

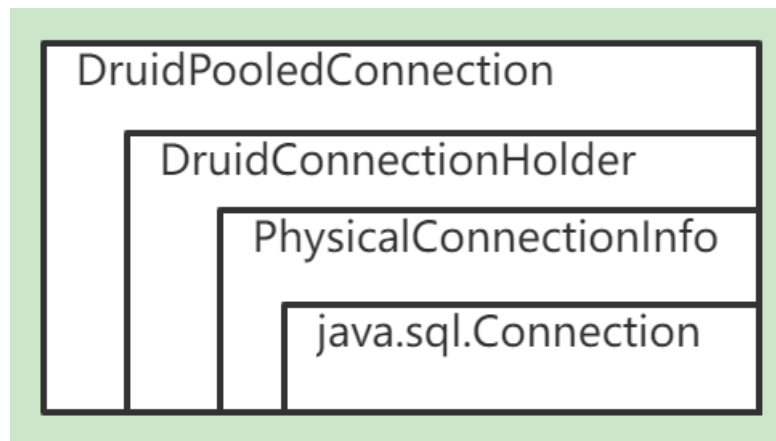
研究问题

为什么要有 `DruidConnectionHolder`？

完成对“池化技术”的部分了解后，选择了研究 `DruidConnectionHolder` 的业务意义，看能否从中学习到一些设计模式或者“如何扩展、增强”等知识。

因此在研究holder的前，我划分好要了解的内容：**1.**类的组成结构（指的是所在包、接口、继承等）。**2.**类本身有哪些变量？**3.**该类与哪些类相关联？

带着以上三个内容，我从前面章节中提到的“connection的组成”开始研究，即下图所示结构：

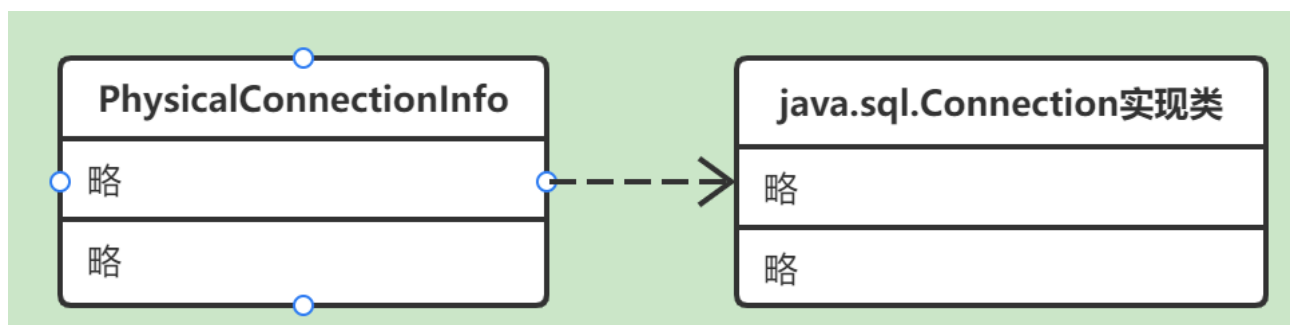


java.sql.Connection

JDBC相关接口，具体内容不展开讨论

PhysicalConnectionInfo

`PhysicalConnectionInfo` 中持有 `java.sql.Connection` 实现类的实例，关系如下图



成员变量

这里指出我所清楚的几个变量，`connectStartNanos`、`connectedNanos`、`initedNanos`、`validatedNanos`。这四个变量分别的含义是“建立连接前的系统当前时间”、“建立连接后的系统当前时间”、“初始化连接后的系统当前时间”、“检验连接后的系统当前时间”。形如用 `connectedNanos` 减去 `connectStartNanos`，便可以知道与数据库建立连接时所消耗的时间。其他变量同理。具体代码在 `DruidAbstractDataSource.createPhysicalConnection()` 中。

方法

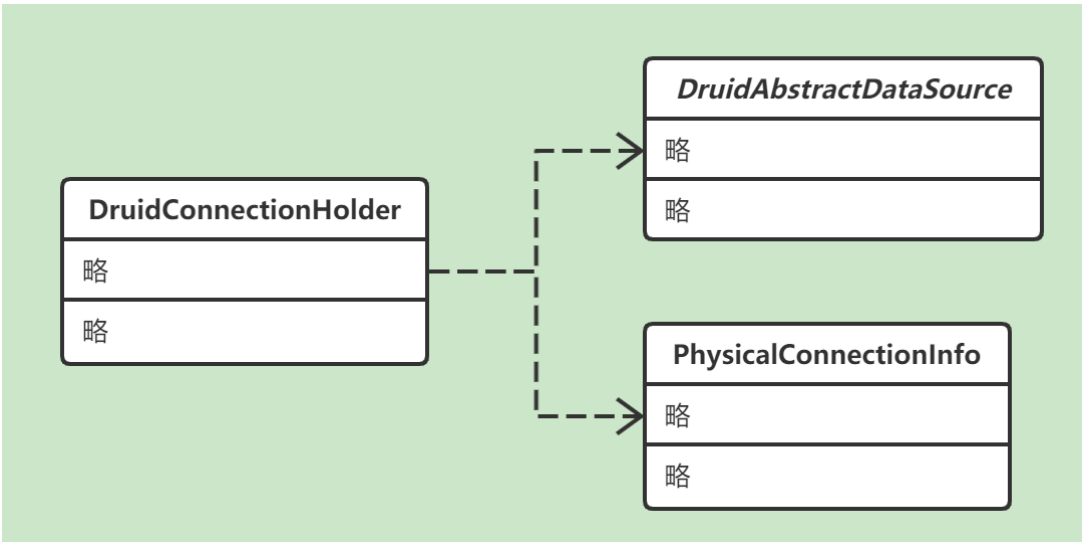
`DruidAbstractDataSource.initPhysicalConnection(Connection, Map<String, Object>, Map<String, Object>)` 和 `DruidAbstractDataSource.validateConnection(Connection)`。若在创建连接池时配置了创建物理连接的初始化方法和检验业务，这两个方法分别会执行对用的业务。

意义

不难看出 `PhysicalConnectionInfo` 存在的目的是扩展JDBC的 `Connection` 的功能。通过关联 `java.sql.Connection` 实现类，增加了“检验物理连接”、“初始化物理连接”的业务方法。因此我对 `PhysicalConnectionInfo` 的理解是：“存放与物理连接相关的内容”。

DruidConnectionHolder

holder持有 `PhysicalConnectionInfo` 和 `DruidAbstractDataSource` 的实例引用，关系如下图



成员变量

目前了解到的有 `lastActiveTimeMillis`、`lastKeepTimeMillis`、`statementPool`。这三个变量分别的作用是“最后活跃时间”、“最后保活时间（存疑）”、“当前连接缓存的执行语句”

方法

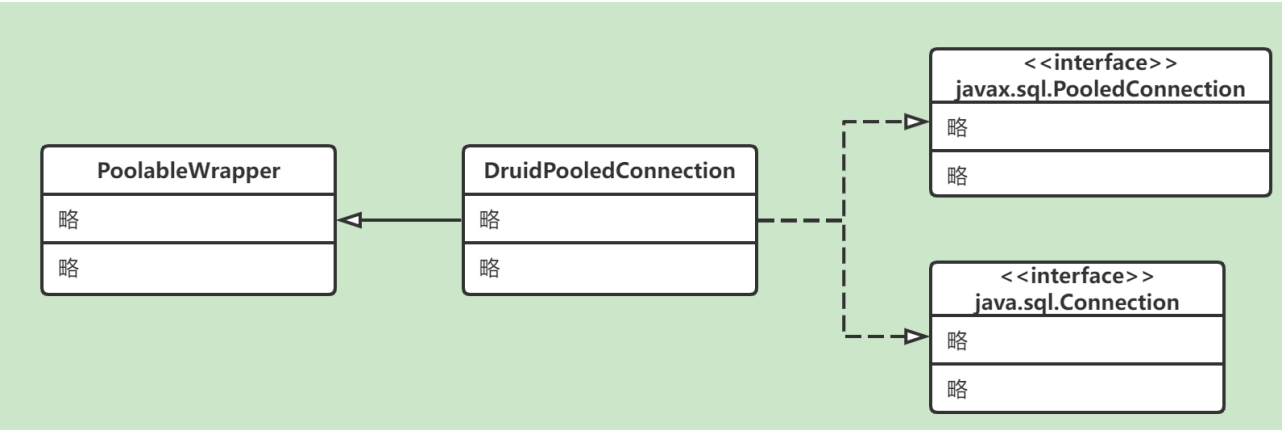
通过了解`toString()`展现的消息来初步了解上面提到的成员变量。

意义

其实在涉及到的成员变量中，就能够看出`holder`主要是负责保存跟执行SQL相关的信息。换句话说来说，`holder`专门用于跟踪应用程序业务信息。

DruidPooledConnection

先给出这个类的关系图



信息划分图

