# H2数据库

H2是一个免费开源，用纯Java开发的关系数据数据库。H2数据数据库可以嵌入到Java应用中作为程序的一部分或以Server-Client模式运行。而且，H2连同Jdbc驱动一起打包在一起，大小也只有1M左右。

## H2支持3种运行模式

### 嵌入式模式

使用内嵌模式时，应用程序使用Jdbc在同一JVM内打开数据库。此时执行效率会比较高，但因为数据在同一时间只能在一上JVM（和class loader）中，所以不允许其他进行同时进行访问。

### 服务器模式

类似于MySQL那种C/S模型，an application opens a database remotely using the JDBC or ODBC APIH2运行在一个独立的进程中，应用程序通过TCP协议与其远程通信。优势就是管理方便，而且可以部署在不同的机器上，使用包括集群等特性。

使用服务器模式时，使用起来与MySQL类似。此时，应用使用JDBC或ODBC远程地打开数据库连接。H2数据库与Java应用可在同一计算机JVM中，也可在不同计算机的不同Jvm中，不同的应用可同时访问同一数据库。

### 混合模式

综合以上两种情况，由应用程序首先启动H2，这时对于应用来说H2工作在嵌入式模式，同时H2启动一个服务（运行不同的进行或JVM），监听TCP某个端口，等待远程连接，这就是服务器模式，便于管理维护。

通常来说，混合模式比较实用。要使用混合模式，只需要在JDBC URL中添加连接参数AUTO\_SERVER=TRUE来建立JDBC连接，就可以开启混合模式。

## JDBC

### driver class

org.h2.Driver

### URL

H2数据库通过不同的JDBC URL可以支持多个数据库连接模式和连接配置。另外，JDBC URL字符串是不区分大小写的。

|  |  |
| --- | --- |
| Topic | URL Format and Examples |
| [Embedded (local) connection](file:///D:\Study\H2\h2\docs\html\features.html#embedded_databases) | jdbc:h2:[file:][<path>]<databaseName> jdbc:h2:~/test jdbc:h2:file:/data/sample jdbc:h2:file:C:/data/sample (Windows only) |
| [In-memory (private)](file:///D:\Study\H2\h2\docs\html\features.html#in_memory_databases) | jdbc:h2:mem: |
| [In-memory (named)](file:///D:\Study\H2\h2\docs\html\features.html#in_memory_databases) | jdbc:h2:mem:<databaseName> jdbc:h2:mem:test\_mem |
| [Server mode (remote connections) using TCP/IP](file:///D:\Study\H2\h2\docs\html\tutorial.html#using_server) | jdbc:h2:tcp://<server>[:<port>]/[<path>]<databaseName> jdbc:h2:tcp://localhost/~/test jdbc:h2:tcp://dbserv:8084/~/sample jdbc:h2:tcp://localhost/mem:test |
| [Server mode (remote connections) using SSL/TLS](file:///D:\Study\H2\h2\docs\html\advanced.html#ssl_tls_connections) | jdbc:h2:ssl://<server>[:<port>]/<databaseName> jdbc:h2:ssl://localhost:8085/~/sample; |
| [Using encrypted files](file:///D:\Study\H2\h2\docs\html\features.html#file_encryption) | jdbc:h2:<url>;CIPHER=[AES|XTEA] jdbc:h2:ssl://localhost/~/test;CIPHER=AES jdbc:h2:file:~/secure;CIPHER=XTEA |
| [File locking methods](file:///D:\Study\H2\h2\docs\html\features.html#database_file_locking) | jdbc:h2:<url>;FILE\_LOCK={FILE|SOCKET|NO} jdbc:h2:file:~/private;CIPHER=XTEA;FILE\_LOCK=SOCKET |
| [Only open if it already exists](file:///D:\Study\H2\h2\docs\html\features.html#database_only_if_exists) | jdbc:h2:<url>;IFEXISTS=TRUE jdbc:h2:file:~/sample;IFEXISTS=TRUE |
| [Don't close the database when the VM exits](file:///D:\Study\H2\h2\docs\html\features.html#do_not_close_on_exit) | jdbc:h2:<url>;DB\_CLOSE\_ON\_EXIT=FALSE |
| [Execute SQL on connection](file:///D:\Study\H2\h2\docs\html\features.html#execute_sql_on_connection) | jdbc:h2:<url>;INIT=RUNSCRIPT FROM '~/create.sql' jdbc:h2:file:~/sample;INIT=RUNSCRIPT FROM '~/create.sql'\;RUNSCRIPT FROM '~/populate.sql' |
| [User name and/or password](file:///D:\Study\H2\h2\docs\html\advanced.html#passwords) | jdbc:h2:<url>[;USER=<username>][;PASSWORD=<value>] jdbc:h2:file:~/sample;USER=sa;PASSWORD=123 |
| [Debug trace settings](file:///D:\Study\H2\h2\docs\html\features.html#trace_options) | jdbc:h2:<url>;TRACE\_LEVEL\_FILE=<level 0..3> jdbc:h2:file:~/sample;TRACE\_LEVEL\_FILE=3 |
| [Ignore unknown settings](file:///D:\Study\H2\h2\docs\html\features.html#ignore_unknown_settings) | jdbc:h2:<url>;IGNORE\_UNKNOWN\_SETTINGS=TRUE |
| [Custom file access mode](file:///D:\Study\H2\h2\docs\html\features.html#custom_access_mode) | jdbc:h2:<url>;ACCESS\_MODE\_DATA=rws |
| [Database in a zip file](file:///D:\Study\H2\h2\docs\html\features.html#database_in_zip) | jdbc:h2:zip:<zipFileName>!/<databaseName> jdbc:h2:zip:~/db.zip!/test |
| [Compatibility mode](file:///D:\Study\H2\h2\docs\html\features.html#compatibility) | jdbc:h2:<url>;MODE=<databaseType> jdbc:h2:~/test;MODE=MYSQL |
| [Auto-reconnect](file:///D:\Study\H2\h2\docs\html\features.html#auto_reconnect) | jdbc:h2:<url>;AUTO\_RECONNECT=TRUE jdbc:h2:tcp://localhost/~/test;AUTO\_RECONNECT=TRUE |
| [Automatic mixed mode](file:///D:\Study\H2\h2\docs\html\features.html#auto_mixed_mode) | jdbc:h2:<url>;AUTO\_SERVER=TRUE jdbc:h2:~/test;AUTO\_SERVER=TRUE |
| [Page size](file:///D:\Study\H2\h2\docs\html\features.html#page_size) | jdbc:h2:<url>;PAGE\_SIZE=512 |
| Changing other settings | jdbc:h2:<url>;<setting>=<value>[;<setting>=<value>...] jdbc:h2:file:~/sample;TRACE\_LEVEL\_SYSTEM\_OUT=3 |

## 数据库管理

组成一个H2的完整数库，包括RDBMS和DB，仅需要两个文件即可：

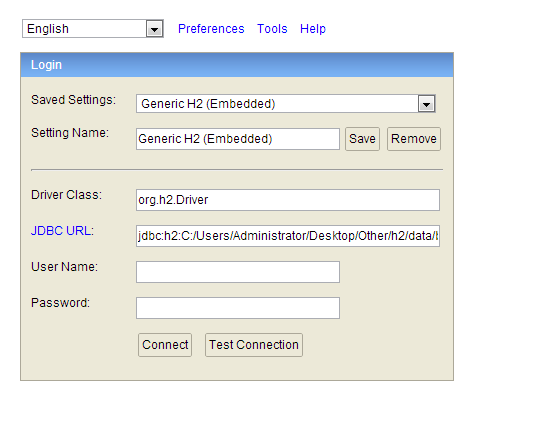
1 H2的驱动文件，也作为RDBMS本身：h2-x.x.xxx.jar,’x’表示版本。

2 数据库文件：xxx.h2.db，’xxx’表示数据库名。

需要直接管理H2数据时（非通过应用程序），打开命令窗口输入：

java -jar h2-x.x.xxx.jar

将自动打开浏览器，进行H2的管理应用程序登录界面。



## 元数据和设置

所有表的元数据和当前设置信息都存在INFORMATION\_SCHEMA模式下的系统表中。

|  |  |
| --- | --- |
| Table | Columns |
| CATALOGS | CATALOG\_NAME |
| COLLATIONS | NAME, KEY |
| COLUMNS | TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, COLUMN\_NAME, ORDINAL\_POSITION, COLUMN\_DEFAULT, IS\_NULLABLE, DATA\_TYPE, CHARACTER\_MAXIMUM\_LENGTH, CHARACTER\_OCTET\_LENGTH, NUMERIC\_PRECISION, NUMERIC\_PRECISION\_RADIX, NUMERIC\_SCALE, CHARACTER\_SET\_NAME, COLLATION\_NAME, TYPE\_NAME, NULLABLE, IS\_COMPUTED, SELECTIVITY, CHECK\_CONSTRAINT, SEQUENCE\_NAME, REMARKS, SOURCE\_DATA\_TYPE |
| COLUMN\_PRIVILEGES | GRANTOR, GRANTEE, TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, COLUMN\_NAME, PRIVILEGE\_TYPE, IS\_GRANTABLE |
| CONSTANTS | CONSTANT\_CATALOG, CONSTANT\_SCHEMA, CONSTANT\_NAME, DATA\_TYPE, REMARKS, SQL, ID |
| CONSTRAINTS | CONSTRAINT\_CATALOG, CONSTRAINT\_SCHEMA, CONSTRAINT\_NAME, CONSTRAINT\_TYPE, TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, UNIQUE\_INDEX\_NAME, CHECK\_EXPRESSION, COLUMN\_LIST, REMARKS, SQL, ID |
| CROSS\_REFERENCES | PKTABLE\_CATALOG, PKTABLE\_SCHEMA, PKTABLE\_NAME, PKCOLUMN\_NAME, FKTABLE\_CATALOG, FKTABLE\_SCHEMA, FKTABLE\_NAME, FKCOLUMN\_NAME, ORDINAL\_POSITION, UPDATE\_RULE, DELETE\_RULE, FK\_NAME, PK\_NAME, DEFERRABILITY |
| DOMAINS | DOMAIN\_CATALOG, DOMAIN\_SCHEMA, DOMAIN\_NAME, COLUMN\_DEFAULT, IS\_NULLABLE, DATA\_TYPE, PRECISION, SCALE, TYPE\_NAME, SELECTIVITY, CHECK\_CONSTRAINT, REMARKS, SQL, ID |
| FUNCTION\_ALIASES | ALIAS\_CATALOG, ALIAS\_SCHEMA, ALIAS\_NAME, JAVA\_CLASS, JAVA\_METHOD, DATA\_TYPE, TYPE\_NAME, COLUMN\_COUNT, RETURNS\_RESULT, REMARKS, ID, SOURCE |
| FUNCTION\_COLUMNS | ALIAS\_CATALOG, ALIAS\_SCHEMA, ALIAS\_NAME, JAVA\_CLASS, JAVA\_METHOD, COLUMN\_COUNT, POS, COLUMN\_NAME, DATA\_TYPE, TYPE\_NAME, PRECISION, SCALE, RADIX, NULLABLE, COLUMN\_TYPE, REMARKS, COLUMN\_DEFAULT |
| HELP | ID, SECTION, TOPIC, SYNTAX, TEXT |
| INDEXES | TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, NON\_UNIQUE, INDEX\_NAME, ORDINAL\_POSITION, COLUMN\_NAME, CARDINALITY, PRIMARY\_KEY, INDEX\_TYPE\_NAME, IS\_GENERATED, INDEX\_TYPE, ASC\_OR\_DESC, PAGES, FILTER\_CONDITION, REMARKS, SQL, ID, SORT\_TYPE, CONSTRAINT\_NAME, INDEX\_CLASS |
| IN\_DOUBT | TRANSACTION, STATE |
| LOCKS | TABLE\_SCHEMA, TABLE\_NAME, SESSION\_ID, LOCK\_TYPE |
| RIGHTS | GRANTEE, GRANTEETYPE, GRANTEDROLE, RIGHTS, TABLE\_SCHEMA, TABLE\_NAME, ID |
| ROLES | NAME, REMARKS, ID |
| SCHEMATA | CATALOG\_NAME, SCHEMA\_NAME, SCHEMA\_OWNER, DEFAULT\_CHARACTER\_SET\_NAME, DEFAULT\_COLLATION\_NAME, IS\_DEFAULT, REMARKS, ID |
| SEQUENCES | SEQUENCE\_CATALOG, SEQUENCE\_SCHEMA, SEQUENCE\_NAME, CURRENT\_VALUE, INCREMENT, IS\_GENERATED, REMARKS, CACHE, ID |
| SESSIONS | ID, USER\_NAME, SESSION\_START, STATEMENT, STATEMENT\_START |
| SESSION\_STATE | KEY, SQL |
| SETTINGS | NAME, VALUE |
| TABLES | TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, TABLE\_TYPE, STORAGE\_TYPE, SQL, REMARKS, LAST\_MODIFICATION, ID, TYPE\_NAME, TABLE\_CLASS |
| TABLE\_PRIVILEGES | GRANTOR, GRANTEE, TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, PRIVILEGE\_TYPE, IS\_GRANTABLE |
| TABLE\_TYPES | TYPE |
| TRIGGERS | TRIGGER\_CATALOG, TRIGGER\_SCHEMA, TRIGGER\_NAME, TRIGGER\_TYPE, TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, BEFORE, JAVA\_CLASS, QUEUE\_SIZE, NO\_WAIT, REMARKS, SQL, ID |
| TYPE\_INFO | TYPE\_NAME, DATA\_TYPE, PRECISION, PREFIX, SUFFIX, PARAMS, AUTO\_INCREMENT, MINIMUM\_SCALE, MAXIMUM\_SCALE, RADIX, POS, CASE\_SENSITIVE, NULLABLE, SEARCHABLE |
| USERS | NAME, ADMIN, REMARKS, ID |
| VIEWS | TABLE\_CATALOG, TABLE\_SCHEMA, TABLE\_NAME, VIEW\_DEFINITION, CHECK\_OPTION, IS\_UPDATABLE, STATUS, REMARKS, ID |

## 常见问题

### 第一次打开数据库是的用户名与密码

第一次打开数据库时，如果使用了用户名与密码，那么在创建数据库时，同时使用提供的用户名和密码创建数据库用户。并在下次打开数据库时要求提供正确的用户名和密码作为验证信息。

**protected** **static** Connection getH2Connect() {

String cls = "org.h2.Driver";

String url = "jdbc:h2:C:/Users/Administrator/Desktop/h2 h2/bpmx";

System.*out*.println(url);

String username = "sa";

String password = "";

**return** *getConnection*(cls, url);

}

# BPMX的H2数据库支持

## 数据连接参数设置

app.properies文件

jdbc.dbType=h2

jdbc.driverClassName=org.h2.Driver

jdbc.url=jdbc:h2:C:/Users/Administrator/Desktop/Other/h2/data/bpmx130202;SCHEMA=BPMX;AUTO\_SERVER=TRUE

jdbc.username=

jdbc.password=

## 添加MyBatis H2分页插件

configuration.xml文件

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE configuration

PUBLIC "-//ibatis.apache.org//DTD Config 3.0//EN"

"http://ibatis.apache.org/dtd/ibatis-3-config.dtd">

<configuration>

<settings>

<setting name=*"cacheEnabled"* value=*"true"*/>

</settings>

<plugins>

<plugin interceptor=*"com.hotent.core.mybatis.OffsetLimitInterceptor"*>

<property name=*"oracle"* value=*"com.hotent.core.mybatis.dialect.OracleDialect"*/>

<property name=*"mssql"* value=*"com.hotent.core.mybatis.dialect.SQLServer2005Dialect"*/>

<property name=*"mysql"* value=*"com.hotent.core.mybatis.dialect.MySQLDialect"*/>

<property name=*"db2"* value=*"com.hotent.core.mybatis.dialect.DB2Dialect"*/>

<property name=*"h2"* value=*"com.hotent.core.mybatis.dialect.H2Dialect"*/>

</plugin>

</plugins>

</configuration>