

# 架构实战营 - 模块4

## 第4课： 如何设计存储架构

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

李运华

前阿里资深技术专家(P9)

# 教学目标

1. 掌握存储架构设计的总体思路
2. 掌握如何估算业务需求所需的存储性能
3. 掌握如何选择合适的存储系统
4. 掌握如何设计出合理的存储方案

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血



厚积薄发

# 目录

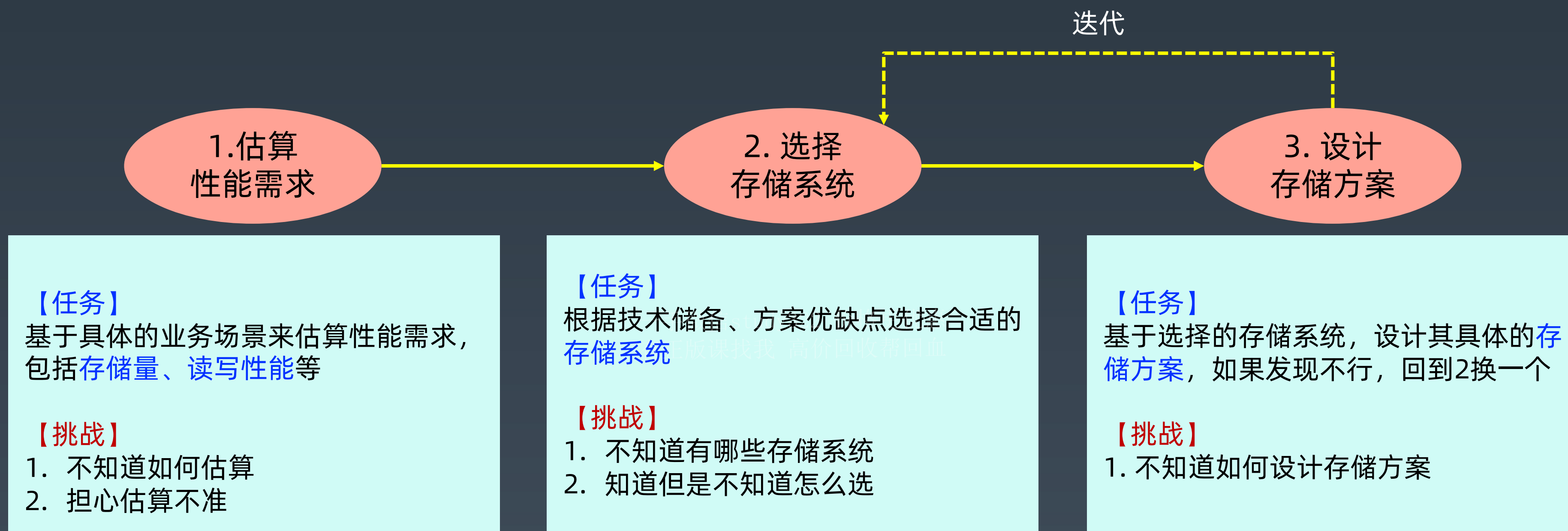
1. 存储架构设计总体思路
2. 如何估算存储性能需求
3. 如何选择存储系统
4. 如何设计存储方案

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 存储架构设计总的思路

一手微信: jkdy322 价格更优惠  
有正版课找我 高价回收帮回血

# 存储架构设计三个步骤

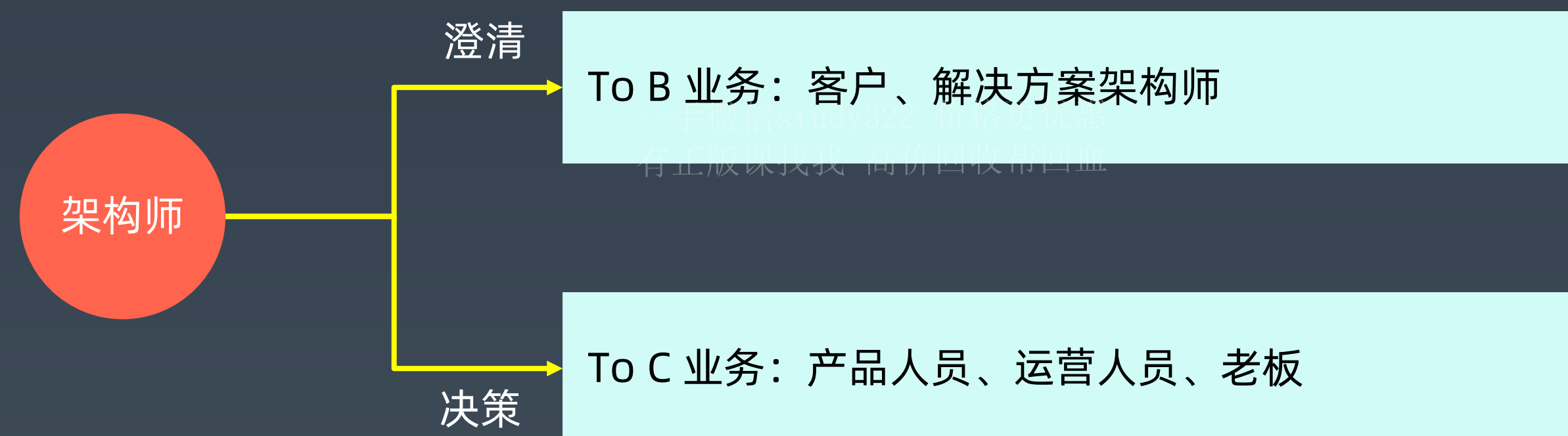
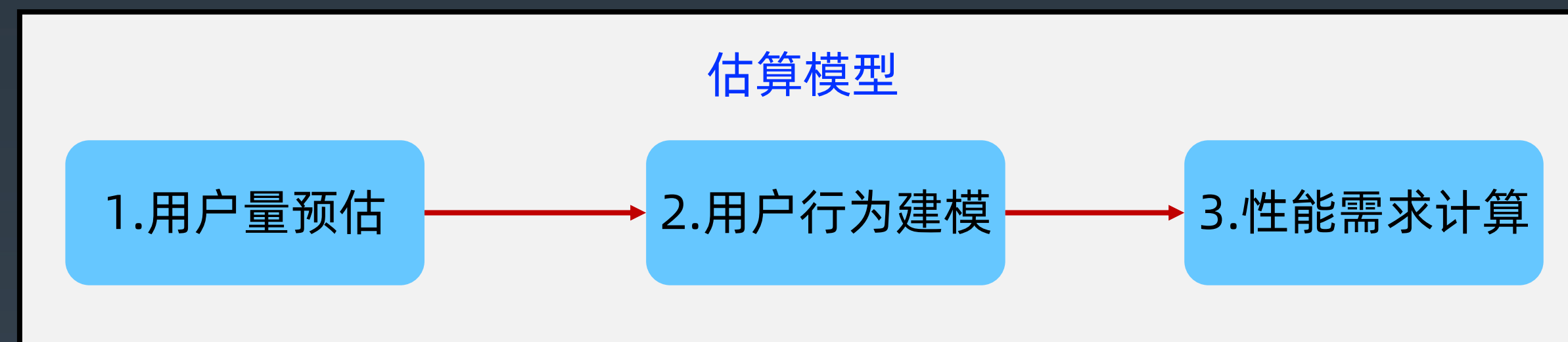


PPT 架构师主要是哪个步骤没做？

# 如何估算业务所需存储性能

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 性能估算步骤



为什么 To B 是“澄清”，To C 是“决策”？

# 用户量预估

## 规划

根据成本、预算、目标等确定

案例：

1. 某个新业务预算投入2000万拉新
2. 年底某业务用户规模达到100万

## 推算

基于已有数据推算

案例：

1. 做1个面向广州在校大学生的购物小程序
2. 香港地铁扫码乘车业务

## 对比

跟已有标杆进行对比

案例：

1. 跟竞争对手比
2. 跟自己已有的同类业务比



注意：用户不一定是人，可以是设备（IoT平台），可以是公司（云平台）



# 用户行为建模

行为

用户的典型行为

数量

采取某种行为的用户数量

频率

用户某种行为的频率

案例：

1. 预计每个月使用钱包付款码的用户有100万，付款笔数达到500万笔
2. 每天使用扫码乘车的用户有500万，平均扫码次数4.6次

加微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 存储性能需求计算

数据量

需要存储的数据总量(G)

请求量

对数据的读写请求量(TPS/QPS)

预留量

预留的增长空间

说明和技巧:

1. 并不是所有数据都一定要用同样的存储方式, 例如当前数据和历史数据可以分开存储
2. TPS/QPS 需要计算出以秒为单位的数值, 并且计算“平均值”和“峰值”
3. 预留增长空间不能太大也不能太小, 如果能做到线性伸缩是最好的



你所在的业务, 峰值和平均值的相差大概多少?

# 存储性能需求计算案例

## 【案例】

用户行为模型：每天使用扫码乘车的用户有500万，平均扫码次数4.6次

## 【部分分析和计算过程示例】

1. 假设总用户数1000万，则用户数据存储量是1000万
2. 每次扫码乘车，都需要访问一次用户数据，则用户数据读取次数：每天500万 \* 4.6 = 2300 万
3. 每次扫码乘车，都会生成一条乘车记录，则单日乘车记录数：500 万 \* 4.6 = 2300万
4. 乘车记录要保存2年，则总数据量为 2300 万 \* 800  $\approx$  200亿
5. 每条乘车记录对应一条支付记录，单日支付记录数 2300万，总数据量为 200 亿
6. 地铁乘车60%集中在早晚高峰的2个小时内，因此乘车记录写入的峰值TPS平均大约为  $2300\text{万} * 60\% / (2 * 3600) \approx 2000$

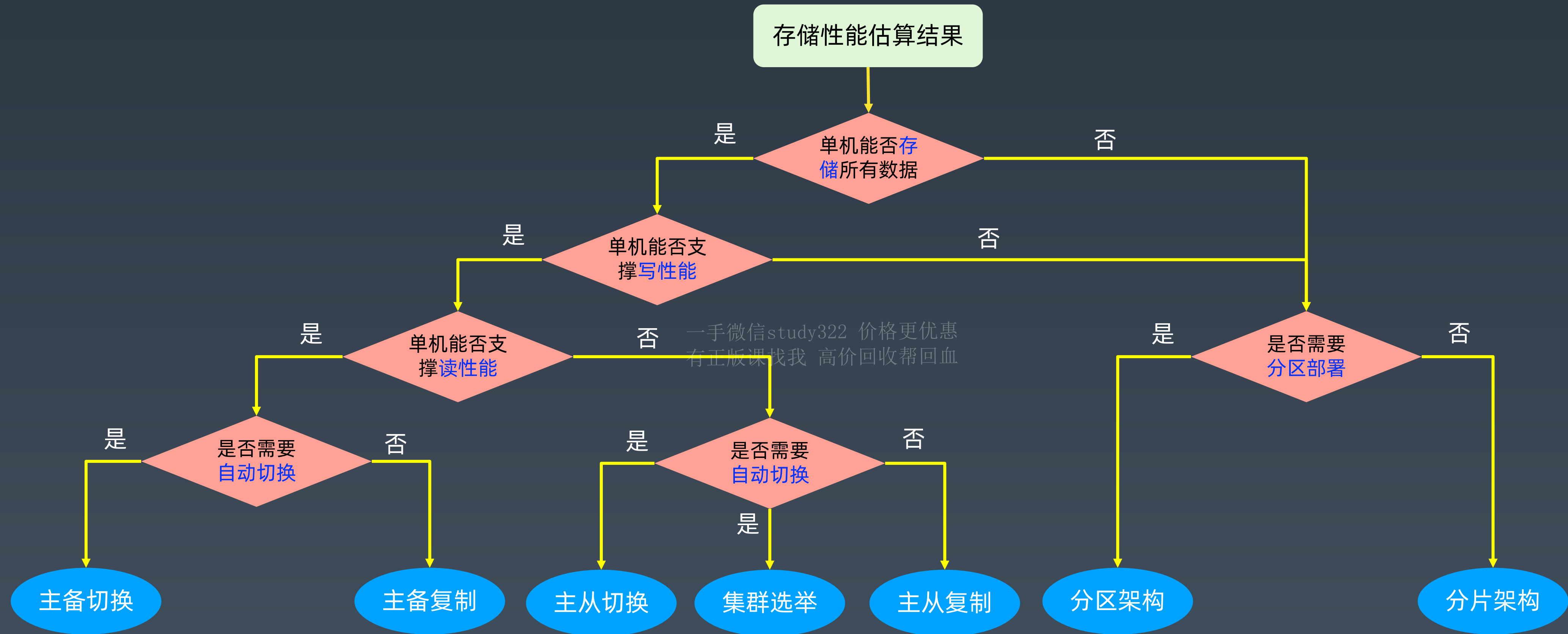


如果这个是香港地铁，你准备预留多少？

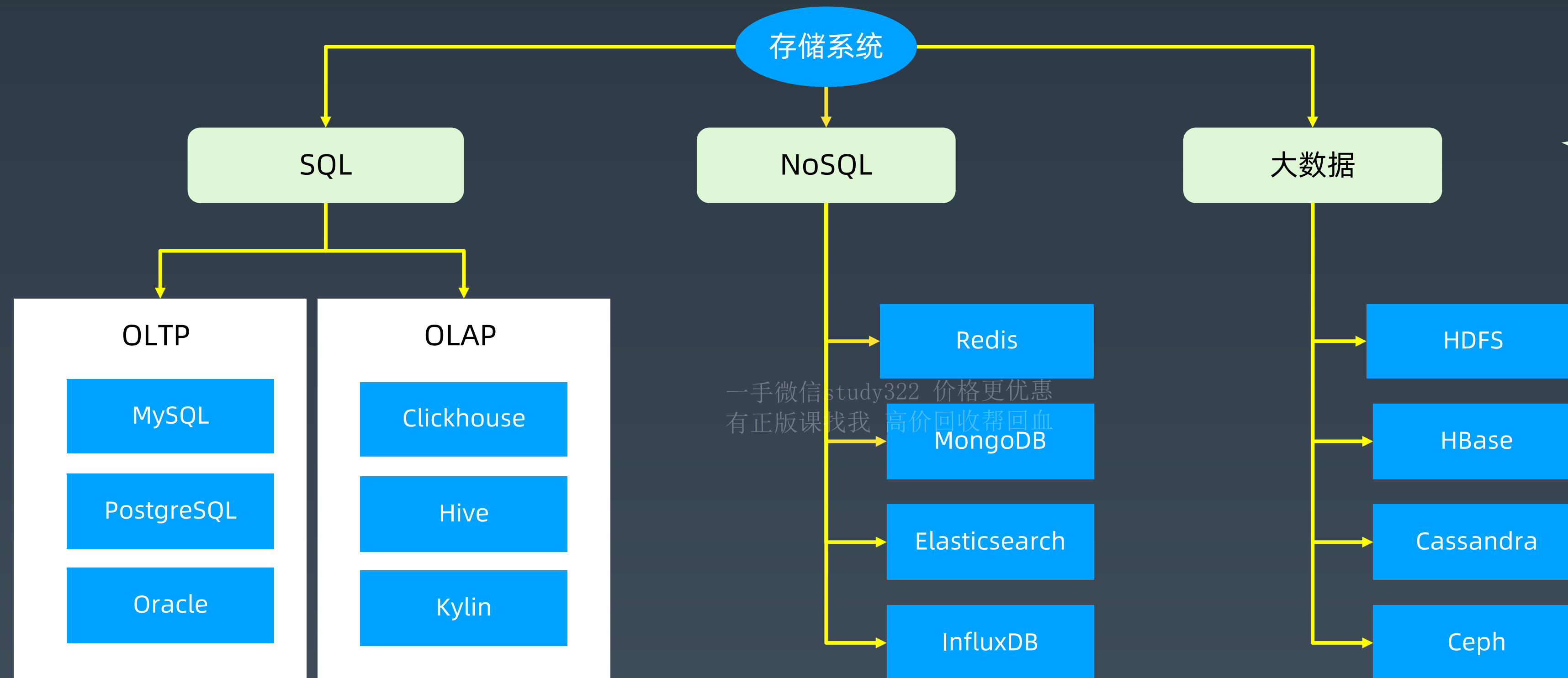
# 如何选择存储架构

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 存储架构选择逻辑



# 常见存储系统分类



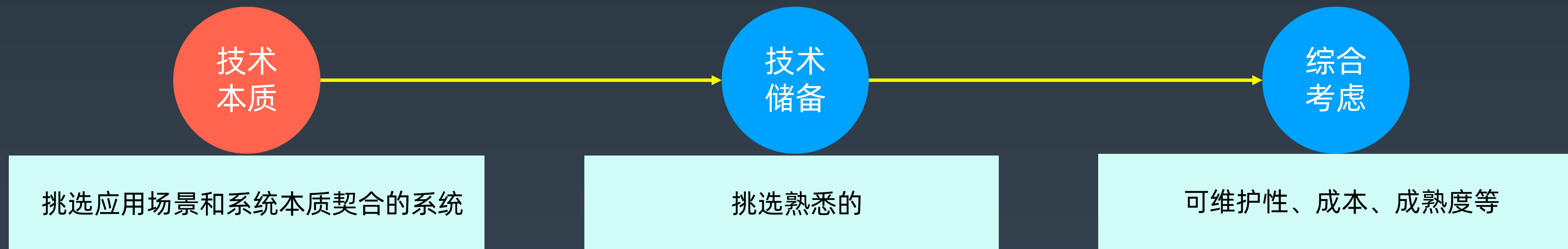
太多了，学不完怎么办？

更多请参考: <https://db-engines.com/en/ranking/relational+dbms>



NoSQL = Not Only SQL, 而不是 No SQL!

# 如何选择合适的存储系统



## 什么是技术本质?

系统的 DNA，有别于其它系统的典型特征。例如：MongoDB 是文档数据库，MySQL 是关系数据库，Redis 是 Remote dictionary Server，Elasticsearch 是倒排索引搜索引擎，HBase 是 “sparse, distributed, persistent multidimensional sorted map” .....

## 技术本质有什么影响?

技术本质决定了其核心应用场景和优缺点，例如 MongoDB 是文档数据库，优点是 Schemaless，缺点是事务支持不好；Elasticsearch 是搜索引擎而不是存储引擎（虽然可以做存储）

举例：

1. 游戏服务器用什么存储玩家数据比较好？
2. 论坛服务器用什么存储帖子数据比较好？



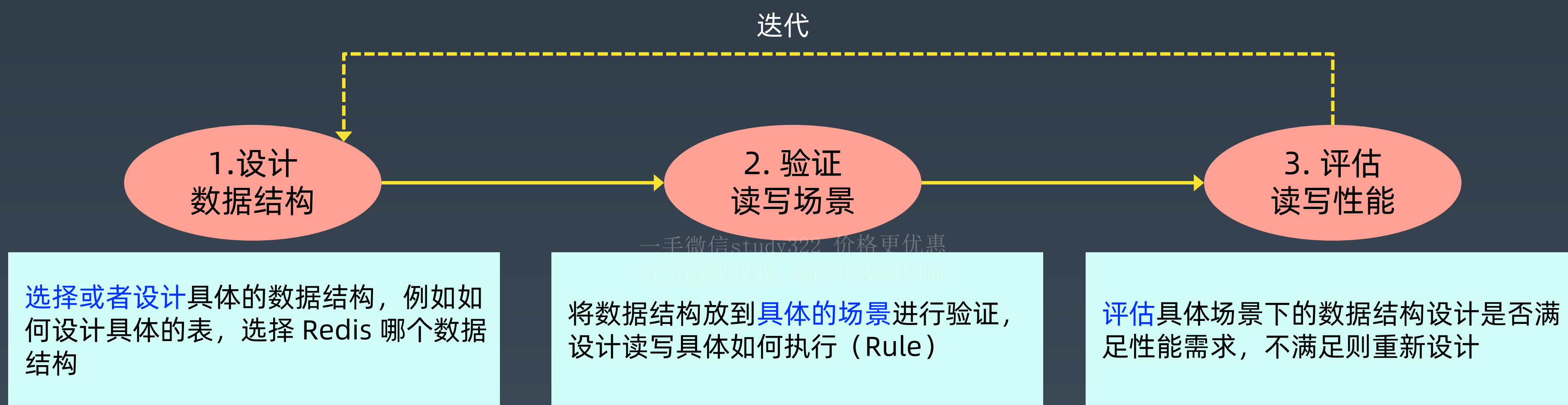
为什么要先理解技术本质，后掌握技术细节？

# 如何设计存储方案

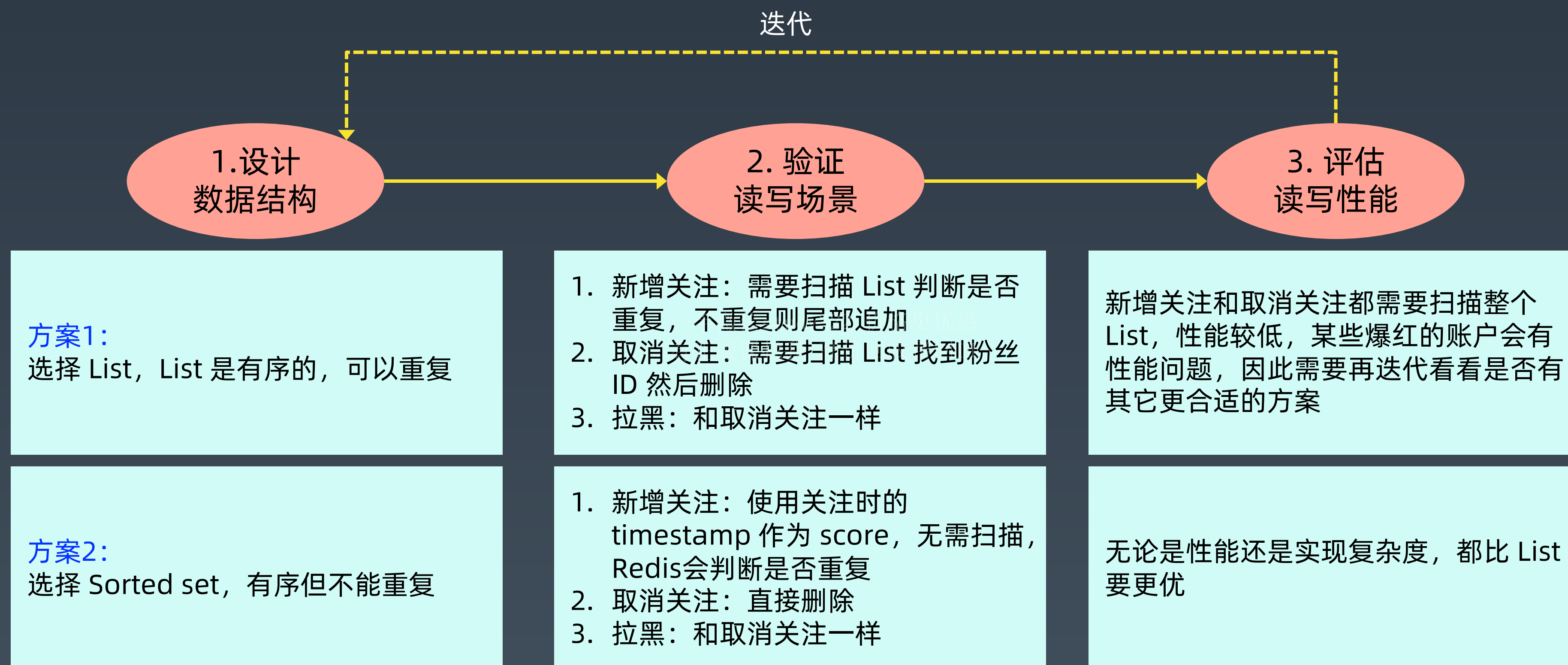
一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血



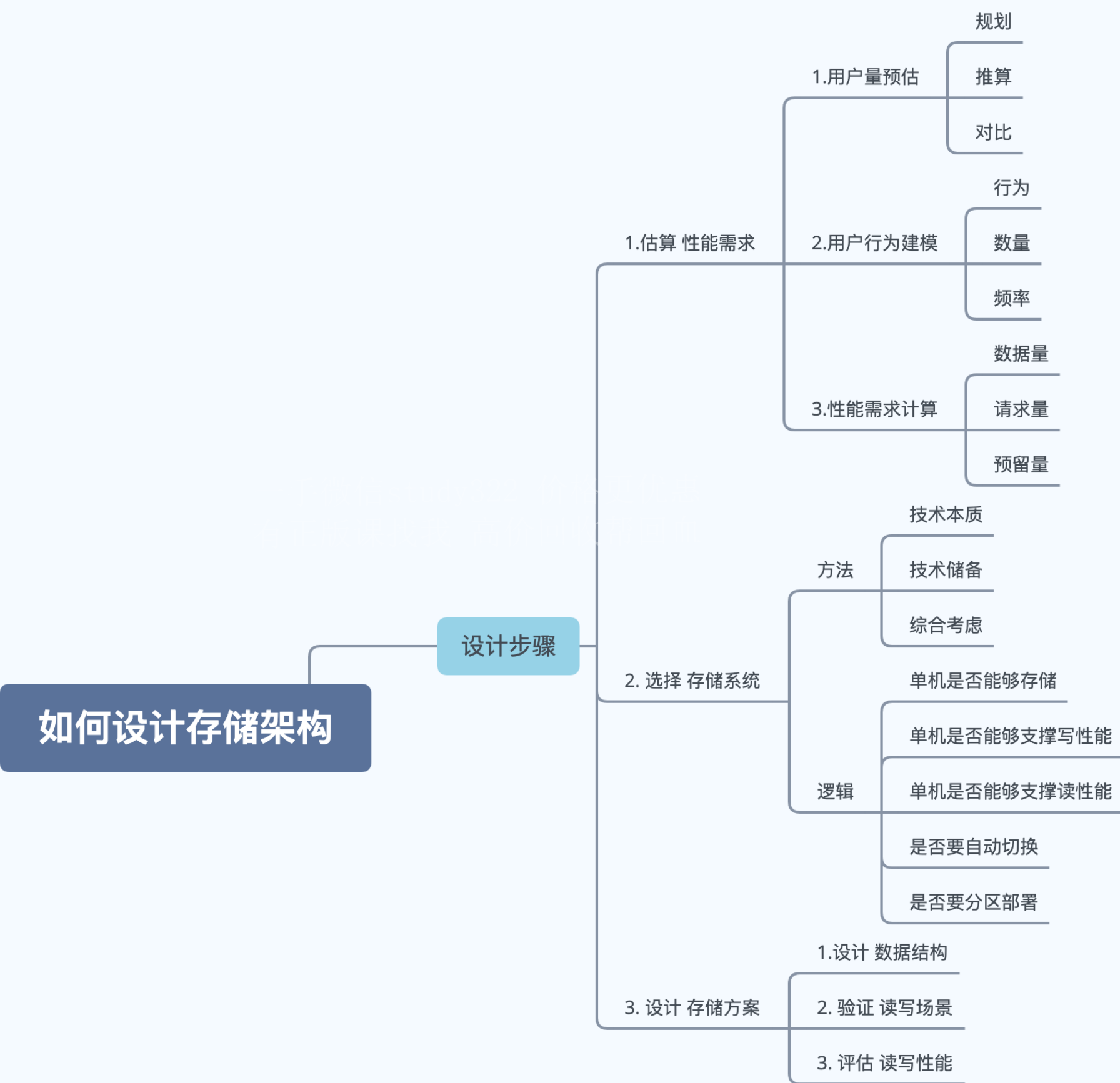
# 存储方案设计三个步骤



# 存储方案设计案例 - Redis 存储粉丝列表



# 本节思维导图



# 随堂测验

## 【判断题】

1. 估算性能需求的时候，通过估算模型就可以比较准确的估算出性能需求
2. 性能估算的时候，预留空间尽量大一些，可以应对业务突发增长
3. 选择存储系统的时候，优先选择功能强大
4. 存储系统太多了，没办法全部学习，可以优先学习和工作相关的成熟系统
5. 存储方案设计的时候，如果某个方案不行，就应该换一个存储系统试试

## 【思考题】

为什么存储系统如此多？

# Q&A





# 茶歇时间



八卦，趣闻，内幕.....

THANKS

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血