可配置组合式数据校验方法①

林 霞, 申端明, 时 迎, 乔德新

(中国石油勘探开发研究院, 北京 100083)

摘 要: 在油气勘探生产信息系统建设过程中, 由于涵盖勘探、开发和工程等业务, 上报数据项多, 数据量大和上报方式多样化的情况, 容易出现上报数据类型不规范、数据精度不统一、数据统计时间不一致、相互关联的数据项不满足业务规则的问题. 为解决这些问题, 总结出数据项校验、列校验、行校验、跨表校验、历史数据校验和奇异值校验六种数据校验方法及组合式数据校验方法, 并在跨业务和跨时间对比与奇异值分析上设计出符合石油勘探生产行业的校验规则. 在此基础上, 研发了可配置易维护的封装技术, 提高了数据校验方法的实施效率.

关键词: 可配置; 组合式; 数据校验; 封装技术

Configurable & Knockdown Data Verification Method

LIN Xia, SHEN Duan-Ming, SHI Ying, QIAO De-Xin

(Research Institute of Petroleum Exploration & Development, Beijing 100083, China)

Abstract: In the construction of some information system for oil&gas exploration and production, it's likely to result in irregularity of reported data types, disunity of data accuracy, inconsistency of data statistics and dissatisfaction of interrelated data items to business rules. This is mainly due to the coverage of exploration, development and engineering business, the large quantity of reported data items and data, as well as the variety of reported ways. In order to solve these problems, we summarize six verification methods including data item, column, row, cross multi-table, historical data, Singular Value, and also their combined verification methods. We also design some verification policy on cross-business and cross-time comparison and Singular value analysis, which is in accord with petroleum exploration and production industry. On this basis, we further develop encapsulated technology of configuration and maintenance. It promotes the implementation efficiency of verification method.

Key words: configurable; combined; data verifying; encapsulated technology

1 引言

在数据上报系统中经常会出现数据格式不正确、数据不规范、重复填报等问题, 目前常用的单一数据项校验已经不能解决这些问题, 本研究针对这些问题总结出可配置组合式数据校验方法. 且已将该项技术在信息系统中得到了广泛的应用.

某公司在未建设信息系统之前一直采用的是电子邮件、传真和电话等方式采集数据, 这种方式可能导致上报内容不规范、上报数据精度不统一、数据统计时间不一致等问题, 信息系统的建设极大的解决了采集数据和汇总统计问题, 然而随着系统规模的不断扩展, 数据项不断增加, 数据量不断增大, 从 1950 年至今累计上报数据 8576 余万项, 管理人员对数据的质量要求越来越高, 目前常用的单一数据项校验已经不能满足管理人员对数据质量[1,2]的需要.

针对系统中存在的数据质量问题, 为解决这些多部门多业务, 业务数据交叉, 数据重复填报且相互矛盾等问题, 本文在单一数据项校验的基础上总结出了数据项校验、列校验和行校验, 首次提出了跨表校验、历史数据校验和奇异值校验共六种校验方式及组合式数据校验[3,4]方法, 并在此基础上, 研发了可配置易维护的封装技术, 提高了数据校验方法的实施效率.

① 收稿时间:2014-09-11;收到修改稿时间:2014-10-28

2 数据校验方法

2.1 数据校验

数据校验是数据清洗中的一个环节, 数据清洗处理检测和消除数据中的错误, 以达到提高数据质量的目的.

石油勘探开发数据的共享, 首先必须保证所共享数据的正确性和有效性. 由于参与报表数据填报的人员众多、使用习惯差异较大, 再加上数据误差等因素, 难免造成数据不完整、不一致、不统一甚至出错的情况. 如果等到数据存入数据库后再进行数据校验, 那么校验和数据修订的难度将大大增加, 并且也无法做到从数据源头来控制数据的正确性. 为了能及时发现数据中的错误, 最大程度上保证数据的完整性和一致性, 需要在数据存入数据库之前对数据进行校验, 系统还可以通过数据校验来锁定数据错误, 并生成错误报告返回给数据填报人员, 以便及时修正[5].

2.2 现有校验方法

目前国内外, 存在着很多形式的数据校验与清洗工具, 如针对专有的数据格式的校验工具 MD5 值校验工具、Cmis30 数据校验工具、CRC 数据校验计算工具等;针对 web 表单数据的 java 开源包 validator 等[6].

此外还有针对运用于客户端/服务器模式的数据库应用系统中的数据校验包括客户端校验和服务器端校验. 客户端校验是在客户端采用 JavaScript、VBSvript 等脚本语言的方式对 Web 系统中的数据进行校验. 通常客户端校验是对输入的数据进行正确性方面的校验, 校验在客户端进行, 校验速度快, 校验过程中不需要与服务器进行交互, 在某种程度上减少了服务器的负载. 服务器端校验通常是对数据进行逻辑性审核, 校验过程中客户端需要同服务器端进行交互, 虽然相对于客户端校验来说, 服务器端校验增加了服务器的负担和网络流量, 但对于具有海量数据、对数据之间逻辑关系的正确性要求较高的大型数据库应用系统, 服务器端校验提供了数据的一致性和完整性的保障[4].

然而这些校验工具或校验方法要么太过于复杂, 要么功能过于单一, 而无法满足报表系统的种类繁多、数据量大、数据关系复杂的校验需求. 基于这些情况, 本文提出了可配置组合式的数据校验方法.

3 可配置组合式数据校验方法设计与实现

3.1 数据校验

某公司信息系统在两个层级用户中使用, 通过系统自动校验和业务人员逐级审核所管辖数据的方式, 及时发现问题并解决. 该系统模拟业务人员检查数据的方法和流程, 对每个上传的数据表进行数据类型、规范性等校验, 只有通过校验的数据才能进入系统, 基于这个现状, 再通过对基于 Struts 的数据校验框架研究[7]的基础上决定采用面向对象的机制实现校验配置封装技术, 构建了易配置易维护的校验平台. Struts 由一组相互协作的类(主件)、Servlet、JSP taglib 等构成.

数据校验框架 validator 是 Jakarta 通用包的一个部分, 是基于 Struts 技术、通过配置外部 XML 文件以引用校验规则和匹配正则表达式为基础的一种校验机制, 可使用相同的校验规则同时实现服务器端和客户端的数据校验.

通过对比, 某公司信息系统底层架构采用面向对象的机制实现校验配置封装技术, 构建了易配置易维护的校验平台, 实现了可配置组合式数据校验方法. 可配置指的是校验是在 XML 中进行的, 这种方式既实现了可配置化又减少了代码的编写. 如图 1 所示, 下面将对系统的校验规则设计进行详细的介绍.

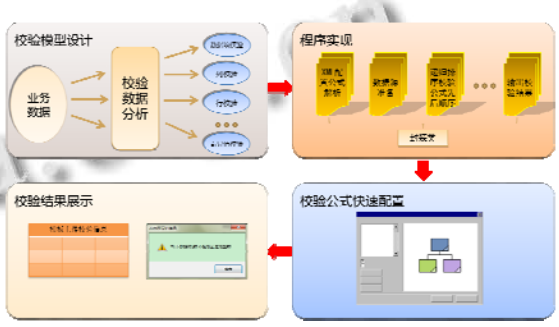

图 1 设计模型

(1)校验模型设计方面, 通过对大批业务数据中需要校验的数据进行归类分析, 整理出数据项校验、列校验、行校验、跨表校验、历史数据校验、奇异值校验六种校验模型.

数据项校验指的是对某一个数据表中的某一个数据项的校验, 包括非空校验、数据类型校验、正则表

达式校验[8,9];列校验指的是数据项与数据项之间关系的校验, 包括对比校验、累计值校验、四则运算校验和复杂公式校验[10];行校验指的是数据表中行与行之间关系的校验, 包括合计值校验和重点数据漏报校验; 跨表校验指的是数据表与数据表中数据项关系的校验, 也叫数据项联合校验;历史数据校验指的是同一张表中跨时间的数据项关系校验;奇异值校验指的是范围值校验和与平均值对比校验.

(2)程序实现方面, 采用 XML 结构性标记语言来记录校验公式规则, 便于进行统一配置和解析, 并采用面向对象[11]的机制将各类方法封装成类, 便于业务层调用.

XML 是基于 SGML[12,13](标准通用标记语言)开发出来的, 因此 XML 具有对之前依赖于 SGML 的系统的兼容性, 并且将 XML 与 WEB 相结合使得显示出来的数据更容易被用户查看和操作. XML 技术主要包括 XML, XSL 和 XLL(Extensible Linking Language, 可扩展的链接语言)[14]这三个标准, 通过这三个标准的相辅相成使得 XML 在描述数据的时候更加强大, 从而保证数据交换过程的顺利进行. XML 即可扩展标记语言, 这里提到的标记意思是根据数据本身对数据进行编码的方法, 简要地说, 就是针对相同的数据因为不同的含义而添加相应的标记[15,16]. 当这些有标记的数据被传送给应用程序时, 应用程序会依据这些标记对这些不同含义数据做不同的处理. 通过此种标记, 程序之间可以处理包含各种信息的文章等. XML 主要包含 XML 文档(按照 XML 所要求的规范些的文件)、DTD 文档(描述 XML 文档的模板)和它的扩展 Schema 模式(用来验证 XML 文档逻辑结构)等标准和规范[17].

(3)校验公式配置方面, 开发了校验公式快速配置工具, 实现在页面上直接配置和在 XML 中快速配置的两种校验方式, 极大的缩短了校验公式配置时间. 通过这种方式使得不会编程的人员也能很方便的配置出校验公式, 对数据进行校验. 基于 XML 的数据交换格式的优点[17]有:①易于扩展;②结构性强;③交互性好;④可格式化;⑤易于处理;⑥灵活性强;⑦与平台无关.

(4)校验结果展示方面, 采用了列表式清晰简洁的布局方式, 通过弹出窗口给出每种校验错误报告, 以便于用户根据报告提示信息迅速找到错误数据并及时更正, 提升用户操作体验. 如图 2 所示:

图 2 错误报告

3.2 技术实现

可配置组合式数据校验方法主要通过程序代码、XML 配置和建立一套统一的运算符三个方面来实现的.

首先是代码实现方便, 包括以下 6 个步骤:

(1)采用了 XML 配置公式解析, 包括以下 3 个类:

①定义 XML 属性对象类 XMLCellStruct()

②定义 XML 列对象类 XMLColumnStruct()

③定义 XML 行对象类 XMLRowStruct()

(2)初始化上传数据对象类 Init();

(3)递归分析校验公式, 包括以下 3 个方法:

①分析校验公式方法 AnalyzeType();

②递归分析列公式方法 AnalyzeCol();

③递归分析行公式方法 AnalyzeRow();

开发技巧:采用堆栈的方式将新产生的校验项前置;校验公式中如果带有新校验公式, 采用递归方法解析当前校验公式中的每个数据项.

(4)加载原始数据方法 FillDataSource();

(5)进行校验计算, 包括以下 2 个方法:

①计算列上的校验公式方法 CalculateOnCol ();

②计算行上的校验公式方法 CalculateOnRow();

开发技巧:使用 JS 进行表达式计算, 会比 C#中运行公式计算的速度快很多.

(6)保存并输出校验结果方法 SaveResult();

其次是配置实现方面, 在配置方面校验公式采用结构性的标记语言 XML 进行快速配置, 一个校验公式的配置信息如图 3 所示.

```
<TD Style="text-align:left;">
  <Control Name="NCNL" ControlType="Text" CheckNull="true" CheckLength="2"
CheckType="CheckInt"
    CheckExp="{NCNL}&gt;={TCYJ}*{DJRC}*220"
CheckReg="^M\d{1,2}|M\d{1,2}L\d{1,2}$" CheckError="错误提示"/>
</TD>
```

图 3　配置信息

图 3 中的 XML 配置各节点属性说明如下表所示:

表 1　节点属性说明

序号	节点属性	属性含义
1	*Control*	校验开始标识
2	*Name*	校验数据项名称
3	*ControlType*	数据项类型
4	*CheckNull*	校验是否为空
5	*CheckLength*	校验精度
6	*CheckType*	校验值类型
7	*CheckExp*	公式校验
8	*CheckReg*	正则表达式校验
9	*CheckError*	校验结果提示

在节点属性说明中提到的正则表达式校验指的是符合某种规则的表达式, 可以将其理解为一种对文字进行模糊匹配的语言[18]. 正则表达式用一些特殊的符号(称为元字符)来代表具有某种特征的一组字符以及指定匹配的次数, 含有元字符的文本不再表示某一具体的文本内容, 而是形成了一种文本模式, 可以匹配符合该模式的所有文本串[19]. 它可以快速地分析大量的文本以找到特定的字符模式;提取、编辑、替换或删除文本子字符串[20]. 在程序语言中引入正则表达式, 可以完成以下功能.

(1)测试字符串的某个模式, 验证用户输入的有效性.

(2)在文本中使用一个正则表达式来标识某些特定的字符, 然后对其进行删除、替换等操作.

(3)利用正则表达式搜索字符串的模式, 然后从字符串中提取一个子字符串.

最后是建立一套统一的运算符, 它包括传统的逻辑运算符和自定义的辅助运算符, 规范校验公式的书写规则, 同时增加用户对校验公式的可读性. 传统的逻辑运算符有 Max, Min, +, -, &等, 系统自定义的辅助运算符如表 2 所示:

表 2　辅助运算符

辅助运算符	含义
YYYY	年
MM	月
DD	日
LD	当月累计天数
PLD	上月累计天数
YD	当年累计天数
MD	年初至当月天数
IsFirstDayInYear	是否某年 1 月 1 日
IsFirstDayInMonth	是否某月 1 日
IsLastDayInMonth	是否某月最后一天
IsTenDays	是否旬末点
……	

可配置组合式数据校验方法正是通过将这三个方面相结合来实现校验数据的目的.

4　实际应用与效果

4.1 实际应用

目前, 可配置组合式数据校验方法已在某油气生产信息系统中得到了较好的应用, 下面就每种校验方式的应用情况进行举例说明.

(1)数据项校验涵盖生产运行、油藏、天然气、煤层气等 12 项业务中 437 个数据项校验. 具体应用如表 3 所示:

表 3　数据项校验

校验类型	公式说明	公式配置
非空校验	站名称不能为空	*CheckNull="true"*
数据类型	库存油量应填整数	*CheckType="CheckNegInt"*
正则表达式	水平井分支名需满足形如"*M1L1*"规范	*CheckReg="^M\d{1,2}|M\d{1,2}L\d{1,2}$"*

(2)列校验涵盖生产运行、天然气、产能建设等 11 项业务中 296 个数据项校验. 具体应用如表 4 所示:

表 4　列校验

校验类型	公式说明	公式配置
对比校验	年产气<300 万方	*CheckExp="NCGYQL<300"*
四则运算	总井数= 产井+注水井	*CheckExp="{HJ}={SCJ}+{ZSJ}"*
复杂公式	采出程度=累积产气量/ [月天数]*[年天数]/地质储量<年初值>*100	*CheckExp="{LJCQL}/{YKFDZCL}<LB:002001><YM:FM>*100"*

(3)行校验涵盖生产运行、油藏报表、产能建设 3 项业务中 35 个数据项校验. 具体应用如表 5 所示:

表 5 行校验

校验类型	公式说明	公式配置
合计值校验	外销合计值要等于供油出口外销之和	*CheckExp="Math.abs({HJ}-!{HJ})<=7"*
漏报校验	重点区块漏报	*CheckExp="isCheckQKDY=True"*

(4)跨表校验涵盖生产运行、油藏、天然气和油藏评价等 6 项业务中 59 个数据项校验. 具体应用如表 6 所示:

表 6 跨表校验

校验类型	公式说明	公式配置
跨业务	气油比=月产气量/井口月产油量	*CheckExp="{SCQYB}={YCQL}/{JKYCYL}"*
跨表	表 1 正钻井井号=表 2 新开钻井-表 2 已完成钻井	*CheckExp="{CX_M}={ZJ_K}-{ZJ_L}"*

(5)历史数据校验涵盖生产运行、油藏、天然气和合作报表 4 项业务中 28 个数据项校验. 具体应用如表 7 所示:

表 7 历史数据校验

校验类型	公式说明	公式配置
累计值	累产油=<上月>累产油+月产油	*CheckExp="{HSLJCYL}<%Y1M12>+{HSYCYL}|1\)?\({HSLJCYL}<%M-1>+{HSYCYL}|2\)"*
跨时间	采出程度=当月累产油/当年一月份时的可采储量	*CheckExp="{HSLJCYL}/{KCCl}<YM:FM>*100"*

(6)奇异值校验具体应用如表 8 所示:

表 8 奇异值校验

说明	公式配置
公式配置	*CheckExp="{RCQ}<=[80]&{RCQ}<=120]"*
校验提示	当日产值在范围平均值 80%~120%区间以外, 系统给出超出平均范围提示

4.2 应用效果

在 2010 年底, 某公司数据采集系统增加校验机制后, 用户及时的修改错误数据, 使 2011 年和 2012 年的数据质量与 2010 年相比得到了显著的提高. 具体体现如下:

(1)2010 年上报数据项:共 1468323;错误 1502;错误率 1‰.

(2)2011 年上报数据项:共 1469126;错误 5;错误率 0.0003‰.

(3)2012 年上报数据项:共 1525735;错误 2;错误率 0.0001‰.

该软件技术在 3 年多的实际应用中, 极大提高了上报数据的准确性、完整性和一致性, 提高了系统数据的可靠性, 提升了系统上报数据的质量, 并且缩短了管理用户整理数据、制作报表、进行统计分析的时间, 得到了系统用户的肯定.

5 总结

可配置组合式数据校验方法利用可配置和组合式的特点, 可配置的特点极大缩短了由于新增和修改校验公式的运维时间, 更不需要修改和开发任何代码程序, 每个数据项的校验公式都是在页面中直接配置出来的;组合式的特点有效地将数据项校验、列校验、行校验、跨表校验、历史数据校验和奇异值校验组合在一起应用, 弥补传统的单一校验方法的不足, 满足系统用户对数据校验的要求. 这种校验方式在一定程度上提高了系统的数据质量. 对于数据管理系统来说, 信息化只是一种工具, 只能解决快速上传、统计汇总等问题, 而系统最关键的还是数据, 正确的数据对领导的分析和决策有很重大的意义, 本文提出的简单有效的校验机制更好的保证了数据的准确, 值得推广和应用.

参考文献

1 苏贤明,沈志宏,刘宁.基于知识规则的 Excel 数据质量校验工具.科研信息化技术与应用,2012,3(3):29–37.

2 张光渝,杨秋辉,詹聪,郭鑫宇,阙舒.开放式 XML 数据的质量分析方法.计算机应用研究,2013,30(7):2082–2086.

3 王怀金.油藏月报数据校验机制的设计与实现.中国信息界,2012,10:63–64.

4 肖明.基于规则的数据校验在数据库应用系统中的实现.计算机与信息技术,2007,21:337–339.

5 郭艳军,王喆,潘懋.一种支持数据校验的 Excel 信息转储元数据模型.计算机应用与软件,2014,31(6):15–17.

6 张仁,沈志宏,黎建辉,施建平.基于规则的土壤数据校验模型研究与实现.计算机系统应用,2010,19(8):78–81.

7 孙麒,郑宁,周志宇.基于 Struts 的数据校验框架的应用研究.计算机工程与设计,2004,25(8):1313–1316.

8 王功明,吴华瑞,赵春江,杨宝祝.正则表达式在电子政务客户端校验中的应用.计算机工程,2007,33(9):269–271.

9 邓富强.基于正则表达式的客户端数据校验方法研究.网络与信息,2010,24(4):49.

10 汪洋.快速校验 Excel 数据的两种方法.电脑知识与技术,2009,5(15):4069,4074.

11 黄捷,古辉.面向对象程序的类信息的抽取规则.计算机系统应用,2011,20(5):51–54.

12 王洪荣,吴保国.异构数据库间数据交换工具的设计与实现.北京林业大学学报,2009,(S2): 102–106.

13 况旭,刘波.XML 的面向对象语言特性.计算机技术与发展,2010,20(1):54–57.

14 Fernandez M, Tan WC, Suciu D. SilkRoute: Trading between relations and XML. Computer Networks, 2007, 33: 723–745.

15 Chen SK, Lo ML, Wu KL. A practical approach to extracting DTD-conforming XML documents from heterogeneous data sources. Information Sciences, 2006, 176(7): 820–844.

16 Chen LY, Tan L, An Y. Design pattern integration method for improving performance of EJB and its applications. Environmental Science and Information Application Technology, 2009, 3(3): 134–136.

17 王军民.基于 XML 的异构数据转换的研究与实现[硕士学位论文].成都:电子科技大学,2008.

18 唐惠丽,郑小妹.正则表达式的研究及在 Web 中的应用.计算机技术与发展,2013,23(2):82–84.

19 周峰,王征.ASP.NET 3.5 网络程序设计案例集锦.北京:水利水电出版社,2009.

20 Goyvaerts J, Levithan S. Regular Expression Cookbook. O'Reilly Media, 2009.