重要通知

- 第10期《系统设计班》第5-7节课程时间调整，更新为：
  - 第5节 美西时间 2月6日 周六 早上 09:30(美东时间2月6日 12:30)
  - 第6节 美西时间 2月7日 周日 早上 09:30(美东时间2月7日 12:30)
  - 第7节 美西时间 2月13日 周六 早上 09:30(美东时间2月13日 12:30)
- 也可在官网-我的课程，查看所在时区的时间。

系统设计公开课

即将开始，小憩片刻

关注微信/微博，获取最新面试题及权威解答

微信: ninechapter

微博: http://www.weibo.com/ninechapter

官网: www.jiuzhang.com

System_Design_1

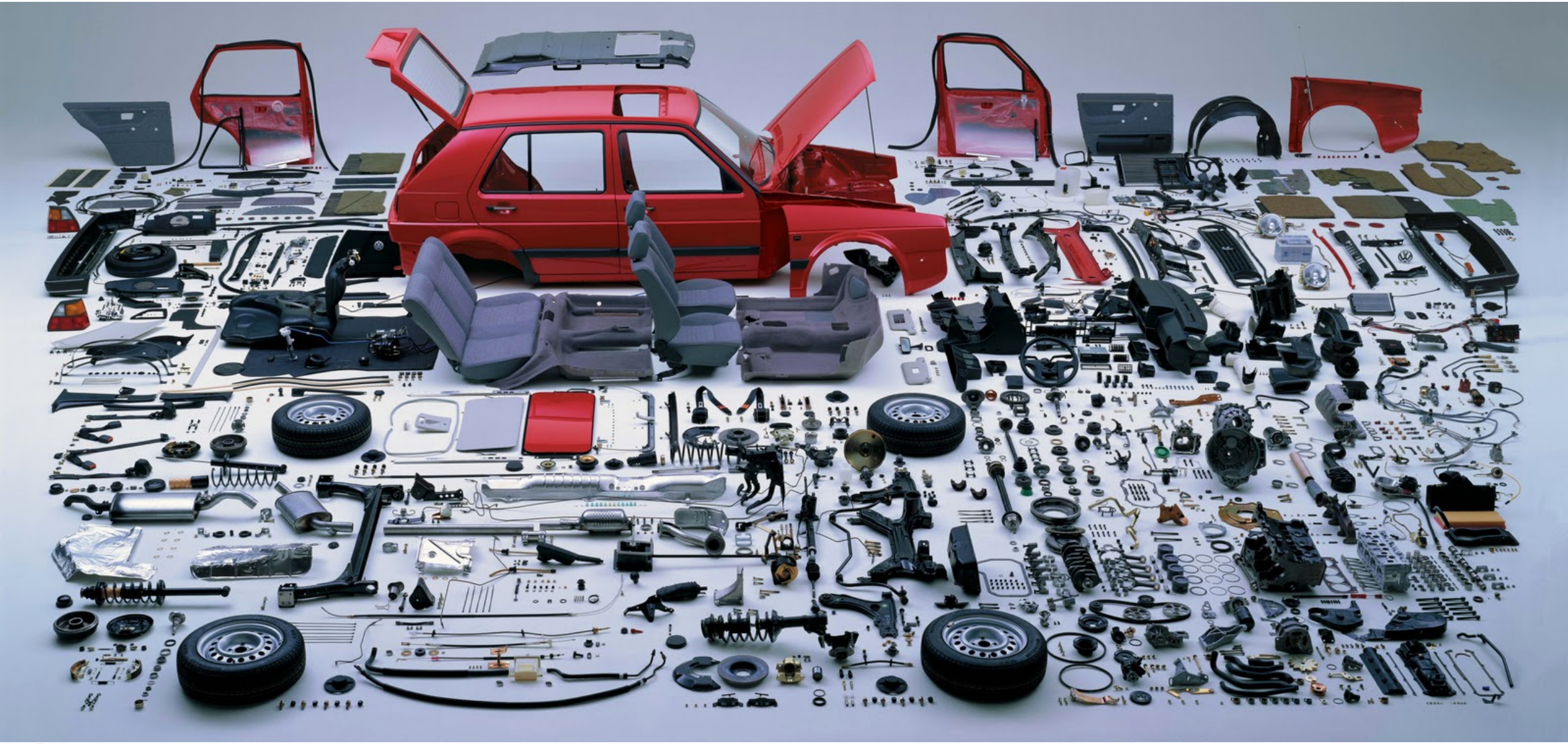
How to design a system?

张无忌

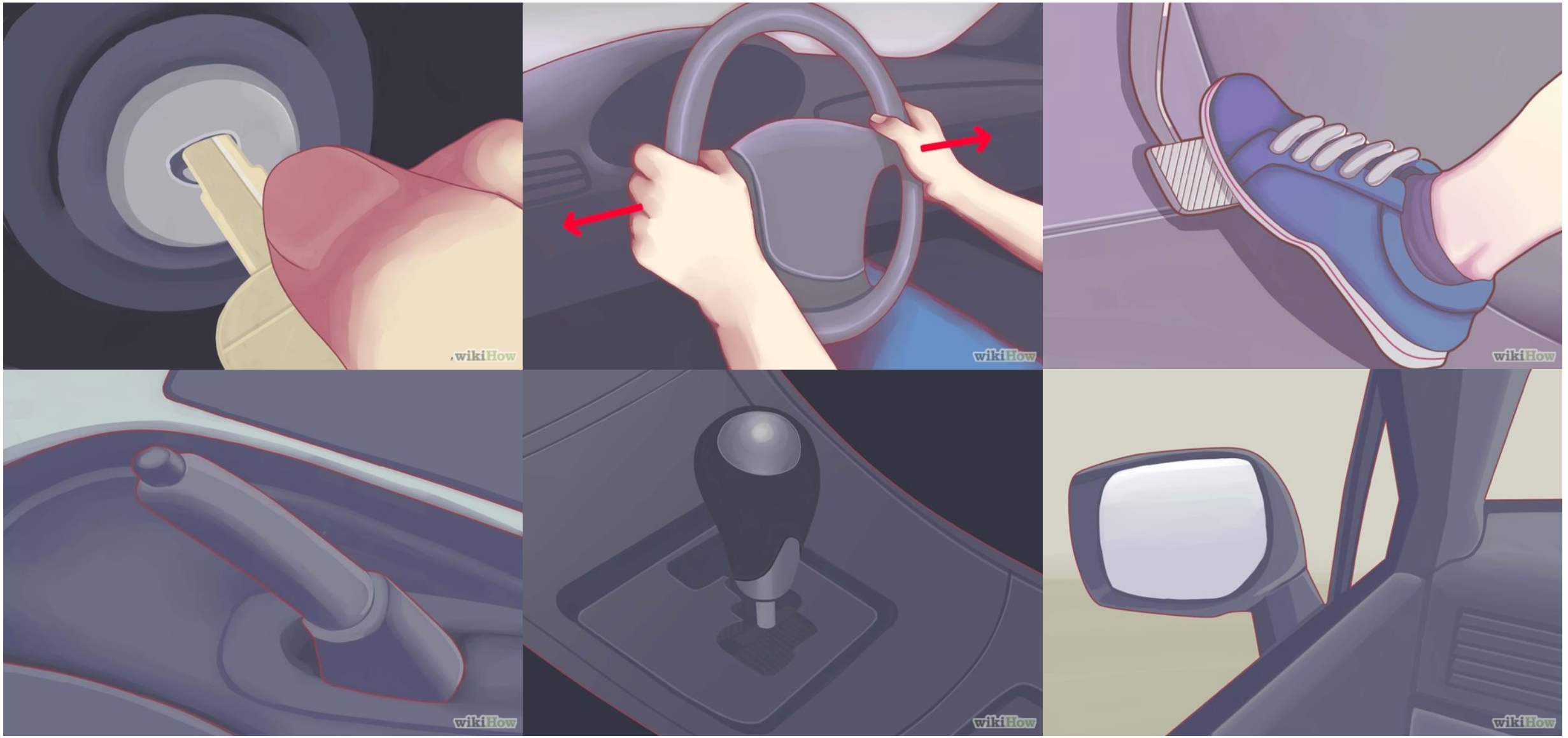
2016-01-16

V5.0.0

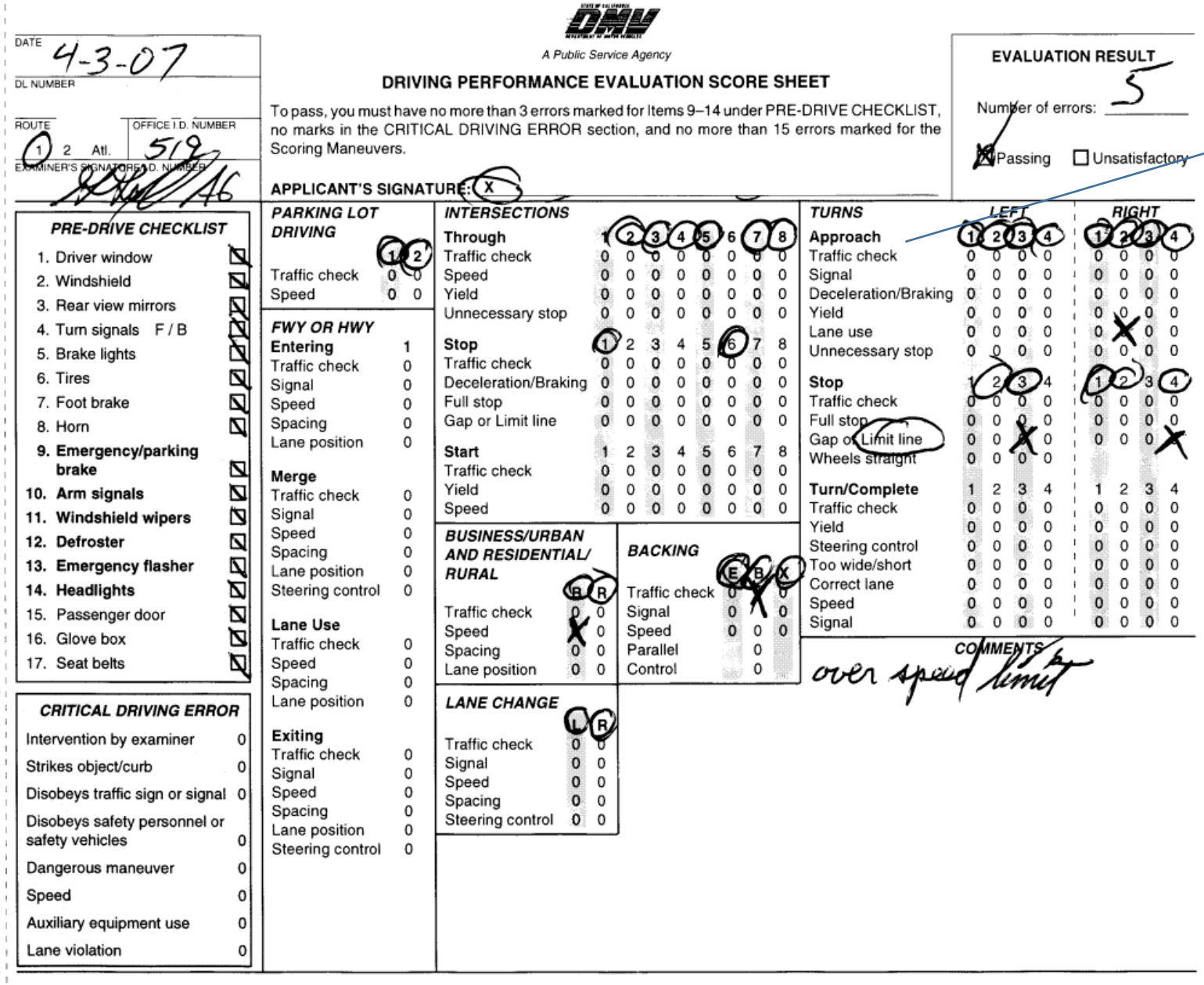
How to drive a car?



How to drive a car?



How to get driver license in one week?



DMV

A Public Service Agency

DRIVING PERFORMANCE EVALUATION SCORE SHEET

To pass, you must have no more than 3 errors marked for Items 9–14 under PRE-DRIVE CHECKLIST, no marks in the CRITICAL DRIVING ERROR section, and no more than 15 errors marked for the Scoring Maneuvers.

DATE 4-3-07

DL NUMBER

ROUTE 1 2 Atl.

OFFICE I.D. NUMBER 519

EXAMINER'S SIGNATURE/I.D. NUMBER 16

APPLICANT'S SIGNATURE: X

EVALUATION RESULT

Number of errors: 5

X Passing ☐ Unsatisfactory

PRE-DRIVE CHECKLIST

1. Driver window	☑
2. Windshield	☑
3. Rear view mirrors	☑
4. Turn signals F / B	☑ ☑
5. Brake lights	☑
6. Tires	☑
7. Foot brake	☑
8. Horn	☑
9. Emergency/parking brake	☑
10. Arm signals	☑
11. Windshield wipers	☑
12. Defroster	☑
13. Emergency flasher	☑
14. Headlights	☑
15. Passenger door	☑
16. Glove box	☑
17. Seat belts	☑

CRITICAL DRIVING ERROR

Intervention by examiner	0
Strikes object/curb	0
Disobeys traffic sign or signal	0
Disobeys safety personnel or safety vehicles	0
Dangerous maneuver	0
Speed	0
Auxiliary equipment use	0
Lane violation	0

PARKING LOT DRIVING

	1	2
Traffic check	0	0
Speed	0	0

FWY OR HWY

Entering	1
Traffic check	0
Signal	0
Speed	0
Spacing	0
Lane position	0
Merge	
Traffic check	0
Signal	0
Speed	0
Spacing	0
Lane position	0
Steering control	0
Lane Use	
Traffic check	0
Speed	0
Spacing	0
Lane position	0
Exiting	
Traffic check	0
Signal	0
Speed	0
Spacing	0
Lane position	0
Steering control	0

INTERSECTIONS

	1	2	3	4	5	6	7	8
Through								
Traffic check	0	0	0	0	0	0	0	0
Speed	0	0	0	0	0	0	0	0
Yield	0	0	0	0	0	0	0	0
Unnecessary stop	0	0	0	0	0	0	0	0
Stop	1	2	3	4	5	6	7	8
Traffic check	0	0	0	0	0	0	0	0
Deceleration/Braking	0	0	0	0	0	0	0	0
Full stop	0	0	0	0	0	0	0	0
Gap or Limit line	0	0	0	0	0	0	0	0
Start	1	2	3	4	5	6	7	8
Traffic check	0	0	0	0	0	0	0	0
Yield	0	0	0	0	0	0	0	0
Speed	0	0	0	0	0	0	0	0

BUSINESS/URBAN AND RESIDENTIAL/ RURAL

	R	R
Traffic check	0	0
Speed	X	0
Spacing	0	0
Lane position	0	0

LANE CHANGE

	L	R
Traffic check	0	0
Signal	0	0
Speed	0	0
Spacing	0	0
Steering control	0	0

BACKING

	E	B	X
Traffic check	0		0
Signal	0	X	0
Speed	0	0	0
Parallel		0	
Control		0	

TURNS

	LEFT				RIGHT			
Approach	1	2	3	4	1	2	3	4
Traffic check	0	0	0	0	0	0	0	0
Signal	0	0	0	0	0	0	0	0
Deceleration/Braking	0	0	0	0	0	0	0	0
Yield	0	0	0	0	0	0	0	0
Lane use	0	0	0	0	0	X	0	0
Unnecessary stop	0	0	0	0	0	0	0	0
Stop	1	2	3	4	1	2	3	4
Traffic check	0	0	0	0	0	0	0	0
Full stop	0	0	0	0	0	0	0	0
Gap or Limit line	0	0	X	0	0	0	0	X
Wheels straight	0	0	0	0				
Turn/Complete	1	2	3	4	1	2	3	4
Traffic check	0	0	0	0	0	0	0	0
Yield	0	0	0	0	0	0	0	0
Steering control	0	0	0	0	0	0	0	0
Too wide/short	0	0	0	0	0	0	0	0
Correct lane	0	0	0	0	0	0	0	0
Speed	0	0	0	0	0	0	0	0
Signal	0	0	0	0	0	0	0	0

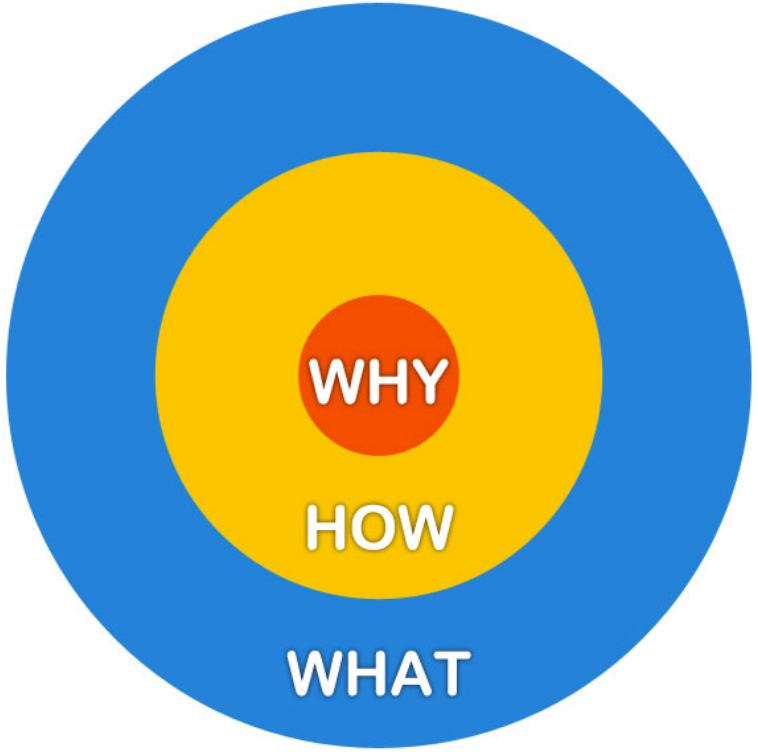
COMMENTS

over speed limit by

DL 32 (REV. 3/2000)

Turn your head exaggeratedly to your left

The secret of WHY



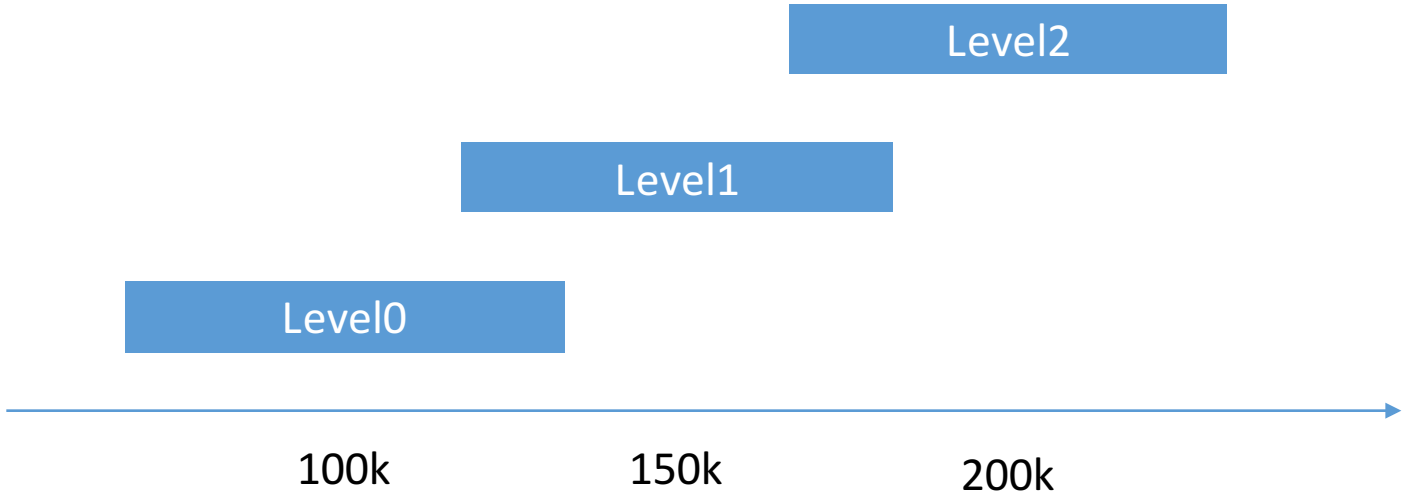
Why do we need system design?

Read More

Novice, http://url.cn/ZrlZtf
Novice, http://url.cn/YrnOHy

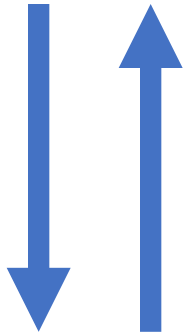
Engineer Level	Ability		Interview Challenge	Base
0 = Fresh Graduate	Function	Algorithm Design	Binary search tree	100k
1	Simple module	System Design	TinyURL	150k
2	Complex module			200k
3 = Manager	Project with 5 people		Payment system	250k
4	Project with 10+ people			300k
5 = Director	Project with 20+ people		Twitter	???

System design is 100% waiting for you in the job/promotion interview.



What is system design?

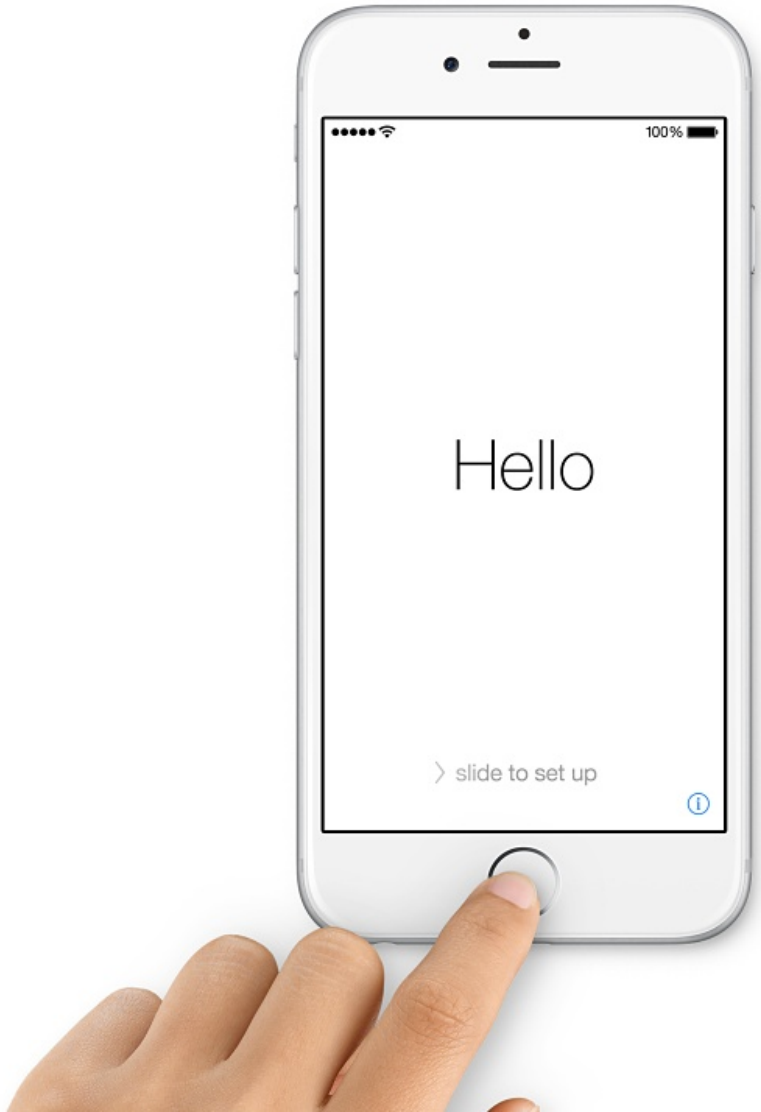
- Definition
  - The process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements.

Type	Transparency	Level	Methodology
Conceptual design	Black	Macro	
Logical design	Gray	-	
Physical design	White	Micro	

What is a good design?

- Healthiness
  - Execution
  - Communication

- Simplicity
  - No more; no less
  - Understandable



It is hard to work out a good design

JUST DO IT.

After class_1, you can answer

- Fundamental questions in system design interviews
  - Please design the system
  - Please evaluate query per second
  - Please scale your system

- Design Netflix/YouTube/Spotify
  - Uber, Google, Alibaba

After class_2, you can answer

- Design data with class and inheritance
  - Google, Alibaba, Facebook, Epic, …
- Design user system
  - Netflix
- Design payment system
  - Yelp, BigCommerce

After class_3, you can answer

- Design crawler
  - Dropbox, Google, Turn, Alibaba
- Design thread-safe consumer and producer
  - Google, Amazon, TripAdvisor, Microsoft, Pure Storage, Dropbox, LinkedIn, Palantir, Intel, Bloomberg, …
- Design TinyURL
  - LinkedIn, Uber, Bloomberg, Hulu

After class_4, you can answer

- What happens when you visit www.google.com in your browser?
  - Uber, Two Sigma, Alibaba, Baidu
- How to increase visiting speed of a webpage?
  - Yelp, Alibaba
- Design “秒杀” System?
  - Alibaba, Baidu, Tencent
- Design rate limiter
  - Yelp, Facebook, Google

After class_5, you can answer

- Design distributed file system(GFS) and database(BigTable)?
  - Google, DrawBridge, Yelp
- Calculate word appearance/inverted index/anagrams with MapReduce
  - Google, Twitter, Drawbridge, Zenefits, Bloomberg, Genapsys, Liveramp, …

After class_6, you can answer

- Design Twitter/Instagram/Facebook
  - Facebook, Uber, Shopkick, Bloomberg
- Design WhatsAPP/Facebook Chat
  - Uber, Facebook, TubeMogul

After class_7, you can answer

- Design typeahead/instant search
  - Facebook, Google, Factset, Expedia, Walmart, Alibaba
- Object oriented design: elevator, parking lot, blackjack
  - Facebook, Google, Twitter, Yelp, Amazon, FaceSet, Groupon, Alibaba, Tencent, Baidu, …
- Achievement system
  - PocketGem

Interviewer

Please design
“Netflix”

Read More

Expert, http://url.cn/fn5mWs
Expert, http://url.cn/cehN0N
Expert/Master, http://url.cn/Ui5CoP
Master, http://url.cn/d2fblH

Try it

http://www.netflix.com

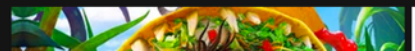
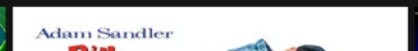
NETFLIX Browse Kids DVD Search Wuji

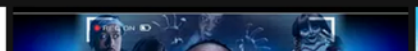
Popular on Netflix

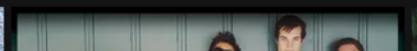
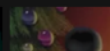
Trending Now

Exciting Movies

Goofy Comedies



Let’s begin by macro design

Scenario: case/interface

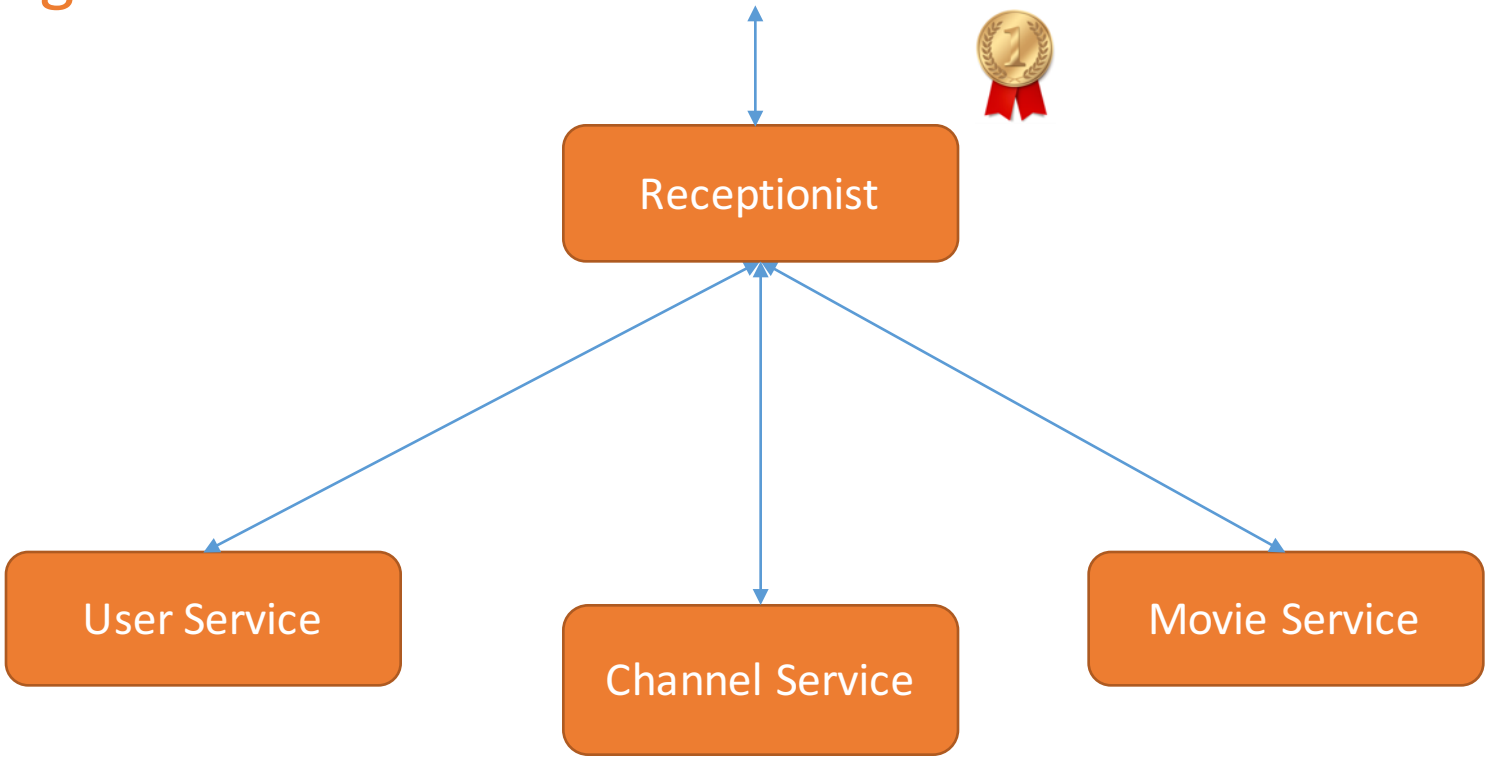
- Step 1: Enumerate 
  - Register/Login
  - Play movie
  - Movie recommendation
- Step 2: Sort 
  - TOP1: Play movie
    - Get channels
    - Get movies in channels
    - Play a movie in a channel

Necessary: constrain/hypothesis

- Step 1: Ask
  - 5,000,000 daily active users
- Step 2: Predict
  - User
    - Average concurrent users = daily_active_users / daily_seconds * average_online_time
      = 5,000,000 / (24*60*60) * (30*60)
      = 104,167
    - Peak users = average_concurrent_users * 6 = 625,000
    - MAX peak users in 3 months = Peak_users * 2 = 1,250,000
  - Traffic
    - Traffic per user = 3 mbps
    - MAX peak traffic = 1,250,000 * 3 mbps = 3.75 Tb/s
  - Memory
    - Memory per user = 10KB
    - MAX daily memory = 5,000,000 * 2 * 10 = 100GB
  - Storage
    - Total movie = 14,000
    - Movie storage = Total_movie * average_movie_size = 14,000*50GB = 700TB

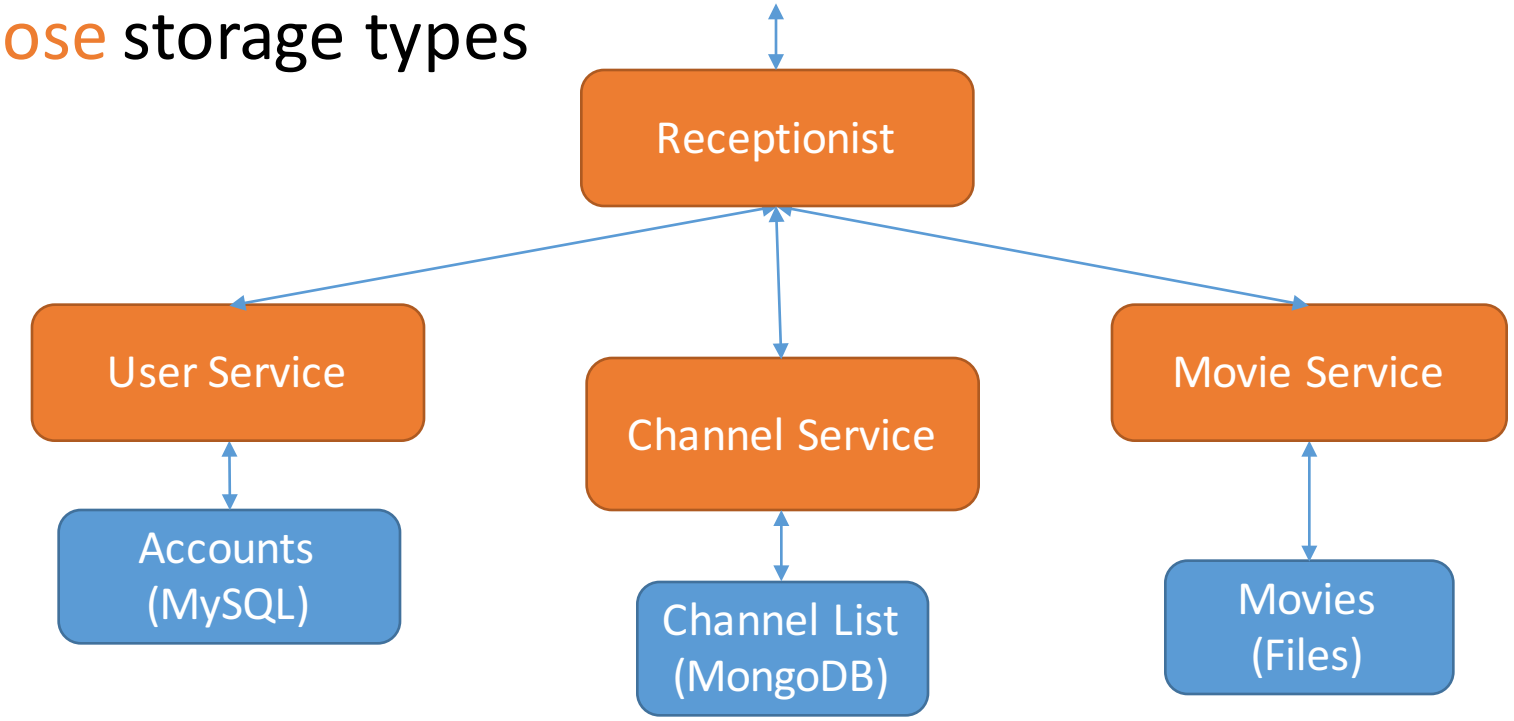
Application: service/algorithm

- Step 1: Replay the case, add a service for each request
- Step 2: Merge the services



Kilobit: data

- Step 1: Append dataset for each request below a service
- Step 2: Choose storage types



Read More Expert, http://url.cn/UVzYRH
Expert, http://url.cn/eRn6Ff

Evolve

- Step 1: Analyze
  - with
    - Better: constrains
    - Broader: new cases
    - Deeper: details

  - from the views of
    - Efficiency
    - Scalability
    - Robustness

- Step 2: Go back by evolving accordingly

Crack a design in 5 steps

- Scenario: case/interface
- Necessary: constrain/hypothesis

- Application: service/algorithm
- Kilobit: data

- Evolve

Philosopher at the gate of Peking University

- Who are you?
- Where are you from?
- Where are you going to?

Interview evaluation in 10 min

Note

- Prepare a pen and a paper
- In 4 directions
  - Give yourself a score of 0-5
    - One-by-one
    - Believe in your intuition

Calculate your scores now

编程 (1/4)

1 （入门）	2 （深度）	3 （广度）	4 （经验）	5 （预测）
能够完成函数级开发	能够完成模块级开发	能够完成项目级开发	做为主程参与大规模系统的研发	能够前瞻性的预测研发的走势并做出准备
现场写代码 评分标准： a) 可读 b) 有效 c) 防御	多线程：解决生产者/消费者类问题 网络：解决爬虫类问题 数据库：使用MySQL/NoSQL 调试：发现并解决BUG 优化：在时间和空间维度优化 评分标准： a) 实例支撑 b) 不需要都会，要在2点上有深度	技术调研：介绍调研的案例 代码整合：介绍整合开源/内部库的案例 突发问题：介绍解决突发问题的案例 评分标准： a) 体现项目级别 b) 至少在2点上有特色	介绍你做为主程的最挑战的项目 评分标准： a) 案例支持技术点 b) 强调整体的把握能力	作为项目的主程，你下一步会怎么做？

设计 (2/4)

1 （入门）	2 （深度）	3 （广度）	4 （经验）	5 （预测）
通过算法与数据结构解决问题	能够设计基本系统	能够设计大规模分布式系统	参与真实系统设计并贡献重要力量	能够前瞻性的预测架构的走势并进行准备
字符串处理算法题	用面向对象思想设计消息系统	设计日活跃1000万的消息系统	介绍你参与过的最复杂的系统设计 追问实例和细节	作为首席架构师，你下一步会做什么？

理解/学习 (3/4)

1 （入门）	2 （深度）	3 （广度）	4 （经验）	5 （预测）
能够快速抓住对方沟通的重点	能够主动学习需要掌握的技能	从更大的维度补充所需要的知识？	具有钻研突破难题的经验	有清晰的职业规划
能否一遍听懂面试官问题？	你在上家公司是如何上手的？	你在上家公司有哪些提升？ 是如何提升的？	在上家公司遇到了哪些挑战（偏项目）？ 是怎么解决的？ 如果重来一次，有什么更好的方案？	你的职业规划是什么？ 你希望在哪方面提升？ 为什么来我们公司？（刨根问底）

总结/表达/教学 (4/4)

1 （入门）	2 （深度）	3 （广度）	4 （经验）	5 （预测）
能够简约的表达重点	能够把自己的知识教给身边的人	能从对方的角度讲问题（跨专业）	具有对公共分享的经验	沟通前瞻性 能够预知对方的需求和疑问点，并主动沟通
介绍你自己？ 你的亮点和要提高的地方是什么？	如何带新人？ 如何让他融入团队？	如何解决意见不一致？ 挑战他的解题答案	在部门/公司的分享案例	Feel it by yourself

Send me your score

(2+1+2+2=7)

Understand your level

Score	Level	Ali	Baidu	Tencent
8-9	L1	P5	T4	T1.2-T2.2
10-13	L2	P6	T5	T2.2-T3.1
14(+)	L3(+)	P7(+)	T6(+)	T3.1-T3.2

Save your score and reevaluate yourself after our class

Structural interview

能力矩阵 （适用P5-P7）	1 （入门）	2 （深度）	3 （广度）	4 （经验）	5 （预测）
编程	能够完成函数级开发	能够完成模块级开发	能够完成项目级开发	做为主程参与大规模系统的研发	能够前瞻性的预测研发的走势并做出准备
	现场写代码 评分标准： a) 可读 b) 有效 c) 防御	多线程：解决生产者/消费者类问题 网络：解决爬虫类问题 数据库：使用MySQL/NoSQL 调试：发现并解决BUG 优化：在时间和空间维度优化 评分标准： a) 实例支撑 b) 不需要都会，要在2点上有深度	技术调研：介绍调研的案例 代码整合：介绍整合开源/内部库的案例 突发问题：介绍解决突发问题的案例 评分标准： a) 体现项目级别 b) 至少在2点上有特色	介绍你做为主程的最挑战的项目 评分标准： a) 案例支持技术点 b) 强调整体的把握能力	作为项目的主程，你下一步会怎么做？
设计	通过算法与数据结构解决问题	能够设计基本系统	能够设计大规模分布式系统	参与真实系统设计并贡献重要力量	能够前瞻性的预测架构的走势并进行准备
	字符串处理算法题	用面向对象思想设计消息系统	设计日活跃1000万的消息系统	介绍你参与过的最复杂的架构设计？ 有哪些经验和改进的思路？	作为首席架构师，你下一步会做什么？
理解/学习	能够快速抓住对方沟通的重点	能够主动学习需要掌握的技能	从更大的维度补充所需要的知识	具有钻研突破难题的经验	有清晰的职业规划
	能否一遍听懂面试官问题？	你进入上家公司时，是如何上手的？	你在上家公司有哪些提升，是如何提升的？	在上家公司遇到了哪些挑战（偏项目）？ 是怎么解决的？ 如果重来一次，有什么更好的方案？	你的职业规划是什么？ 你希望提升的方向在哪里？ 为什么来我们公司？
总结/表达/教学	能够用简约的话表达重点	能够把自己的知识教给身边的人	能从对方的角度讲问题（跨专业）	具有对公共分享的经验	沟通前瞻性，能够预知对方的需求和疑问点，并主动沟通
	介绍你自己？ 你的亮点和需要提高的地方是什么？	你是如何带新人的？ 如何让他融入团队？	如何解决意见不一致？ 挑战他的解题答案	在部门/公司的分享案例	Feel it by yourself

Let’s try micro design

Interviewer: design recommendation module

- Each user likes a set of movies
  - u1 = { m3, m5, m7, m11}
  - u2 = { m1, m2, m3, m4, m5, m6, m7, m8, m9}
- Similarity(u1, u2) = 3

- For a user, find his top-1 similar user

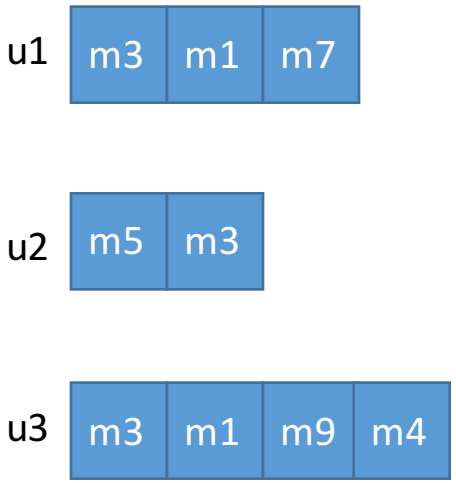
Scenario: interface

```
class Recommender{

    public:
        int FindSimilarUser(int userID);

};
```

Necessary: constrain/hypothesis

- Predict
  - Max peak users = 1,250,000
  - Calculation frequency = 1 /10min/user
  - Peak QPS (Queries Per Second) = Max_peak_users * Calculation_frequency
    = 1,250,000 * 1/(10*60)
    = 2083/s

Algorithm + Data



Key steps:

For u1 and u2,

m3 != m5

m3 == m3

m1 != m5

m1 != m3

m7 != m5

m7 != m3

Similarity(u1,u2)= 1

For u1 and u3,

m3 == m3

m3 != m1

m3 != m9

m3 != m4

m1 != m3

m1 == m1

m1 != m9

m1 != m4

m7 != m3

m7 != m1

m7 != m9

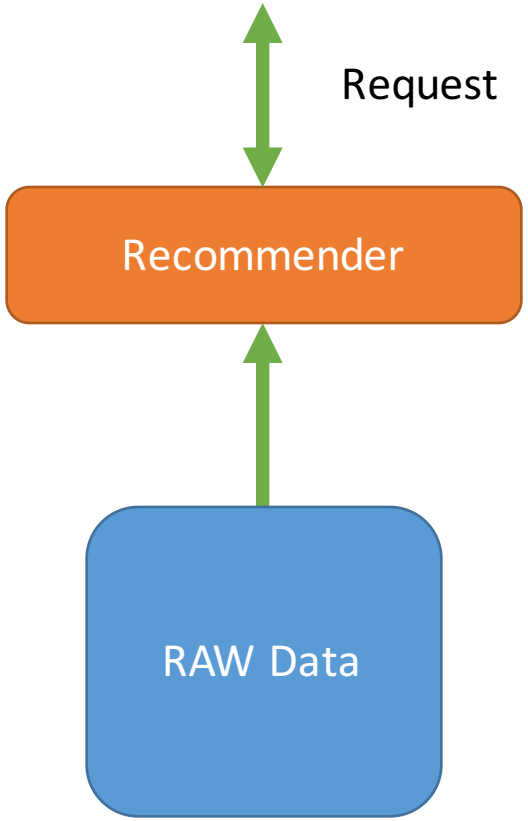
m7 != m4

Similarity(u1,u3)= 2

Performance = 0.2s/query

MAX capability = 5qps

System design (v1)



How to go from Level_0 to Level_1?

Engineer Level	Ability		Interview Challenge	Package
0 = Fresh Graduate	Function	Algorithm Design	Binary tree	10W
1	Simple module	System Design	TinyURL	15W

- Evolve from the view of
  - Performance
  - Scalability
  - Robustness

Interviewer: improve efficiency

QPS=2083

Improve performance with inverted index

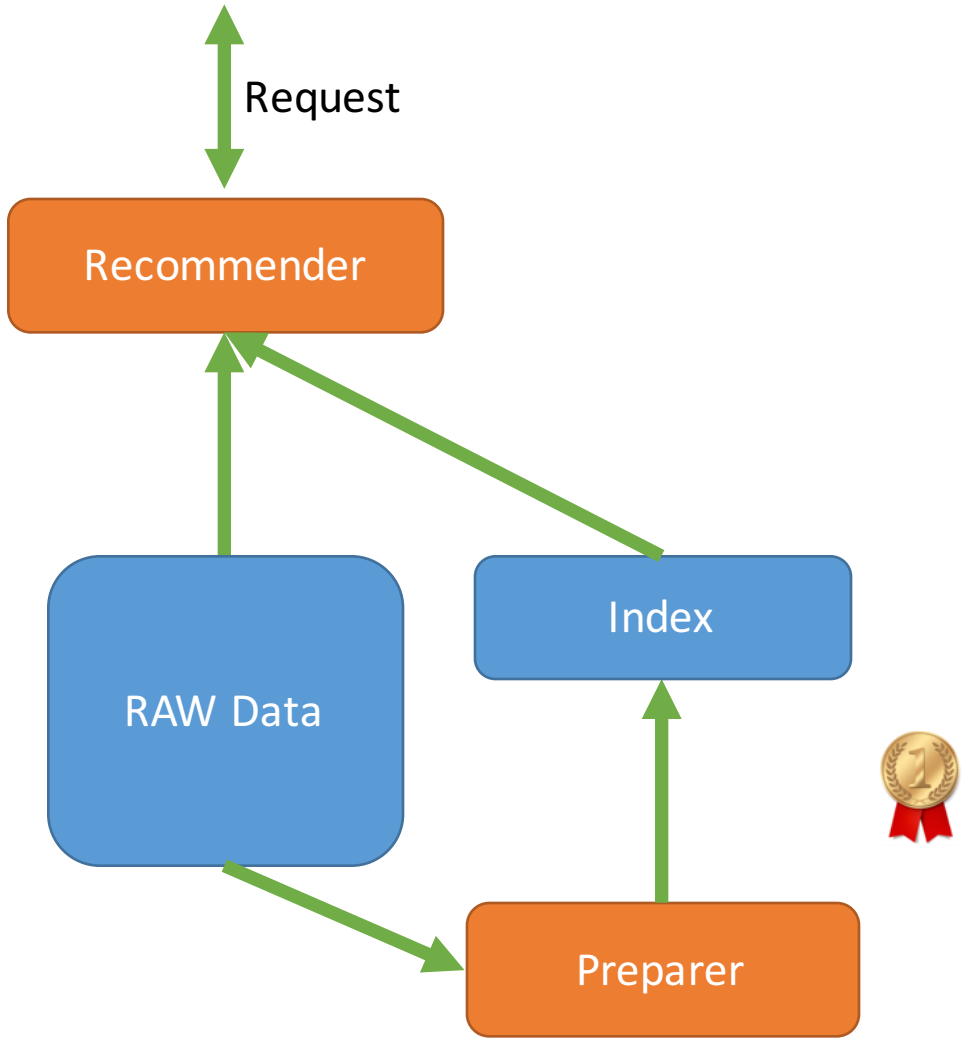
Key steps:

For movie liked by u1,
  For m3,
    Similarity(u1,u2) += 1
    Similarity(u1,u3) += 1
  For m1,
    Similarity(u1,u3) += 1
  For m7,
    none

Similarity(u1,u2)= 1
Similarity(u1,u3)= 2

Performance = 0.02s/query
MAX capability = 50qps

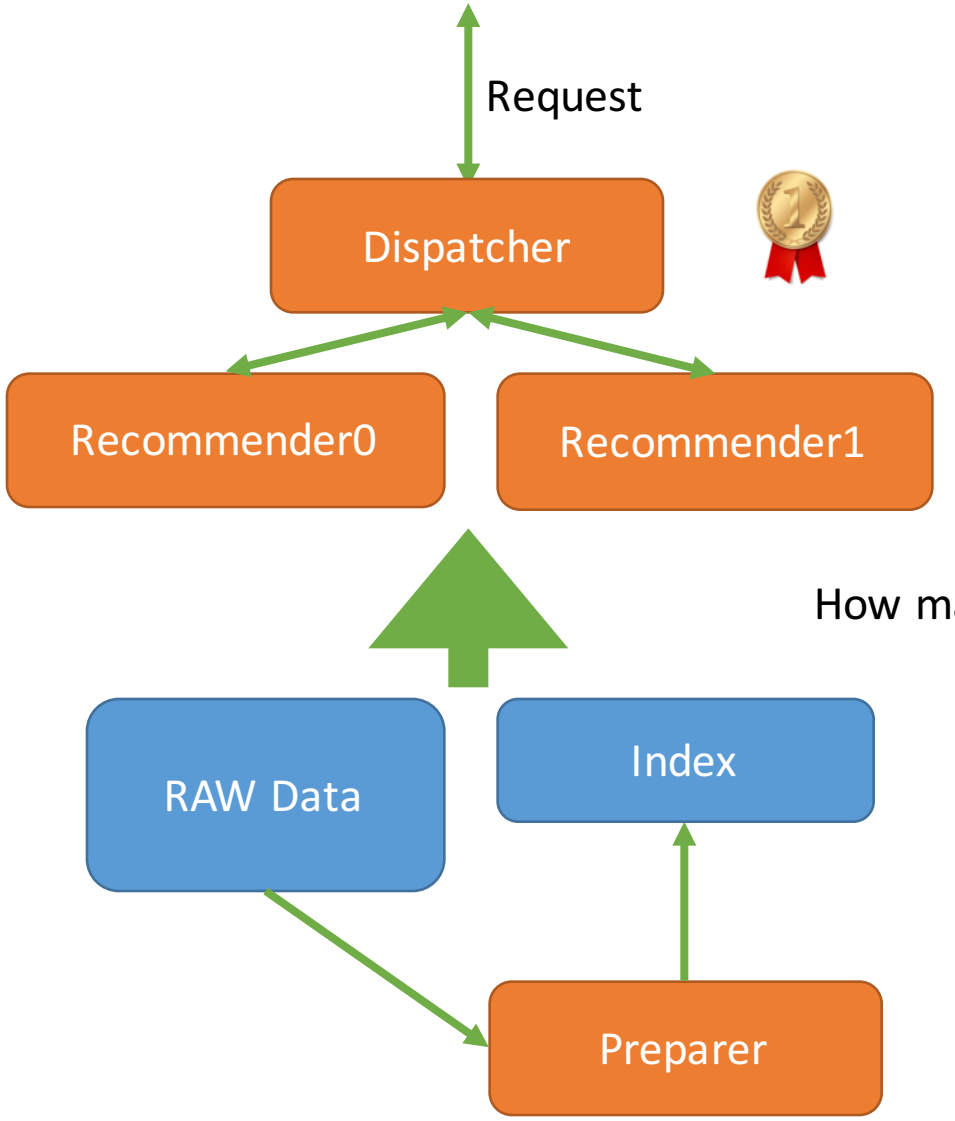
System design (v2) with preparer



Interviewer: improve scalability

QPS=2083

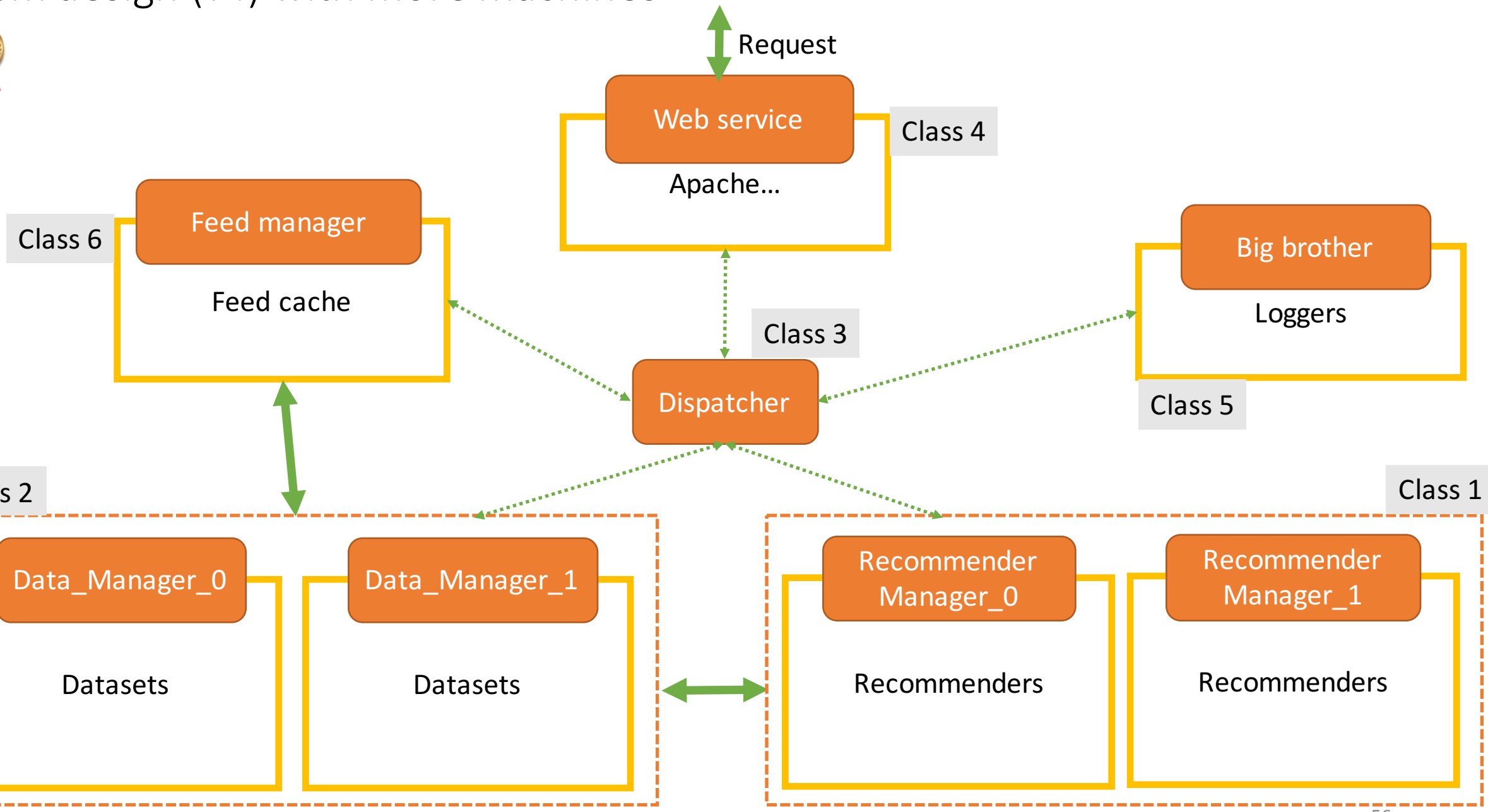
System design (v3) with dispatcher



How many recommenders do you need to achieve 2083qps?

Interviewer: improve robustness

System design (v4) with more machines



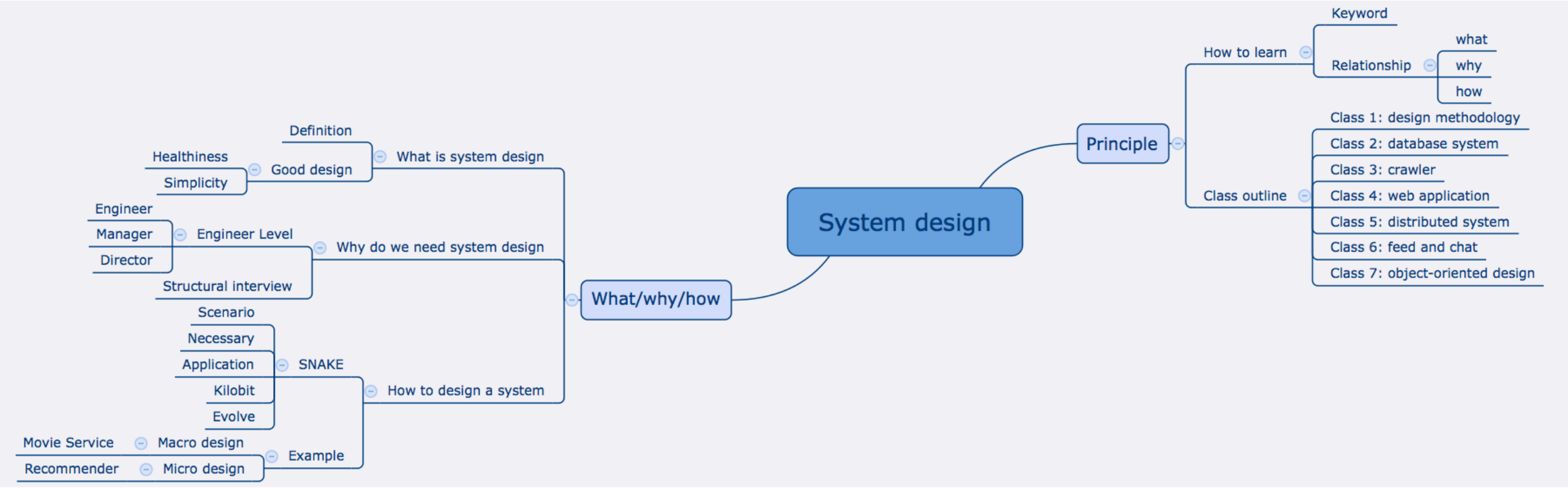
Review: SNAKE

- Scenario: case/interface
  - Enumerate & Sort
- Necessary: constrain/hypothesis
  - Ask & Predict
- Application: service/algorithm
  - Replay & Merge
- Kilobit: data
  - Append & Choose
- Evolve
  - Analyze & Go back

We are building a real system

Class	Challenge	Key points	Time (US)
1	What/why/how to design a system?	Snake	2016.01.16
2	Design account/payment system	Data/database	2016.01.23
3	Implement entertainment news to increase user activities	Crawler	2016.01.24
4	Reduce failure rate to increase user satisfaction	Web application	2016.01.30
5	Scale our system into Petabyte	Distributed systems	2016.02.06
6	Implement chat and moments to increase activities among users	Feed and chat	2016.02.07
7	Design achievement system to award our users	Object oriented design	2016.02.13

Summary



After class_1, you can answer

- Fundamental questions in system design interviews
  - Please design the system
  - Please evaluate query per second
  - Please scale your system

- Please design Netflix/YouTube/Spotify
  - Uber, Google, Alibaba

What is NineChapter?

- Our class
  - http://www.jiuzhang.com/course/2/
- More class
  - http://www.jiuzhang.com/course/
- Solution/Lintcode
  - http://www.jiuzhang.com/solutions/
- QA
  - http://www.jiuzhang.com/qa/

Last thing: the secret of study

- Keywords (Points)
- Relationships
  - What
  - Why
  - How

Homework

- Please design Netflix?
  - http://www.jiuzhang.com/qa/50/

QA

关注微信/微博，获取最新面试题及权威解答

微信: ninechapter

微博: http://www.weibo.com/ninechapter

官网: www.jiuzhang.com