

数据库系统辅助测试工具

王敏¹, 陈亚光²

(1. 武昌理工学院 信息工程学院, 湖北 武汉 430223;

2. 中南民族大学 生物医学工程学院, 湖北 武汉 430074)

摘要: 提出了一种数据库系统测试的辅助工具, 用于解决规范化管理的数据库系统测试项目中测试数据构建及测试结果保留效率低下的问题。其具体做法是, 采用 Excel 工作表规范设计输入数据和预期结果, 再利用该工具完成一些简单设定, 可实现构造数据库测试数据、获取测试结果及实现执行结果与预期结果比较等功能。经多个数据库使用结果表明, 该工具能生成规范的测试结果, 并能减轻测试人员负担, 提高了测试效率。

关键词: 数据库系统; 测试工具; 单元测试; 回归测试

中图分类号: TP311.52

文献标识码: A

文章编号: 1674-7720(2013)03-0013-03

An aided test tool for database system

Wang Min¹, Chen Yaguang²

(1. College of Information Engineering, Wuchang University of Technology, Wuhan 430223, China;

2. College of Biomedical Engineering, South-Central University For Nationalities, Wuhan 430074, China)

Abstract: In order to improve the efficiency of constructing test data and saving test results in database system test, develop an aided tool for standardized management of the testing project. The specific approach is that design the input data and expected results by Excel worksheet standardly. Next, the tool can achieve construction of the database test data, saving test results and comparing with the expected results by simple setting in tool's form. Using results from several database system test project show that, the tool can generate standardized test results, reduce the burden on the testers and improve the efficiency of the test.

Key words: database system; test tool; unit test; regression test

规范化管理的软件测试项目, 从过程上划分为: 单元测试、集成测试、系统测试和验收测试。如果在各个阶段的测试中发现缺陷, 在进行缺陷对应后还必须进行回归测试。因此, 规范化管理的软件测试项目, 其测试用例的设计必须包括输入数据、执行条件和预期结果。针对数据库系统的测试项目, 其测试用例的输入数据、执行条件和预期结果绝大部分体现在数据库中的数据上, 而这些数据在测试用例设计时, 通常采用与关系数据库模型类似的 Excel 工作表来呈现。规范化管理的软件测试项目, 针对各测试阶段的每一个测试用例都必须留下完备的测试结果, 而针对数据库系统的测试结果很大一部分是数据库中的数据。为便于清晰地呈现这些结果, 通常是将测试结果存放在 Excel 工作表中。本文提出的测试工具正是对应这种规范化管理的测试项目而设计开发的。

采用自动化测试工具进行测试, 能减轻测试人员负担、提高测试效率及测试的准确性^[1-2]。虽然 20 世纪 90 年代以来人们就充分认识到测试工具的重要性, 软件开发机构也相继开发出了一些卓有成效的测试工具产品, 但由于测试工具购入的成本及使用培训成本较高, 商用测试工具给软件项目带来的时间效率及经济效率受到一定的限制; 同时, 由于软件项目本身的复杂性, 很难将测试工具充分贯穿于软件测试项目。因此, 在实际项目开发中, 很多软件开发组织并不采用商用软件测试工具, 而是针对项目特点, 自行研发针对性强的轻量级辅助测试工具来改善测试过程。研究者针对数据库系统的测试进行了大量探索, 但探索的内容主要体现在测试方法和测试脚本的开发方面^[1-4], 对实际测试过程中细节部分的测试效率问题探索较少。而本文提出的测试工具

主要针对在规范化管理的测试项目中,需要频繁构建数据库输入和获取测试结果这一特点而设计开发的,它能改善测试过程、提高测试效率。

1 应用需求

由于数据库系统会频繁涉及到对数据库数据的操作,因此,测试用例的设计多数体现在数据库数据的设计上。各测试用例的实施过程为制作测试数据、测试实施和结果判断。其中,制作测试数据需要用设计的测试用例数据去更新数据库,测试实施后需要判断数据库中相关数据是否与预期结果相吻合。作为测试证据的测试结果需要记录下的信息包含:测试执行前数据库相关表的数据和测试执行后相关表的数据。由此可见,在一个测试用例的测试过程中,需要分3次操作数据库,其中1次为更新数据,另外2次为抽出数据。采用SQL-Server的数据库系统,通常使用企业管理器来操作数据库数据;采用Oracle的数据库系统,通常使用Object Browser来操控数据库数据。目前在测试过程中,采用这些工具操控数据,基本上还是人工操作,因此,仍然存在误操作及操作效率低下的问题。如果能将测试过程中对数据的各种操作交给计算机来完成,可提高测试效率,避免误操作的发生,并起到辅助测试的作用。为实现这一辅助测试功能,下面通过对餐饮管理系统点菜画面的单元测试的用例设计和测试结果保留模式进行分析,探索计算机辅助测试的方法。

1.1 测试用例设计模式

图1是采用Excel设计的点菜画面的单元测试项目书,鉴于篇幅,这里只显示了说明所需要的测试项目,对其他项目进行了隐藏。图中用例编号(Case No)31用于测试点菜画面的保存按钮按下,在输入检查无误的情况下,画面中的数据是否正确登录数据库。

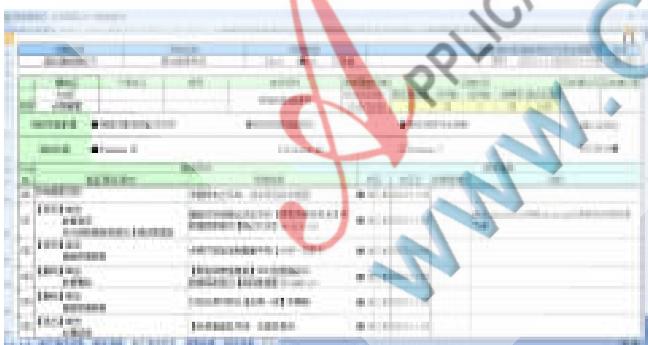


图1 测试项目书

在进行数据库系统的测试用例设计时,由于涉及的数据项较多,不便在测试项目书中列举数据库中的数据,通常采用Excel工作簿的另一张工作表来描述数据库相关的输入数据和预期结果,常用的设计格式如图2所示。其中的信息包含:测试用例编号、测试执行前后标记、表名、字段名及数据明细。其中的数据明细不是来自于数据库,而是来自于人工设计;在测试执行前需要用图2中执行前的数据去更新数据库,采用人工方式可直

接按行拷贝数据粘贴到数据库表中,如果要求表中只包含所设计的测试数据,还需要先删除表中的数据再拷贝粘贴数据。当涉及到多个表时,就需要人工不断在Excel工作表和数据库工具之间来回切换,操作十分繁琐。如果按照一定的规范(图2的样式)设计测试用例,便可采用编程手段实现将Excel工作表中的数据自动写入数据库。

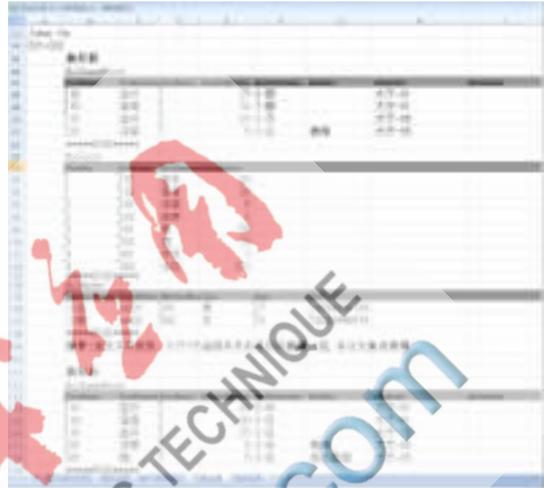


图2 测试用例数据格式

1.2 测试结果保留模式

很多项目也把测试结果称作测试证据,留下测试结果除了便于测试人员判断结果的正误,很大程度上留下测试的证据也是为了便于管理。因此,保留的测试结果应详细体现实施每一个测试用例的输入数据、执行条件、执行过程及执行结果,这些信息应足以体现测试用例的执行过程,并且按照这些信息能再现测试结果。数据库系统的测试结果通常采用Excel工作表来记录,它既能保留截图信息,又能直观地保留数据库中的数据。图3是点菜画面的Case No31-32的测试结果,包含测



图3 测试结果保留格式

试执行前后数据库数据及画面显示内容的截图。对其分析可知,采用编程手段实现抽出数据库中数据并按图3的格式保留数据是可以做到的。

2 工具的实现

分析测试用例设计模式和测试结果保留模式可得,在测试过程中对数据的处理包含两部分:(1)将工作表中设计的测试数据追加到数据库;(2)将数据库数据作为测试结果抽出到 Excel 工作表中。也就是通过编程将图2中的数据追加到数据库以及将数据库中的数据抽出可以实现如图3所示将测试结果显示在工作表中。因此,数据库辅助测试工具可设计成如图4(此截图为数据抽出处理的设置画面)所示的画面。图4操作界面中包含有:对数据库的连接关闭处理区、对 Excel 文件的打开关闭处理区、功能选项区、SQL 编辑区及格式设定区。其主要功能就是实现输入数据做成、测试结果做成及测试配置的保存。下面详细分析各部分的实现。

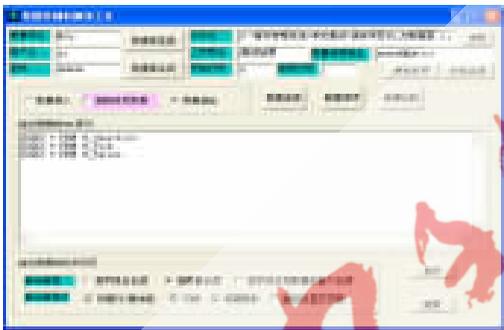


图4 数据库辅助测试工具主画面设计

2.1 输入数据做成处理

输入数据做成即是将设计的测试数据追加到数据库中,图2的数据包含描述性文字(执行前、执行后)、表名、字段名、字段值及数据结束标志(*****END*****),数据结束标志用于标识某数据表数据结束。利用这些数据可以提取出需要更新的表名、字段名及数据。利用图4所示工具,建立数据库连接、指定工作表及向数据库追加数据的开始和结束行号,功能选项设为“数据插入”。单击执行按钮的处理流程如图5所示。图中,“删除现存数据”复选按钮用于选择更新数据前是否删除数据库中现存的数据,若选择删除现存数据,则需要读取数据库中所有用户表名,逐一删除后再向数据库追加数据;过程①实现读取工作表中的表名及字段名,形成 INSERT 语句到 VALUES 为止的前半部分 SQL 语句,当遇到数据结束标志时,再次调用过程①重置 SQL 语句的前半部分;过程②实现读取该行单元格数据,并将数据拼接到过程①形成的 SQL 语句,拼接完成则执行该语句,实现数据向数据库的追加。

2.2 测试结果做成处理

测试结果做成处理主要完成用例执行前后数据库相关表数据的获取,其实质就是根据 SQL 语句抽出数

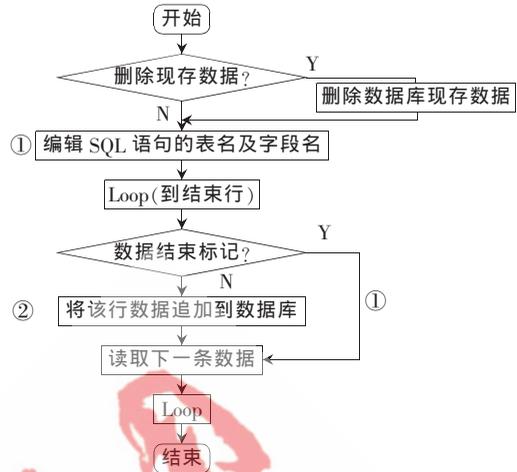


图5 输入数据做成的处理流程

据,将数据结果保留到 Excel 工作表中。如图4的参数设定,这里只需要指定开始行号,即指定从工作表的哪一行开始保留抽出数据,逐一读取“抽出数据的 SQL 语句”中的 SQL 语句,从语句中取出表名,根据表名获取字段名写入工作表,再执行 SQL 语句抽出数据写入工作表,同时按照“抽出数据的格式设定”进行工作表数据的格式设定,所得测试结果规范且便于结果判定。

2.3 测试结果比较处理

为了验证测试结果是否与预期结果相吻合,本测试工具还设计了数据比较功能。单击主画面的“结果比较”按钮,打开如图6所示的测试结果比较画面,在画面中分别设定参与比较的文件名、工作表名、开始及结束行号,即可对指定部分的数据进行比较,从而检验测试结果是否与预期结果一致。

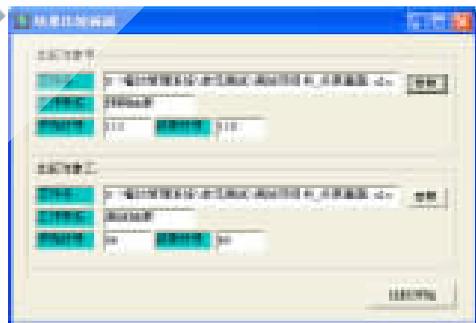


图6 测试结果比较画面

2.4 测试配置保存处理

图4所示的执行画面中,完成数据插入功能画面设定内容较少,但完成数据抽出功能涉及的画面设定较多。为便于回归测试,本工具还设计了配置保存功能,用于将画面现有配置保存到指定的 ini 文件中,便于在需要再次执行某测试用例时,可通过“选择配置”实现读入指定 ini 文件内容进行画面参数的设定。这一功能将保证回归测试及验收测试的测试效率。

本文介绍的数据库辅助测试工具在多个项目的测

试中已被采用,实践证明了采用此辅助测试工具进行测试,可快速生成输入数据及保留测试结果,其保留的测试结果与单元测试的结果要求的吻合度较高,其效率在单元测试过程中体现更为突出;同时在验收测试中需要重现测试结果时,采用本工具也能快速完成测试,提高了测试效率;不仅能产生规范的测试文档,还能减轻测试负担。

参考文献

- [1] 曹娜. 大型关系数据库自动化测试工具的设计与实现 [D]. 武汉: 华中科技大学, 2009.
- [2] 王逸兮. 一个数据库功能性自动化测试系统的设计与实

现[D]. 武汉: 华中科技大学, 2011.

- [3] 曾强, 骆德全, 杨昊苏, 等. 数据库系统测试工具综述 [J]. 科技风, 2008(16): 74.
- [4] 王振铎, 陈晓范. 数据库系统测试方法与技术探讨[J]. 中国民航飞行学院学报, 2008(1): 49.

(收稿日期: 2012-10-11)

作者简介:

王敏, 女, 1968年生, 副教授, 主要研究方向: 软件测试技术。

陈亚光, 男, 1951年生, 教授, 主要研究方向: 计算机应用技术。

