

개략적인 규모 추정

2의 제곱수

2의 x 제곱	근사치	이름
10	1천	1KB
20	1백만	1MB
30	10억	1GB
40	1조	1TB
50	1000조	1PB

제대로 된 계산 결과를 얻기 위해 데이터 볼륨의 단위를 2의 제곱수로 표현하면, 어떻게 되는지를 알기 위한 값들.

응답 지연 값

연산명	시간
L1 캐시 참조	0.5ns
분기 예측 오류(branch mispredict)	5ns
L2 캐시 참조	7ns
뮤텍스(mutex) 락/언락	100ns
주 메모리 참조	100ns
Zippy로 1 KB 압축	10,000ns = 10μs
1 Gbps 네트워크로 2 KB 전송	20,000ns = 20μs
메모리에서 1 MB 순차적으로 read	250,000ns = 250μs
같은 데이터 센터 내에서의 메시지 왕복 지연시간	500,000ns = 500μs
디스크 탐색(seek)	10,000,000ns = 10ms
네트워크에서 1 MB 순차적으로 read	10,000,000ns = 10ms
디스크에서 1 MB 순차적으로 read	30,000,000ns = 30ms
한 패킷의 CA(캘리포니아)로부터 네덜란드까지의 왕복 지연시간	150,000,000ns = 150ms

ns = nanosecond(나노초), μs = microsecond(마이크로초), ms = millisecond(밀리초)
1나노초 = 10⁻⁹초
1마이크로초 = 10⁻⁶초 = 1,000나노초
1밀리초 = 10⁻³초 = 1,000μs = 1,000,000ns

각 연산에 따른 지연 시간을 정리해놓은 값들.
이 값을 통해서 대략적인 비교가 가능하다.

- 메모리는 빠르지만, 디스크는 느리다
- 디스크 탐색은 최대한 피하라
- 단순한 압축은 빠르다

- 데이터를 인터넷에 보내기 전에, 최대한 압축
- 데이터 센터는 여러 지역에 분산되어 있기 때문에, 센터들간의 데이터 전송비용은 비싸다.

가용성 수치

가용률	하루당 장애시간	주당 장애시간	개월당 장애시간	연간 장애시간
99%	14.40분	1.68시간	7.31시간	3.65일
99.9%	1.44분	10.08분	43.83분	8.77시간
99.99%	8.64초	1.01분	4.38분	52.60분
99.999%	864.00밀리초	6.05초	26.30초	5.26분
99.9999%	86.40밀리초	604.80밀리초	2.63초	31.56초

표 2-3

가용성 수치는, 고가용성에서 지속적으로 중단 없이 운영될 수 있는 능력을 수치로 나타낸 것.
각 %에 따라, 장애시간등을 알 수 있고 9가 많을수록 좋다고 보면 된다.

예제 : 트위터 QPS와 저장소 요구량 추정

가정

- MAU → 3억명
- 50%의 사용자는 매일 트위터를 사용
- 평균적으로 사용자들은 매일 2건의 트윗을 올림
- 미더어를 포함하는 트윗은 10%
- 데이터는 5년간 보관

추정

QPS

- $DAU = 3억 * 50\% \Rightarrow 1.5억$
- $QPS = 1.5억 * 2 트윗 / 24시간 / 3,600초 = 약 3,500$
- 최대 QPS = $QPS * 2 = 7,000$

미디어 저장을 위한 요구량

- 평균 트윗 크기
 - tweet_id = 64byte
 - text = 140byte
 - media = 1mb

- 미디어 저장소 요구량
 - $1.5\text{억} * 2 * 10\% * 1\text{mb} = 30\text{TB} / \text{일}$
- 5년간 요구량
 - $30\text{TB} * 365 * 5 = \text{대략 } 55\text{PB}$

팁

- 근사치를 활용한 계산
 - 면접시에, 복잡한 계산은 어렵기 때문에 대략적인 계산을 통해 간소화하자.
- 가정을 적어두어, 나중에 살펴보자
- 단위를 붙이자.
 - 스스로 헷갈리지 않게, 습관을 들여야 함.
- 많이 출제되는 개략적 규모 추정 문제는 아래와 같다.
 - QPS
 - 최대 QPS
 - 저장소 요구량
 - 캐시 요구량
 - 서버 수 추정

시스템 설계 면접 공략법

- 시스템 설계는 단순해보이지만, 결국 복잡한 것을 설계하게 된다.
- 이러한 복잡한 시스템을 완벽하게 설계하는 것을 기대하지 않는다.
- 결국 모호한 문제를 협력하여 해결해나가는 과정이다.
 - 답도 없고, 정해진 결말도 없다.
 - 설계 과정에서 내린 결정을 증명하고 방어하는 자리
 - 또한, 면접관의 피드백을 건설적으로 처리할 능력을 보여주는 자리

⇒ 시스템 설계 면접은 결국, “협력에 적합한 사람인지, 압박이 심한 상황에서 잘 헤쳐 나갈 자질이 있는지, 모호한 문제를 건설적으로 해결할 능력이 있는지” 를 살펴보는 자리.

효과적 면접을 위한 4단계 접근법

문제 이해 및 설계 범위 확정

- 바로 답을 내지 마라
 - 면접은 퀴즈쇼가 아니고, 정답이 없다.
- 깊이 생각하고 질문하여, 요구사항과 가정을 분명히 하라
- 아래와 같은 질문을 생각해보자.
 - 구체적으로 어떤 기능을 만들어야하는가?
 - 제품 사용자는 얼마나 되나?
 - 회사의 규모는 얼마나 빠르게 커지리 예상하나?
 - 회사의 기술 스택은 무엇인가?

개략적인 설계안 제시 및 동의 구하기

- 설계안에 대한 최초 청사진 제시 및 의견 구하기
- 화이트보드나 종이에 핵심 컴포넌트를 포함하는 다이어그램을 그려라
- 설계안이 시스템 규모에 관계된 제약사항들을 만족하는지 개략적으로 계산해보아라

상세 설계

- 시스템에서 전반적으로 달성해야 할 목표와 기능 범위 확인
- 전체 설계의 개략적 청사진 마련
- 해당 청사진에 대한 면접관의 의견 청취
- 상세 설계에서 집중해야 할 영역들 확인

마무리

- 면접관이 시스템 병목구간, 혹은 더 개선 가능한 지점을 찾아낼 주문 가능
 - 개선점은 언제나 있기 마련, 비판적으로 사고하고 개선점을 찾아내자.
- 설계를 다시 요약해주는 것도 도움이 될 수 있음.
 - 면접을 환기하는 역할

- 오류가 발생하면 무슨 일이 생기는지 따져보면, 흥미로울 것
- 운영 이슈도 논의해보자
 - 메트릭은 어떻게 수집하고, 모니터링 할 것인가.
 - 로그는 어떻게 관리하고, 배포할 것인가
- 미래에 닥칠 규모 확장 요구에 대해 어떻게 대처할 것인지
- 필요하지만 다루지 못한 세부적인 사항들도 제안

해야 할 것

- 질문을 통해 확인해라
- 문제의 요구사항을 이해하라
- 정답이나 최선의 답은 없다. 요구사항을 정확하게 이해했는지 확인해라
- 면접관이 사고 흐름을 이해할 수 있도록하라
- 가능하다면 여러 해법을 함께 제시해라
- 개략적 설계에 면접관이 동의하면, 각 컴포넌트의 세부사항을 설명하기 시작하라
- 면접관의 아이디어를 이끌어내라. 면접관이 팀원처럼 협력하도록해라
- 포기하지 마라

하지 말아야 할 것

- 전형적인 문제도 대비하지 않고, 면접장에 가지 않기
- 요구사항이나 가정들을 분명히 하지 않은 상태에서 설계를 제시 X
- 처음부터 특정 컴포넌트의 디테일에 집착 X
 - 개략적인 설계를 마친 뒤, 세부사항으로 나아가라
- 다시 말하지만, 소통을 주저 X
- 설계안을 내놓는다고 면접이 끝인게 X
 - 의견을 일찍 그리고 자주 구해라

시간 배분

보통 시스템설계 면접은 45분 혹은 1시간이 충분하지 않을 수 있다. 이러한 관점에서 시간관리는 필수적이고, 대략 아래와 같은 시간을 배분하자.

- 문제 이해 및 설계 범위 확정 : 3분 ~ 10분
- 개략적 설계안 제시 및 동의 구하기 : 10분 ~ 15분
- 상세 설계 : 10분에 ~ 25분
- 마무리 : 3분 ~ 5분