参考：<https://segmentfault.com/q/1010000006095431?_ea=1030472>

[面试题：谈谈业务中使用分布式的场景](https://segmentfault.com/q/1010000006095431)

面试的时候，面试官直接问，说说项目中使用分布式的场景？感觉现在的面试，高并发和分布式是面试的标配啊，说实话，虽然项目的底层用了8台机器做集群，但是使用分布式场景的业务场景确实不多，回答的不是很好。  
问题：  
1.个人感觉单讲分布式的话，包含有分布式存储、分布式事务，分布式锁，分布式锁有memcache、redis、DB等实现方式。以后遇到这种问题，如何回答才能达到面试官满意的程度？？  
(个人感觉面试官是想知道：对于分布式这种问题的架构设计方面的理解以及在场景中如何使用)

============================================================

总结答案如下：  
一、解决java集群的session共享的解决方案： SSO单点登录  
1.客户端cookie加密。（一般用于内网中企业级的系统中，要求用户浏览器端的cookie不能禁用，禁用的话，该方案会失效）。  
2.集群中，各个应用服务器提供了session复制的功能，tomcat和jboss都实现了这样的功能。特点：性能随着服务器增加急剧下降，容易引起广播风暴；session数据需要序列化，影响性能。  
3.session的持久化，使用数据库来保存session。就算服务器宕机也没事儿，数据库中的session照样存在。特点：每次请求session都要读写数据库，会带来性能开销。使用内存数据库，会提高性能，但是宕机会丢失数据(像支付宝的宕机，有同城灾备、异地灾备)。  
4.使用共享存储来保存session。和数据库类似，就算宕机了也没有事儿。其实就是专门搞一台服务器，全部对session落地。特点：频繁的进行序列化和反序列化会影响性能。  
5.使用memcached来保存session。本质上是内存数据库的解决方案。特点：存入memcached的数据需要序列化，效率极低。

二、分布式事务的解决方案:  
1.TCC解决方案：try confirm cancel。

首先，需要了解系统为什么使用分布式。  
随着互联网的发展，传统单工程项目的很多性能瓶颈越发凸显，性能瓶颈可以有几个方面。  
一、应用服务层：随着用户量的增加，并发量增加，单项目难以承受如此大的并发请求导致的性能瓶颈。  
二、底层数据库层：随着业务的发展，数据库压力越来越大，导致的性能瓶颈。  
针对上面两点，我觉得可以从两方面解决。  
应用服务层：  
 应用服务层的解决方案有几种：  
 应用系统集群：  
 应用系统集群最简单的就是服务器集群，比如：tomcat集群。应用系统集群的时候，比较凸显的问题是session共享，session共享我们一是可以通过服务器插件来解决。另外一种也可以通过redis等中间件实现。  
 服务化拆分：  
 服务化拆分，是目前非常火热的一种方式。现在都在提微服务话。通过对传统项目进行服务化拆分，达到服务独立解耦，单服务又可以横向扩容。服务化拆分遇到的经典问题就是分布式事务问题。目前，比较常用的分布式事务解决方案有几种：消息最终一致性、TCC补偿型事务、尽最大能力通知。具体的你可以参考下这篇博客[分布式事务解决方案](http://www.roncoo.com/article/detail/124243)  
底层数据库层：  
 如果系统的性能压力出现在数据库，那我们就可以读写分离、分库分表等方案进行解决，由于这方面我的经验也不够多，所以，你可以参考下其他的一些文献。  
[Mycat 数据库分库分表中间件](http://mycat.io/)   
[MySQL高可用读写分离集群](http://www.roncoo.com/details?cid=3117ffd4c74b4a51a998f9276740dcfb)

* [2016年07月31日更新](https://segmentfault.com/q/1010000006095431/a-1020000006127244)
* [评论](javascript:void(0);)
* [赞赏](javascript:;)
* [编辑](javascript:;)

[PeterHu](https://segmentfault.com/u/peterhu)148 声望

答案对人有帮助，有参考价值6答案没帮助，是错误的答案，答非所问

谢邀，不是大神，而且感觉题主问的问题好大，不太好回答。

不过据我了解，技术面试一般都是层层渐进，面试官试探候选人技术体系的**深度**，所以说理论上在一个话题上聊的越多，越代表候选人的能力越优秀。很多**高并发**和**分布式**的处理问题其实都是经验问题，因为不同业务场景不同数据量的高并发处理情况完全不同，并没有完全通用的解决方案。

所以面试的时候，理解面试官描述的业务场景非常重要，这也是分析性能瓶颈的关键信息，根据木桶原理，肯定在某一个关键点存在瓶颈。

举个简单例子，**随着业务增长，一个直连数据库的服务器集群遇到了性能瓶颈，应该怎么解决？**

这个时候你要首先分析性能瓶颈出在哪里，首先，要考虑数据库本身的设计是否合理，索引是否起到了作用，分析SQL执行计划，是否可以把数据库进行水平拆分或者垂直拆分来分摊压力。

进而还要分析能否使用分布式的读写分离数据库，引发怎么进行数据同步，数据分发的问题等等。

数据库层分析完之后，在应用层也可以分析，一般是通过缓存来提高查询性能，涉及缓存的命中率问题，缓存的更新问题，缓存的多节点 hash 问题等等。

总得来说，要搞清楚业务场景，然后针对某个具体的问题，去尝试提出解决方案。比如新浪微博，是采用推模式还是拉模式？如果是推模式的话，一个几千万粉的大v发条微博，是不是要给这几千万人推？如果是拉取模式，一个人关注的用户多了每次拉取的时候会不会出现性能问题？

* [2016年07月28日更新](https://segmentfault.com/q/1010000006095431/a-1020000006106031)
* [1 评论](javascript:void(0);)
* [赞赏](javascript:;)
* [编辑](javascript:;)

[letcheng](https://segmentfault.com/u/letcheng)383 声望

答案对人有帮助，有参考价值1答案没帮助，是错误的答案，答非所问

分布式有很多种，比如你把你的项目分成了多个模块，每个模块一个jvm 通过rpc调用来完成整体的工作，这是分布式，比如你redis单台无法抗那么多并发或者无法有那么多内存做缓存，可以做redis集群，这也是分布式，再比如，你的数据库单台无法满足你现在的数据量或者并发量，那么分库分表，通过jta实现事务这也是分布式，再加上什么日志同步，负载均衡，加hdfs存储数据备份，存储日志，加eslaticsearch 分析搜索日志 都可以谈。

* [2016年07月29日回答](https://segmentfault.com/q/1010000006095431/a-1020000006111143)
* [评论](javascript:void(0);)
* [赞赏](javascript:;)
* [编辑](javascript:;)

[yuwenbao](https://segmentfault.com/u/heisijianpan)316 声望

答案对人有帮助，有参考价值1答案没帮助，是错误的答案，答非所问

别人掏出一个笼统的问题，回答者如果跟着笼统，就挂啦···建议回答的时候要有逻辑。  
如果是我，我会2种做法：  
1.找一个我之前做过的项目，跟他讨论瓶颈，然后再引出理论知识，最后给出解决方案。这种做法容易出现自己讲自己的，最好和面试官有些交互。重视面试官的在谈话过程中的疑问，审清楚面试官问的是什么

2.第二种做法就是先摊牌，把比较经典的分布式问题拿出来，比如流量的负载。事务的保证。。。。然后就问题给出业务场景。