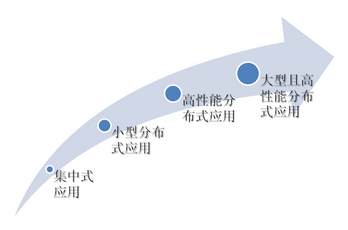
[**Java应用从集中式步入分布式**](http://www.blogjava.net/BlueDavy/archive/2009/01/04/249763.html)

记得Martin大叔在《企业应用架构模式》中特别强调：“能够不分布式的应用就不要分布式”，这句话没什么问题，尤其对于做过分布式应用的人而言，就更会有深刻的体会了，但这个世界偏偏就没有那么简单，大多数人都会碰到分布式应用的场景，尤其是对于大型应用而言，从集中式步入分布式是不可避免的，只是也许是小型分布式的，也许是大型分布式的；也许是有高性能要求的，也许是没有的，在这篇blog中我们来看看java应用从集中式步入分布式后到底会带来些什么挑战。

blog按照应用从集中式演变为大型且高性能的应用的过程来叙述，这也是因为这几类应用对于设计、开发人员的要求也是从低到高的。



Level 1：小型分布式应用

对于做习惯了集中式应用的同学而言，当碰到小型分布式应用场景时，最明显的问题就是如何和远程的系统通讯，如何发消息给远程的系统，告诉它做什么事情，并且得到事情处理的结果，对于java开发人员而言，更习惯的考虑方式也许是如何去调用远程系统提供的接口，在集中式的应用中可以采用A a=new AImpl();这样的方式调用，但远程就没办法这么调用了。

在小型分布式应用中，最需要解决的就是上面所说的远程系统通讯以及远程系统调用的问题了，为此就得学习相关的知识体系。

对于远程系统通讯而言，至少应该知道下TCP/IP、UDP/IP是个大概什么样的概念，BIO/NIO是什么概念，另外就是需要知道下基于java如何去实现tcp/ip+BIO(NIO)、udp/ip+BIO(NIO)的通讯，相应的会需要了解下Java序列化、服务器端采用线程池方式处理等知识，在掌握了这些知识后，基本上做一个小型的java远程通讯的应用不会有什么太大的问题。

对于通过远程系统调用就可实现的小型分布式应用而言，也许更为简单，可以只掌握怎么用spring-remoting、axis或hessian这类的框架就可以了，直接就可实现，当然，如果能了解下rmi、webservice是什么机制就更好了，另外，毕竟远程调用很难做到完全和本地调用一样，尽管所有的远程调用框架都想做到完全透明，但远程调用时仅仅异常就比本地调用会多出好几种来，例如网络异常、超时异常等，因此在调用远程系统时，需要明确的了解这些异常的区分，当然，或者可以认为都是同样的远程业务异常，但在追查错误时，则需要明确区分。

从上面这些知识点可以看出，应用即使是从集中式应用步入小型分布式应用，无论是对于开发、部署还是测试而言，都复杂了很多。

Level 2：高性能分布式应用

对于高性能分布式应用而言，要求就高很多了，因为要做到高性能，仅了解怎么用是不够的，通讯和调用方面的技术都需要了解到底层的具体机制和实现。

对于远程通讯方式的分布式应用而言，需要了解BIO/NIO具体有什么不同，具体机制是怎么样的，以及使用或参考目前主流的通讯框架，像mina、grizzly等，看看它们在做NIO实现时有什么高超的技巧，除了了解这些外，还需要掌握线程池应如何配置；长连、短连什么概念，如何实现；连接怎么管理；序列化如何优化等等。

对于远程调用的分布式应用而言，则在上面的基础上，还需要了解rmi、webservice等的实现机制是怎么样的，另外就是spring-remoting、hessian、axis的实现机制是怎么样的，有什么可改进的地方，在明确了可改进的地方后需要自己选择相应的方案去实现。

从上面这些知识点可以看出，要写出高性能的分布式应用真的不是盖的，确实挺困难的。

Level 3：大型高性能分布式应用

对于大型高性能分布式应用而言，难度就较之之前两个level更大很多，它面临的问题非常的多：

1、稳定性的问题

对于大型分布式应用而言，保持稳定是其重点，要做到稳定，就必须对整个应用的通讯过程的实现技术有全面的了解，以能够做到在使用时尽量的正确，另外就是当出现问题时能够尽快的查明原因。

2、统一、标准通讯的问题

大型分布式应用必然会涉及到很多种不同的应用，应该能够做到对于任何的应用都采用同样的方式进行通讯或调用，这样才能确保使用的简单。

3、分布式事务的问题

当应用变成大型分布式应用后，分布式事务的问题会不可避免的出现，而且这个时候还要求做到高性能，因此通常是不能用传统的数据库事务来实现的，目前对于大型应用的分布式事务而言，流行的有BASE和CAP策略，和传统的ACID策略不同。

4、如何管理这些应用的问题

大型分布式应用，意味着应用非常的多，在这样的场景下，怎么样管理好这么多的应用呢，例如这些应用之间的依赖关系是怎么样的，应用运行期的状况是怎么样的等等。

还有很多各种各样的问题，因此，我们更要说，要做个能支撑大型高性能的分布式应用真的非常非常的不容易，对于实现大型分布式应用而言，目前SOA是个可参考的思想，但其具体如何实现就需要各位架构师根据情况来做出具体的选择了。

从以上简单的描述来看，可以看出，分布式应用比集中式应用真的难很多，至少它是在原有的集中式应用的基础上增加了更多的挑战和对于知识体系的更高的要求，当然，对于大型应用而言，它带来的好处也是非常明显的，因此就让我们痛并快乐着吧。