**作者:***[hsm\_computer](https://www.cnblogs.com/JavaArchitect/)*

# **地址：https://www.cnblogs.com/JavaArchitect/p/10011253.html**

# **[以我的亲身经历为例，告诉大家写简历和面试的技巧（面向高级开发和架构师）](https://www.cnblogs.com/JavaArchitect/p/10720897.html)**

    之前我也写过不少介绍面试相关的博文，有些文章的点击量还不低，这些面试相关的经验大多都是从我面试别人的过程中提炼出来了。

    在18年底到19年3月，陆续面试了几家公司，有成有不成的，最终进了一家比较满意的知名外企。在本文里，我将介绍我这段时间里更新简历和面试的相关经历。可能面试相关的一些方法，大家能从我之前的博文里看到，但在本文里，大家更能从相关实战性的例子中进一步体会到面试技巧的使用方法。

#### **1  刚开始一定得通过一些小公司练手**

    我打算要换工作，就更新了简历，这时就有不少猎头来找我，刚开始，我不敢面大公司，只是去些规模小的互联网公司。因为一方面，当时我很久不面试了，面试实战技能难免有些荒废，另一方面，离我之前的面试毕竟过了一阵，我不知道当前市面上哪些技术比较值钱。

    而且，我在面试前会准备一些针对项目和值钱技术的说辞，这些说辞总要经过实战来演练下。事实上，在和小公司沟通的过程中，进一步提升了我的相关表述能力和相关话术，相当于用小公司来踩坑。相反，如果用大公司的面试失败来换得这些进步，就未免有些可惜了。

    所以，我和猎头明说，我想去大公司，但之前，你得让我到一些小公司去练练手，同时，我面试好以后，会告诉你这些小公司的面试题和相关技巧。这样一些猎头就肯先推荐我去小公司了。

#### **2  投简历前，微调简历描述的技巧**

     1 公司A在职位介绍里，提到对分布式组件有要求，尤其是负载均衡集群。这块技术我在通用版简历中只是笼统写了下，有nginx+lvs实际经验，既然这个公司有类似要求，我就在简历中细化了这块：用到了lvs+keepalive集群和nginx，它们分别工作在4层和7层，（后省略相关技术使用点描述），最后介绍下效果：用了这个负载均衡组件外，我们系统承受住了xx的并发量。

    2 在公司B的职位介绍里，提到了高并发的需求，那么我就着重写了我压测的相关经验，包括压测环境怎么部署，用到哪些组件，再同时列了些压测后改善系统性能的若干方法。

    3 公司C比较小，需要一位比较资深的能带团队的架构师，那么我在简历中除了介绍技术外，还着重写了带团队的若干经验。

    对大家来说，其实面试任何一家公司，它们的职位要求绝对不可能一样。这时大家最好是微调下简历。这种微调，绝对不是胡乱编写，比如上个项目干的是测试，简历中改成开发。而是挖掘下侧重点。

    就像我之前给出的案例，对于分布式技术，我上个项目确实用过，但简历中篇幅有限，我不可能面面俱到地写，但既然你们公司有要求，那么我就针对性地详细介绍。

    其实这块谈不上是面试技巧，而是语文技巧。首先是审题，理解职位介绍的主要内容，然后再像写作文那样，详细扩展下上个项目里相关技术点。甚至这只是中考语文的考点而不是高考的，但我自己的体会是，在这段时间里针对每份职位介绍我都做了微调，所以基本上是投一份简历就能得到一次面试机会。

#### **3  在换工作前，需要定下预期目标，同时要观察公司的情况**

    我见过不少人，面试成了就进去了，结果虽然钱达到预期，但工作环境和行业背景都不是自己想要的，对自己未来发展很不利。

    目前不少公司的情况，比如行业背景（比如金融保险电商等）还有工作强度（加班情况）在网上都能查到，用什么技术（比如分布式技术还是云还是AI）通过打听也能问到。互联网是万能的，甚至如果仔细些，一些离职率，每年工资的增长幅度，甚至这个公司是第几轮融资，一般也能看到。

    而且，大家在面试的时候，可以通过现场观察以及和面试官交流，旁敲侧击地打探到上述情况。在这个基础上，大家需要结合自己当前的需求和未来3到5年的发展方向，找若干类适合自己的公司，下面就给些我个人的和我所见到的例子。

    1 （这是我个人的观点）比如我不大愿意进创业型公司，也比较排斥长时间的996，所以这类公司我只是拿来练手。

    2  有段时间，不少P2P金融公司踩雷，所以哪怕面试时，这类公司的面试官会说得天花乱坠（很少有说自己不好的），但这类公司得慎重。

    3  有次我到一个现场去面试，发现公司规模不大，是在一个园区里占1个楼层，且大多是业务人员，开发的不多，而且人事总会说自己公司正在融资，那么这个公司就有可能不稳定，在有其它选项时，我就没去。

    4 我认识一个人，他想在Java分布式这块有所发展，他5年后的目标是架构师，在面试过程中，他就详细问了未来新进组的技术栈，等确定可以了再入职。

    换工作是个折腾人的事情，虽然没有十全十美的事情，进新公司后一定会发现和当初的预期有所不同，但如果事先不调查，那么入职后后悔这还是小事，耽搁自己的发展才是大事。

#### **4  面试前，我总要查看职位介绍，以此准备项目介绍**

    在上文里，我一直介绍的是准备简历和定目标，从这部分开始，将介绍准备面试。在之前的博文里，我一直强调项目介绍的重要性，所以当我自己出去面试的时候，这块绝对重视。在准备项目介绍前，我还是会通读应聘职位的技术介绍，然后按如下的步骤来准备。

    1 介绍项目的基本情况，比如这个项目是干嘛的，有哪些模块流程是什么，自己做了哪些，持续了多久，有多少个人参与。这块切记，可以谈下业务，但别深入，因为面试官不熟悉，也不想熟悉候选人的业务，这块时间控制在1分钟之内。

    2 通过介绍项目用到的软件，以及项目的管理方式，来进一步验证项目的真实性，否则很难和培训班给出的项目有差别。

    我介绍的时候，就说，项目管理用Maven，部署用jenkins，静态扫描用Sonar，任务管理和bug管理用jira，平时采用敏捷的项目迭代方式，每天有站会，大约1月一个迭代版本。这块大家可以根据自己的情况来介绍，时间也别太长，估计用1分钟也就够了。

    3 这块最重要，是结合业务，抛出项目用到的技术，但别展开。

    比如我见过一个职位介绍，里面写到需要有数据库优化的经验，那么我就说，项目里我用过MyCat作为分库分表，（不展开技术），结果上线后，我们数据库能承受住每秒2000个并发请求（说下用好的结果）。

    又如一个JD里说要用到微服务技术，那么我就说，项目里用到了Spring Cloud框架，用到了Ribbon，Eureka等组件，容器是Docker。用好以后，在发布时会发现，各模块之间的调用耦合性大大降低。

    这块为什么不用展开？第一，当前还在介绍项目，如果展开技术，就会主次不分，面试官会对你的逻辑思路产生质疑。第二，由于在JD里有要求，面试官一定会深入来问，别人问你说总比自己自吹自擂要好。 第三，就好比之前考文科内容，当前在介绍项目时，先应当把所有的点说出来，面上自有机会发挥。

#### **5  准备项目介绍时，我是如何准备值钱技术的**

    这块其实和介绍项目经验时密不可分，这里就拿我介绍过的Netty组件来说明。

    1 准备如何过渡到相关技术的说辞。

    这里一定别让面试官感觉你在自说自话。比如我准备的方案1是，在介绍好项目背景后，如果面试官没打断，我就询问，能否介绍其中的Netty细节？得到允许后再说。方案2是把这些技术关联到面试官可能会问的问题上，比如问及网络通讯时介绍Netty，这个事先我会整理一个问题列表，遇到此类问题，我会抛出Netty说辞。对于其他技术也可以照此准备。

    2 结合业务需求点，说下为什么要用这个技术，以及用了有什么好处。

    比如为了优化网络通讯协议，所以用基于TCP协议的Netty，我业务模块里的xxx功能是用到netty，再稍微结合业务和Netty展开下。这样的话，面试官就会感觉这个技术你不是仅仅在学习中用到，而且在项目业务里用到。

    3 准备下这个技术里我踩过的坑。

     比如通讯过程中我遇到了半包和粘包问题，对此我更是深入了底层代码说明，还有Netty里的堆外内存管理问题，我准备了一个由于管理不善而导致的OOM问题的排查和解决过程（这类问题网上相关的也很多）。

    4 我再准备了用了这个技术后对项目的帮助。

    比如Netty是基于TCP协议的，它要比Http协议要轻，所以通讯性能高，且Netty内部的Reactor线程模型对系统的IO帮助很大。

    其实准备到这个程度，一般的面试官就不怎么再问了，哪怕面试官再问了一个我不知道的问题，我也可以理直气壮地说不知道了，毕竟我之前已经介绍了不少底层代码，再精通Netty的人也未必能面面俱到地在脱稿的情况下回答出任何问题。

    但为了以防万一，我更准备了“源于项目但高于项目”的说辞，一方面证明我平时还会不断专研，另一方面进一步证明我熟悉并能合理使用Netty的底层代码。这方面我准备了如下的内容：

    1 Netty的重要组件以及一个请求过来时各模块工作的时序图。2 Netty内部零拷贝的原理和好处，对此还专门准备了代码。3 Netty读写索引的工作方式。4 Netty异步处理的机制。

    大家可以想象下，如果在面试中，大家很自信地边写底层代码边解释问题，面试官会对大家有什么评价？其实这些都是可以准备的，我一直没明白，为什么有些人面试前不做任何准备？这些准备的素材又不是找不到。

    下面我再略讲下我MyCAT的准备要点。

    1 还是讲下业务背景，为什么要用？2 如何使用，无非是设置分库规则，改写SQL语句等。3 准备下踩到的坑，比如自增长主键在每台机器上都要保证唯一。4 然后再结合些底层代码，准备下一条SQL语句是如何分发到对应的分库上的，然后执行好以后又如何返回的。

    基本上我面试时说到这个程度后，面试官就不怎么问了。

#### **6  事前评估好自己的薄弱点，想办法让面试官忽略**

    哪怕事先再准备，我们也不能保证自己一定能达到职位介绍里的要求，这时候就得准备一个自己薄弱技术的列表，然后想办法在面试中绕过去，如果绕不过去，就找些替代方案。

    比如我对消息通讯组件不熟悉，在面试前我哪怕看代码看文档也无法进一步理解，我就在面试中，故意不提这个环节，抛出其它值钱技术，引诱面试官来问，一般交流一个技术点大约耗时10分钟（边画图边介绍再提问），所以面试过程中详细交流到的最多也就5个点，少的话就3个，在不少面试里就根本没涉及到这块。

     有两场面试，技术面试官问及到了，我就说项目里用过，同时多介绍些使用场景，比如用Kafka进行异步消息通知，通过消息中间件来保证交易的幂等性。这样面试官听到我介绍了应用场景，就至少会认为我用过。只有一场面试，面试官在kafka方面问得很细，问到如何集群内防止消息重发的细节，我就没回答上。

    也就是说，我通过刻意回避外加介绍业务应用场景，很大程度上隐藏了我了解不深的技术。相反，如果我在简历中大大咧咧地写上用过kafka，那么可能每场面试都会被问到，毕竟在面试中被问倒不是件好事。

    又如，我对分布式集群里的失效转移底层细节始终没理解透彻，那么在相关问题上就得非常小心。

    有一场面试里，我被问到dubbo+zk集群，这块如果细问，一定会问到失效转移的选举算法，我对此不熟，那么我就先说dubbo的通讯协议，序列化方式以及通过设置超时时间实现服务降级的思路，然后再说zk的部署。最终面试官果然问到选举算法，但之前我已经结合业务谈了dubbo底层代码和zk的相关部署，所以不理解选举算法并没有产生太大的影响。

    这里用到的技巧是“偷换概念”，一个问题有多个回答方向，你先让面试官感觉到你在很多领域非常熟悉，那么在个别领域不熟悉就无伤大雅了。

#### **7  和猎头和人事的沟通要点**

    1 猎头和我们是一伙的，我们成了猎头也能拿到提成，所以和猎头谈时一定得坦诚，把你优点和弱点都说，猎头会帮忙想办法（比如问面试题）或者是帮忙优化简历。

    2 面试前，一定和猎头谈次，或者问下对这家公司该说什么话，以及之前人的成功失败经验，如果问题，去问下面试题（包括笔试题和口头技术问题）。

    3 面试后，一定第一时间和猎头沟通，这时个人直接去问不合适。而且还可以通过猎头说，哪怕这个岗位没面上，还可以尝试这个公司的其它岗位。

    4 谈工资时一定得先听猎头的建议，不少人对当前行情不了解，或者信心不足，工资没要足，这时让猎头谈是比较合适的，因为猎头就干这个的。一般来说，跳槽后，工资大约有20%到50%的涨幅，有些公司会要前家公司的收入证明，相关问题也可以问猎头。

    但和人事谈，就要小心了，毕竟人事是公司开工资的。

    1 人事会问之前公司的离职原因，这时千万别让人事感觉出你不稳定，你可以说为了进一步提升自己的能力，此类问题网上太多，大家可以自己查。

    2 在谈发展方向时，别和JD上的背离。比如JD上是招个技术开发人才，你却和人事说，干了1年后想转Team leader管理岗位，或者说同时想做测试，那么人事就会感觉到，你未来的发展规划和公司想要的不匹配。

    总之，面试时人事会非常nice地和你交流，有时候甚至是套话，看你是否稳定，是否能很好地和团队协作，能否加班，我就见过有人卡在人事这关，甚至还不知道为啥被拒。

    而且一般你在和人事谈工资前，最好向猎头打探下这个职位的工资范围，在此基础上，如果你感觉面试非常好，或者他们着急要人，或者你感觉加班会很多，可以适当多要些，相反如果你非常想进这个公司，那么就自己斟酌。

#### **8  总结：不能常跳槽，但得经常更新自己的技能**

    在上文里，我通过最近换工作的经历，向大家介绍了相关经验，比起我的两本书，[Java Web轻量级开发面试教程](https://item.jd.com/12136095.html" \o "科技IT爆款好书，全场每满200减100，为新学期助力加油！戳我》》" \t "https://www.cnblogs.com/JavaArchitect/p/_blank)和[Java核心技术及面试指南](https://item.jd.com/12421187.html" \o "一本书讲透Java语言核心概念、编程思想、开发实战及面试技巧" \t "https://www.cnblogs.com/JavaArchitect/p/_blank)，自认为本文介绍的技巧更有实战性。

    我不推荐经常换工作，比如一年换个两家，那么公司会质疑稳定性，但对于我们做IT的人来说，如果手头技术停滞不前，那么自己的竞争高工资的能力也会大大降低。

    那么目前该了解哪些技术？以及在这个技术方面我当前掌握得如何？这固然可以通过工作来检验，但工作中接触到的技术毕竟是有限的，这时就可以通过面试来检验自己当前的学习情况，同时也可以通过面试题去了解自己的努力方向。