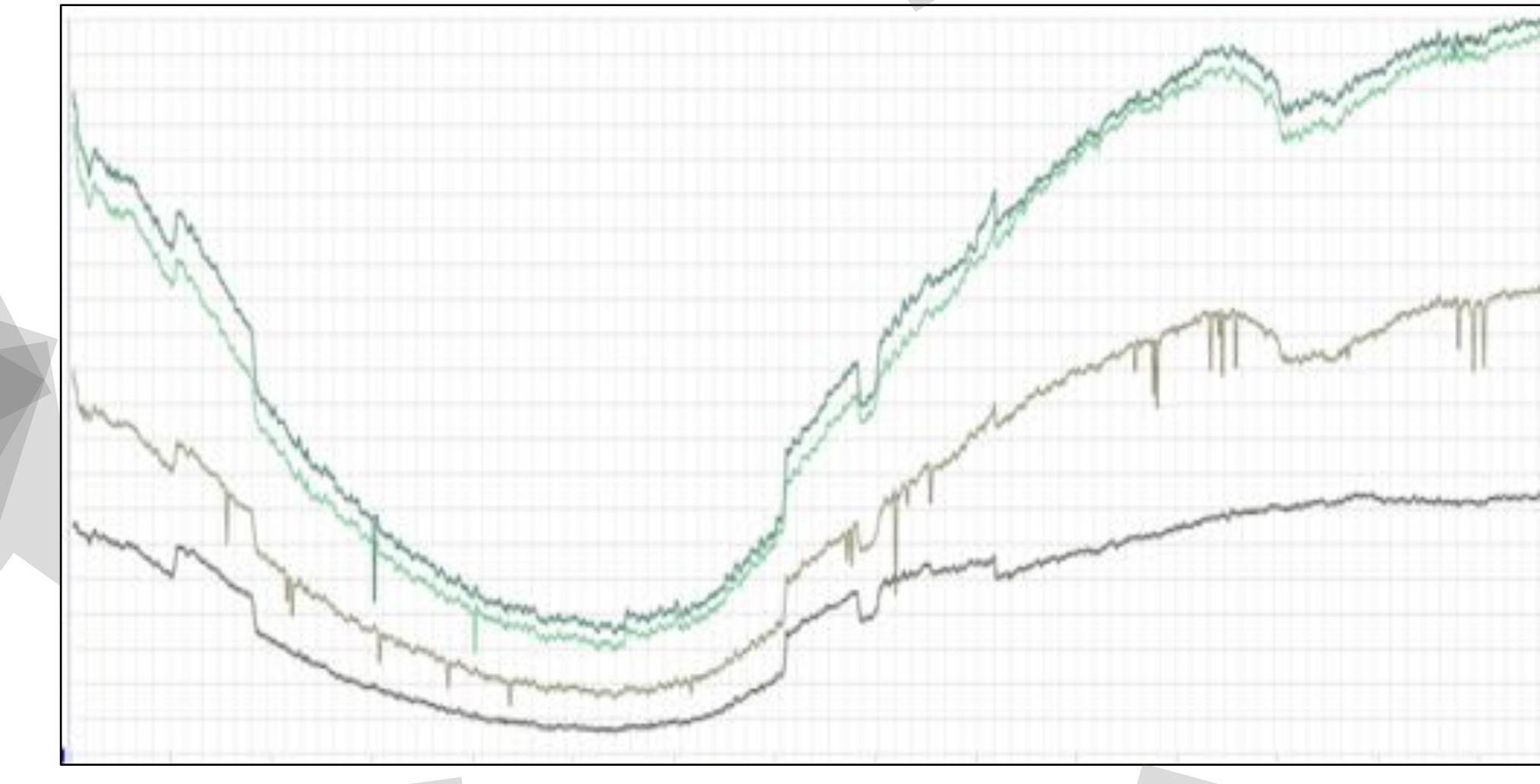
각 서비스 내부에서 일어나는 변화 집중 파악

문제 분석 관점 : Macro Analysis



각 서비스 담당자의 분석 영역

각 서비스 내부에서 일어나는 변화 집중 파악



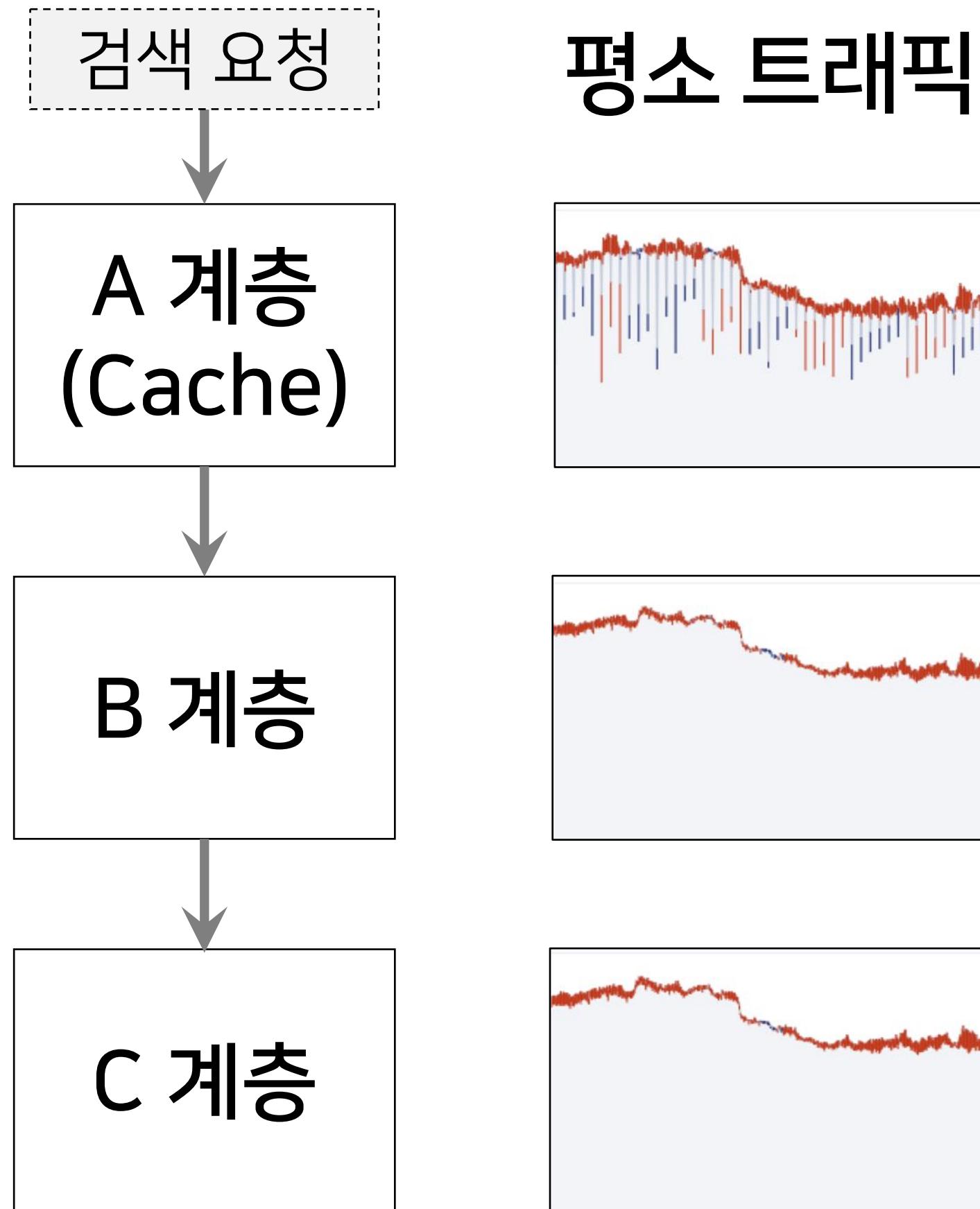
외부 요인

SRE 분석 영역

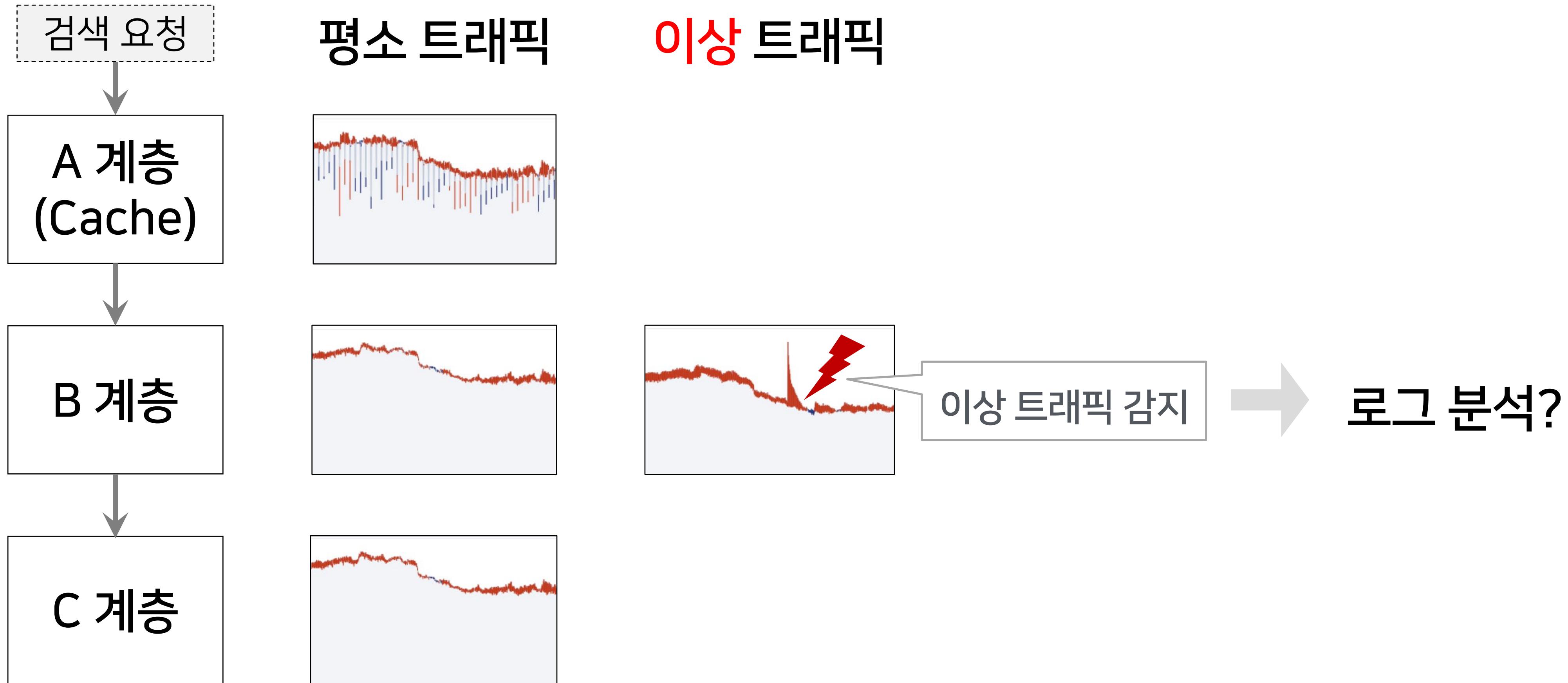
외부 요인들까지 고려하여
종합적인 관점에서 분석

외부 요인

장애관제 방식 : Blackbox Monitoring



장애관제 방식 : Blackbox Monitoring



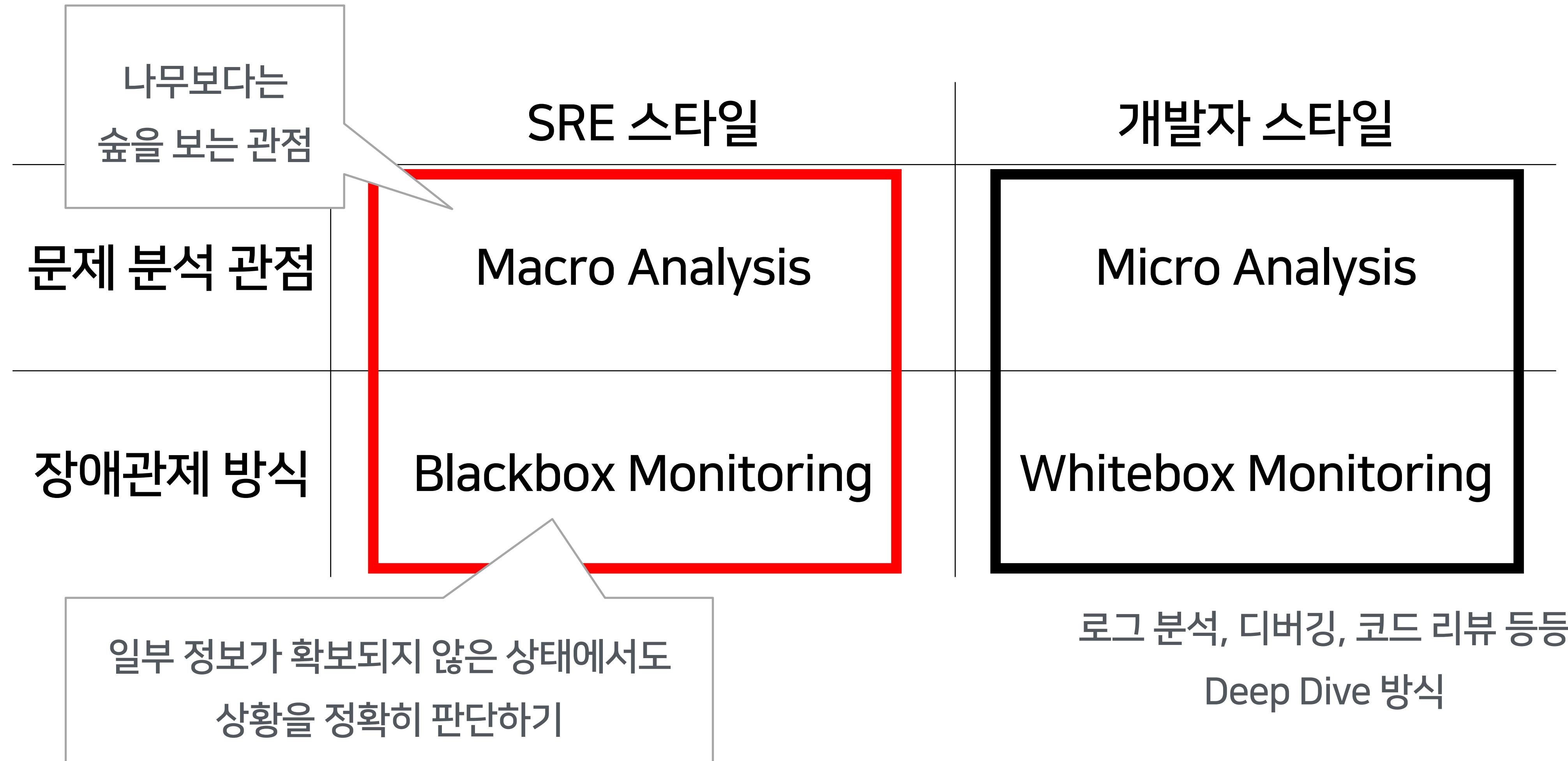
장애관제 방식 : Blackbox Monitoring



SRE의 문제 해결법

	SRE 스타일	개발자 스타일
문제 분석 관점	Macro Analysis	Micro Analysis
장애관제 방식	Blackbox Monitoring	Whitebox Monitoring

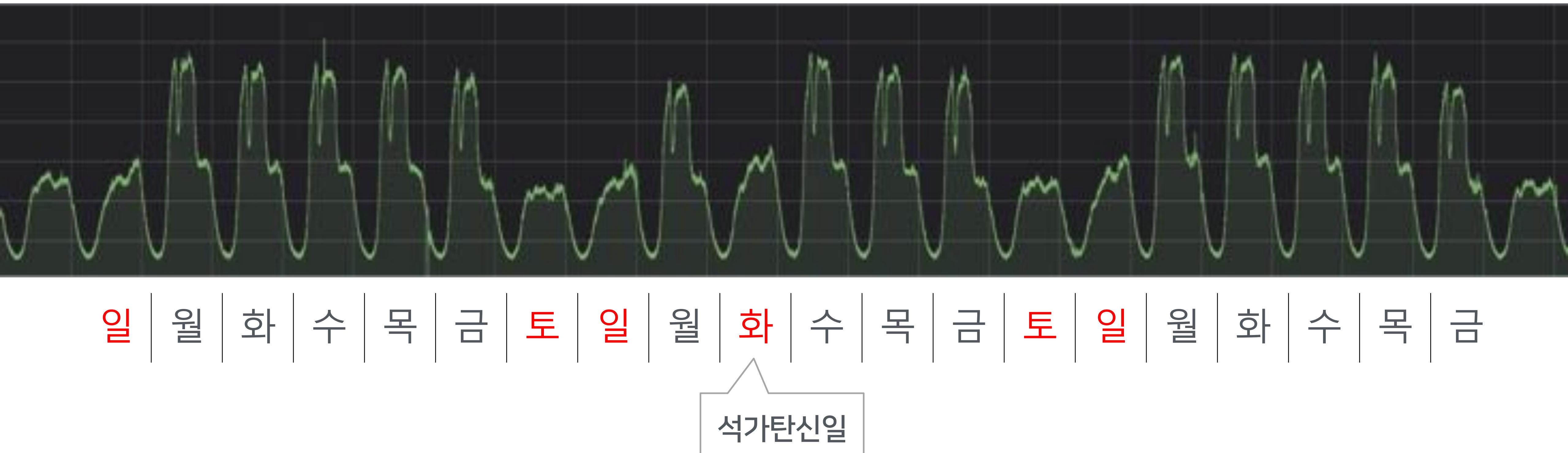
SRE의 문제 해결법



3. 실제 사례 소개

우리 일상과 밀접한 네이버 검색시스템

사람들의 생체 주기를 따라가는 트래픽 (PC 통합검색)



요일 별 트래픽 패턴

평일 PC 통합검색

오전, 오후에 증가

점심시간, 저녁시간에 감소



주말 / 공휴일 PC 통합검색

새벽 외 비슷한 트래픽 유지

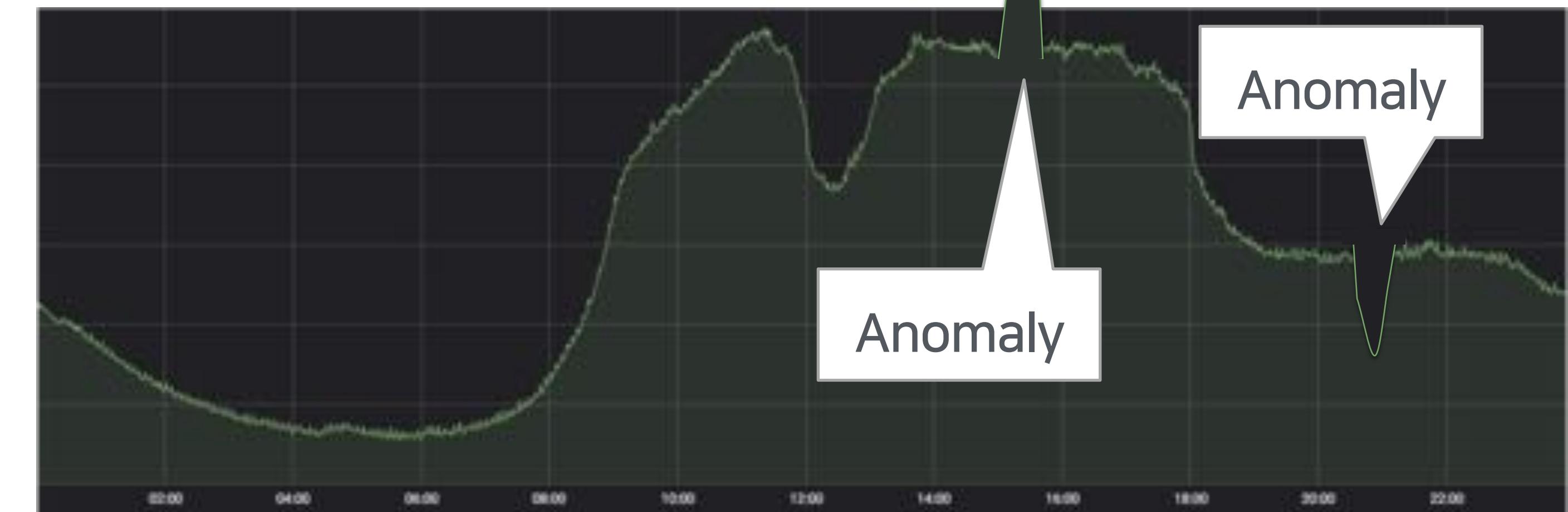


요일 별 트래픽 패턴

평일 PC 통합검색

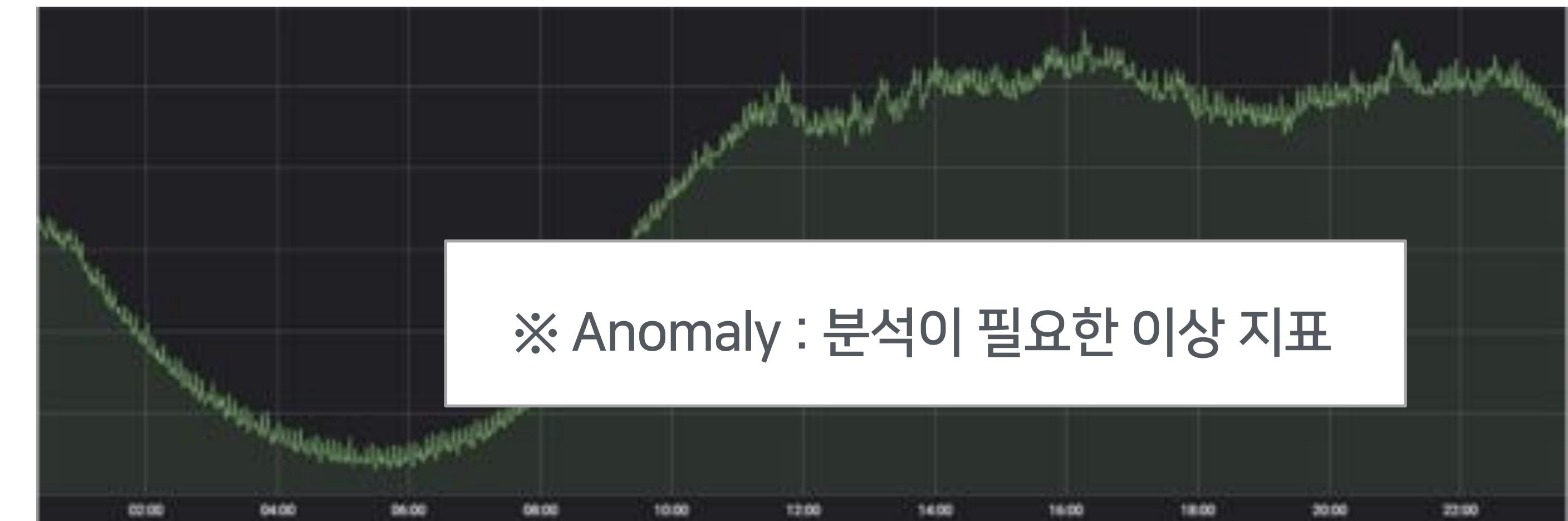
오전, 오후에 증가

점심시간, 저녁시간에 감소



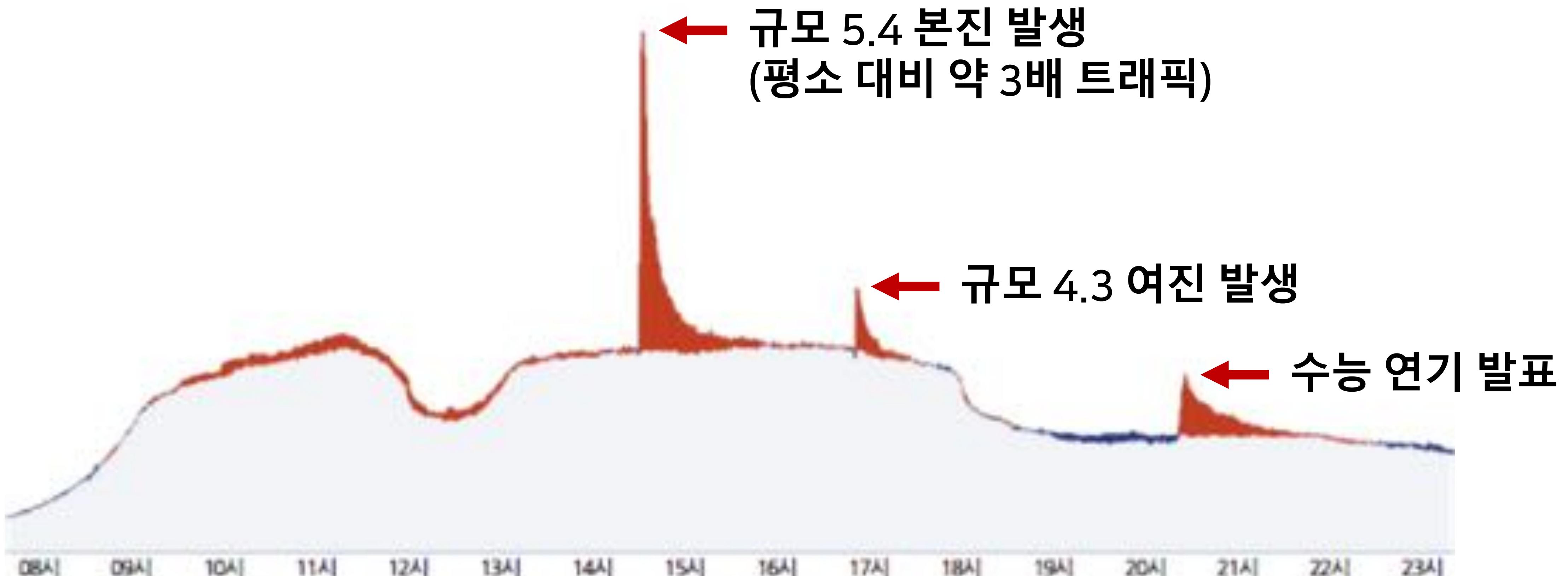
주말 / 공휴일 PC 통합검색

새벽 외 비슷한 트래픽 유지



※ Anomaly : 분석이 필요한 이상 지표

2017년 11월 15일 포항 지진



당시 PC 통합검색 트래픽 변화

스포츠 이벤트

평창 동계 올림픽

화제가 되거나 (컬링 등)
금메달을 획득한 종목
(스켈레톤 등)에 대한
검색 트래픽 대량 유입



2018년 2월

FIFA 월드컵

우리나라 대표팀 경기에
많은 관심 집중.
축구 경기 특유의
전반전, 후반전 패턴 발생



6월

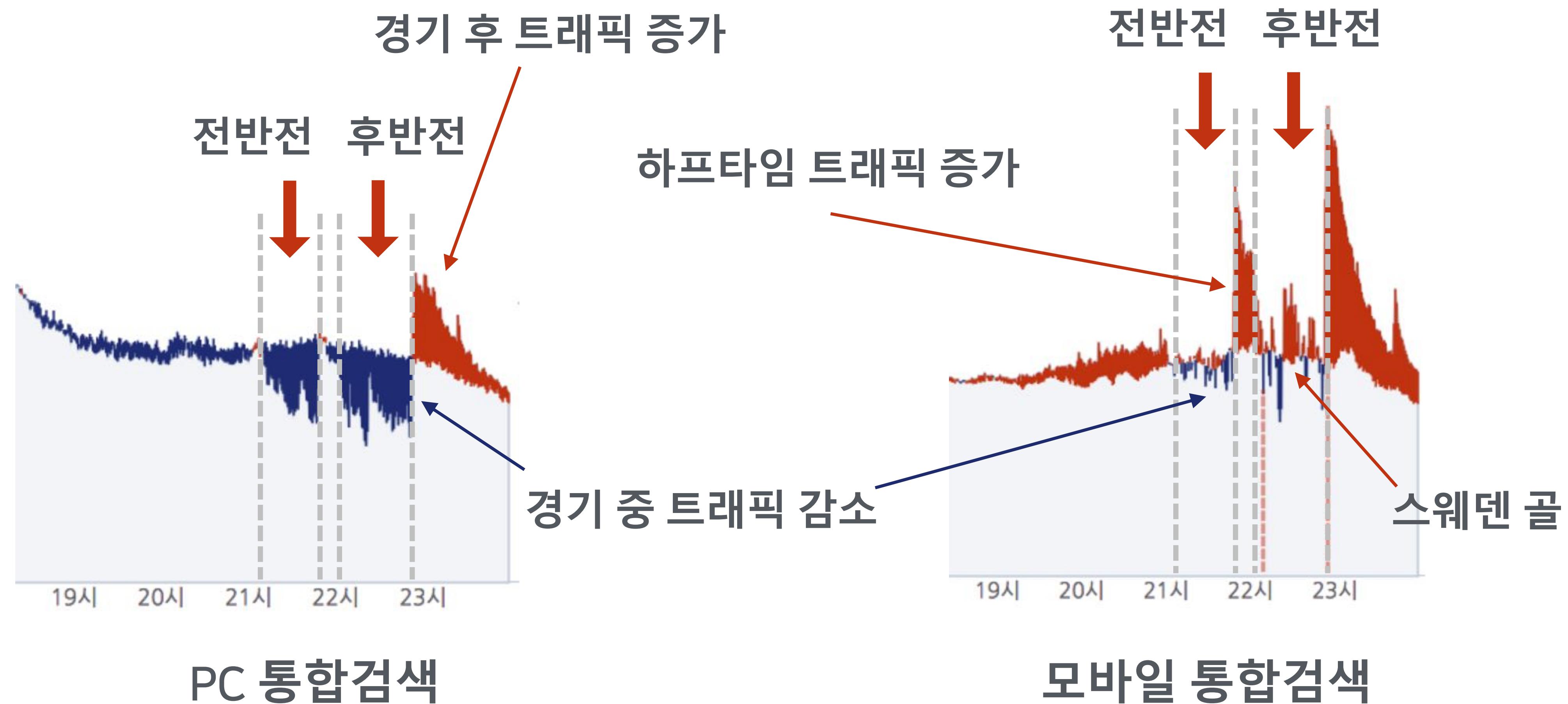
아시안 게임

병역 혜택이 걸린 종목은
경기 진행 상황에 따라 많은
검색 트래픽 유입

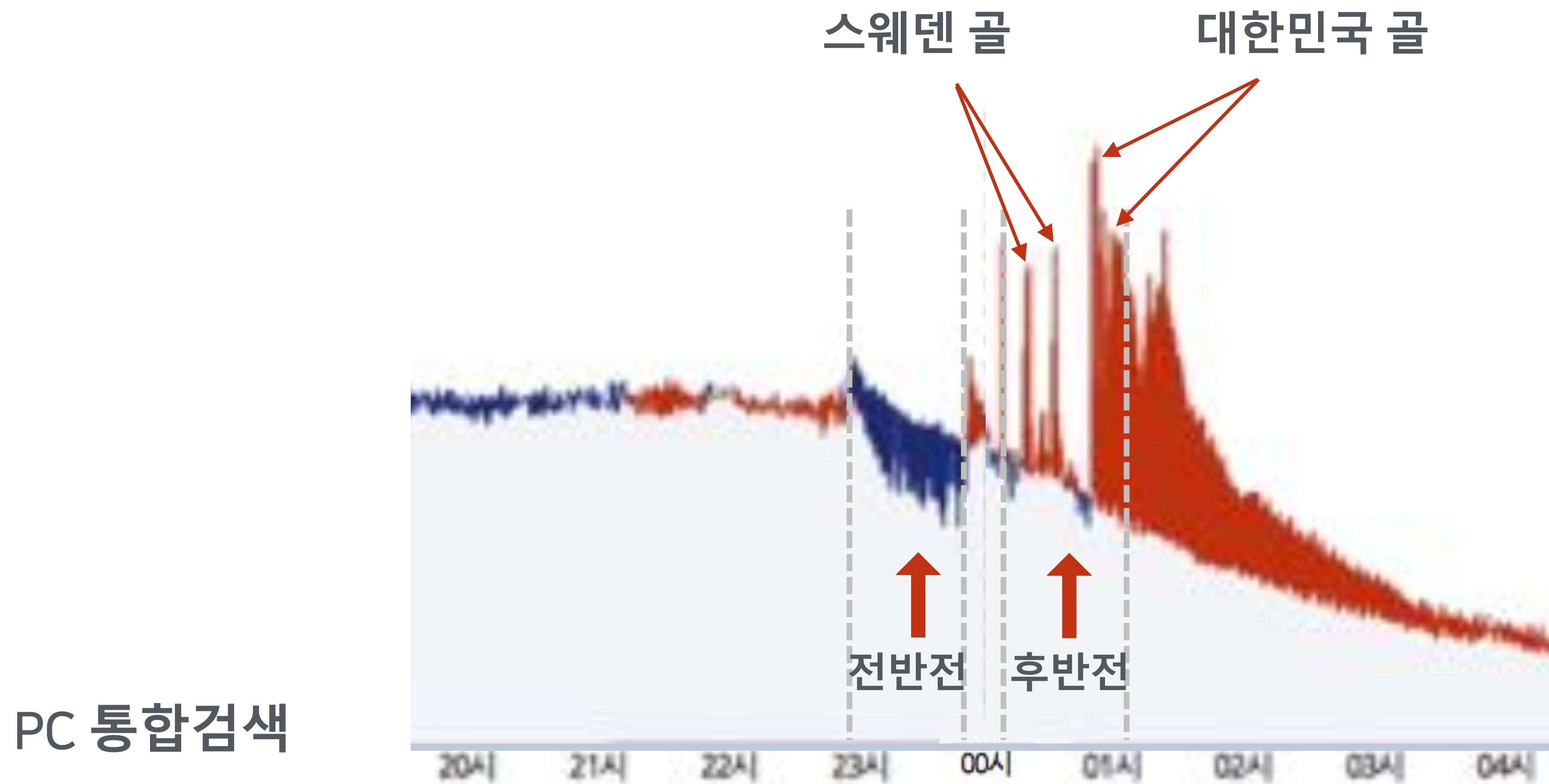


8월

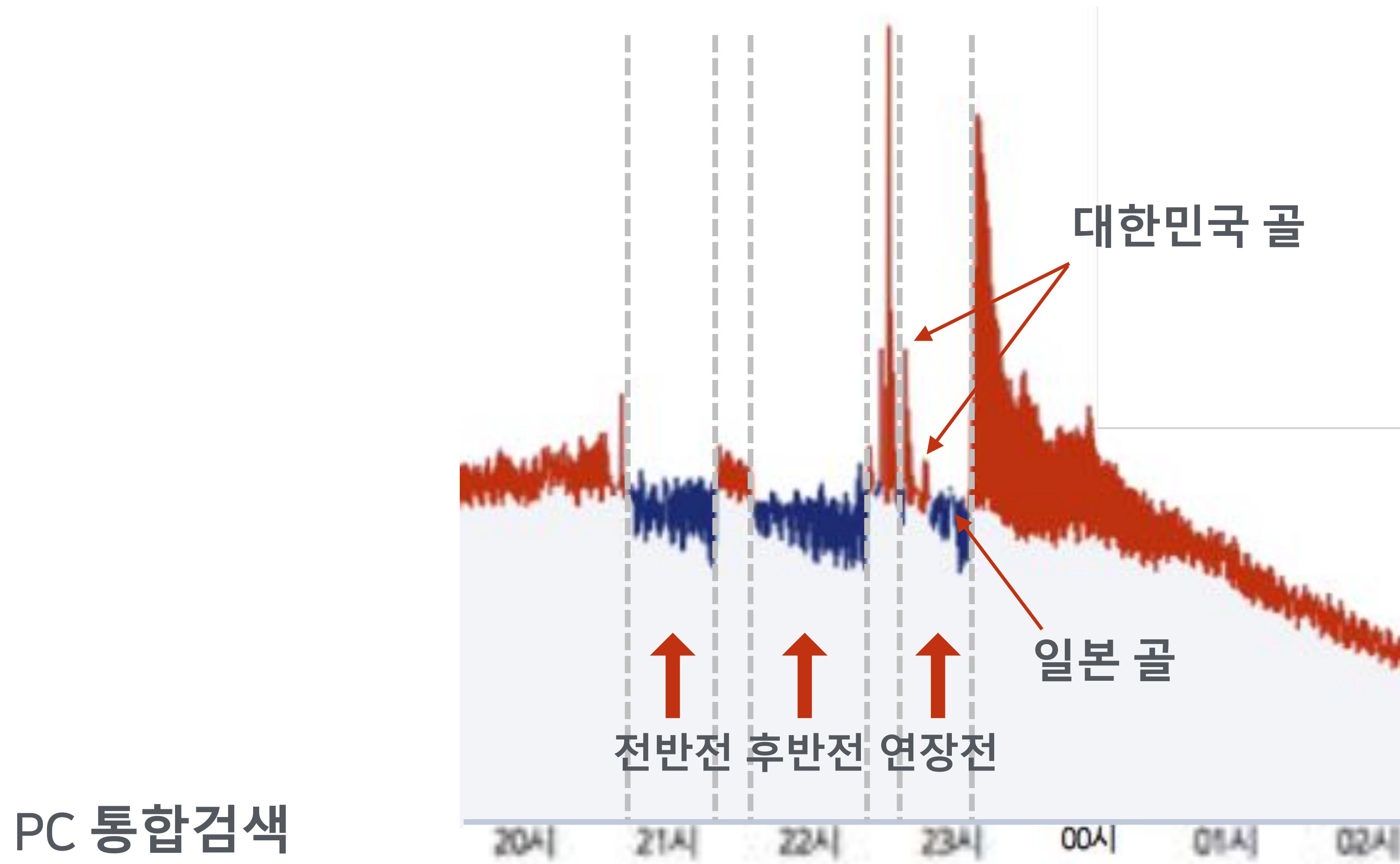
FIFA 월드컵 주요 경기 (6월 18일 대한민국 vs 스웨덴)



FIFA 월드컵 주요 경기 (6월 27일 대한민국 vs 독일)



아시안 게임 축구 결승 (9월 1일 대한민국 vs 일본)



방송 / 연예인 이슈 (6월 16일 기안84)



출처: MBC '나 혼자 산다' 방송화면 캡쳐

그 외 흔히 겪는 다양한 이슈들

개편과 버전 변경

버전 변경 시 트래픽 양상 변화, 통계 무효화

조작 실수나 버그 유입 가능성

분석용 데이터 수집 / 부하 테스트

실제 사용자 트래픽이 아니지만 서비스에 영향을 줄 수 있음

지표 수집 문제

지표 수집에 문제가 생겨서 거짓 경보가 발생하는 경우 많음

Mapping 정보에 문제가 생기거나 수집 자체가 잘 안되는 경우 등 원인은 매우 다양

DEVIEW
2018

4. Search Reliability Engineer

엔지니어와 시스템

SRE들이 주로 하는 고민
지금까지 겪어본 적 없는 문제들

언제 어디서 어떻게 발생할지 모르는 Incident
모든 경우의 수를 다 자동화하거나 시스템화 하는 것은 불가능

※ Incident : 예기치 않았던 이벤트나 사건

안정적인 사람이 만들고 운영하는 안정적인 시스템

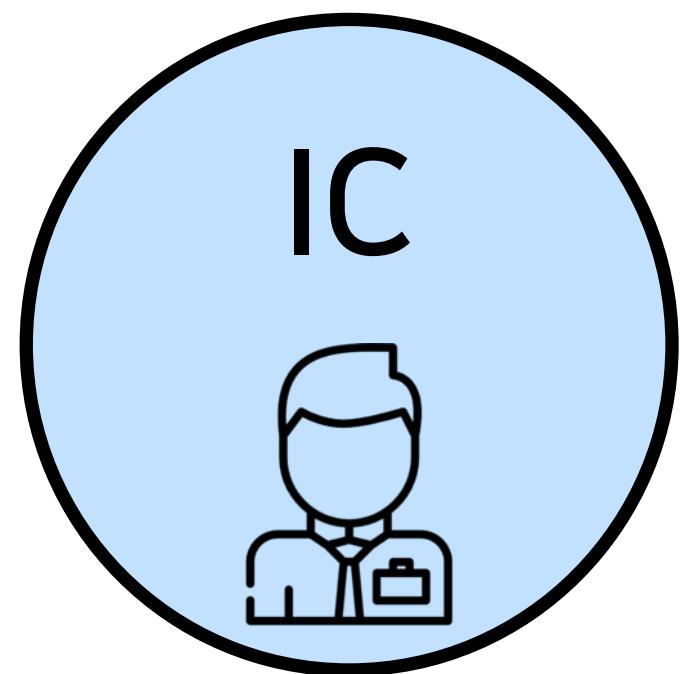
SRE는 시스템이 안정적으로 돌아가게 만들기 위한 '모든' 활동

개발자 / 엔지니어의 심리적, 정신적 안정감도 시스템의 Reliability에 큰 영향을 줌

새벽, 주말의 장애관제

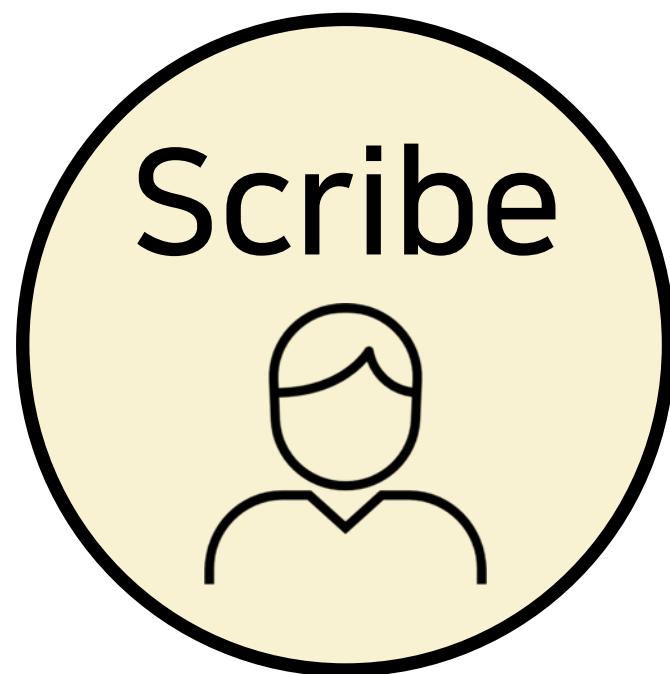
전체 구성원이 365일 24시간 대응할 수는 없음

일단, Incident 발생 시 필요한 두 가지 역할 정의



Incident Commander (IC)

- 전체 Incident 통제 역할
- 상황 판단과 의사 결정
- 리더 or 외부 조직과 대화

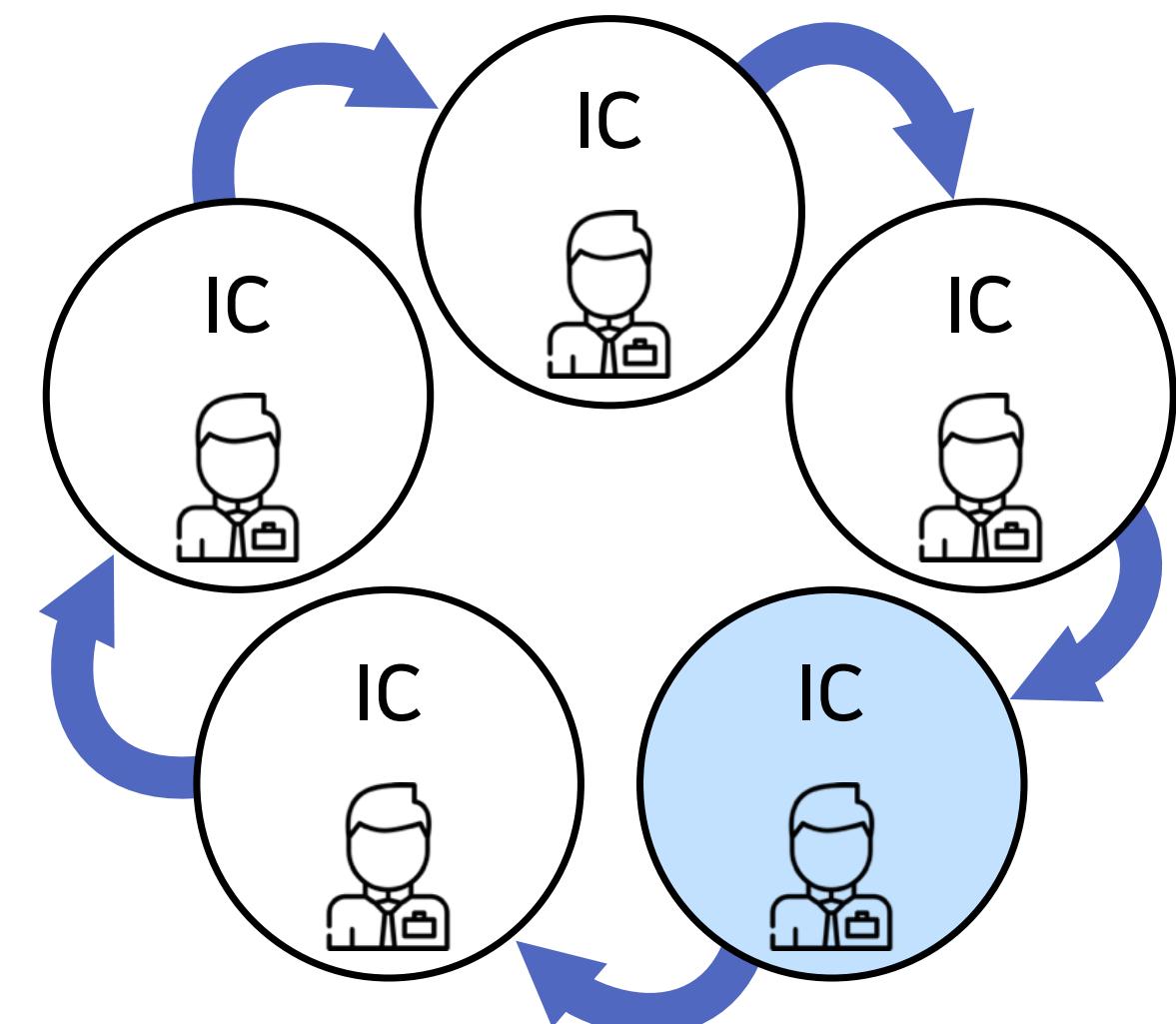


Scribe

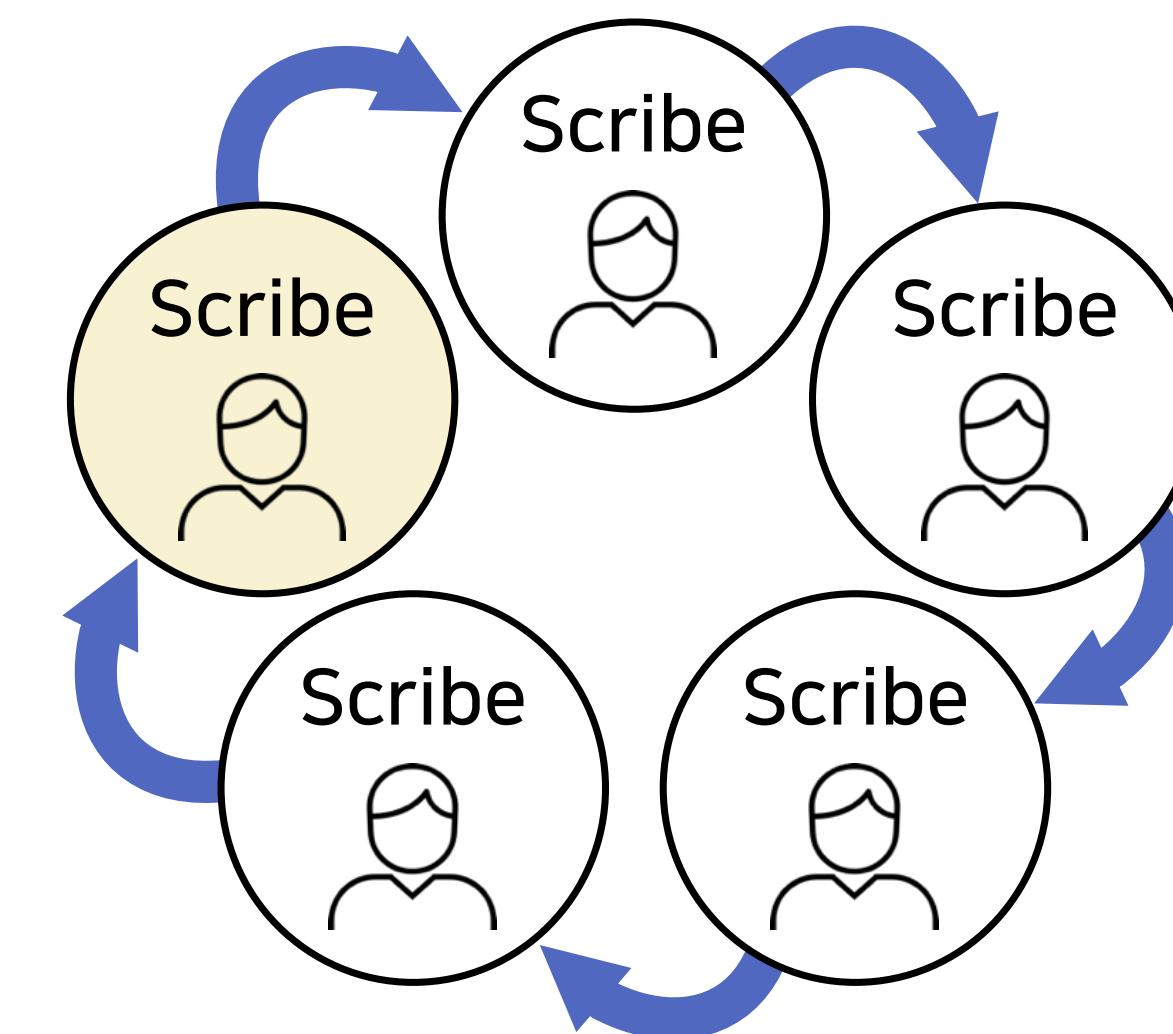
- 서기, 필경사 역할
- 판단에 필요한 지표 수집
- 상황 정리 및 기록, 전파

새벽, 주말의 장애관제

Incident Commander (IC)
Rotation Pool

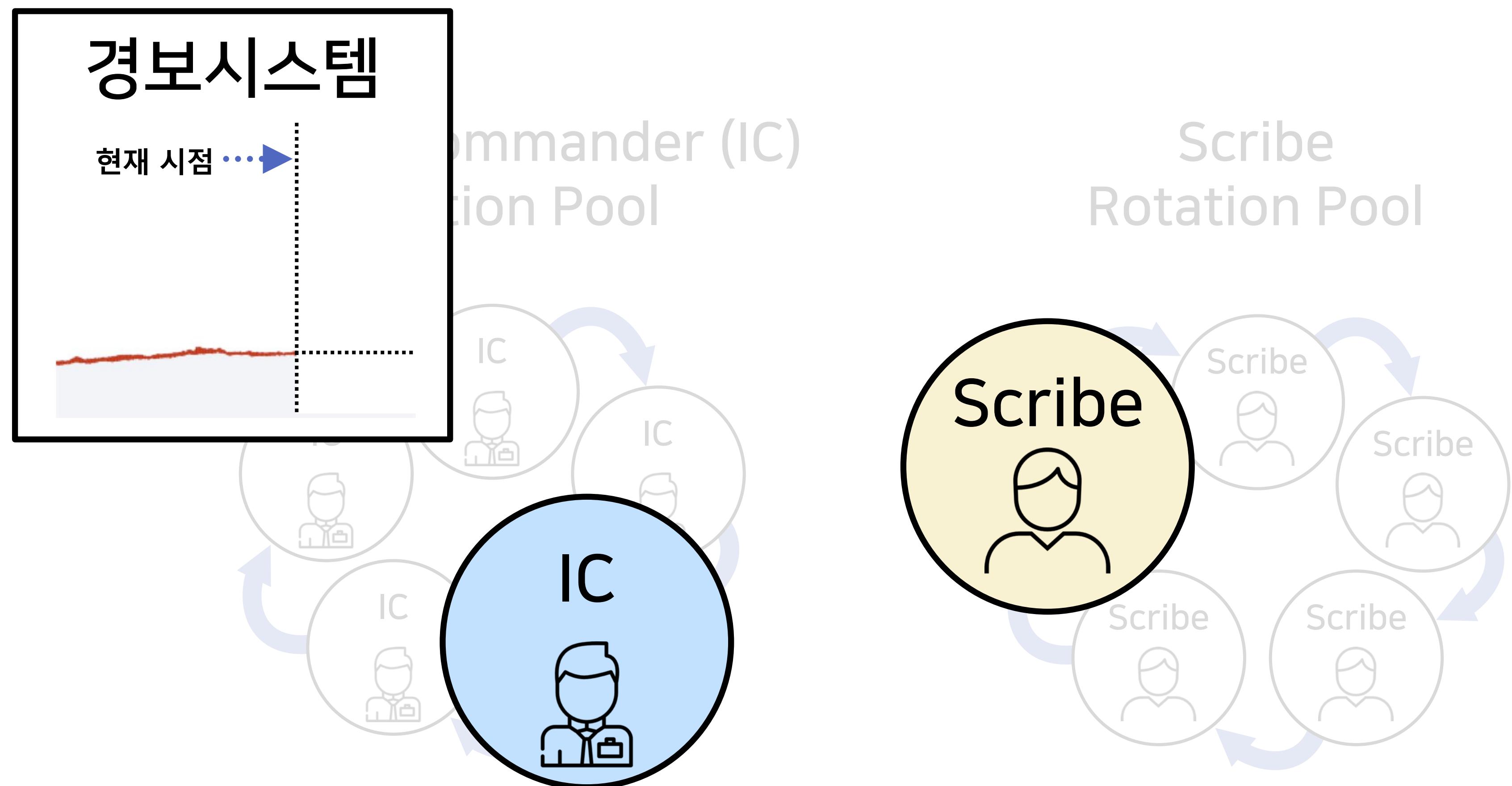


Scribe
Rotation Pool

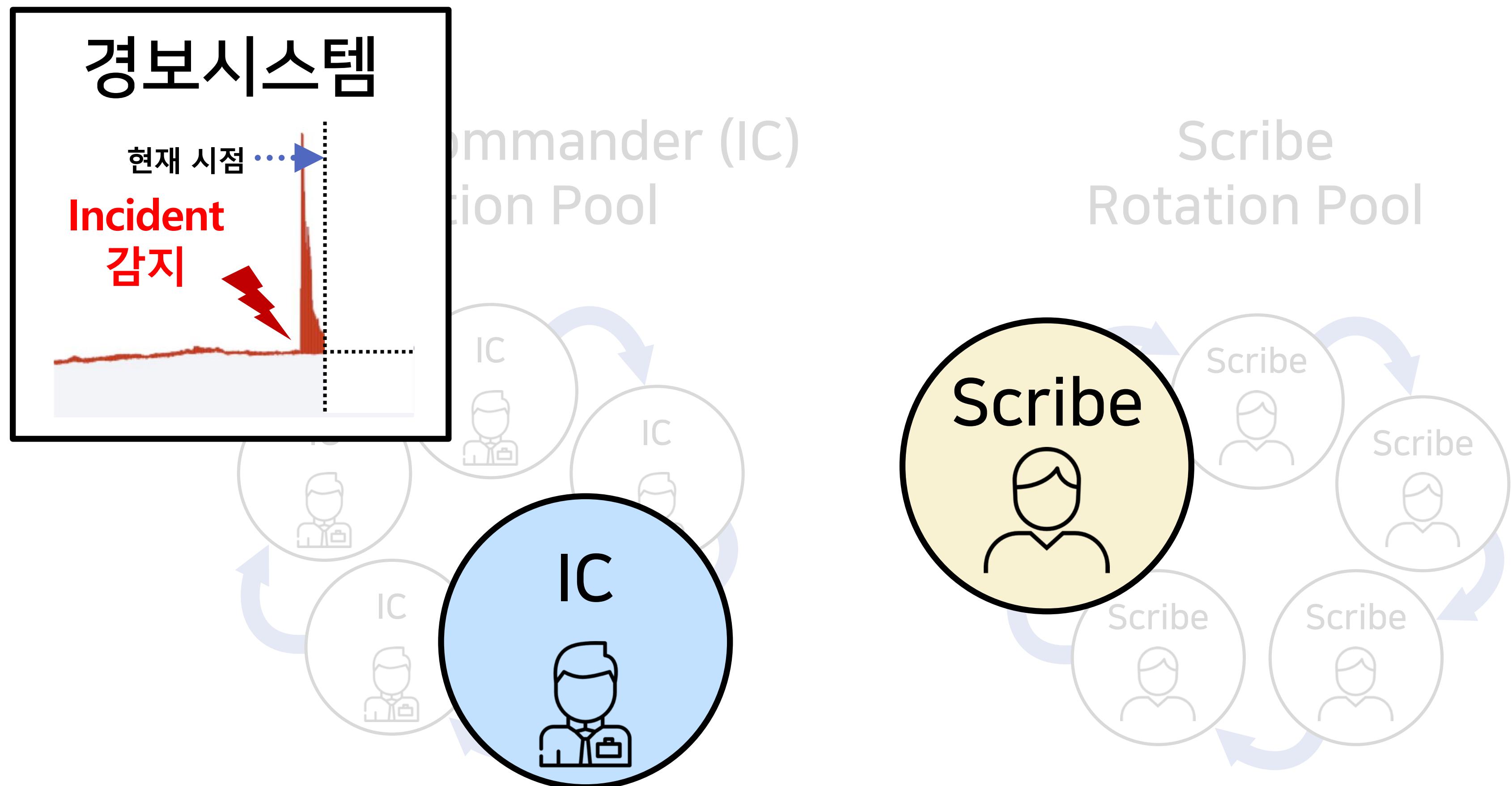


별도의 구성원으로 개별 Rotation

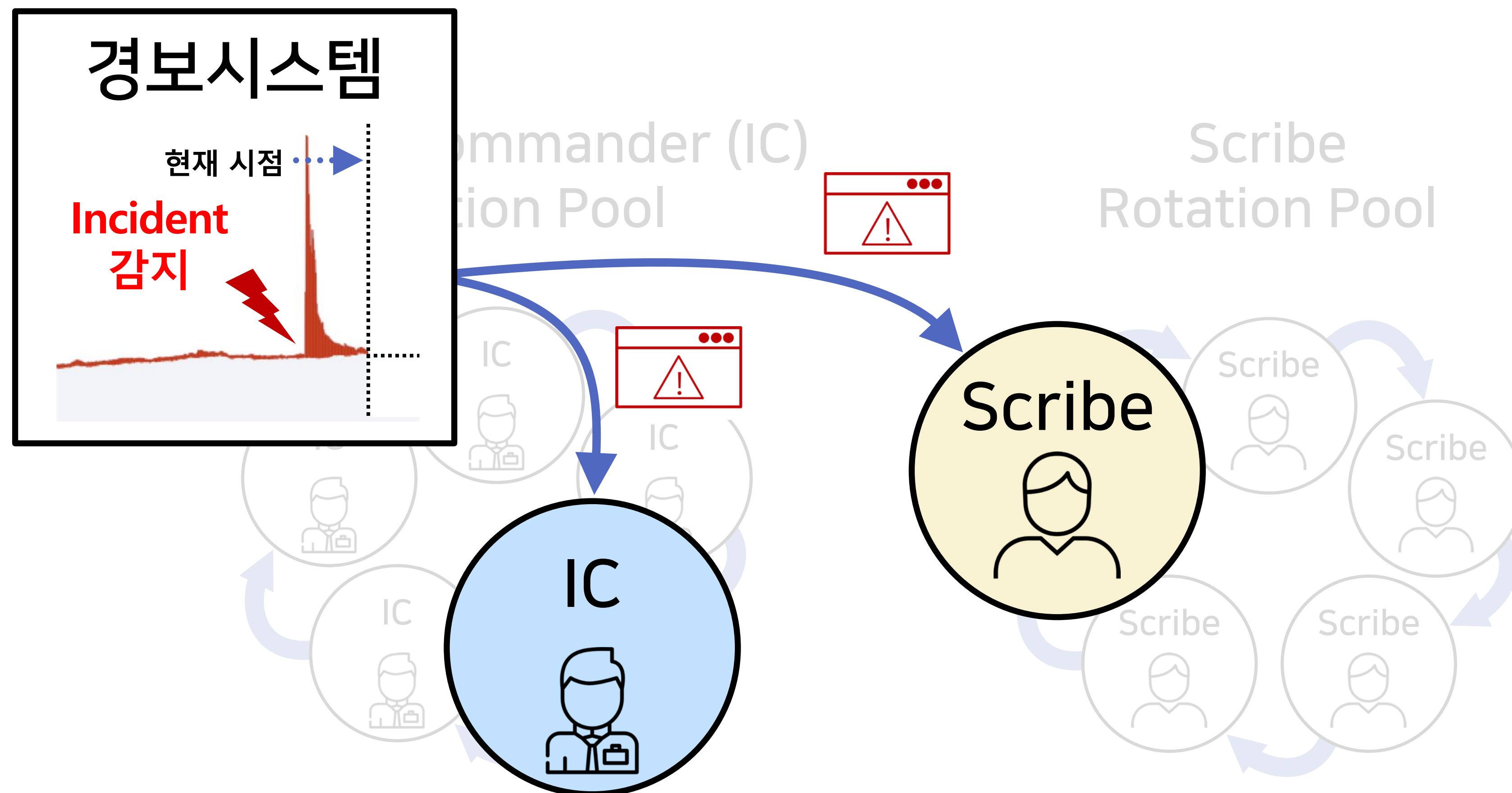
새벽, 주말의 장애관제



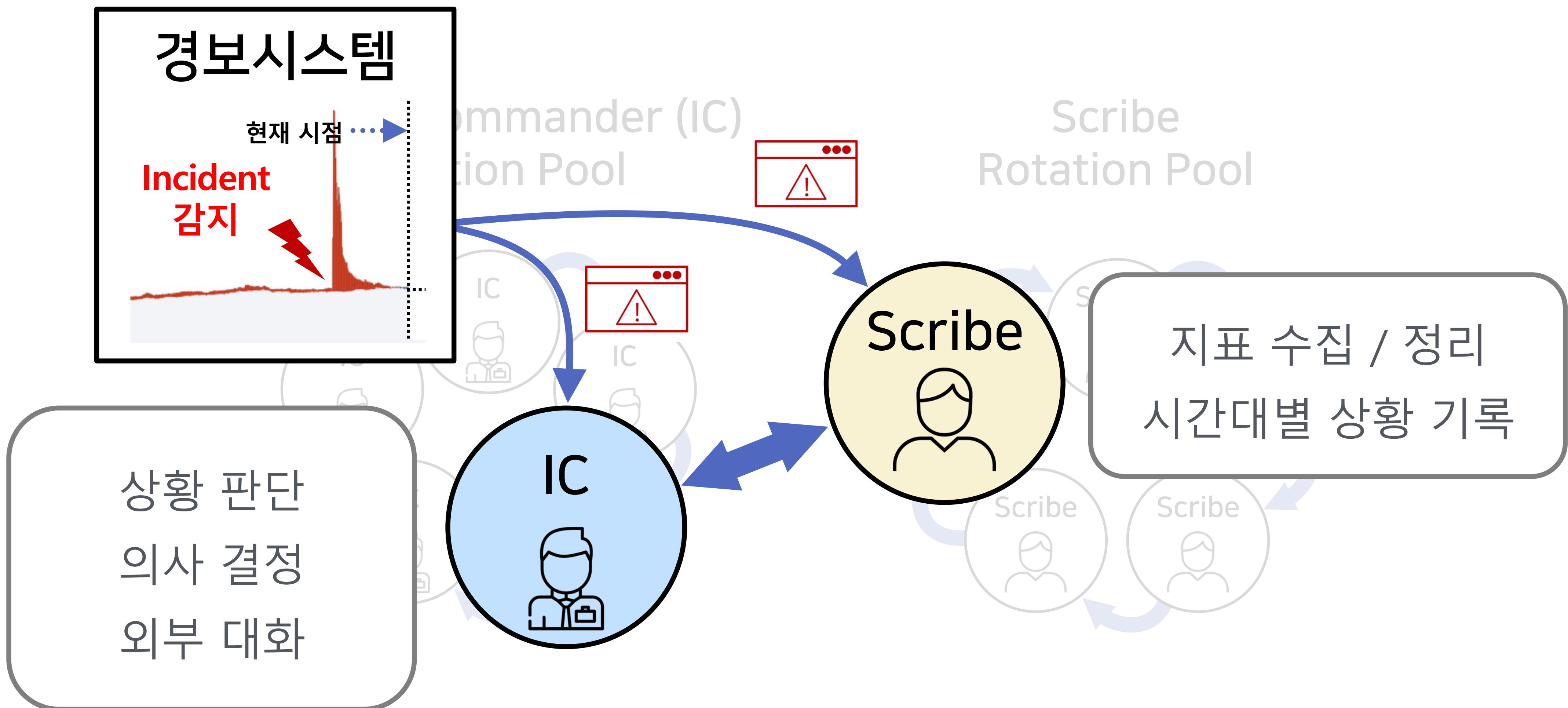
새벽, 주말의 장애관제



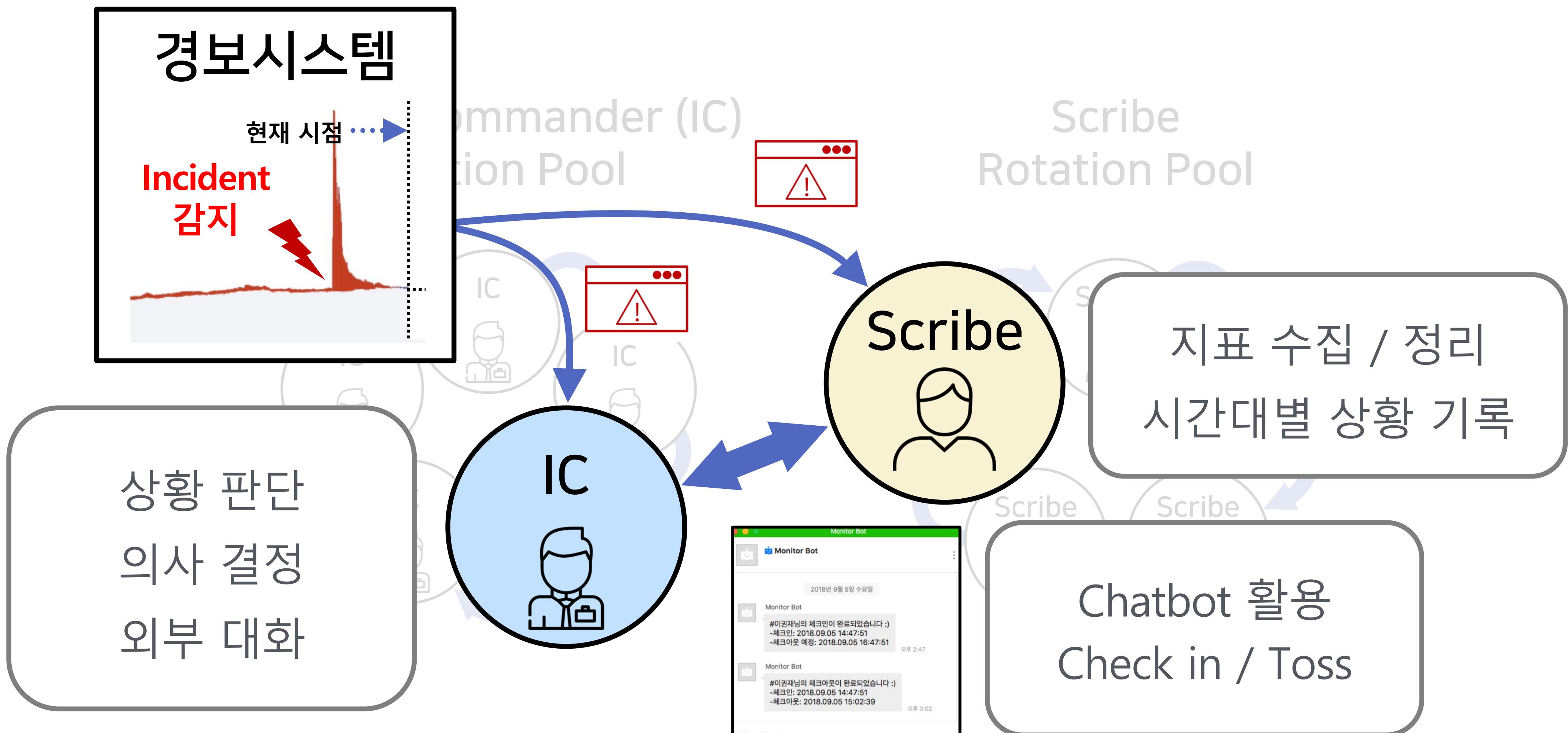
새벽, 주말의 장애관제



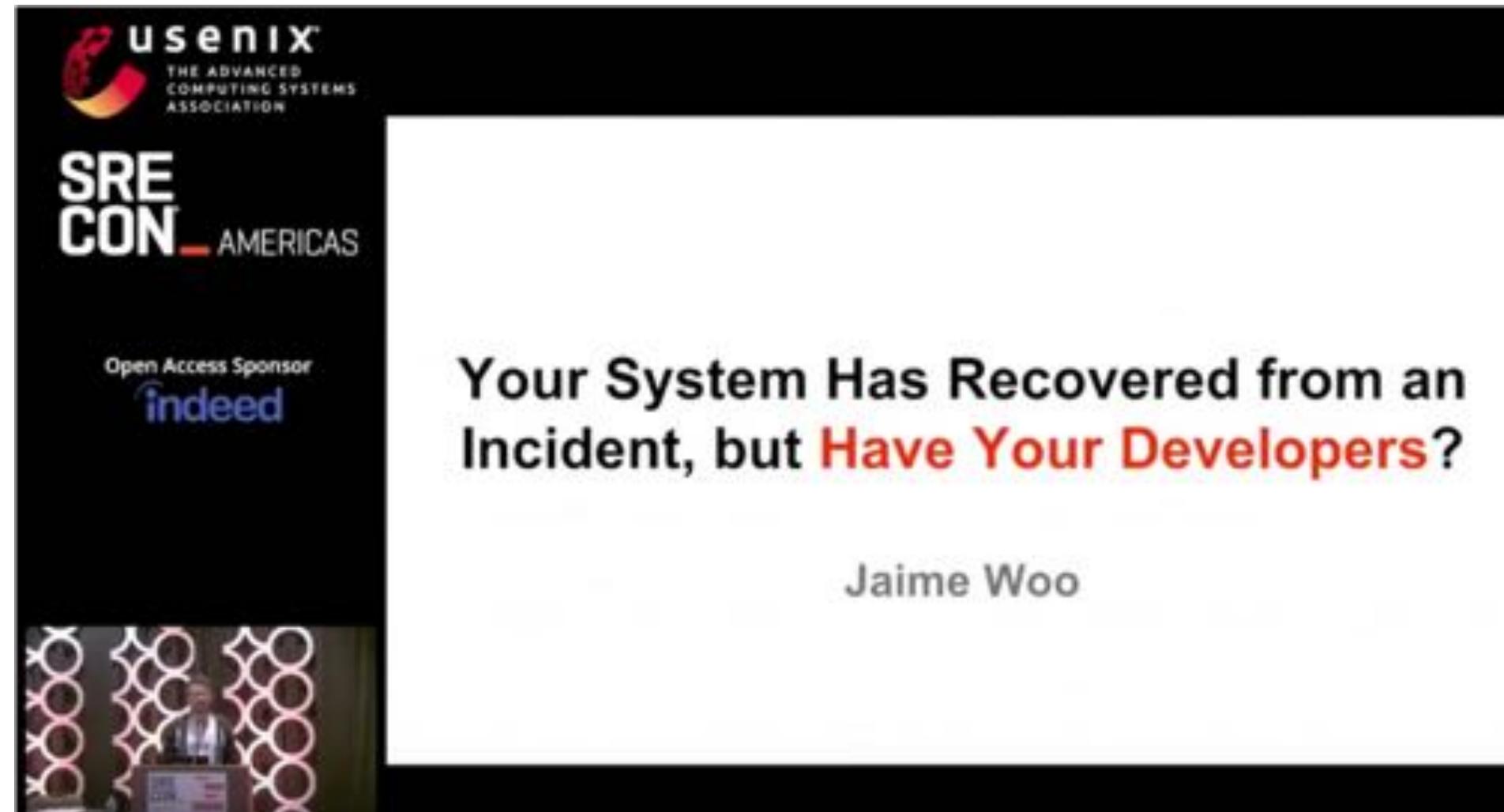
새벽, 주말의 장애관제



새벽, 주말의 장애관제



해외 사례



해외 개발자 / 엔지니어 설문조사

Incident 처리 후 많은 사람들이
분위기, 의욕, 집중력 등 각종 문제 경험

주요 내용

심리적 안전 (psychological safety)의 중요성
의료계, 코미디, 스포츠 등 다른 분야 참고

제안하는 해결책

각자 스트레스 처리하는 법 익히기
언제나 문제가 발생할 수 있다는 마음가짐
Incident에서 교훈 얻기
동료들의 도움의 중요성

SRE 문화

비난 없는 사후분석 (Blameless Postmortem)

현실에서는 단순히 비난이 없는 것 보다 사실에 기반하여 분석하는 것이 더 중요

SRE 홍보 활동

글쓰기, 정기 Report 발행, 교육 등 SRE 문화 전파

NAVER Search & Tech
(https://blog.naver.com/naver_search)

지진에도 흔들리지 않는 네이버 검색 시스템 - 1편

지진에도 흔들리지 않는 네이버 검색 시스템 - 2편

네이버 검색의 스마트한 경보 시스템

네이버 검색시스템의 노력

정책적, 제도적 해결책 고민

비상 상황 대응 방법 구체화

구성원 별 역할 체계화

SRE 조직 인원 보강

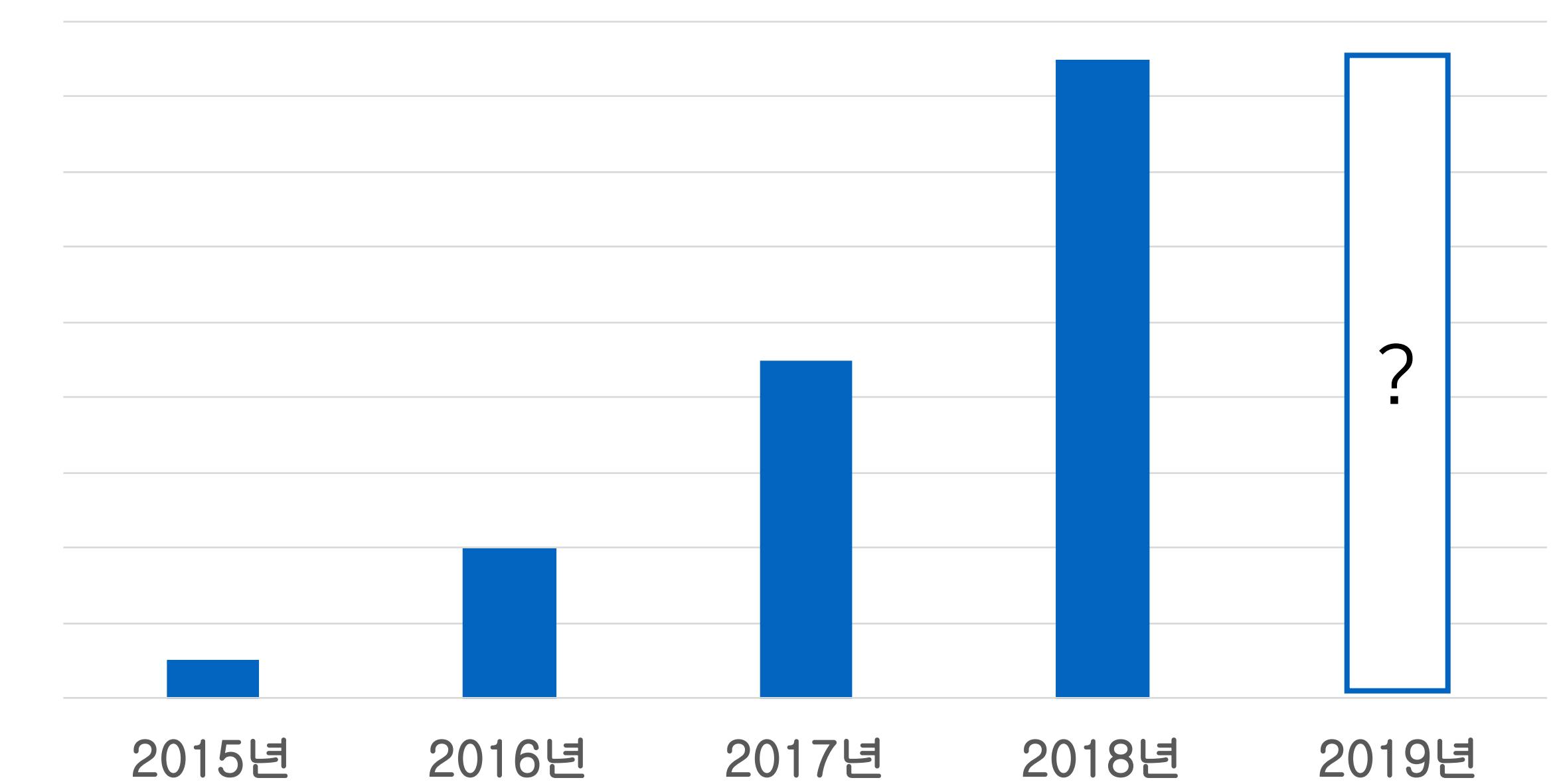
계속 발전하는 SRE

Anomaly Detection, Simulation 등

다양한 측면에서 발전을 위해 노력 중

SRE 인원 현황

2015년	2016년	2017년	2018년	2019년
1명	4명	9명	17명	?



마치며

Search Reliability Engineering

주변에서 아무도 해보지 않은 분야

참고할 사례가 적음

직접 두들겨 맞으면서 개선점을 찾아가는 방법 뿐

채용도 하고 있습니다!

질문은 Slido에 남겨주세요.

sli.do

#devview

TRACK 1

Thank you