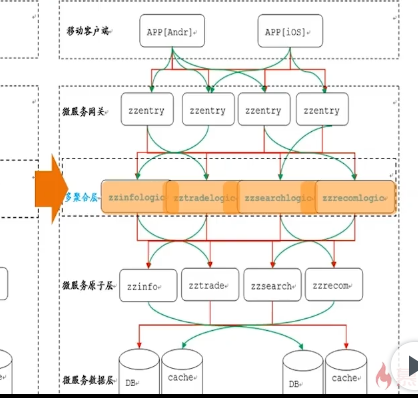
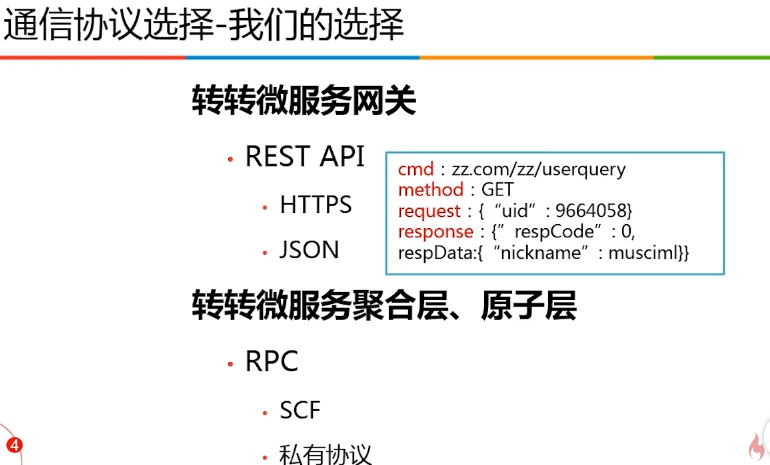


将之前的单个模块的聚合层进行扩展到多个模块和原子层类似，一一对应

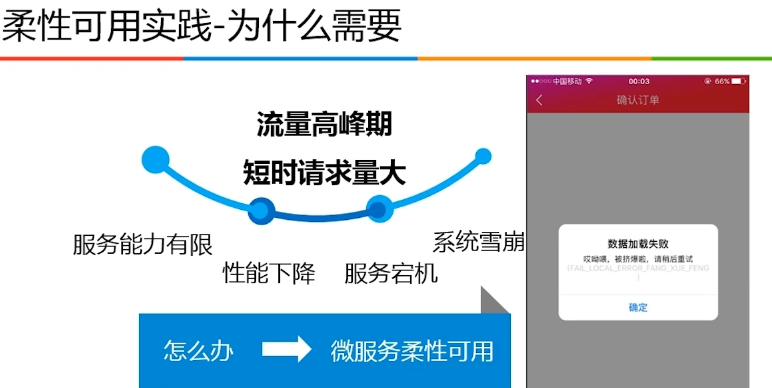




通信协议选择：、



存在的问题以及解决方案：



柔性可用、服务降级



系统降级：拒绝部分请求，关闭部分服务（一般是**拒绝部分请求**）

以下为三种拒绝策略：



数据层降级：

更新请求:当高峰期来时，将数据更新的消息放到消息队列中

并且更新数据直接更新缓存、不进数据库

读操作：直接读缓存，读不到的数据不再去数据库中查询

数据补齐：当高峰期过去后，恢复之前的操作，不再降级，将消息队列、缓存中的更新数据消息一点一点的同步到数据库中



最后的服务治理

就是要对整个微服务系统进行监控：

进程监控：（可以采用Alive监控）作用为监控整个系统是否存活，不存活要及时唤醒重启

语义监控：有时候系统存活状态但是有可能不能正常请求，可以时不时的模拟用户请求，看

返回的结果是否是期望值；

异常监控：对系统的异常信息进行监控，通常每一段时间的错误量是平均的，有时候错

误量会飙升，那说明当前的系统就是有问题了，就需要进行报警处理

数据波动监控：对一段时间的流量进行监控，如果流量骤增或骤减那说明需要引起高度重视

了

数据资源监控：对所有的机器资源进行监控，比如硬盘容量、内存等等

可以用开源的鹰眼（其实这应该就是我之前学的链路追踪系统）服务治理系统，这种系统需要综合系统的每个点进行配置。



最后还有微服务的实时监控策略

请求平均耗时：对每天的请求量进行监控，并且可以将昨天和当天的各个模块的请求量

进行对比，可以看出来各个版本的请求耗时情况，或者说是时间段的

请求异常条数：跟请求监控类似，将每天的异常监控起来