

## 几种常见的内嵌数据库 - CSDN博客

因笔者最近有一个项目要使用内嵌数据库，在对内嵌数据库选型上纠结了一段时间，以下则是对各个内嵌数据库的整理总结给大家共享，有不当之处还请指正。

- Sqlite

SQLite第一个Alpha版本诞生于2000年5月. 至今已经有10个年头了. SQLite也迎来了一个版本 SQLite 3已经发布.

SQLite是一款轻型的数据库，是遵守ACID的关联式数据库管理系统，它的设计目标是嵌入式的，而且目前已经在很多嵌入式产品中使用了它，它占用资源非常的低，在嵌入式设备中，可能只需要几百K的内存就够了。它能够支持 Windows/Linux/Unix等等主流的操作系统，同时能够跟很多程序语言相结合，比如Tcl、C#、PHP、Java等，还有ODBC接口。同时SQLite支持以加密方式存储数据信息。

优点：占用内存非常小，适合内存受限设备（如手持设备）。

缺点：官方已停止开发，后续升级维护困难。

授权方式：public domain

官方网站：<http://www.sqlite.org/>

- Apache derby

Apache Derby 是用 Java 语言编写的，所以可以在任何存在合适的 Java 虚拟机（JVM）的地方运行。这意味着 Derby 实际上可以跨操作系统上运行，包括 Microsoft Windows、Macintosh、Linux 和 UNIX平台。支持 SQL 99 和 SQL 2003语法。

Derby 也可以在三个 Java 平台的任何一个上运行：Java 2 Platform, Micro Edition (J2ME)、Java 2Platform, Standard Edition (J2SE) 和 Java 2 Platform,Enterprise Edition (J2EE)。Derby 软件绑定在 Java 档案（JAR）文件中，只有2 MB 大小。由于内存占用这么小，所以 Derby 数据库可以容易地与应用程序绑定在一起。

从 Cloudscape Version10 开始，IBM 的 Cloudscape 解决方案都以 Derby 代码为基础。

可以从 IBM 下载 Cloudscape。下载 Cloudscape 不需要任何许可费用，但是可以从 IBM 购买对它的支持；在 Sun JDK6开始，Apache derby 作为 sun jdk内置数据库使用。

优点：完全由java开发，适合JavaEE平台；得到Sun，IBM和其他框架（如hibernate）支持，应用前景良好。

缺点：因完全使用java开发，不适用于在其他编程语言内置使用。

授权方式：Apache License 2.0

官方网站：<http://db.apache.org/derby/>

- hsqldb

Hsqldb是一个开放源代码的JAVA数据库，它具有标准的SQL语法和JAVA 接口，它可以自由使用和分发，非常简洁和快速的。

HSqldb 是由 Tomas Muller 的 Hypersonic SQL 后续开发出来的项目, hsql 已经停止研发了, hypersonic db 是纯 java 所开发的数据库, 可以透过 jdbc driver 来存取, 支持 ANSI-92 标准的 SQL语法, 而且他占的空间很小, 大约只有 160K, 拥有快速的数据库引擎, 也提供了一些工具, 例如 web-server, 缓冲查询, 及一些管理工具. 他是属于 BSD 的 license, 可以自由下载, 并且可以安装使用在商业产品之上。

HSqldb非常适合在用于快速的测试和演示的Java程序中。做单元测试也非常理想。

HSqldb不适合管理大型数据，例如百万数量级记录的数据库应用。HSqldb简介它具有 Server模式，进程内模式(In-Process)和内存模式(Memory-Only)三种。运行Hsqldb需要hsqldb.jar包，它包含了一些组件和程序。

优点：完全由java开发，适合JavaEE平台的测试环境。

缺点：官方已停止开发，后续升级维护困难。

授权方式：BSD License

官方网站：<http://hsqldb.org>

- MS Access

基于文件的数据库，由Microsoft office提供，非常适合window操作系统中使用，可以通过ADO API读写数据。

优点：使用ADO API访问，非常适合在window平台开发的程序使用

缺点：不支持跨平台

授权方式：私有授权协议

- Oracle Berkeley DB

Berkeley DB是历史悠久的嵌入式数据库系统，主要应用在UNIX/LINUX操作系统上，其设计思想是简单、小巧、可靠、高性能。本文是对DB开发的一个入门级指南，重点讨论了DB的核心数据结构和数据访问算法，并通过实际的代码演示如何使用DB。最后有一个对DB的简单总结，并提出作者对工具选择的一些感想。

DB最初开发的目的是以新的HASH访问算法来代替旧的hsearch函数和大量的dbm实现（如AT&T的dbm，Berkeley的 ndbm，GNU项目的gdbm），DB的第一个发行版在1991年出现，当时还包含了B+树数据访问算法。在1992年，BSD UNIX第4.4发行版中包含了DB1.85版。基本上认为这是DB的第一个正式版。在1996年中期，Sleepycat软件公司成立，提供对DB的商业支持。在这以后，DB得到了广泛的应用，当前最新版本是 Berkeley DB 11gR2。

DB支持几乎所有的现代操作系统，如LINUX、UNIX、WINDOWS等，也提供了丰富的应用程序接口，支持C、C++、JAVA、PERL、TCL、PYTHON、PHP等。DB的应用十分广泛，在很多知名的软件中都能看到

其身影。

值得注意的是DB是嵌入式数据库系统，而不是常见的关系/对象型数据库，对SQL语言不支持，也不提供数据库常见的高级功能，如存储过程，触发器等。

优点：数据存取效率高，适合数据缓存。

缺点：使用键值对方式保存数据，应用范围窄；不支持SQL语法。

授权方式：BSD License

官方网页：

<http://www.oracle.com/technology/products/berkeley-db/index.html>

- db4o

db4o 是一个开源的纯面向对象数据库引擎，对于Java 与 .NET 开发者来说都是一个简单易用的对象持久化工具，使用简单。同时，db4o 已经被第三方验证为具有优秀性能的面向对象数据库。

db4o 的目标是提供一个功能强大的，适合嵌入的数据库引擎，可以工作在设备，移动产品，桌面以及服务器等各种平台。db4o 是 100% 原生的面向对象数据库，直接使用编程语言来操作数据库。程序员无需进行 OR 映射来存储对象，大大节省了程序员在存储数据的开发时间。

使用 db4o 仅需引入 400 多 k 的 jar 文件或是 dll 文件，内存消耗极小。

优点：对象数据库，适合面向对象编程思想；占用内存小，适合内存受限使用环境。

缺点：在大规模应用环境下，存在性能瓶颈。

授权方式：GPL

官方网页：<http://www.db4o.com>