

# 给学习 J2EE 的朋友一些值得研究的开源项目

江苏 无锡 缪小东

这篇文章写在我研究 J2SE、J2EE 近三年后。前 3 年我研究了 J2SE 的 Swing、Applet、Net、RMI、Collections、IO、JNI.....研究了 J2EE 的 JDBC、Servlet、JSP、JNDI.....不久我发现这些好像太浮浅了：首先，我发现自己知道的仅仅是 java 提供的大量的 API，根本不能很好地使用它；其次，我根本就没有学到任何有助于写程序的知识，此时我也只不过能写个几页的小程序。出于这个幼稚的想法我研究了 JDK 中 Collections、Logger、IO.....的源代码，发现这个世界真的很神奇，竟然有如此的高手——利用 java 语言最最基本的语法，创造了这些优秀的 Framework。从此一发不可收拾，我继续研究了 J2EE 的部分，又发现这是一个我根本不能理解的方向（曾经有半年停滞不前），为什么只有接口没有实现啊！后来由于一直使用 Tomcat、Derby 等软件突然发现：哦！原来 J2EE 仅仅是一个标准，只是一个架构。真正的实现是不同提供商提供的。接着我研究了 MOM4J、OpenJMS、Mocki、HSQLD.....发现这些就是 J2EE 的实现啊！原来软件竟会如此复杂，竟会如此做....规范和实现又是如何成为一体的呢？通过上面的研究发现：原来 J2EE 后面竟然有太多太多理念、太多太多的相似！这些相似就是其背后的理念——设计模式！（很幸运，在我学 java 的时候，我一般学 java 的一个方向就会读一些关于设计模式的书！很幸运，到能领略一点的时候能真正知道这是为什么！）其实模式就是一种思维方式、就是一种理念.....模式是要运用到程序中的，只有从真正的项目中才能领会模式的含义.....

学得越多，发现懂得越少！在学习过程中发现一些很有用，很值得学习的开源项目，今天在此推荐给大家。

## 一、JavaServlet 和 JSP 方向

很多人都是从 Servlet 和 JSP 步入 J2EE 的。它就是 J2EE 的表现层，用于向客户呈现服务器上的内容。J2EE 很重要的方面。不罗嗦了！大家都知道的！下面就开始推荐吧！

### 1. Jakarta Tomcat

Apache 基金会提供的免费的开源的 Servlet 容器，它是的 Jakarta 项目中的一个核心项目，由 Apache、Sun 和其它一些公司（都是 IT 界的大鳄哦）及个人共同开发而成，全世界绝大部分 Servlet 和 Jsp 的容器都是使用它哦！由于 Sun 的参与和支持，最新的 Servlet 和 Jsp 规范总能在 Tomcat 中得到体现。

不过它是一个非常非常全的 Servlet 容器，全部源码可能有 4000 页，对于初学者或者一般的老手可能还是比较大了！在你有能力时推荐研究！下载地址：<http://jakarta.apache.org/tomcat/index.html>

下面推荐两个小一点的吧！

### 2. Jetty

Jetty 是一个开放源码的 HTTP 服务器和 Java serverlet 容器。源代码只有 1000 页左右，很值得研究。有兴趣可以去 <http://jetty.mortbay.com/> 下载看看。我曾经翻了一下，只是目前没有时间。（都化在博客上了，等博客基本定型，且内容完整了，再干我热衷的事件吧！）

### 3. Jigsaw

Jigsaw 是 W3C 开发的 HTTP，基于 Java 的服务器，提供了未来 Web 技术发展的蓝图。W3C 知道吧！（太有名气了，很多标准都是它制订的！有空经常去看看吧！）下载网址：<http://www.w3.org/Jigsaw> 代码仅仅 1000 页左右。

### 4. Jo!

Jo!是一个纯 Java 的实现了 Servlet API 2.2, JSP 1.1, 和 HTTP/1.1 的 Web 服务器。它的特性包括支持 servlet tag,支持 SSI, 高级线程管理, 虚拟主机, 数据缓存, 自动压缩 text 或 HTML 文件进行传输, 国际化支持, 自动重新加载 Servlet、Jsp, 自动重新加载 web 工程文件(WARs), 支持 WAR 热部署和一个 Swing 控制台。jo! 可以被用做 jboss 和 jakarta avalon-phoenix 的 web 容器。下载地址 <http://www.tagtraum.com/>。我极力推荐大家在研究 Tomcat 之前研究该软件, 主要是其比 Tomcat 小多了, 且开发者提供比较全的手册。该方向研究这两个也就可以了！

## 二、JDBC 方向

很多人都喜欢 JDBC，数据库吗！很深奥的东西，一听就可以糊弄人。其实等你真正研究了数据库的实现后发现，接口其实真的太简单，太完美了！要想设计如此优秀的框架还是需要学习的。下面就推荐几个数据库的实现吧！

### 1. Hypersonic SQL

Hypersonic SQL 开源数据库方向比较流行的纯 Java 开发的关系型数据库。好像不是 JDBC 兼容的，JDBC 的很多高级的特性都没有支持，不过幸好支持 ANSI-92 标准 SQL 语法。我推荐它主要是它的代码比较少 1600 页左右，如此小的数据库值得研究，而且他占的空间很小，大约只有 160K，拥有快速的数据库引擎。推荐你的第一个开源数据库。下载地址：<http://hsqldb.sourceforge.net/>。

### 2. Mckoi DataBase

McKoiDB 和 Hypersonic SQL 差不多，它是 GPL 的 license 的纯 Java 开发的数据库。他的 JDBC Driver 是使用 JDBC version 3 的 Specifaction。他也是遵循 SQL-92 的标准，也尽量支持新的 SQL 特色，并且支持 Transaction 的功能。两个可以选一个吧！下载地址：<http://mckoi.com/database/>。

### 3. Apache Derby

学 Java 的数据库我建议使用 Apache Derby，研究数据库想成为一个数据库的高手我建议你先研究 Apache Derby。Apache Derby 是一个高质量的、纯 Java 开发的嵌入式关系数据库引擎，IBM® 将其捐献给 Apache 开放源码社区，同时 IBM 的产品 CloudSpace 是它对应的产品。Derby 是基于文件系统，具有高度的可移植性，并且是轻量级的，这使得它非常便于发布。主要是没有商业用户的很好的界面，没

有其太多的功能。不过对于我们使用数据库、研究数据库还是极其有用的。对于中小型企业说老实话你也不要用什么 Oracle、SqlServer 了，用 Derby 就可以了，何况是开源的呢！只要能发挥其长处也不容易啊！下载地址：<http://incubator.apache.org/derby>。

不过在没有足够的 ability 前，不要试图读懂它！注释和源代码 15000 页左右，我一年的阅读量！能读下来并且能真正领会它，绝对高手！你能读完 Derby 的源代码只有两种可能：1.你成为顶尖的高手——至少是数据库这部分；2.你疯了。选择吧!!!! 作为我自己我先选择 Hypersonic SQL 这样的数据库先研究，能过这一关，再继续研究 Derby！不就是一年的阅读量吗！我可以化 3 年去研究如何做一个数据库其实还是很值得的！有的人搞 IT 一辈子自己什么都没有做，也根本没有研究别人的东西！

作为一个 IT 落后于别国若干年的、从事 IT 的下游产业“外包”的国家的 IT 从业人员，我认为还是先研究别人的优秀的东西比较好！可以先研究别人的，然后消化，学为己用！一心闭门造车实在遗憾！

### 三、JMS 方向

JMS 可能对大家来说是一个比较陌生的方向！其实 JMS 是一个比较容易理解，容易上手的方向。主要是 Java 消息服务，API 也是相当简单的。不过在企业应用中相当广泛。下面就介绍几个吧！

#### 1. MOM4J

MOM4J 是一个完全实现 JMS1.1 规范的消息中间件并且向下兼容 JMS1.0 与 1.02。它提供了自己的消息处理存储使它独立于关系数据与语言，它的客户端可以用任何语言开发。它可以算是一个小麻雀，很全实现也比较简单！它包含一个命名服务器，一个消息服务器，同时提供自己的持续层。设计也相当的巧妙，完全利用操作系统中文件系统设计的观念。代码也很少，250 页左右，最近我在写该实现的源代码阅读方面的书，希望明年年中能与大家见面！下载地址：<http://mom4j.sourceforge.net/index.html>。

#### 2. OpenJMS

OpenJMS 是一个开源的 Java Message Service API 1.0.2 规范的实现，它包含有以下特性：

1. 它既支持点到点（point-to-point）（PTP）模型和发布/订阅（Pub/Sub）模型。
2. 支持同步与异步消息发送。
3. JDBC 持久性管理使用数据库表来存储消息。
4. 可视化管理界面。
5. Applet 支持。
6. 能够与 Jakarta Tomcat 这样的 Servlet 容器结合。
7. 支持 RMI, TCP, HTTP 与 SSL 协议。
8. 客户端验证。
9. 提供可靠消息传输、事务和消息过滤。

很好的 JMS 方向的开源项目！我目前也在研究它的源代码！学习它可以顺便研究 JNDI 的实现、以及网络通信的细节。这是我 JMS 方向研究的第二个开源项目。代码量 1600 页左右吧！下载地址：<http://openjms.sourceforge.net/index.html>

### 3. ActiveMQ

ActiveMQ 是一个开放源码基于 Apache 2.0 licenced 发布并实现了 JMS 1.1。它能够与 Geronimo，轻量级容器和任 Java 应用程序无缝的给合。主要是 Apache 的可以任意的使用和发布哦！个人比较喜欢 Apache 的源代码！下载地址：<http://activemq.codehaus.org/>

### 4. JORAM

JORAM 一个类似于 openJMS 分布在 ObjectWeb 之下的 JMS 消息中间件。ObjectWeb 的产品也是非常值得研究的！下面我还会给大家另外一个 ObjectWeb 的产品。下载地址：<http://joram.objectweb.org/>  
我个人推荐：OpenJMS 和 ActiveMQ！

## 四、EJB 方向

EJB 一个比较“高级”的方向。Sun 公司曾经以此在分布式计算领域重拳出击。不过自从出现了 Spring、Hibernate.....后似乎没落了！这个方向单独开源的也比较少，主要 EJB 是和 JNDI、JDBC、JMS、JTS、JTA 结合在一起的是以很少有单独的。下面推荐两个不过好像也要下载其它类库。

### 1. EasyBeans

ObjectWeb 的一个新的项目，一个轻量级的 EJB3 容器，虽然还没有正式发布，但是已经可以从它们的 subversion 仓库中检出代码。代码量比较小 600 页左右，熟读它可以对网络编程、架构、RMI、容器的状态设计比较了解了！即学会 EJB 又能学习其它设计方法何乐而不为哦！下载地址：<http://easybeans.objectweb.org/>

### 2. OpenEJB

OpenEJB 是一个预生成的、自包含的、可移植的 EJB 容器系统，可以被插入到任意的服务器环境，包括应用程序服务器，Web 服务器，J2EE 平台，CORBA ORB 和数据库等等。OpenEJB 被用于 Apple 的 WebObjects。听起来很好，我目前没有研究过。不知道我就不推荐了。下载地址：<http://www.openejb.org/>

## 五、J2EE 容器

上面谈了这么多，都是 J2EE 的各个方向的。其实 J2EE 是一个规范，J2EE 的产品一般要求专业提供商必须提供它们的实现。这些实现本身就是 J2EE 容器。市场上流行的 J2EE 容器很多，在开源领域流行的只有很少，很少。其中最著名的是 JBoss。

### 1. JBoss

在 J2EE 应用服务器领域，Jboss 是发展最为迅速的应用服务器。由于 Jboss 遵循商业友好的 LGPL 授权分发，并且由开源社区开发，这使得 Jboss 广为流行。另外，Jboss 应用服务器还具有许多优秀的特质。

其一，它将具有革命性的 JMX 微内核服务作为其总线结构；

其二，它本身就是面向服务的架构（Service-Oriented Architecture，SOA）；

其三，它还具有统一的类装载器，从而能够实现应用的热部署和热卸载能力。因此，它是高度模块化的和松耦合的。Jboss 用户的积极反馈告诉我们，Jboss 应用服务器是健壮的、高质量的，而且还具有良好的性能。为满足企业级市场日益增长的需求，Jboss 公司从 2003 年开始就推出了 24\*7、专业级产品支持服务。同时，为拓展 Jboss 的企业级市场，Jboss 公司还签订了许多渠道合作伙伴。比如，Jboss 公司同 HP、Novell、Computer Associates、Unisys 等都是合作伙伴。

在 2004 年 6 月，Jboss 公司宣布，Jboss 应用服务器通过了 Sun 公司的 J2EE 认证。这是 Jboss 应用服务器发展史上至今为止最重要的里程碑。与此同时，Jboss 一直在紧跟最新的 J2EE 规范，而且在某些技术领域引领 J2EE 规范的开发。因此，无论在商业领域，还是在开源社区，Jboss 成为了第一个通过 J2EE 1.4 认证的主流应用服务器。现在，Jboss 应用服务器已经真正发展成具有企业强度（即，支持关键级任务的应用）的应用服务器。

Jboss 4.0 作为 J2EE 认证的重要成果之一，已经于 2004 年 9 月顺利发布了。同时，Jboss 4.0 还提供了 Jboss AOP（Aspect-Oriented Programming，面向方面编程）组件。近来，AOP 吸引了大量开发者的关注。它提供的新的编程模式使得用户能够将方面（比如，事务）从底层业务逻辑中分离出来，从而能够缩短软件开发周期。用户能够单独使用 Jboss AOP，即能够在 Jboss 应用服务器外部使用它。或者，用户也可以在应用服务器环境中使用它。Jboss AOP 1.0 已经在 2004 年 10 月发布了。很有名吧！可以下载一个用一下，下载地址：<http://www.jboss.org/>

关于 JBoss 的使用资料也非常多，甚至比商业软件的还多。有机会研究吧！

## 2. JOnAS

JOnAS 是一个开放源代码的 J2EE 实现，在 ObjectWeb 协会中开发。整合了 Tomcat 或 Jetty 成为它的 Web 容器，以确保符合 Servlet 2.3 和 JSP 1.2 规范。JOnAS 服务器依赖或实现以下的 Java API：JCA、JDBC、JTA、JMS、JMX、JNDI、JAAS、JavaMail。下载地址：<http://jonas.objectweb.org/>

## 3. Apache Geronimo

Apache Geronimo 是 Apache 软件基金会的开放源码 J2EE 服务器，它集成了众多先进技术和设计理念。这些技术和理念大多源自独立的项目，配置和部署模型也各不相同。Geronimo 能将这些项目和方法的配置及部署完全整合到一个统一、易用的模型中。作为符合 J2EE 标准的服务器，Geronimo 提供了丰富的功能集和无责任 Apache 许可，具备“立即部署”式 J2EE 1.4 容器的各种优点，其中包括：

1. 符合 J2EE 1.4 标准的服务器。
2. 预集成的开放源码项目。
3. 统一的集成模型。
4. 可伸缩性、可管理性和配置管理功能。

我一直比较推荐 Apache 的产品。主要是可以任意自由地使用。下载地址：<http://incubator.apache.org/projects/geronimo/>

## 六、其它

讲了这么多大家可能很厌烦了！是不是很多很多啊！其实不然，我们不会的太多太多了！不会的太多太多了。不管你是不是 J2EE 高手，还是 J2SE 高手，有些东西你要绝对很精明的。例如：1.Java 的 Collections Framework 就是 java 的数据结构了，不仅要吃透它，还要能按照需要扩展它，利用其思想创建一个自己的数据结构。2.网络编程肯定要会吧，现在以及以后很多程序都是不在同一台机器上的，不会网络怎么行哦！3.IO 肯定要会的吧！你的程序难道不用输入输出数据啊！整个 IO 包加 NIO 也有 600 多页的源代码哦！4.JDBC 你要会吧！数据库都不会，在你的企业应用中你的数据又保存到哪里啊！文件中——太落后了吧！典型的没有学过 J2EE。尽管数据库背后也是采用文件保存的。5.Serverlet、JSp 你要是做网页做网站，肯定要能做到。问你一个简单的问题，网页中如何实现分页啊！有具体方法的就在本文章后发言吧！6. Ant 要会吧！java 语言中发布的工具，类似与 c 中的 make 工具。7.JUnit 用过吧！单元测试软件。你不要啊！你的软件就没有 bug！你牛！（建议大家研究研究其源代码，很有用的框架，包含大量的设计模式，源代码不到 100 页！看了只能感叹——高手就是高手）细心的朋友可以看到在你使用的很多 IDE 工具中都有 JUnit 哦！就是它。

一切的一切才刚刚开始！有兴趣，有需要你可以研究数据库连接池的框架，如：C3P0、Jakarta DBCP、DBPool....可以研究 J2EE 框架 Spring..... Web 框架 Struts.....持久层框架 Hibernate.....甚至开发工具 Eclipse.....Sun 领导的点对点通信的 JXTA.....报表工具 JFreeChart、JasperReports.....分布式网络编程的 CORBA、网络通信的 JGROUPS、XML 解析的 xerces.....（在不经意间开源已经步入你的电脑，不信啊！你 JDK 的安装目录 jdk1.6.0\src\com\sun\org\apache 就是 Xerces，一个 XML 解析的著名的开源项目）

不管怎么样我还是建议从基本的做起，学精 J2SE，熟读它的源码，准确了解其设计理念，然后分头击破 J2EE——一口吃不成一个胖子！不要贪多贪广！脚踏实地就可以了！

假如你是初学者，请你到我的博客看那些我给你推荐的书籍！见《Java 推荐读物与源代码阅读》这篇文章。假如你是以为 J2SE 高手想进入 J2EE 方向，请先了解设计模式，然后搞懂其各个方向的 API，接着可以研究各个方向的开源项目提高自己！充实自己！接着就是服务于你的行业，做你想干的事！谢谢！你已经是高手了，不需要来听我唠叨了！不过我还是恳切希望你能到我的博客给我唠叨！

谢谢大家！浪费大家 N 多时间！

<http://blog.163.com/miaoxiaodong78/>欢迎您！