

2주차

개략적인 규모 추정

개략적인 규모추정은 보편적으로 통용되는 성능 수치상에서 사고 실험을 행하여 추정치를 계산하는 행위로서, 어떤 설계가 요구사항에 부합할 것인지 보기 위한것

개략적인 규모추정을 효과적으로 하려면?

→ 규모 확장성을 표현하는 데 필요한 2의 제곱수, 응답 지연 값, 가용성에 관계된 수치들을 기본적으로 잘 이해하고 있어야한다.

2의 제곱수

최소단위는 1바이트, 8비트로 구성 (1바이트 = 8비트)

ASCII 문자 하나가 차지하는 메모리 크기가 1바이트이다.

	근사치	이름	축약형
2^{10}	1천	1킬로바이트	1KB
2^{20}	1백만	1메가바이트	1MB
2^{30}	10억	1기가바이트	1GB
2^{40}	1조	1테라바이트	1TB
2^{50}	1000조	1페타바이트	1PB

1KiB는 뭘까???? 2의 10승을 1KiB로 썼던 기억이 있는데....

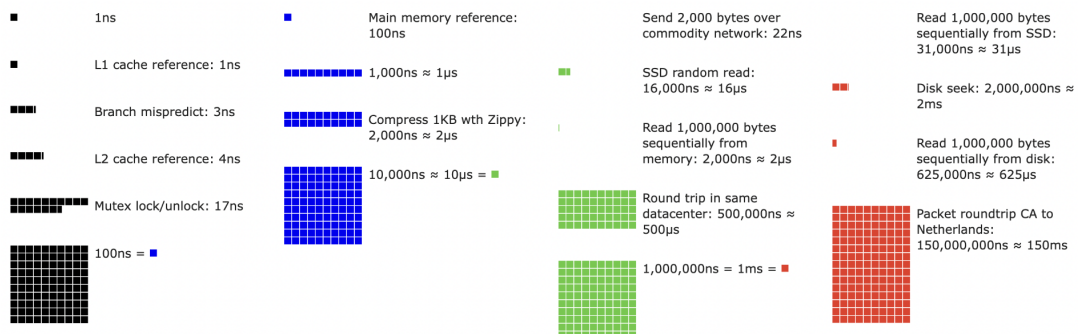
바이트 크기 v · d · e · h					
SI 접두어		전통적 용법		이진 접두어	
기호(이름)	값	기호	값	기호(이름)	V값
kB (킬로바이트)	$1000^1 = 10^3$	KB	$1024^1 = 2^{10}$	KiB (키비바이트)	2^{10}
MB (메가바이트)	$1000^2 = 10^6$	MB	$1024^2 = 2^{20}$	MiB (메비바이트)	2^{20}
GB (기가바이트)	$1000^3 = 10^9$	GB	$1024^3 = 2^{30}$	GiB (기비바이트)	2^{30}
TB (테라바이트)	$1000^4 = 10^{12}$	TB	$1024^4 = 2^{40}$	TiB (테비바이트)	2^{40}
PB (페타바이트)	$1000^5 = 10^{15}$	PB	$1024^5 = 2^{50}$	PiB (페비바이트)	2^{50}
EB (엑사바이트)	$1000^6 = 10^{18}$	EB	$1024^6 = 2^{60}$	EiB (엑스비바이트)	2^{60}
ZB (제타바이트)	$1000^7 = 10^{21}$	ZB	$1024^7 = 2^{70}$	ZiB (제비바이트)	2^{70}
YB (요타바이트)	$1000^8 = 10^{24}$	YB	$1024^8 = 2^{80}$	YiB (요비바이트)	2^{80}

킬로 이진 바이트는 원래 Kibbyte (KiB)가 따로 있어서

1KB는 1000 즉, 10의 3승을 의미하고 2의 10승은 1KiB(1키비바이트)라고 하지만,

킬로바이트 용어의 혼란 문제는 컴퓨터 하드 드라이브가 기가바이트로 용량이 커져서 대중화되는 동시에 1,000 과 1,024 둘 다 사용되었으며, 제조사는 10의 제곱을 사용하였지만 고객은 2의 제곱으로 용량을 계산하여 상당한 차이가 생길 수 있었다.

프로그래머가 알아야하는 응답지연 값 (2020)



- 메모리는 빠르지만 디스크는 아직도 느리다
- 디스크 탐색은 가능한 피하라
- 단순한 압축 알고리즘은 빠르다
- 데이터를 인터넷으로 전송하기 전에 가능하면 압축하라
- 데이터 센터는 보통 여러 지역에 분산되어있고 센터들간에 데이터를 주고 받는데 시간이 걸린다

가용성에 관계된 수치들

- 고가용성 : 시스템이 오랜 시간 동안 지속적으로 중단 없이 운영될 수 있는 능력
- 고가용성을 표현하는 값은 퍼센트로 표현하는데, 100퍼센트는 시스템이 단 한번도 중단된 적 없었음을 의미한다. 대부분의 서비스는 99퍼센트에서 100퍼센트의 값을 갖는다.
- SLA (Service Level Agreement) : 서비스 사업자와 고객사이에 맺어진 합의

시스템 설계 면접 공략법

시스템 설계 면접은 두명의 동료가 모호한 문제를 풀기 위해 협력하여 그 해결책을 찾아내는 과정에 대한 시뮬레이션일 뿐

시스템 설계 면접은

- 지원자의 설계 능력의 기술적 측면
- 지원자가 협력에 적합한 사람인지
- 압박이 심한 상황에서도 잘 헤쳐 나갈 자질이 있는지
- 모호한 문제를 건설적으로 해결할 능력이 있는지
- 좋은 질문을 던질 능력이 있는지

확인하기 위한 자리이다.

효과적인 면접을 위한 4단계 접근법

1. 문제 이해 및 설계 범위 확정

- 깊이 생각하고 질문하여 요구사항과 가정들을 분명히 하라. !!!
- 올바른 질문을 하는 것, 적절한 가정을 하는 것, 시스템 구축에 필요한 정보들을 모으는 것이 중요하다!

EX) 구체적으로 어떤 기능을 만들어야하나?

제품 사용자 수는 얼마나 되나?

회사의 규모는 얼마나 빨리 커지리라 예상하나? 석달, 여섯 달, 일년뒤의 규모는 얼마가 되리라 예상하는 가?

회사가 주로 사용하는 기술 스택은 무엇이냐? 설계를 단순화 하기 위해 활용할 수 있는 기존 서비스로는 어떤 것이 있는가?

2. 개략적인 설계안 제시 및 동의 구하기

- 면접관에게 개략적인 설계안을 제시하고, 면접관의 동의를 구하자
- 핵심 컴포넌트를 포함하는 다이어그램을 그려라 (클라이언트, API, 웹 서버, 데이터 저장소, 캐시, CDN, 메시지 큐같은 것들이 포함될 수 있다)
- 이 최초 설계안이 제약사항들을 만족하는지 개략적으로 계산해보고 면접관에게 이러한 추정이 필요한지 물어보자

3. 상세설계

- 시스템에서 전반적으로 달성해야할 목표와 기능 범위를 확인
- 전체 설계의 개략적 청사진 마련
- 해당 청사진에 대한 면접관의 의견 청취
- 상세 설계에서 집중해야할 영역들 확인

면접관과 해야할 일

- 컴포넌트들의 우선순위 정하기
- 단축 URL 생성기 설계 → 해시 함수의 설계를 구체적으로 설명하기
- 채팅 시스템 → 지연시간을 줄이고 사용자의 온/오프상태를 표시할 것인지
-

4. 마무리

- 개선가능한 부분을 찾고 비판적으로 설계를 바라보자
- 설계를 다시 한 번 요약하자
- 오류를 따져보자
- 운영 이슈도 논의해보자
- 확장성에 대해 논의해보자

해야할것

- 질문을 통해 확인하라
- 문제의 요구사항을 이해하라
- 면접관과 소통하라
- 가능한 여러 해법을 제시하라
- 면접관이 개략적 설계에 동의한다면 컴포넌트의 세부사항을 설명하기 시작하라, 가장 중요한 것 부터
- 포기하지 마라

하지 말아야 할 것

- 전형적인 면접 문제들에도 대비하지 않은 상태에서 면접장에 가지마라
- 요구사항이나 가정들을 분명히하지 않은 상태에서 설계를 제시하지 마라
- 처음부터 특정 컴포넌트의 세부사항을 깊이 설명하지 마라
- 힌트 청하기를 주저하지 마라
- 소통을 주저하지 마라
- 의견을 일찍 , 자주 구하라