

一种新的数据持久层设计与实现

程文波^{1,2}, 卢涵宇³, 陈劲松^{1,2}

(1 宜宾学院 物理与电子工程学院, 四川 宜宾 644000; 2 宜宾机械电子工程技术研究中心, 四川 宜宾 644000;
3 贵州大学 计算机与信息工程系, 贵州 贵阳 550003)

摘 要: 针对目前数据库模型改动将导致业务层和数据库层的大量数据修改的不足这一问题, 提出建立新的框架即双重数据持久层框架的设计方法, 该方法联合运用 iBATIS 数据持久层框架和数据字典技术来实现。当数据库模型需要变更时, 可较方便地通过操作数据字典, 来实现对数据持久层和数据库模型的同步维护, 较大程度提高了工作效率, 可移植性增强。

关键词: 数据字典技术; iBATIS 框架; 双重数据持久层框架; 低耦合

中图分类号: TP311.52

文献标识码: A

文章编号: 1000-7180(2011)07-0028-03

A New Design and Application of Data Persistence Layer

CHENG Wen-bo^{1,2}, LU Han-yu³, CHEN Jin-song^{1,2}

(1 School of Physics and Electronic Engineering, Yibin University, Yibin 644000, China;

2 Mechanical and Electronics Engineering Research Center, Yibin University, Yibin 644000, China;

3 Department of Computer and Information Engineering, Guizhou University, Guiyan 550003, China)

Abstract: According to the shortages of a large number of workload will be triggered in the business layer and database layer by reason of the database model changes, this paper puts forward a new framework design pattern which uses the iBATIS data persistence layer framework and the data dictionary technology, establishing jointly double persistence layer during the database system development. When the database model needs to change, it can easily to realize synchronous maintenance of the data persistence layer and the database model through the operation of the data dictionary. The method has high efficiency and good portability.

Key words: data dictionary technology; iBATIS framework; double data persistence layer framework; low coupling

1 引言

在数据持久层框架引入之前, 系统开发基本采用传统的单层或双层设计模式, 应用程序与其使用和操纵的数据源并未分离, 业务逻辑接口的开发维护受到极大束缚, 如何将复杂的数据持久化, 寻求“高内聚、低耦合”的架构已经成为系统开发领域的热门研究课题^[1-5]。

数据持久化是一种对象化服务, 即通过操作对象的方法访问数据库, 而不必在程序中书写 SQL (结构化查询) 语句。目前, 已经存在许多开源的使用

O/R Mapping (对象关系数据库映射) 组件来解决数据持久化问题的框架, 例如 Hibernate 和 iBATIS 持久层框架。基于这些持久层开发机制的框架均存在相似的问题, 即维护非常繁琐且工作量巨大。

针对数据库模型改动会导致业务接口层和数据库层的大量修改工作的问题, 提出在数据库系统开发过程中建立双重持久层的设计方法, 即在数据持久层框架的基础上, 建立数据字典持久层, 同时建立关于数据字典的业务层, 通过对数据字典的操作, 实现对数据库模型和数据持久层的同步维护, 可以较大程度提高维护效率, 增强可移植性、可扩展性和可维护性。

收稿日期: 2010-10-10; 修回日期: 2010-11-12

基金项目: 国家“八六三”计划项目(2006AA06Z110)

2 iBATIS 架构

iBATIS 一词来源于“internet”和“abatis”的组合,是由 Clinton Begin 在 2001 年发起的开源项目.最初侧重于密码软件的开发,现在是一个基于 Java 的持久层框架,它融合了多种操作关系数据的理念和方法,是一个混合型解决方案(Hybrid Solution)^[6]. iBATIS 架构图如图 1 所示.

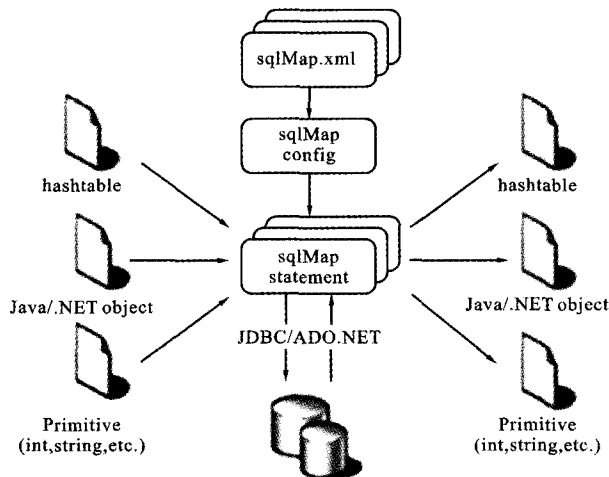


图 1 iBATIS 架构

相对 Hibernate 和 Apache OJB 等“一站式”O/R Mapping 解决方案而言, iBATIS 是一种“半自动化”的 O/R mapping 实现. 即就是说, iBATIS 并不为程序员在运行期自动生成 SQL 执行, 而是由程序员根据具体的 SQL 需要来编写, 然后通过映射配置文件, 将 SQL 所需的参数, 以及返回的结果字段映射到指定的 POJO(简单的 Java 对象). 可见, 虽然通过 SQL 映射器(SQL Mapper)保证了在对象和数据库之间数据传输时两者与映射器之间的相对独立性, 但若数据库模型发生变动时, 仍需要对映射关系进行修改. 相对于将数据库直接映射为 Java 对象, 并完全隐藏 SQL 的 O/R M 工具 Hibernate 而言, 它是以牺牲 SQL 开发的工作量和数据库的移植性来换取系统设计的自由度.

3 数据字典

数据字典是关于数据信息的集合, 是对数据流图中包含的所有元素定义的集合, 通常包括数据项、数据结构、数据流、数据存储和处理五部分. 数据字典旨在建立一组严密一致的定义, 负责操作模式和子模式的内容、文件间的联系、数据项的长度、类型、用户标识符、口令、索引等等非用户数据信息, 以帮助改进分析员和用户的通信, 用户可以用 SQL 语句

访问数据库数据字典.

通过对数据字典技术的运用, 不仅可以实现对数据库本身的有效维护和操作, 同时也可以为系统开发建立标准和依据^[7]. 但从本质上讲, 数据字典技术依然采用的是传统的两层模式, 是在数据库的基础上建立了一个关于数据库中数据的数据库, 并不能真正的实现业务层与数据库层的独立. 基本数据字典如表 1 所示.

表 1 基本数据字典

名称	包含字段	关键字
数据表字典	表名; 表别名; 关键字; 父表名; 表格有效性规则等	表名
属性字段字典	字段名; 字段编码; 字段类型; 长度; 小数位; 单位; 允许空值; 默认值; 代码转换; 字段有效规则等	字段名
表格字段字典	表名; 字段名称; 字段编号; 是否使用; 用户可见性等	表名+字段名
索引信息字典	表名; 索引名; 索引表达式等	表名+索引名
查询检索方法字典	检索名; 检索编号; 数据表名; 字段名等	检索名

4 双重持久层设计

4.1 iBATIS 数据持久层分层设计

iBATIS 数据持久层是一个持久层框架, 持久层位于应用程序的业务逻辑层和数据库之间, 它隐藏了关于数据如何被存储和提取的全部细节, 这种分离策略保证了数据持久化时这些细节不会混入其它层.

持久层内部又分为抽象层、持久化框架和驱动程序或接口层. 抽象层为持久层提供一致且有意义的接口, 一旦合适的抽象层建立就绪, 整个持久化方法的改变就不再涉及该层, 也不会引起其它任何依赖层的改变; 持久层框架负责与数据库程序或接口交互, 提供存储、获取、更新、查找以及管理数据的方法, 其中的 SQL Maps 采用 XML 配置文件将 Java Bean 映射成 SQL 语句, 允许开发者在不使用 JDBC API 以及不耦合 Java 代码与 SQL 语句的情况下, 实现 Java 对象和 DBMS 的访问操作^[5]; 驱动程序或接口层负责与数据库进行数据交换. 典型的分层设计策略如图 2 所示.

4.2 融合数据字典技术的双重持久层分层设计

文中提出在数据持久层框架上融合数据字典技术, 即在图 2 的基础上, 增加数据字典持久层, 从而构建并实现了双重数据持久层框架. 改进持久层框架分层策略如图 3 所示.

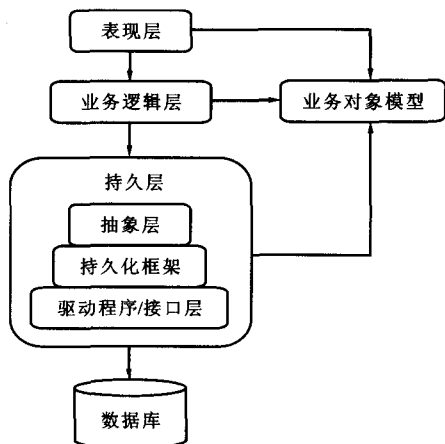


图2 Law of Demeter 典型分层策略

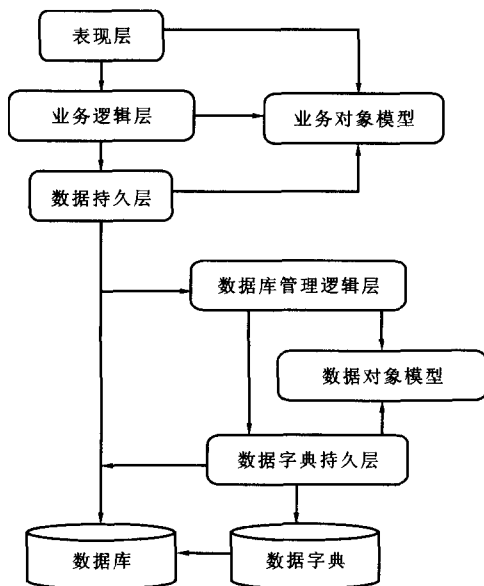


图3 改进持久层框架的分层策略

改进的数据持久层架构融合了 iBATIS 持久层框架的思想,通过建立数据表与数据持久层中的 XML 文件的映射对应关系,可以实现对数据表的修改. XML 文件由 Data Map 或 SQL Map 和 Data Access Object(数据访问对象 ADO)两部分构成,针对不同数据字典,其配置应有所区别.下面给出为数据表字段字典配置 XML 文件的伪代码:

```

<resultMaps>
<resultMap id="Select-TABLE_FIELD-Result"
class="TABLE_FIELD">
<result property="TableName" column="Ta-
bleName" />
<result property="FieldName" column="Field-
Name" />
.....
</resultMap>
</resultMaps>

```

```

<statements>

```

```

<!--编写对数据表操作的代码-->

```

```

<delete id="TableFiled_Delete" parameterClass="
TABLE_FIELD">

```

```

delete from TABLE_FIELD where TableName = #
TableName # and FieldName = # FieldName #

```

```

.....

```

```

</delete>

```

```

</statements>

```

在数据库系统开发过程中,若采用新的持久层框架方法进行设计,即在数据持久层的基础上,建立数据字典持久层和关于数据字典的业务层,即可方便实现利用数据字典技术对数据库的维护,来完成对数据库模型和数据持久层的操作.

5 结束语

在系统开发过程中采用数据持久层框架,并融入数据字典技术,实现了双重数据持久层框架设计的思想,不仅可以实现业务层与数据库层的相互独立,而且降低了对数据持久层框架维护的难度.通过在某矿井下人员定位与安全监测系统中的应用表明,基于新的数据持久层框架较好地解决了以往对数据库和数据持久层维护繁琐的问题,且具有较高效率和较强的移植性和扩展性.

参考文献:

- [1] 于洪志. 民族文献信息库中基于 iBATIS 的持久层应用[J]. 微计算机信息, 2010, 26(3): 21-22, 67.
- [2] 李澎林, 朱国清, 吴斌. 基于 iBATIS SQL Map 的数据持久层实现应用研究[J]. 浙江工业大学学报, 2008, 2: 72-75.
- [3] 吴卫平, 王丽芳, 蒋泽军. 基于 ORM 的数据持久层框架研究[J]. 微电子学与计算机, 2008, 25(7): 188-190.
- [4] 陈景霞, 陈桦, 张鹏伟. 利用 XML 扩展基于 MVC 模式的 Web 应用框架的研究[J]. 微电子学与计算机, 2005, 22(4): 38-41.
- [5] 张俐. 基于 iBATIS 和设计模式的数据持久层的应用[J]. 安徽大学学报: 自然科学版, 2010, 34(3): 43-48.
- [6] Clinton B, Brandon G, Larry M. iBATIS 实战[M]. 叶俊, 译. 北京: 人民邮电出版社, 2008.
- [7] 韩志军, 汪新庆, 吴冲龙. 数据库系统数据字典的设计与实现[J]. 微机发展, 1999, 9(2): 30-32.

作者简介:

程文波 男, (1979-), 博士, 讲师. 研究方向为现代信号与信息处理.