

## 架构实战营模块6

# 第5课：中台深入剖析和实现技巧

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. 全面理解中台概念
2. 掌握中台落地的问题
3. 掌握中台落地的技巧

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血



橘生淮南则为橘，生于淮北则为枳！

# 目录

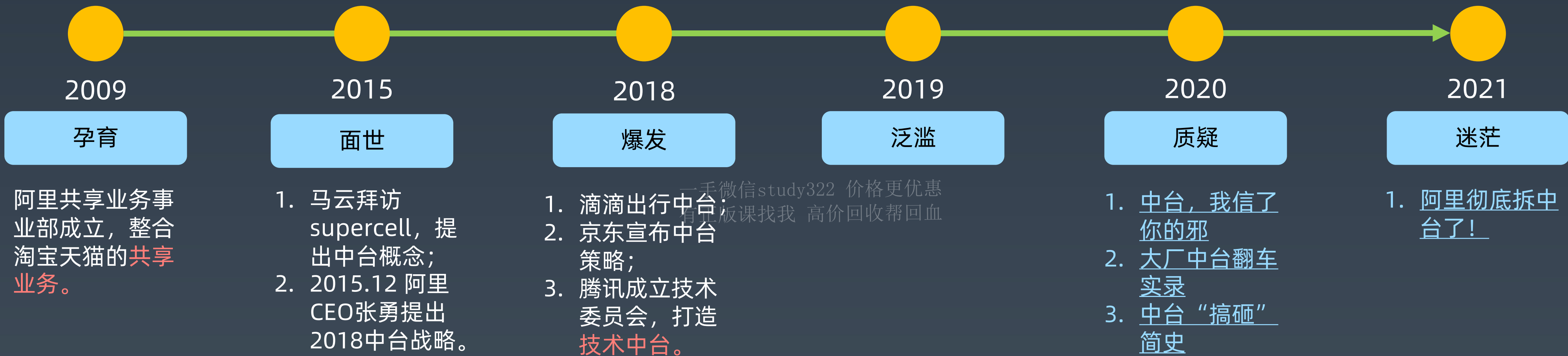
1. 什么是中台
2. 中台带来的问题
3. 中台系统落地技巧

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

# 1. 什么是中台

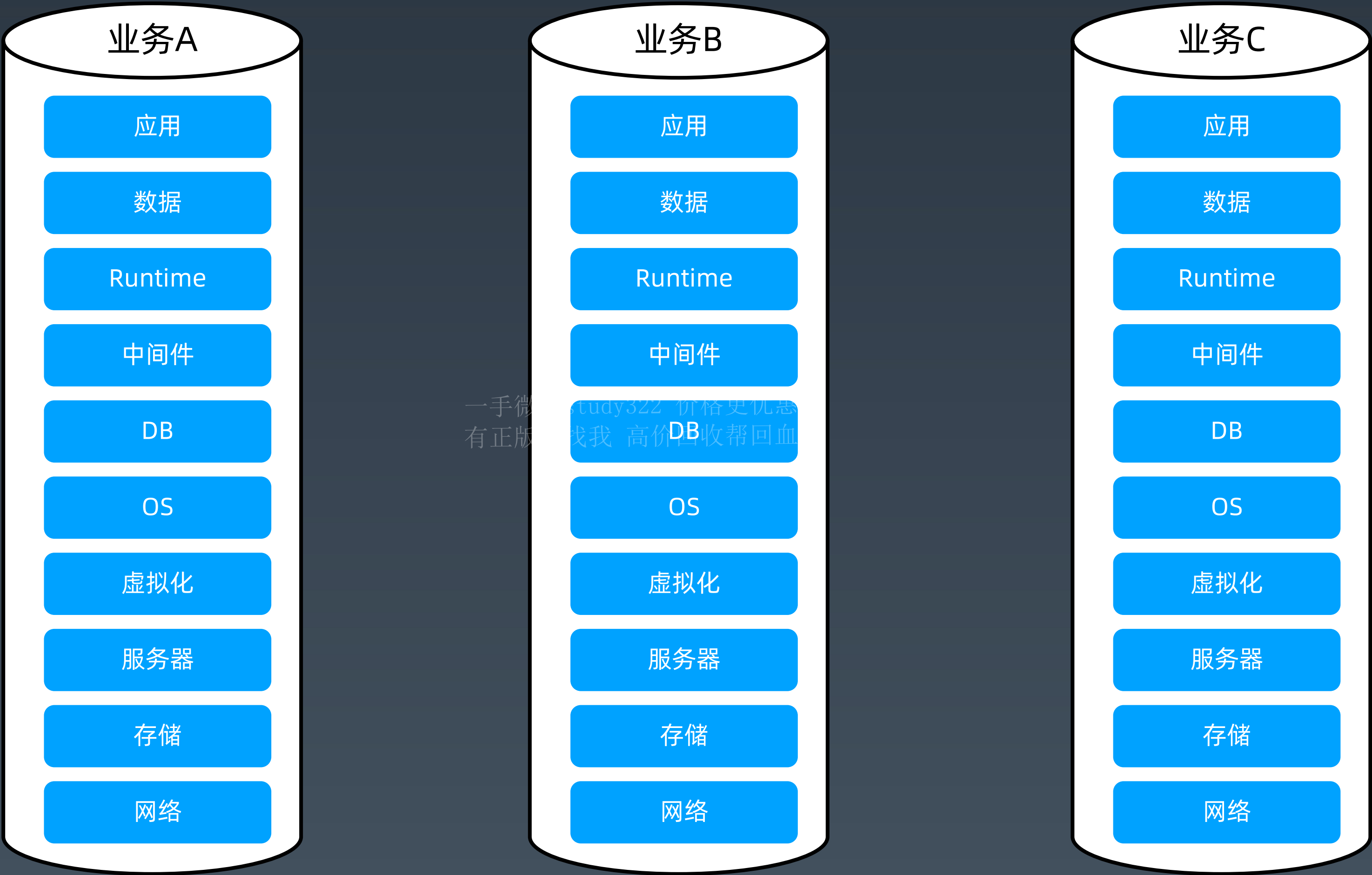
一手微信study321 价格更优惠  
有正版课找我 高价回收帮回血

# 中台发展史

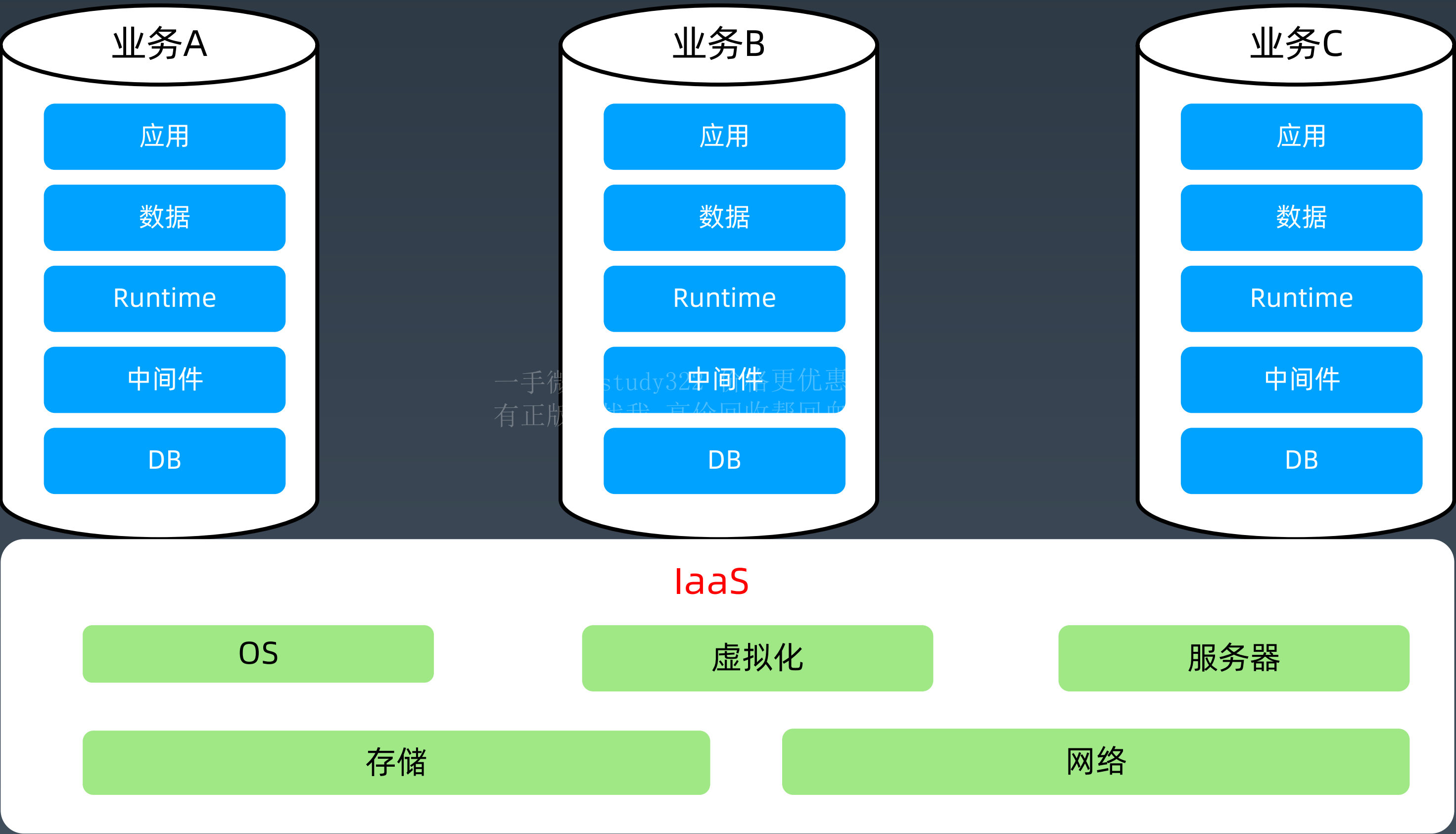


对照面向复杂度架构设计理论，中台是用来应对什么复杂度的？

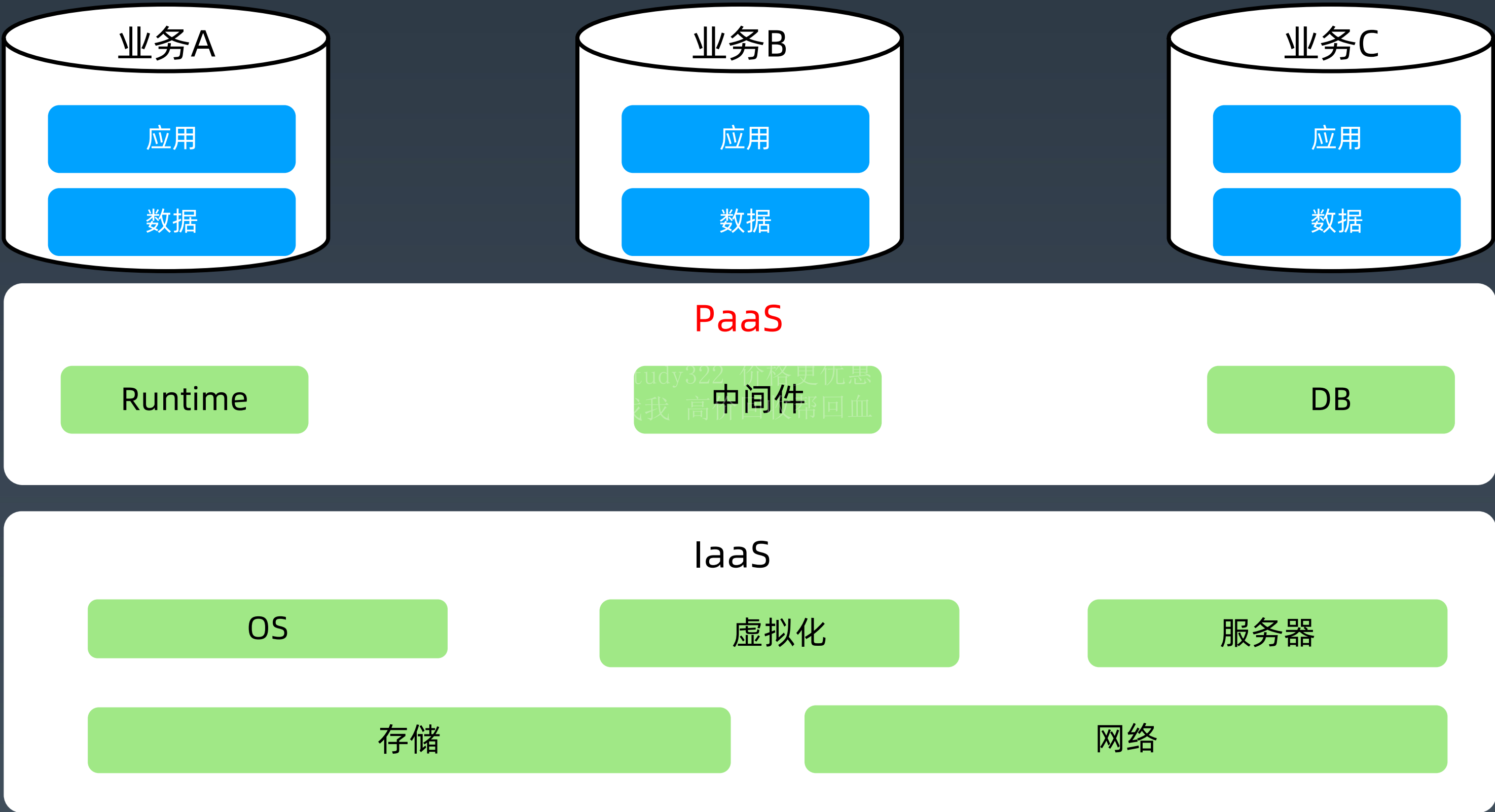
# 大烟囱架构



# laaS 架构



# PaaS 架构

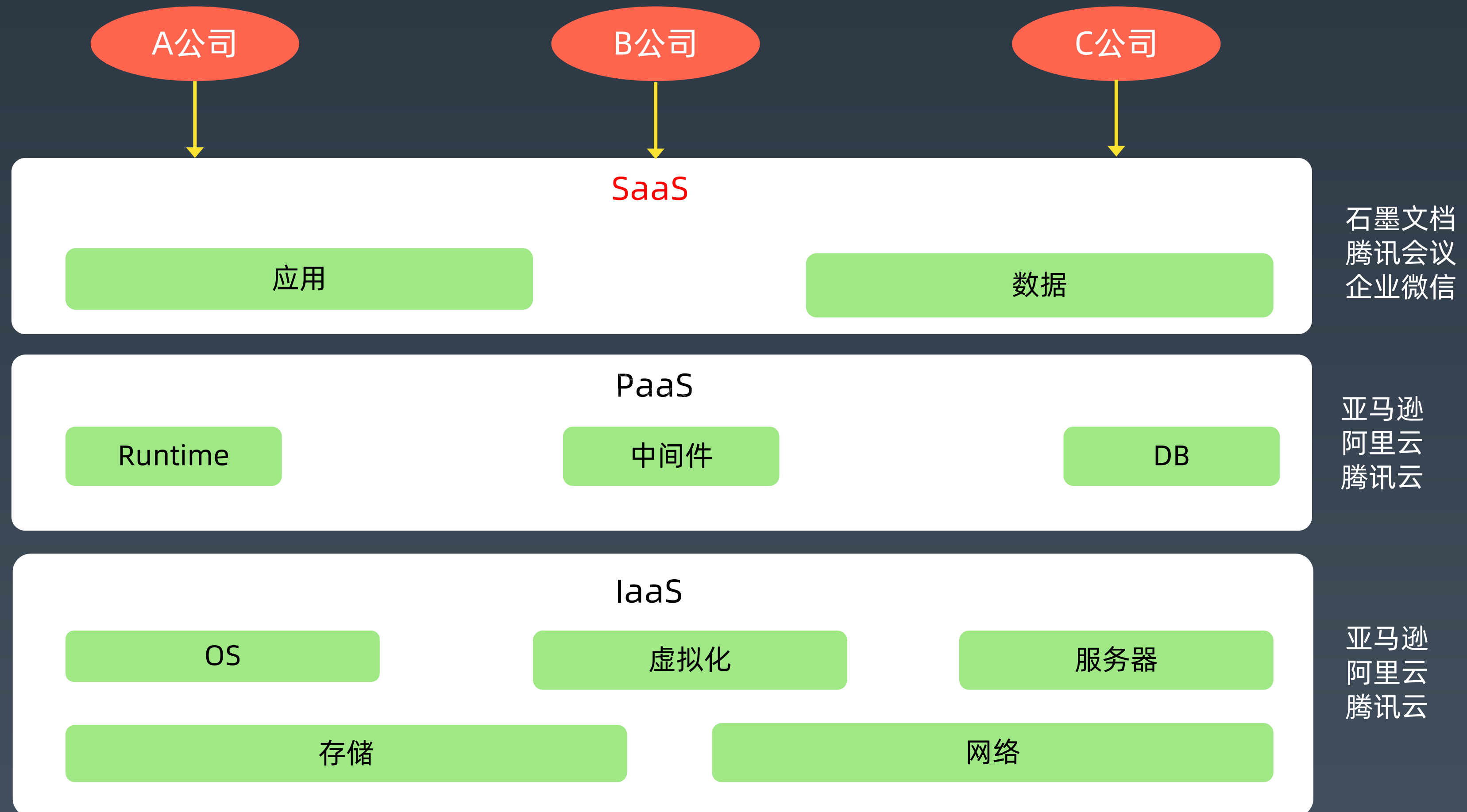


亚马逊  
阿里云  
腾讯云

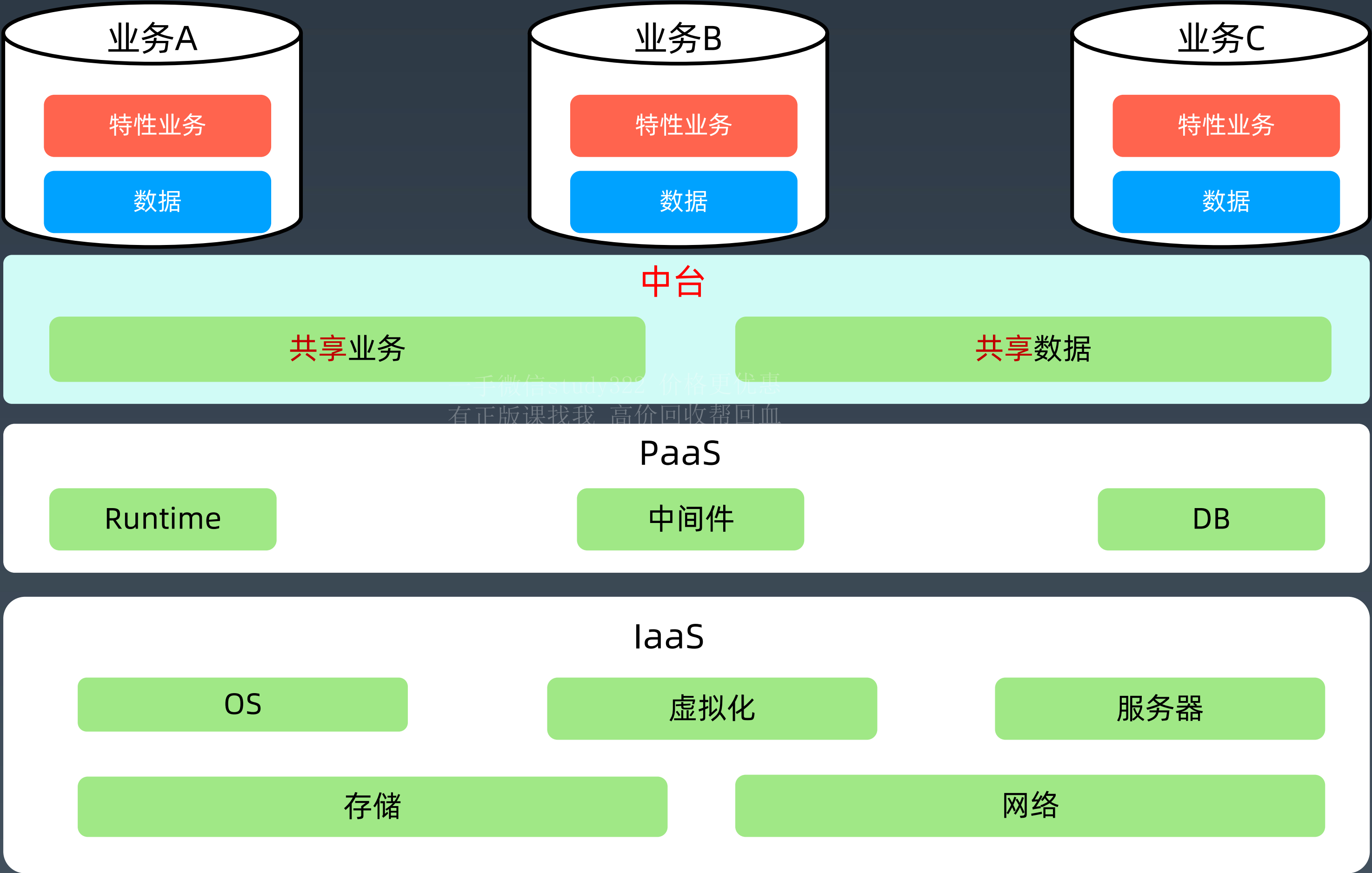
亚马逊  
阿里云  
腾讯云



# SaaS 架构



# 中台架构



# 中台定义

中台就是“企业级的能力复用平台”。  
—— Thoughtworks 首席咨询师王健

中台是将系统的通用化能力进行打包整合，通过接口的形式赋能到外部系统，从而达到快速支持业务发展的目的。  
—— [百度百科](#)

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

中台架构，是将企业的核心能力随着业务不断发展以数字化形式沉淀到平台，形成以服务为中心，由业务中台和数据中台构建起数据闭环运转的运营体系，供企业更高效的进行**业务探索和创新**，实现以数字化资产的形态构建企业核心差异化竞争力。  
—— 阿里官方定义



你觉得哪种定义合理？

# 理解中台概念 - 业务中台

业务  
相关

业务中台是企业内部业务相关的能力共享，IaaS、PaaS、SaaS都不是中台。

跨业务

业务中台肯定是跨业务的，单个业务不需要中台这个概念。

相似  
业务

相似的业务才可以构建在同一中台上，差异太大的业务，中台没有意义。

一手微信study322 价格更优惠  
有正课跟我 高价回收帮回血

业务中台，是将企业内多个相似业务的通用业务能力沉淀到平台，以减少重复建设，提升业务开发效率的一种架构模式。

—— 架构实战营



这里的中台定义与阿里的中台定义核心区别是什么？

# 理解中台概念 - 数据中台

所有  
业务

数据中台应该是支持**所有**业务的。

数据  
打通

业务间的数据需要打通。

例如通过“统一用户 ID”来关联同一用户在多个业务上的数据。

数据  
复用

不同业务间的数据可以复用，提升整体的运营效率。

例如：  
美团可能根据你看电影的数据来向你推荐外卖的商品。

数据中台，是将企业所有业务的数据沉淀到同一平台，支持业务间数据打通以及数据复用，**提升企业运营效率**的一种架构模式。

——架构实战营



数据中台最难的是哪部分？

# 中台的价值



相似业务的能力共享，避免大量重复开发，提升开发效率。

- 1. 业务相似度越高，业务中台价值越大，建议相似度达到60%以上的多个业务共建中台，例如“快车+专车”，“淘宝+天猫+咸鱼”，“火山+抖音+西瓜”。
- 2. 评估业务相似度，需要依赖业务专家，而不是一个单纯的技术工作。
- 3. 强行将相似度低的业务塞进一个中台，不但不会提升开发效率，还会大大降低效率（参考第6课的实战案例）。



数据打通和复用，避免数据孤岛，提升运营效率。

- 1. 使用数据中台的业务越多，数据中台价值越大。
- 2. 数据中台的价值体现在：统一数据平台、跨业务的数据打通、跨业务的数据复用（挖掘）。
- 3. 跨业务的数据复用：理想很丰满，现实比较骨感，受限于业务熟悉度和组织结构的相关约束。

# 中台的一些说法澄清



业务中台、数据中台、技术中台、电商中台.....哪个是假冒的中台？

数据中台和数据平台有什么区别？搭建一套完整的 Spark+Flink+Hadoop 等技术栈的大数据处理系统，属于数据中台建设么？

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

为什么 SaaS 不是中台？

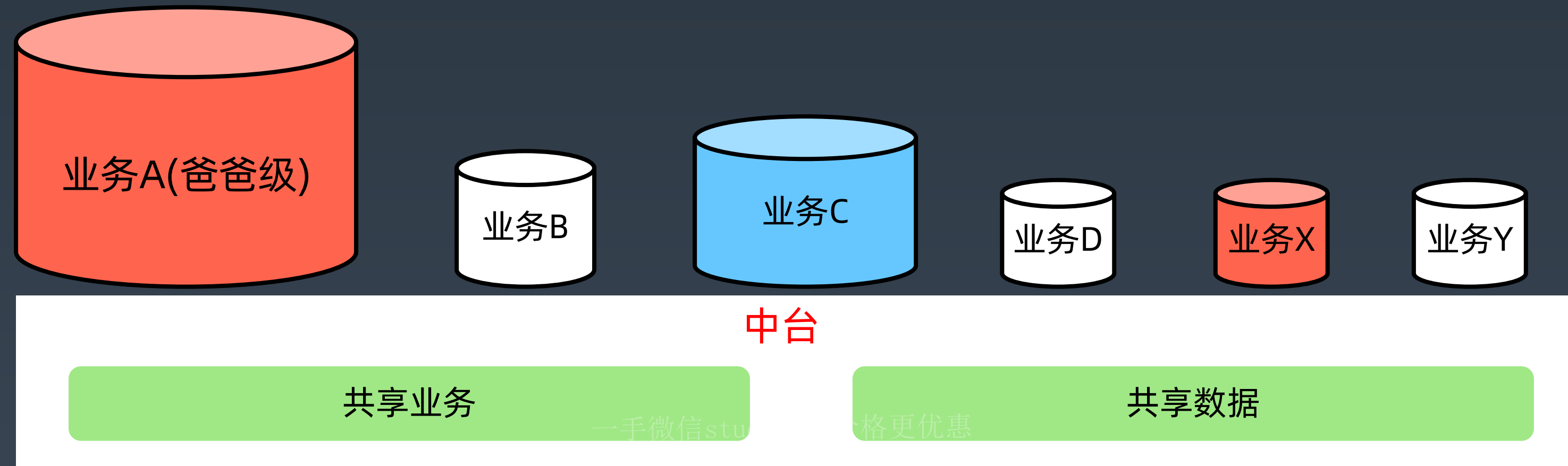
老板说要建一套公司级的中台，支撑电商、物流、ERP、CRM 等各种业务。

## 2. 中台带来的问题

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血



# 小业务抱中台大腿，中台抱大业务大腿



## 【现象】

1. 业务A：爸爸级别的业务，中台 KPI 关键；
2. 业务C：保3.5争3.75的重要抓手；
3. 业务X：有 P11 大佬站台的业务，汇报重点；
4. 其它：爱理不理，没事做就支持一下的业务。

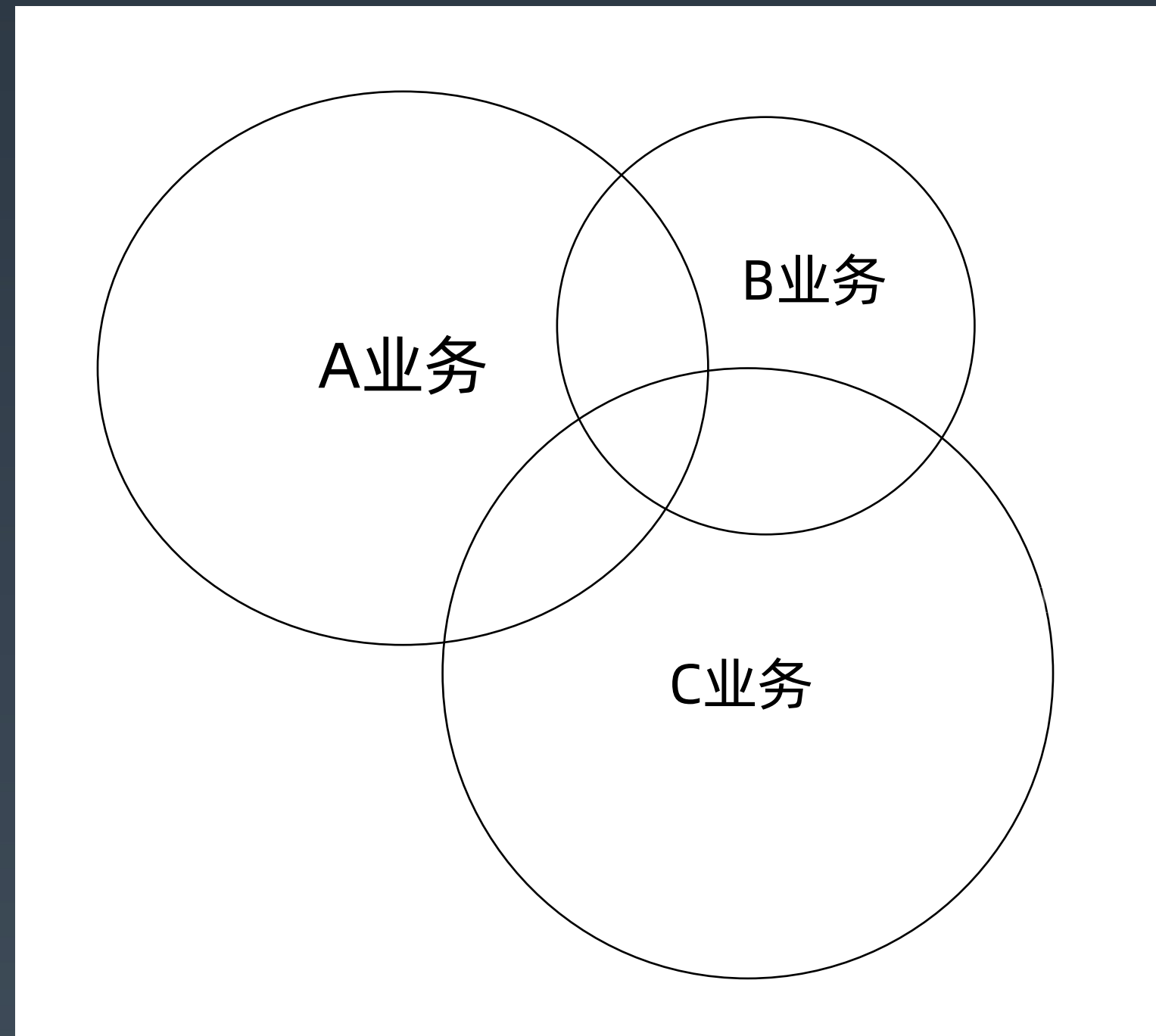
## 【应对方法】

目前没有看到很好的应对方法，中台建设最后就是一个组织结构问题（康威定律）



即使业务Y是创新业务，如果没有大佬站台，中台也不太会大力支持，原因是什么？

# 中台与业务的边界难以明确



## 【现象】

1. 没有任何明确的规则，都是靠人肉讨论。
2. 在已有业务基础上比较容易提炼，创新业务几乎无法判断。
3. 创新业务对中台 KPI 没有帮助，大部分情况下都会被拒掉，由业务自己实现。
4. 中台适合做“**组合式创新**”，没法做“**颠覆式创新**”。

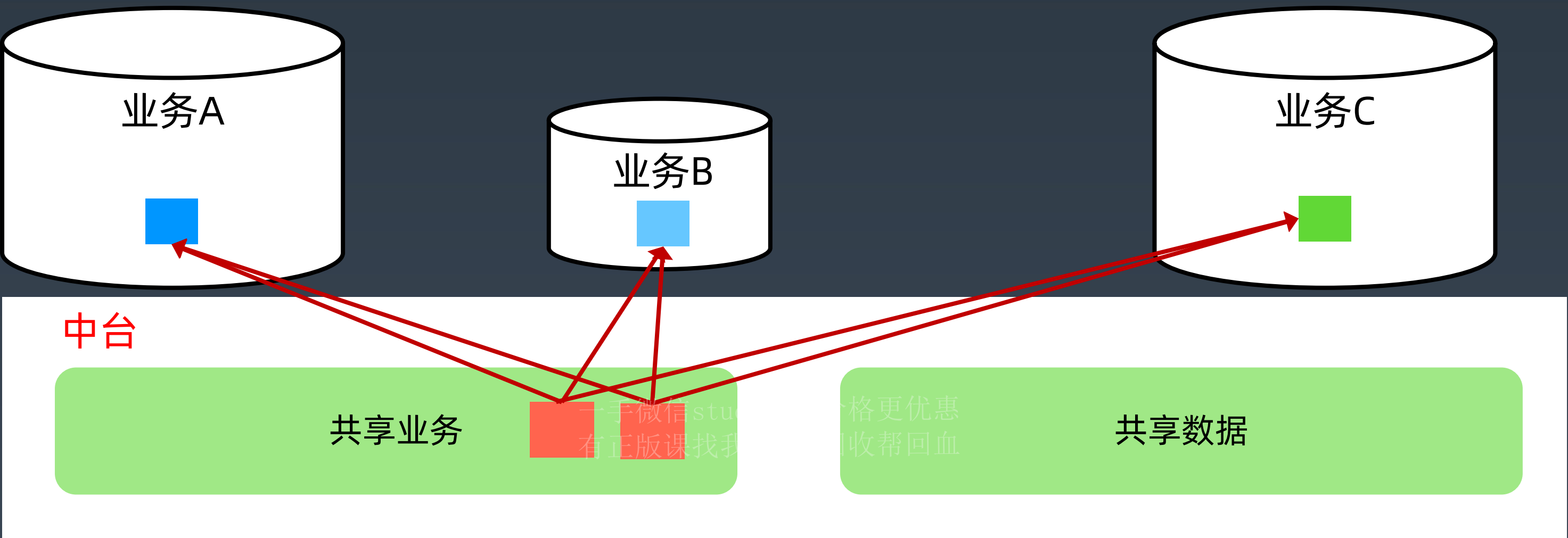
## 【应对方法】

业务上和组织上目前没有很好的解决方法，技术上可以采用本课后面的技巧来缓解问题。



中台设计最难的不是领域划分，而是中台和业务的边界划分！

# 中台的全流程效率并不高



## 【现象】

1. 每个业务功能都要讨论边界；
2. 每个业务功能都要考虑对所有业务的影响。

## 【应对方式】

业务上没有什么解决方法，技术上可以通过本课后面讲的技巧来缓解问题。

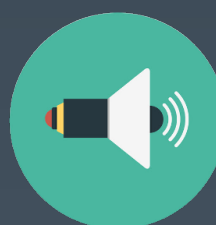
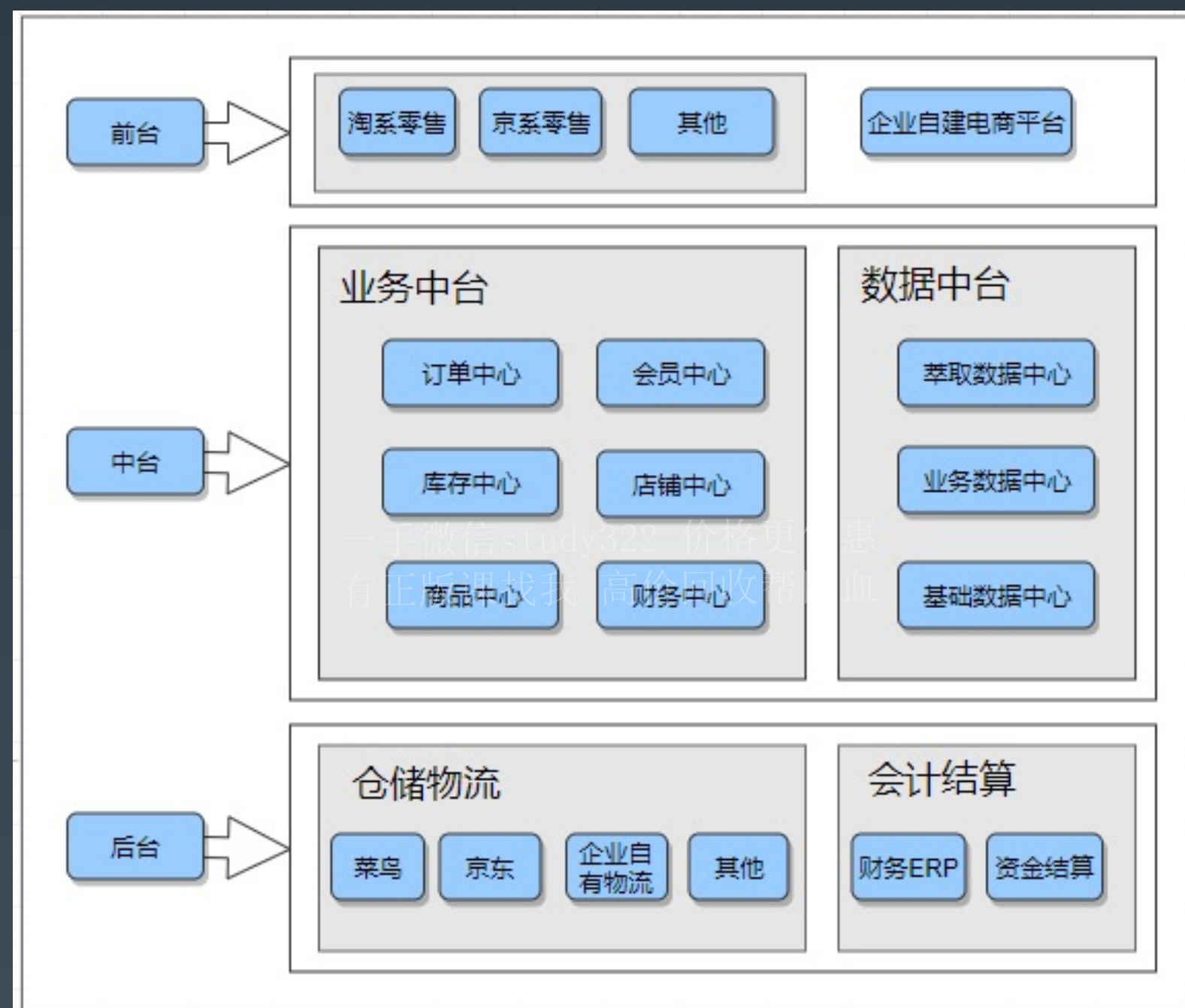


小需求30人讨论，中需求50人讨论，大需求100人讨论！

# 3. 中台落地技巧

一手微信study822 价格更优惠  
有正版课找我 高价回收帮回血

# 微服务搭建业务中台



微服务不一定是中台，中台一定是微服务！

# 用 Pipeline 封装不同业务流程

香港钱包用户



香港钱包扫码 pipeline

handler-解码

handler-校验

handler-路由

澳门钱包用户



澳门钱包扫码 pipeline

handler-解码

handler-校验

handler-码补充

handler-路由

扫码微服务

# Pipeline 代码示例

arch-training ~/code/arch-training

src

main

java

com

demo

controller

Request

Scanner

core

Handler

Checker

Decoder

Message

Pipeline

PipelineFactory

Router

hk

HKChecker

HKRouter

mo

MOAppender

MOChecker

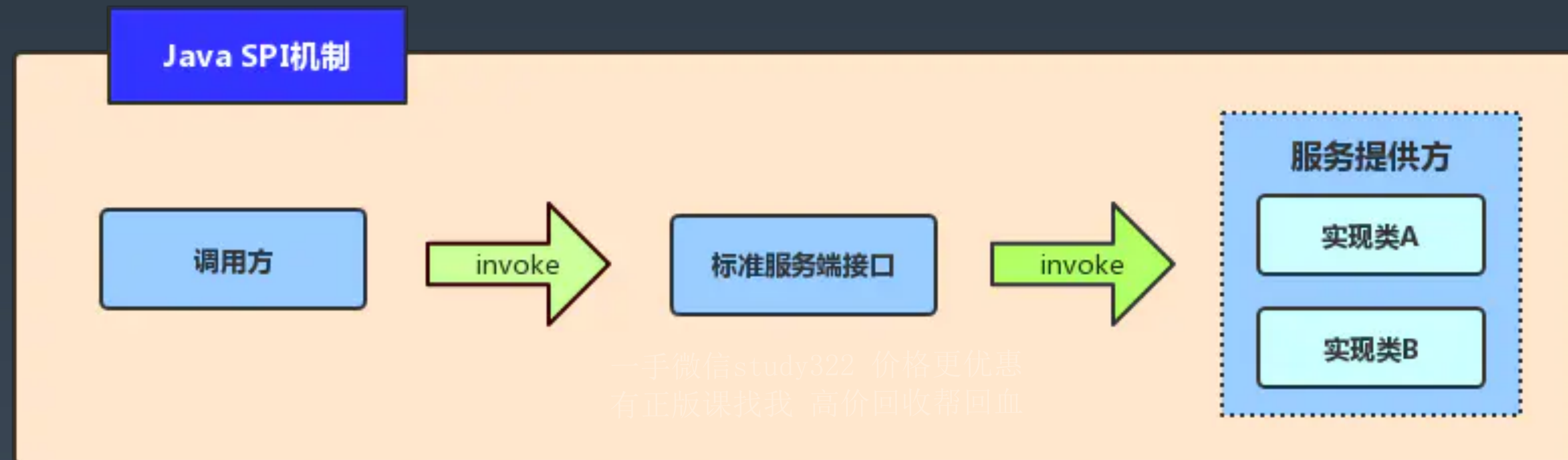
MORouter

resources

```
1 package com.demo.controller;
2
3 import com.demo.core.Message;
4 import com.demo.core.Pipeline;
5 import com.demo.core.PipelineFactory;
6
7 public class Scanner {
8     @ public void scan(Request request){
9         String bizType = request.getBizType();
10        Pipeline pipeline = PipelineFactory.getPipeline(bizType);
11
12        Message message = Message.init(request);
13        pipeline.process(message);
14    }
15 }
16
```

项目地址: <https://github.com/yunhua-lee/arch-training>

# 用 SPI 封装不同业务

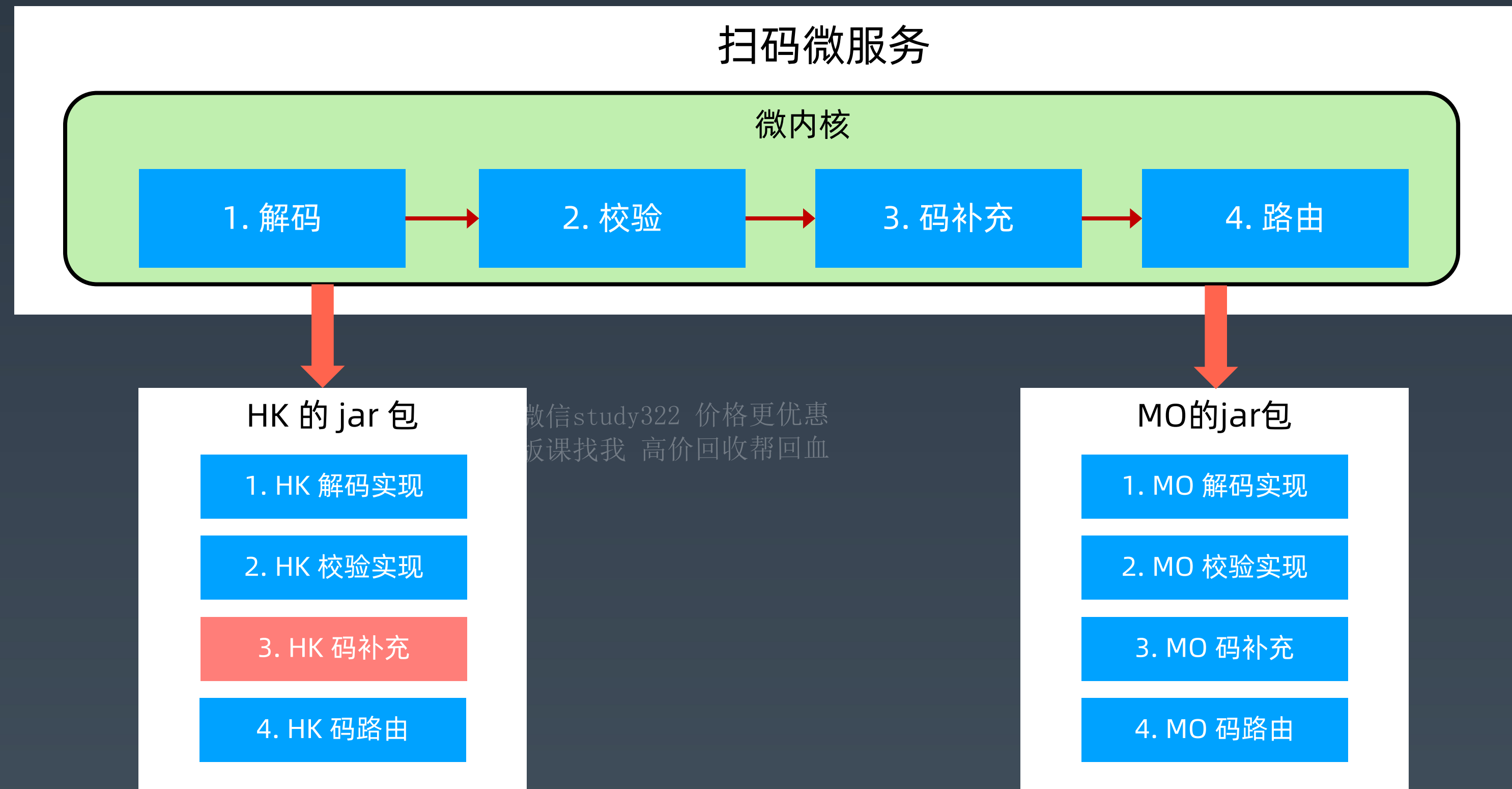


SPI 全称 [Service Provider Interface](#)，是 Java 提供的一套用来被第三方实现或者扩展的接口，它可以用来启用框架扩展和替换组件。

SPI 的作用就是为这些被扩展的 API 寻找服务实现。



# 用 SPI 封装案例



为什么 HK 没有码补充的功能也要实现码补充的 SPI 接口?

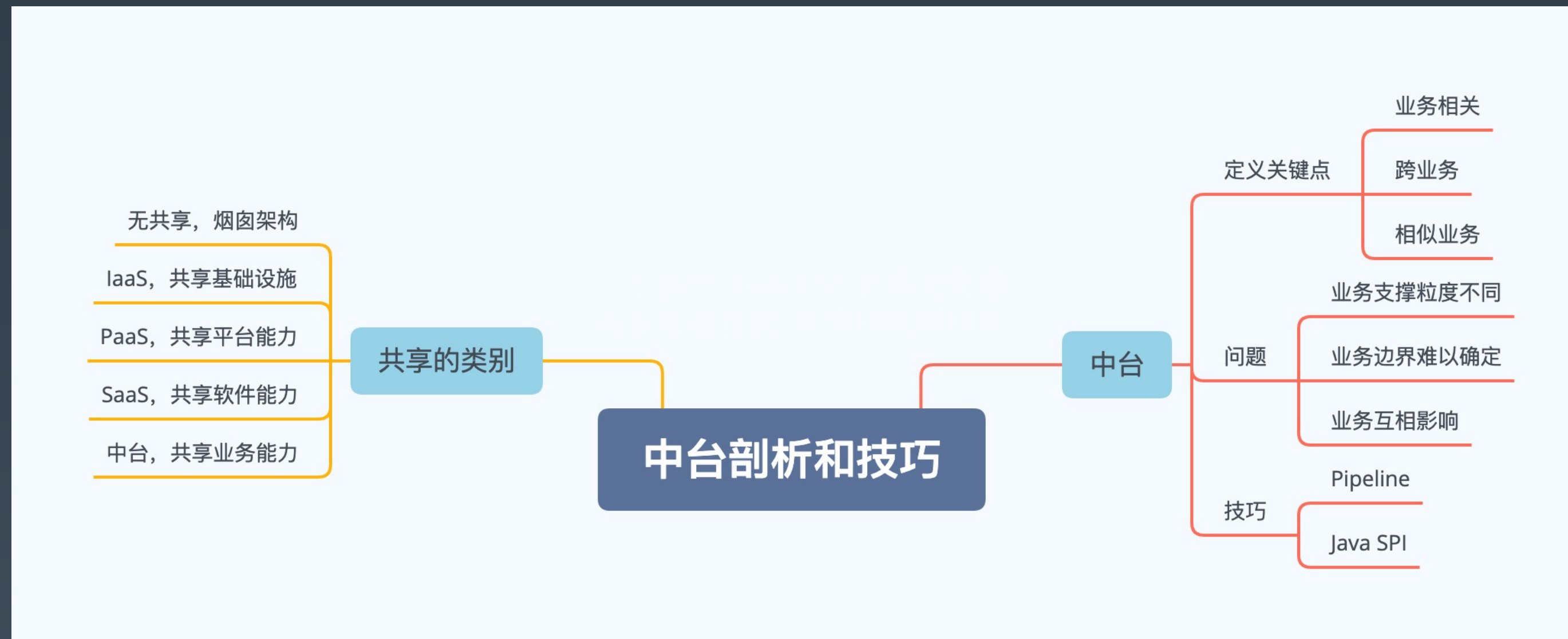
# Pipeline 和 SPI 方案对比

	Pipeline	SPI	对比
开发模式	中台团队负责框架和实际的业务代码实现。	中台团队负责框架，业务团队负责业务代码实现。	SPI 看起来好些。
开发难度	中台团队全部搞定，开发难度低。	业务团队需要熟悉中台团队的设计和原理，并且需要明确边界，开发难度高。	Pipeline 好一些。
部署方式	统一部署。	业务代码更新只需要发布 jar 包。	SPI 更好一些。
业务隔离	Pipeline 做业务隔离，代码级别隔离。	微内核+插件做业务隔离，插件级别隔离。	SPI 好一些。



如果是你，你会选择哪种模式？

# 本节思维导图



# 随堂测验

## 【判断题】

1. 公司级的能力复用都可以称为中台，例如技术中台。
2. 中台一定是跨多个业务的，所以应该建一套公司级的业务中台来支撑所有业务。
3. 中台能够减少重复开发，因此效率肯定比不采用中台要高。
4. 中台落地最难的是很难清晰地界定业务和中台的边界。
5. SPI 模式优点更多，应该选择 SPI 模式。

一手微信study322 价格更优惠  
有正版课找我 高价回收帮回血

## 【思考题】

中台里面是业务共享，数据互通，那数据互通的目的和作用是什么？

# 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