

自动气象站运行状态查询和统计系统的设计与实现

崔明¹, 卢会国¹, 裴翀², 胡学英², 刘银锋²

(1 成都信息工程学院, 四川 成都 610225; 2. 中国气象局 气象探测中心, 北京 100081)

摘要: 在 Microsoft Visual Studio 2008 开发环境下, 通过用 C# 语言连接 SQL SERVER 2000 数据库, 实现了五个试验自动气象站的分钟状态文件统计入库。系统可以对各试验自动气象站的分钟运行状态进行查询和统计, 以便及时发现样机设计和运行中存在的问题。为试验阶段对样机的评估提供了一定的帮助。

关键词: 自动气象站; 运行状态; 查询统计; 系统设计

中图分类号: TP315 **文献标识码:** A **文章编号:** 1009-3044(2012)16-3764-05

Design and Implementation of Automatic Weather Station's Query and Statistics System in Running Status

CUI Ming¹, LU Hui-guo¹, PEI Chong², HU Xue-ying², LIU Yin-feng²

(1. Chengdu University of Information Technology, Chengdu 610225, China; 2. Atmospheric Observation Technology Center CMA, Beijing 100081, China)

Abstract: The system realizes the minute operation and automatic data that from five test automatic weather stations analysis and storage f by using C # to connect SQL SERVER2000 database in the development environment of Microsoft Visual Studio 2008. And the system provides function like query and statistics minute operation information that from each test automatic weather station, so that timely discovery the existing problems prototype designs and operations. It provides some help for evaluation of the prototype in the test stage.

Key words: automatic weather station; running status; query and statistics; system design

我国自动气象站站点数量庞大, 其中包括 2400 多个基本业务站和 30000 多个区域站点。如此多的站点设备维护是异常困难的。而设备正常无故障运行是保障探测数据的质量状况、气象预报准确性和决策服务有效性的前提。气象行业专项项目“地面观测网自动化运行监控技术研究”在现有自动站的基础上通过对设备进行改造设计, 实现了对自动站各部件运行状态的实时监控。改造后的样机除了上传正常的观测要素数据外, 还每 10 分钟上传样机的运行状态文件, 文件按一定的格式记录该 10 分钟内自动站各部件的运行状态和其它参数信息。在 SQL SERVER 2000 数据库支持下, 采用 C# 语言, 在 Microsoft Visual Studio 2008 开发环境下设计的自动气象站运行状态查询和统计系统就是对试验样机上传的分钟状态文件进行解析入库, 并实现对样机运行状态的查询和统计。

1 系统框架设计

为了对样机的设计改造进行检验, 项目在全国范围内选取了五个试验场地进行为期一年的外场试验。分钟数据文件和状态文件每隔 10 分钟由台站上传到国家局服务器。在试验阶段将数据由服务器下载到本地计算机后在进行解析入库, 为系统的查询统计提供支持。查询界面通过对分钟数据进行访问和统计, 实现对自动站运行状态的查询和故障部件的统计分析。系统结构图如图 1。

2 C#和SQL SEVER 数据库

2.1 C#语言概述^[1]

C# (C SHARP) 是一种最新的、面向对象的编程语言。它使得程序员可以快速地编写各种基于 MICROSOFT .NET 平台的应用程序。使用 C# 开发应用程序比使用 C++ 简单, 因为其语法简单。但 C# 是一种强大的语言, C++ 能完成的任务利用 C# 也能完成。C# 的优点是, 它是唯一为 .NET Framework 设计的语言, 是在移植到其他操作系统上的 .NET 版本的主要语言。用 C# 编写 Windows 应用程序具有我们熟悉的 Windows 外观和操作方式, 利用 VS2008 的 Windows Forms 模块就能生成这样的应用程序。Windows Forms 模块是一个控件库, 其中的控件 (如按钮、工具栏、菜单栏等) 可以用以建立 Windows 用户界面。

另外, 在进行数据库访问是, 可以通过 Active Data Objects.NET (ADO.NET) 部分或 C# 得 LINQ 功能。

2.2 SQL SERVER 数据库相关概念^[2-3]

SQL 2000 是一个高性能的客户端/服务器结构的关系型数据库管理系统。SQL 的一个主要有待你就是与主流客户端/服务器开

收稿日期: 2012-03-28

基金项目: 气象行业专项项目“地面观测网自动化运行监控技术研究”资助

作者简介: 崔明 (1987-), 男, 在读硕士, 研究方向为大气探测信息处理。

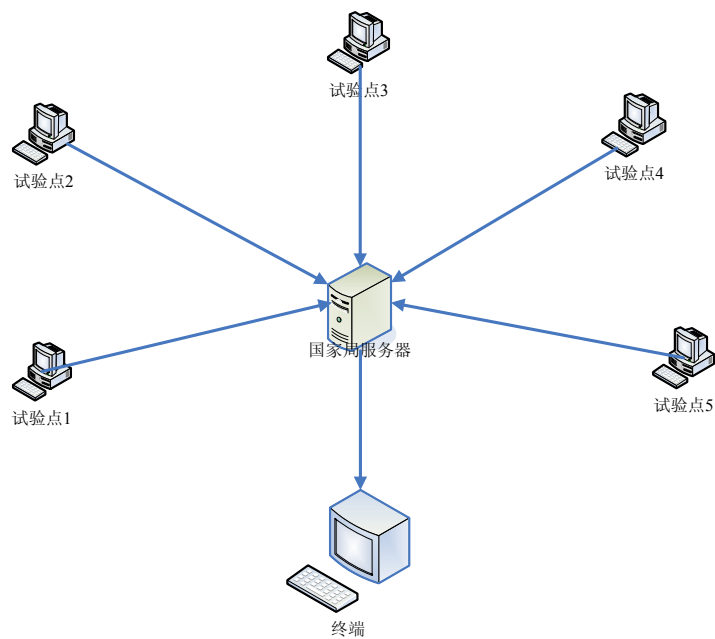


图1 系统结构图

关工具和桌面应用程序的紧密集成。可以在 Visual Basic.NET, Visual C#.NET 和 Visual C++ 中访问 SQL Server2000 数据库。

2.2.1 表

SQL Server2000 数据库的所有数据包含在称为表的对象中。每个表代表某类对用户有意义的对象。SQL Server 中的表有行和列两个主要组件,如图2所示。

台站号	观测时间	部件数量	FRU数量	系统部件状态
JY001	2011-11-25 7:30	010	030	0000000001
JY001	2011-11-25 7:40	010	030	0000000001
JY001	2011-11-25 7:50	010	030	0000000000
JY001	2011-11-25 8:00	010	030	0000000000
JY001	2011-11-25 8:10	010	030	0000000000
JY001	2011-11-25 8:20	010	030	0000000001
JY001	2011-11-25 8:30	010	030	0000000000
JY001	2011-11-25 8:40	010	030	0000000000

图2 SQL Server2000 表实例

2.2.2 SELECT 查询语句

数据查询是用来描述从数据库中获得数据和操纵数据的过程,是对已经存在与数据中的数据按照指定条件检索的过程。SELECT 语句从数据库中检索出数据,然后以一个或多个结果集的形式将其返回给用户。SELECT 语句的完整语法比较复杂,其主要子句可总结为:

```
SELECT select_list [DISTINCT] [TOP n]
[INTO new_table_name]
[FROM able_list]
[WHERE search_conditions]
[GROUP BY group_by_list]
[HAVING search_conditions]
[ORDER BY order_list [ASC | DESC]]
```

虽然 SELECT 语句的完整语法比较复杂,但该系统中主要应用 SELECT 语句描述结果集的 4 个主要属性,即,结果集的列的数量和属性、结果集从中检索数据的表、条件语句、结果集得分组。

3 数据库主要数据表

分钟观测数据表(odata_OM):该表存储5个试验站点的分钟观测数据,每个站点每分钟一条记录,是通过上传的分钟数据文件进行解析后导入数据库的;10分钟观测数据表(odata_OT):此数据表内存储站点每十分钟的观测数据,每个站点每十分钟一条记录,是后期数据质量控制的基础数据表;小时观测数据表(odata_OH):表内存有试验站点的小时观测数据,每个站点每小时一条记录,是与基本业务站进行数据对比的基础数据表;分钟状态数据表(rdata_OM_JY、rdata_OM_NC、rdata_OM_NJ、rdata_OM_XN、rda-

ta_OM_ZQ):每个实验站点各一个分钟状态数据表,表内存有个站点的分钟运行状态,这5个表是“状态查询”和“状态统计”功能的基础,具有相同的表结构。除了这些主要表外,还有文件名错误表(files_Namecheck)、数据格式错误表(element_Check):表内分别存有在进行文件名格式检查是发现的错误信息和数据格式错误信息;站点信息表(site_Info):存放五个实验站点的基本信息。

4 主要程序设计

4.1 数据库操作类

```
class DataCon
{
public SqlConnection getcon()
{
string M_str_sqlcon = "Data Source=.;Initial Catalog=db_SAMTR;Integrated Security=True";
SqlConnection myCon = new SqlConnection(M_str_sqlcon);
return myCon;
}
public DataSet getds(string M_str_sqlstr, string M_str_table)
{
SqlConnection sqlcon = this.getcon();
SqlDataAdapter sqlda = new SqlDataAdapter(M_str_sqlstr, sqlcon);
DataSet myds = new DataSet();
sqlda.Fill(myds, M_str_table);
return myds;
}
}
```

其中方法 getcon()用来建立数据库连接, M_str_sqlcon 为连接字符串;方法 getds(string M_str_sqlstr, string M_str_table)创建 DetSet 对象,将数据显示到窗体的 DataGridView 控件中,参数 M_str_sqlstr 为 SQL 语句。

4.2 主要数据操作

```
string sqlstatus = @"INSERT INTO "+station+"(台站号,观测时间,部件数量,FRU 数量,系统部件状态,FRU 状态,采集状态,采集信息,地温状态,地温信息,温湿度状态,温湿度信息,风状态,风信息,雨状态,雨量信息,供电状态,供电信息) VALUES(@台站号,@观测时间,@部件数量,@FRU 数量,@系统部件状态,@FRU 状态,@采集状态,@采集信息,@地温状态,@地温信息,@温湿度状态,@温湿度信息,@风状态,@风信息,@雨状态,@雨量信息,@供电状态,@供电信息)";
```

```
SqlCommand command = new SqlCommand(sqlstatus, conn);
```

上述 sqlstatus 为状态数据入库时的 SQL 语句,通过执 command.ExecuteNonQuery()将处理后的数据存入不同的数据表中。

```
DataCon dataconn = new DataCon();
```

```
DataSet myds = dataconn.getds("select * from " + station + " where 观测时间>=" + strbegin + "' and 观测时间<=" + strend + "'", "" + station + "");
```

```
dgvShow.DataSource = myds.Tables["" + station + "");
```

```
DataSet myds1 = dataconn.getds("select 系统部件状态,count(*)as 总数 from " + strstation + " where 观测时间>=" + strbegin + "' and 观测时间<=" + strend + "' group by 系统部件状态 ", "" + station + "");
```

```
dataGridView1.DataSource = myds1.Tables["" + station + "");
```

执行上述程序分别实现对运行状态的查询显示及统计显示。其中 station 对应站点的数据表名称;strbegin、strend 要查询的开始时间和结束时间,通过 DateTimePicker 控件来选择。查询结果会显示在窗体的 DataGridView 控件中。

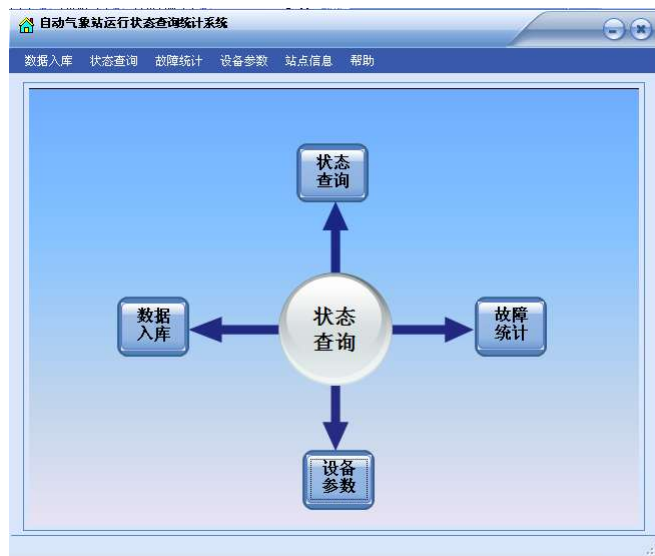


图3 系统主界面

5 系统界面设计^[4]

系统采用C/S(Client/Server)结构,初步主要有数据入库、状态查询、故障统计、设被参数查询显示等功能(如图3所示。)。通过公用数据库操作类实现对数据库的连接及对数据库中数据表的操作。

5.1 数据入库界面

点击主界面中的“数据入库”按钮进入数据入库界面。首先通数据库操作公用类建立与数据库的连接。在选择站点数据库后,通过System.IO名称空间提供的文件读写类StreamReader对数据文件进行解析入库。在将数据文件读取并按照规定格式处理后,通过执行SQL语句将处理后的数据存入已有的数据表中。在处理不同类型(观测数据、状态数据)、不同时间(分钟、10分钟、小时)的数据时程序会执行不同的SQL语句,将数据存入对应的表中。同时在解析文件时会对文件名的格式和数据格式进行初步的判断。如果存在错误,错误信息会存入已有的表文件名错误表(files_Namecheck)、数据格式错误表(element_Check)中。此外还实现了对数据文件的备份处理,见图3。



图4 数据入库界面



图5 状态查询界面

5.2 状态查询界面

在该查询界面中选择查询站点和时间段后,查询结果会显示到窗体的 DataGridView 控件中。点击 DataGridView 数据单元格,此时的系统部件和 FRU(可更换单元)的数据和解析后的结果会会显示到相应的位置,如图 5 所示。实现了对任意时间样机运行状态的查询。为试验阶段样机的阶段性评估及改进维护提供了帮助。

5.3 故障统计界面

该界面根据选择的站点和时间,对数据库表进行检索并按照系统部件状态和 RRU 状态进行统计,将查询结果显示到窗体中。通过点击显示结果,可以将统计的故障信息和故障总数显示在界面中,如图 6 所示。



图 6 故障统计界面

6 结束语

由于该系统是在项目试验阶段为分析数据提供帮助而设计的,在设计和功能上还存在许多不足。在后期的的开发中会逐步增加观测数的查询显示、数据质量控制、对比分析、到报率统计等功能,以便为样机的运行评估提供必要的支持。但是该系统的设计实现了对试验样机观测数据、状态数据的解析入库、状态查询及故障统计分析,为监控样机运行和维护保障提供了一定的帮助。同时通过对数据的分析统计可以成项目阶段性报告为项目顺利完成提供帮助。

参考文献:

[1] Waston K. C#入门经典[M].齐立波,译.北京:清华大学出版社,2008:3-457.
[2] 姜力,高群.SQLServer数据库设计与管理[M].北京:北京大学出版社,2006:22-170
[3] 格鲁夫.SQL完全手册[M].3版.北京:电子工业出版社,2010:48-178.
[4] Johansen B.Windows应用高级编程—C#篇[M].张哲峰,译.北京:清华大学出版社,2002:172-248.