



(12) 发明专利申请

(10) 申请公布号 CN 105512042 A

(43) 申请公布日 2016. 04. 20

(21) 申请号 201510981715. 4

(22) 申请日 2015. 12. 22

(71) 申请人 广东金赋信息科技有限公司

地址 528200 广东省佛山市南海区桂城深海路 17 号瀚天科技城 A 区 5 号楼三楼

(72) 发明人 区洪杨 张伟昌

(74) 专利代理机构 北京孚睿湾知识产权代理事

务所(普通合伙) 11474

代理人 李丹丹

(51) Int. Cl.

G06F 11/36(2006. 01)

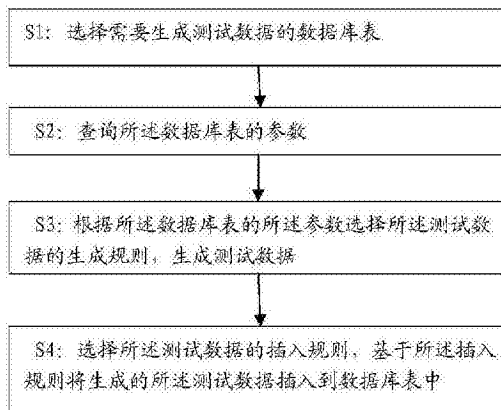
权利要求书2页 说明书6页 附图3页

(54) 发明名称

一种数据库的测试数据的自动生成方法、装置及测试系统

(57) 摘要

本发明提供了一种数据库的测试数据的自动生成方法、装置及测试系统,所述方法包括:选择需要生成测试数据的数据库表;查询所述数据库表的参数;根据所述数据库表的所述参数选择所述测试数据的生成规则,生成测试数据;选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到数据库表中。本发明解决了字段关联外键识别的技术难题,提供了多种灵活的测试数据生成规则,提出智能化插入规则,使数据插入速度大幅提高,提供了用户界面,易于操作。



1. 一种数据库的测试数据的自动生成方法,其特征在于,包括以下步骤:
 - S1、选择需要生成测试数据的数据库表;
 - S2、查询所述数据库表的参数;
 - S3、根据所述数据库表的所述参数来选择所述测试数据的生成规则,生成测试数据;以及
 - S4、选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到所述数据库表中。
2. 根据权利要求1所述的自动生成方法,其特征在于,所述参数包括:所述数据库表的字段名称、字段类型、字段是否唯一、字段是否可为空、以及字段是否为关联外键。
3. 根据权利要求2所述的自动生成方法,其特征在于,通过编程语句获取所述字段名称、字段类型、字段是否唯一以及字段是否可为空,其中,所述编程语句包括java语句。
4. 根据权利要求2所述的自动生成方法,其特征在于,所述步骤S2还包括获取所述字段是否为关联外键的步骤,用于获取所述字段是否为关联外键,包括:
 - S21、获取该数据库表的建表语句;以及
 - S22、解析所述建表语句,确定所述建表语句中是否存在外键,如果存在,所述字段为关联外键,并基于所述建表语句确定与所述外键相关联的其他数据库表。
5. 根据权利要求1所述的自动生成方法,其特征在于,所述测试数据的所述生成规则包括:根据随机数生成、通过输入生成、根据配置文件生成、根据数据库中已有的数据库表中的数据生成,其中,生成每个所述测试数据的所述生成规则相同或者不同。
6. 根据权利要求1-5之一所述的自动生成方法,其特征在于,所述步骤S4还包括生成所述插入规则的步骤,用于生成所述插入规则,包括:
 - S41、生成存储有插入规则的插入配置文件;
 - S42、初始化所述插入配置文件,在所述插入配置文件中初始化每条插入数据的初始插入量;
 - S43、确定待插入的字段的数量、大小和类型,使用所述初始插入量进行数据的插入,并确定初始插入速度;以及
 - S44、调整所述初始插入量,并记录相应的插入速度,进行多次调整以确定最大插入速度及相应的数据插入量,使用与所述最大插入速度相应的所述数据插入量来更新所述插入配置文件中的所述初始插入速度,生成所述插入配置文件。
7. 一种数据库的测试数据的自动生成装置,其特征在于,包括:
 - 选择模块,用于选择需要生成测试数据的数据库表;
 - 查询模块,用于查询所述数据库表的参数;
 - 测试数据生成模块,用于根据所述数据库表的所述参数来选择所述测试数据的生成规则,生成测试数据;以及
 - 测试数据插入模块,用于选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到所述数据库表中。
8. 根据权利要求7所述的自动生成装置,其特征在于,所述参数包括:所述数据库表的字段名称、字段类型、字段是否唯一、字段是否可为空、以及字段是否为关联外键。
9. 根据权利要求8所述的自动生成装置,其特征在于,通过编程语句获取所述字段名

称、字段类型、字段是否唯一以及字段是否可为空,其中,所述编程语句包括java语句。

10.根据权利要求8所述的自动生成装置,其特征在于,所述查询模块包括关联外键获取模块,用于获取所述字段是否为关联外键,所述关联外键获取模块包括:

获取模块,用于获取该数据库表的建表语句;以及

解析模块,用于解析所述建表语句,确定所述建表语句中是否存在外键,如果存在,所述字段为关联外键,并基于所述建表语句确定与所述外键相关联的其他数据库表。

11.根据权利要求7所述的装置,其特征在于,所述测试数据的所述生成规则包括:根据随机数生成、通过输入生成、根据配置文件生成、根据数据库中已有的数据库表中的数据生成,其中,生成每个所述测试数据的所述生成规则相同或者不同。

12.根据权利要求7-11之一所述的装置,其特征在于,所述测试数据插入模块还包括插入规则生成模块,用于生成所述插入规则,所述插入规则生成模块包括:

插入配置文件生成模块,用于生成存储有插入规则的插入配置文件;

初始化模块,用于初始化所述插入配置文件,在所述插入配置文件中初始化每条插入数据的初始插入量;

确定模块,用于确定待插入的字段的数量、大小和类型,使用所述初始插入量进行数据的插入,并确定初始插入速度;以及

调整模块,用于调整所述初始插入量,并记录相应的插入速度,进行多次调整以确定最大插入速度及相应的数据插入量,使用与所述最大插入速度相应的所述数据插入量来更新所述插入配置文件中的所述初始插入速度,生成所述插入配置文件。

13.一种数据库测试系统,包括用户界面和数据库,其特征在于:还包括如权利要求7-12之一所述的数据库的测试数据的自动生成装置;

其中,用户通过所述用户界面操作所述测试数据的自动生成装置,生成所述测试数据并插入到所述数据库中。

一种数据库的测试数据的自动生成方法、装置及测试系统

技术领域

[0001] 本发明涉及测试数据的自动生成方法、装置及测试系统,特别是涉及一种在数据库中自动地生成测试数据的自动生成方法、装置及测试系统。

背景技术

[0002] 在软件测试中非常重要的一个工作就是测试数据的准备和维护,制作测试数据是软件测试执行前重要和必要的一个环节,而这个工作恰恰是繁琐、费时、重复而极容易出错的,所以找到一种通用的、可视化的方法来生成测试数据是非常有意义和重要的事。尤其在数据库自动化测试和性能测试领域,需要大量测试数据来进行测试,测试数据的需求量会是十万级、百万级甚至千万级,并且需要的数据一般不会是固化单一数据,而是有一定规则的数据,数据之间还会存在着各种的关系,因此测试数据的准备工作需要耗费大量的人力和时间。而且,生成和插入的测试数据量越大,测试数据规则就越复杂,所耗费的人力就越多,现有的测试工具数据生成规则单一,基本上靠人工来生成和维护,生成的数据可靠性差,数据之间的整合性差,给测试工作带来了很大的困扰。

[0003] 特别是在现有的数据库测试数据生成时,无法对数据库表中的字段是否关联外键进行判断,导致生成的测试数据不准确,无法检验数据库的表之间的关联数据是否正确。

[0004] 现有技术中,将生成的测试数据通过java的批处理操作插入到数据库中,由于java的批处理插入数据库会涉及效率问题,字段数量、大小、类型等不同,会直接影响到插入效率,而批量插入量不同也会影响到插入效率。

[0005] 现有的数据库测试无法进行可视操作,操作效率低下,准确率低下。

发明内容

[0006] 本发明的目的就是为了弥补上述现有技术中的缺陷,提供了一种数据库的测试数据的自动生成方法,该方法包括以下步骤:

[0007] S1、选择需要生成测试数据的数据库表;

[0008] S2、查询所述数据库表的参数;

[0009] S3、根据所述数据库表的所述参数来选择所述测试数据的生成规则,生成测试数据;以及

[0010] S4、选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到所述数据库表中。

[0011] 更进一步地,所述参数包括:所述数据库表的字段名称、字段类型、字段是否唯一、字段是否可为空、以及字段是否为关联外键。

[0012] 更进一步地,通过编程语句获取所述字段名称、字段类型、字段是否唯一以及字段是否可为空,其中,所述编程语句包括java语句。

[0013] 更进一步地,所述步骤S2还包括获取所述字段是否为关联外键的步骤,用于获取所述字段是否为关联外键,包括:

- [0014] S21、获取该数据库表的建表语句;以及
- [0015] S22、解析所述建表语句,确定所述建表语句中是否存在外键,如果存在,所述字段为关联外键,并基于所述建表语句确定与所述外键相关联的其他数据库表。
- [0016] 更进一步地,所述测试数据的所述生成规则包括:根据随机数生成、通过输入生成、根据配置文件生成、根据数据库中已有的数据库表中的数据生成,其中,生成每个所述测试数据的所述生成规则相同或者不同。
- [0017] 更进一步地,所述步骤S4还包括生成所述插入规则的步骤,用于生成所述插入规则,包括:
- [0018] S41、生成存储有插入规则的插入配置文件;
- [0019] S42、初始化所述插入配置文件,在所述插入配置文件中初始化每条插入数据的初始插入量;
- [0020] S43、确定待插入的字段的数量、大小和类型,使用所述初始插入量进行数据的插入,并确定初始插入速度;以及
- [0021] S44、调整所述初始插入量,并记录相应的插入速度,进行多次调整以确定最大插入速度及相应的数据插入量,使用与所述最大插入速度相应的所述数据插入量来更新所述插入配置文件中的所述初始插入速度,生成所述插入配置文件。
- [0022] 本申请还提供了一种数据库的测试数据的自动生成装置,包括:
- [0023] 选择模块,用于选择需要生成测试数据的数据库表;
- [0024] 查询模块,用于查询所述数据库表的参数;
- [0025] 测试数据生成模块,用于根据所述数据库表的所述参数来选择所述测试数据的生成规则,生成测试数据;以及
- [0026] 测试数据插入模块,用于选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到所述数据库表中。
- [0027] 更进一步地,所述参数包括:所述数据库表的字段名称、字段类型、字段是否唯一、字段是否可为空、以及字段是否为关联外键。
- [0028] 更进一步地,通过编程语句获取所述字段名称、字段类型、字段是否唯一以及字段是否可为空,其中,所述编程语句包括java语句。
- [0029] 更进一步地,所述查询模块包括关联外键获取模块,用于获取所述字段是否为关联外键,所述关联外键获取模块包括:
- [0030] 获取模块,用于获取该数据库表的建表语句;以及
- [0031] 解析模块,用于解析所述建表语句,确定所述建表语句中是否存在外键,如果存在,所述字段为关联外键,并基于所述建表语句确定与所述外键相关联的其他数据库表。
- [0032] 更进一步地,所述测试数据的所述生成规则包括:根据随机数生成、通过输入生成、根据配置文件生成、根据数据库中已有的数据库表中的数据生成,其中,生成每个所述测试数据的所述生成规则相同或者不同。
- [0033] 更进一步地,所述测试数据插入模块还包括插入规则生成模块,用于生成所述插入规则,所述插入规则生成模块包括:
- [0034] 插入配置文件生成模块,用于生成存储有插入规则的插入配置文件;
- [0035] 初始化模块,用于初始化所述插入配置文件,在所述插入配置文件中初始化每条

插入数据的初始插入量；

[0036] 确定模块,用于确定待插入的字段的数量、大小和类型,使用所述初始插入量进行数据的插入,并确定初始插入速度;以及

[0037] 调整模块,用于调整所述初始插入量,并记录相应的插入速度,进行多次调整以确定最大插入速度及相应的数据插入量,使用与所述最大插入速度相应的所述数据插入量来更新所述插入配置文件中的所述初始插入速度,生成所述插入配置文件。

[0038] 本申请还提供了一种数据库测试系统,还包括所述数据库的测试数据的自动生成装置;

[0039] 其中,用户通过所述用户界面操作所述测试数据的自动生成装置,生成所述测试数据并插入到所述数据库中。

[0040] 本发明的技术效果在于:解决了字段关联外键识别难的技术难题,提供了多种灵活的测试数据生成规则,提出智能化插入规则,使数据插入速度大幅提高,并提供了用户界面,易于用户操作,能够大量且高速地生成符合要求的测试数据,提高了软件开发与测试的效率。

附图说明

[0041] 图1为本发明的数据库的测试数据的自动生成方法的流程图;

[0042] 图2为本发明的确定字段是否关联外键的流程图;

[0043] 图3为本发明的生成插入规则的流程图;

[0044] 图4为本发明的数据库的测试数据的自动生成装置的结构图;

[0045] 图5为本发明的查询模块的结构图;

[0046] 图6为本发明的测试数据插入模块的结构图;

[0047] 图7为本发明的数据库测试系统的示意图。

具体实施方式

[0048] 下面,参照附图来说明本发明涉及的实施方式,不过本发明不局限于此。以下说明的实施方式的结构要素能够适当地组合。此外,也存在不使用一部分结构要素的情况。

[0049] 数据库的测试数据的自动生成方法

[0050] 图1示意性地示出了本发明的数据库的测试数据的自动生成方法。该自动生成方法用于在软件开过与测试过程中,自动地在目标数据库中,依据一定的生成规则和插入规则来生成测试数据。具体而言,该方法包括以下步骤:S1、选择需要生成测试数据的数据库表;S2、查询所述数据库表的参数;S3、根据所述数据库表的所述参数来选择所述测试数据的生成规则,生成测试数据;以及S4、选择所述测试数据的插入规则,基于所述插入规则将生成的所述测试数据插入到所述数据库表中。

[0051] 下面,详细地说明本发明的数据库的测试数据的自动生成方法的详细步骤。

[0052] 数据库表以及字段

[0053] 数据库是以一定方式储存在一起、能为多个用户共享、具有尽可能小的冗余度、与应用程序彼此独立的数据集合。数据库由数据库表组成,其中,数据库表是数据库中存储数据的一种数据组织形式,具有多个字段,举例而言,用户需要借助数据库来保存一个班级中

所有学生的信息,该情况下,可以以班级为单位创建一个数据库,命名为“class”,这个数据库中包括一个学生基本信息表,命名为“stu”,一个是学生的科目考试成绩表,命名为“score”。学生基本信息表包括“姓名”、“性别”、“学号”这三个字段,每个字段都可以存储一个特定类型的信息,比如“张三”、“男”、“200110010”。科目考试成绩表与学生基本信息表相同,包括“学号”、“科目”、“成绩”这三个字段,可以存储例如“200110010”、“语文”、“80”。其中,每个字段都具有特定的属性,代表不同的含义。这里,在进行测试数据准备的时候,就需要知晓学生基本信息表与科目考试成绩表这两个表之间的关系以及每个字段的属性,换言之,就需要知晓待插入数据的插入规则。否则,准备的插入数据之间就不符合数据内在的逻辑结构,从而导致所准备的数据是无效的,或者至少是质量不高的数据。

[0054] 数据库表字段的参数

[0055] 为对数据库表进行操作,比如插入或者修改数据库表中的数据,需要获得其参数,参数一般包括:字段名称、字段类型、字段是否唯一、字段是否可为空、字段是否为关联外键。数据库表中字段属性唯一的就可作为主键,主键的要求是所存储的数据是唯一的就可以,可以是一个字段,也可以是组合字段。关联外键的含义是通过一个数据库表中的某个字段可以外部连接到其他的数据库表,如上述学生基本信息表和科目考试成绩表这两个表的“学号”字段,可以是关联外键。

[0056] 一般来说,通过编程语句可以获取字段名称、字段类型、字段是否唯一及字段是否可为空,编程语句可以是Java语句、C、C++、VC++等各种编程语言,本发明中对于编程语言没有限定,可以是任何的编程语言,只要能够实现对数据库表生成语句实现解析就可以。

[0057] 字段是否是关联外键的确定

[0058] 但是,现有的编程语言例如Java语句无法获取一个字段是否关联外键的信息,本申请提出了确定字段是否关联外键的方法,如图2所示,包括:S21、获取该数据库表的建表语句;S22、解析所述建表语句,确定所述建表语句中是否存在外键,即是否存在类似于“FOREIGN KEY”的语句,如果存在,相应字段为关联外键,并基于所述建表语句确定与所述外键相关联的其他数据库表。

[0059] 该步骤一般作为查询步骤的子步骤,当然也可以独立执行。

[0060] 测试数据的生成规则

[0061] 为灵活的生成测试数据,就需要对测试数据的生成规则进行定义,测试数据的生成规则是指按照什么方式来生成测试数据,本发明中,测试数据的生成规则包括但不限于根据随机数来生成测试数据、通过手工输入来生成测试数据、通过配置文件来生成测试数据、引用数据库中已有的数据表中的数据来生成测试数据等。其中,每一字段的数据可通过不同的方式生成,也可以通过相同的方式生成,在此并不限定于生成方式,可以根据具体的需求来定。

[0062] 测试数据的插入规则

[0063] 由于使用编程语言的批处理插入数据库会涉及效率问题,字段数量、大小、类型等不同,会直接影响到插入效率,而批量插入量不同也会影响到插入效率。因此,本发明能够通过程序智能选择去配置批量插入量来达到一个最优插入速度来插入到数据库。而智能选择批量插入量是通过配置智能选择插入规则文件去实现,由于插入规则是能够具体规则的配置,因此该智能选择规则可以通过不断的测试之后不断地去优化,直到寻找到最优

的规则文件和最快的插入速度。图3示出了生成所述插入规则的步骤,用于生成所述插入规则,包括:S41、生成存储插入规则的插入配置文件;S42、初始化所述插入配置文件,在所述插入配置文件中初始化每次插入数据的初始插入量;S43、确定待插入的字段的数量、大小和类型,使用所述初始插入量进行数据的插入,并确定初始插入速度;以及S44、调整所述初始插入量,并记录相应的插入速度,进行多次调整,以确定最大插入速度及相应的数据插入量,使用与最大插入速度相应的数据插入量,来更新插入配置文件中的初始插入速度,生成插入配置文件。

[0064] 这样通过智能的计算,针对不同的数据表、数据类型等,就能够确定出最合适的插入量,从而提高插入速度。

[0065] 本发明的上述方法可通过计算机程序实现,计算机至少具备CPU(Central Processing Unit:中央处理单元)、存储控制程序的ROM(Read Only Memory:只读存储器)、作为CPU的作业区域使用的RAM(Random Access Memory)以及非易失性存储器。非易失性存储器包括快闪存储器以及硬盘驱动器中的至少一个。所述计算机程序可以存储在存储器中,存储器可以为以下中任何一个:随机存储器、只读存储器、光或磁存储器,或者能够存储数字信息的任何合适存储器。所述存储器可以是系统存储器、可为系统访问的其他存储器或协同处理器的特定本地存储器。但是不限于此,也可以通过特定的物理电路来实现,物理电路包括集成电路、大规模集成电路、VLSI(超大规模集成电路)、FPGA等。

[0066] 根据本发明的数据库的测试数据的自动生成方法,解决了字段关联外键识别难的技术难题,能够提供多种灵活的测试数据生成规则,实现智能化插入,同时能够大量且高速地生成符合要求的测试数据,提高了软件开发与测试的效率。

[0067] 数据库的测试数据的自动生成装置

[0068] 下面,基于图4-图6来说明本发明提供的数据库的测试数据的自动生成装置。图4示出了本发明的数据库的测试数据的自动生成装置1,包括:选择模块11,用于选择需要生成测试数据的数据库表;查询模块12,用于查询数据库表的参数;测试数据生成模块13,用于根据数据库表的参数选择测试数据的生成规则,生成测试数据;以及测试数据插入模块14,用于选择测试数据的插入规则,基于插入规则将生成的测试数据插入到数据库表中。

[0069] 在查询模块12中设有关联外键获取模块121,如图5所示,其包括:获取模块1211,用于获取该数据库表的建表语句;解析模块1212,用于解析建表语句,确定建表语句中是否存在“FOREIGN KEY”,如果存在,所述字段为关联外键,并基于所述建表语句确定与所述外键相关联的另一数据库表。

[0070] 一般来说,关联外键获取模块121可设置于查询模块12中,当然,其可以是一个独立的模块。

[0071] 如图6所述,在测试数据插入模块14中包括插入规则生成模块141,其中包括:插入配置文件生成模块1411,用于生成存储插入规则的插入配置文件;初始化模块1412,用于初始化插入配置文件,在插入配置文件中初始化每条插入数据的初始插入量;确定模块1413,用于确定待插入的字段的数量、大小和类型,使用初始插入量进行数据的插入,并确定初始插入速度;调整模块1414,用于调整初始插入量,并记录相应的插入速度,经过多次调整,以确定最大插入速度及相应的数据插入量,使用与最大插入速度相应的所述数据插入量更新插入配置文件中的所述初始插入速度,生成插入配置文件。

[0072] 一般来说,插入规则生成模块141可设置于测试数据插入模块14中,当然,其可以是一个独立的模块。

[0073] 本发明的数据库的测试数据的自动生成装置1可以通过集成电路技术实现,如VLSI(超大规模集成电路)、FPGA等,也可以通过计算机程序模块实现,程序模块包括用来执行特定的任务或实施特定的抽象数据类型的例行程序、程序、构件、数据结构等等。而且,本领域技术人员将意识到,本发明方法可以通过其它计算机系统配置而被实践,包括单处理器或多处理器计算机系统、小型计算机、大型计算机、以及个人计算机、手持计算设备、基于微处理器的或可编程的消费者电子设备等等,它们中的每个设备可以操作地被耦合到一个或多个相关联的设备。

[0074] 根据本发明的数据库的测试数据的自动生成装置,解决了字段关联外键识别难的技术难题,能够提供多种灵活的测试数据生成规则,实现智能化插入,同时能够大量且高速地生成符合要求的测试数据,提高了软件开发与测试的效率。

[0075] 数据库的测试数据的自动生成系统

[0076] 下面,基于图7来说明本发明提供的数据库的测试数据的自动生成系统。图7示出了,本申请的一种数据库测试系统,包括用户界面和数据库,还包括如上所述的数据库的测试数据的自动生成装置;其中,用户通过用户界面操作测试数据的自动生成装置,生成测试数据并插入到数据库表中。用户使用该用户界面,通过该自动生成装置可以登录数据库,查询数据库中的数据库表及其字段,该自动生成装置会在该用户界面上返回数据库表字段、字段的类型、字段是否唯一、字段是否可为空、字段是否关联外键、该数据库表的建表语句等。用户在该界面中选择字段的生成规则和/或插入规则,然后选择执行操作,该自动生成装置根据用户配置的生成规则及插入规则在数据库表中插入(生成)数据,然后返回结果界面。用户界面是人机交互接口,供用户输入指定信息,并且按照特定的显示规则显示设备呈现给人的各种结果。在本申请中,用户通过用户界面可以输入要登陆的数据库的服务器名、数据库名、用户以及密码等信息,返回给用户的信息就是指上述数据库表字段、字段的类型、字段是否唯一、字段是否可为空、字段是否关联外键、该数据库表的建表语句等信息。

[0077] 根据本发明的数据库的测试数据的自动生成系统,解决了字段关联外键识别难的技术难题,能够提供多种灵活的测试数据生成规则,实现智能化插入,同时能够大量且高速地生成符合要求的测试数据,提高了软件开发与测试的效率,同时由于能够通过可视化用户界面来选择目标数据库表以及生成规则以及插入规则,对于用户而言,操作形象直观,能够提高软件开发与测试的效率。

[0078] 在本申请中,根据存储模型划分,数据库类型主要可分为:网状数据库(Network Database)、关系数据库(Relational Database)、树状数据库(Hierarchical Database)、面向对象数据库(Object-oriented Database)等,但是不限于此。其中,关系数据库包括例如Oracle、DB2、Sybase、MS SQL Server、Informax、MySQL等,但是不限于此。

[0079] 对于本领域技术人员而言,显然本发明不限于上述示范性实施例的细节,而且在不背离本发明的精神或基本特征的情况下,能够以其他的具体形式实现本发明。因此,无论从哪一点来看,均应将实施例看作是示范性的,而且是非限制性的,本发明的范围由所附权利要求而不是上述说明限定,因此旨在将落在权利要求的等同要件的含义和范围内的所有变化囊括在本发明内。不应将权利要求中的任何附图标记视为限制所涉及的权利要求。

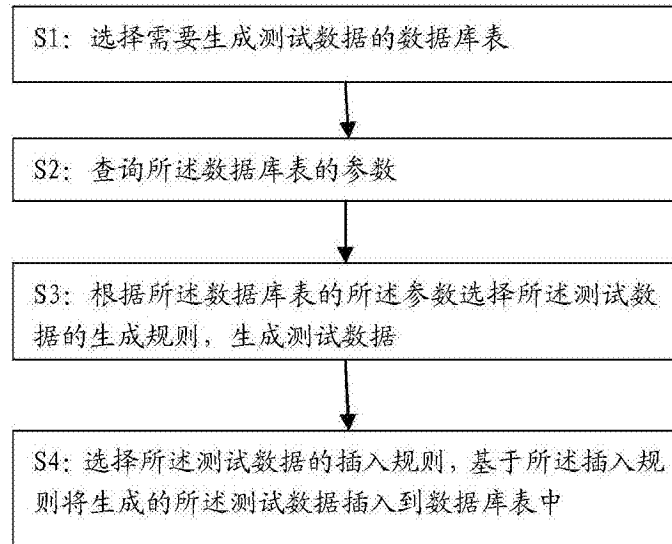


图1



图2

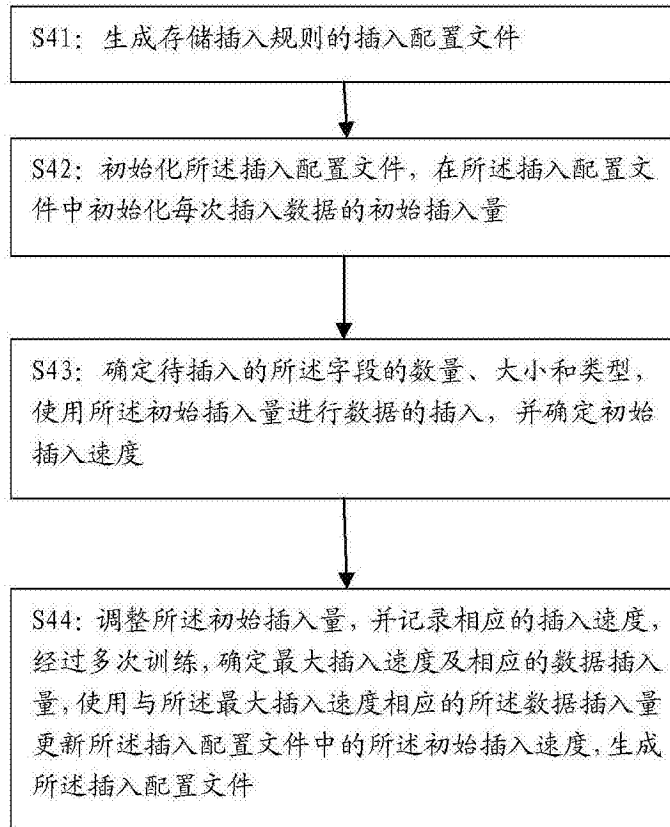


图3

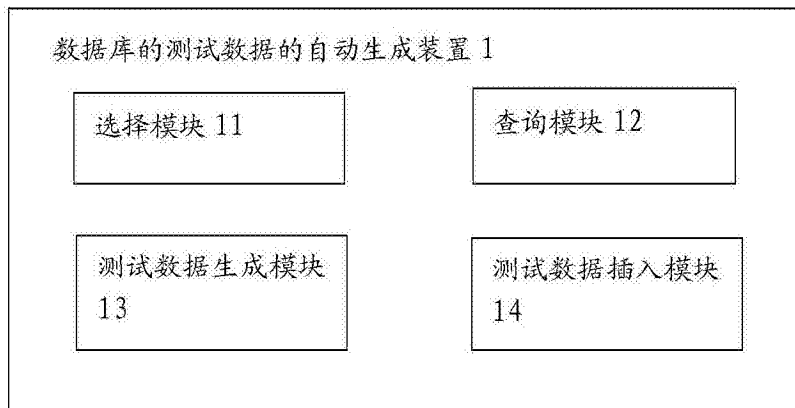


图4

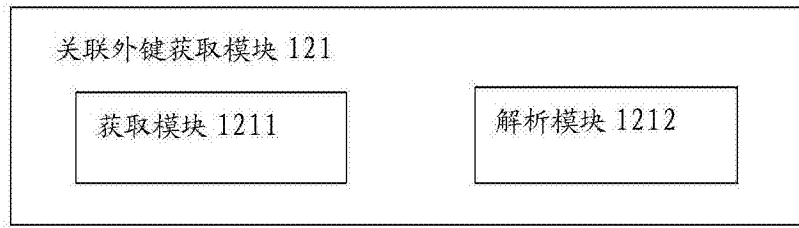


图5

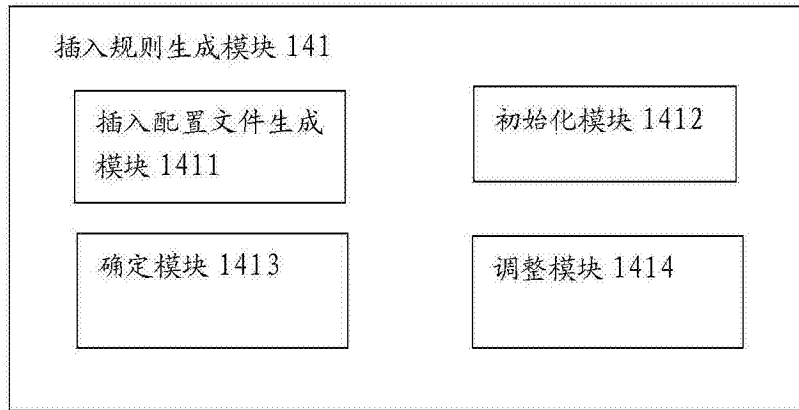


图6

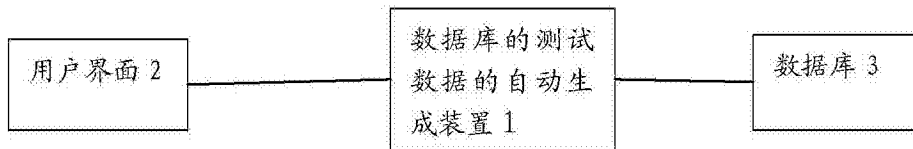


图7