Research on system of equipment management based on Cloud and Internet of things

Tang xiaojiao

Tangshan Key Laboratory of Mechantronics, Tangshan College, Hebei Tangshan, 063000, China

tyra619@163.com

Keywords: Cloud and Internet of things; Handhold Terminal; equipment management; B/S Architecture

Abstract. There are many problems in the traditional equipment management, such as lower efficiency, decentralized information and unable to realize Fault prediction and diagnosis. The system of equipment management based on Cloud and Internet of things was researched. Handhold Terminal based on B/S Architecture was developed, which can collect the state information of all kinds of equipment. The system can perform quantitative measurement, qualitative analysis, fault diagnosis and maintenance. And it can perform data synchronous communication with the server by wireless network, which includes data up-download and update. The system can improve the reliability of equipment and reduces the incidence of failure.

I. Introduction

In the traditional mode of production, the management system for large production equipment is very laggard [1]. There are many problems, such as lower production efficiency, lower informatization level, higher accident frequency because of inadequate monitoring [2.3]. The technology of Cloud and Internet of things combine Internet of things with Cloud calculating. The sensing devices in Internet of things will generate a large amount of information. Cloud calculating can storage, retrieval and use these Massive Data. The system of equipment management based on Cloud and Internet of things takes on a huge network structure, which can realize intelligent recognition, positioning, tracking, monitoring and management.

II. Structure of system for equipment management

The structure of system for equipment management is shown in Fig.1. ZigBee network is arranged in the industrial field. Wireless sensor network node is installed on key equipments. Handhold Terminal is used to Spot Check some important devices real time. The network structure of Internet of things is build based on ZigBee technology and all of the Massive Data are sent into Cloud server.

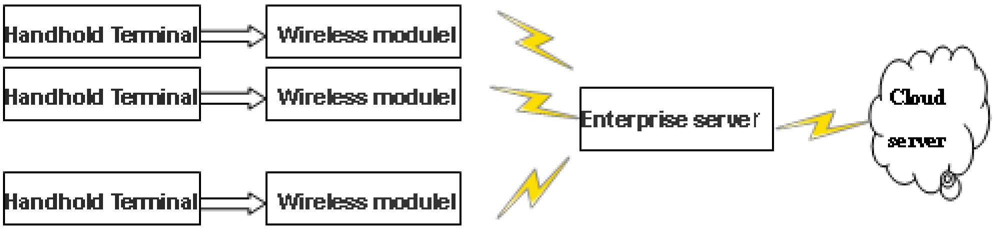

Fig.1 structure of the system

III. Condition monitoring process of the system

The core of device monitoring is to measure all kinds of parameters of devices, such as temperature, vibration, pressure. If the devices work in normal, it’s just need to finish the communication between data collecting with server. When there is abnormal data, it’s needed to inquire fault cause according to fault table, and then to examine and repair the device according to standard operation sheet, as shown in Fig.2.

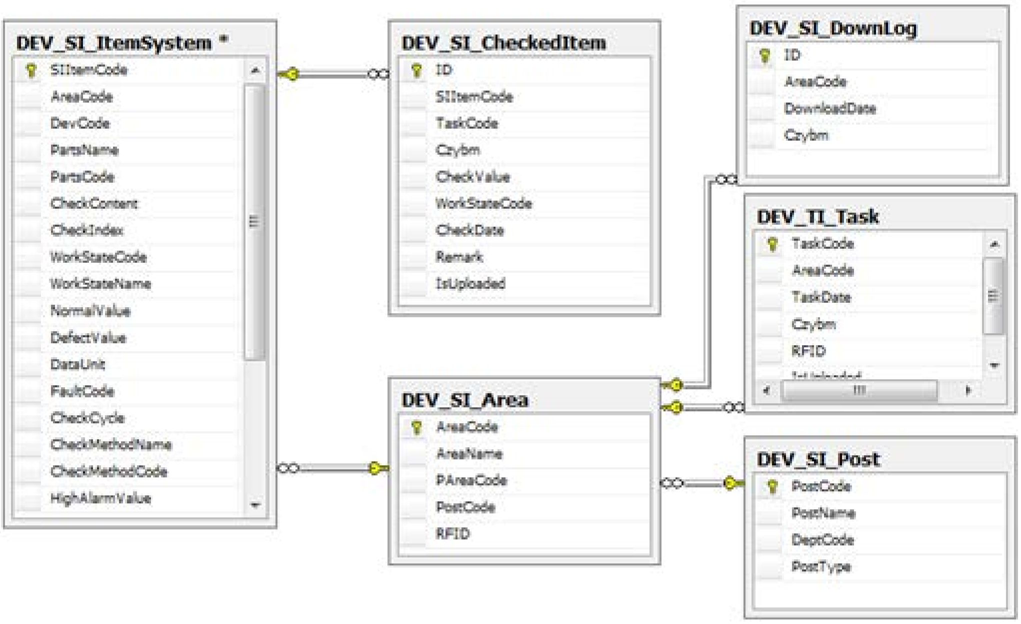

Fig.2 standard operation sheet

Taking a coal company's production process as an example, using Tree View control in Visual Studio, a dendrogram of monitoring process is build [4.5]. First determine monitoring points and then traverse every point of the Spot Checking tree with recursive method from the root node as beginning. The Structure of three layers tree is build, including region layer, device layer and measuring layer. For example, if the first layer is the region of daily check for coal mining, the second layer is the device of shearer and the third layer is the measurement of sound and temperature of cutting part. It's realized as follows:

```
private void UpdateItemState()
{      int i, j, k;
 //    Construct the field node
            for (i = 0; i < classDatas.pDataPerson.Posts[gIdxPost].Areas.Count; i++)
{TreeNode aNode = new TreeNode();
              aNode.Text = classDatas.pDataPerson.Posts[gIdxPost]. Area s[i].AreaName;
              aNode.Tag = TreeIdxToTag(i, -1, -1, -1);
              aNode.SelectedImageIndex = aNode.ImageIndex = SetupImgIdx(0, gIdxPost, i, 0, 0);
              treeView1.Nodes.Add(aNode);
              aNode.Nodes.Clear();
// Construct device node
            for (j = 0; j < classDatas.pDataPerson.Posts[gIdxPost].
Areas[i].Devices.Count; j++){
                TreeNode dNode = new TreeNode();
                dNode.Text = classDatas.pDataPerson.Posts[gIdxPost].
Areas[i].Devices[j].DevName;
                  dNode.Tag = TreeIdxToTag(i, j, -1, -1);
                  dNode.SelectedImageIndex = dNode.ImageIndex = SetupImgIdx (1, gIdxPost, i, j, 0);
                  aNode.Nodes.Add(dNode);
                  dNode.Nodes.Clear();
// Construct measuring node
                  for (k = 0; k < classDatas.pDataPerson.Posts[gIdxPost].
Areas[i].Devices[j].Items.Count; k++) {
                    TreeNode iNode = new TreeNode();
                    iNode.Text = classDatas.pDataPerson.Posts[gIdxPost].
Areas[i].Devices[j].Items[k].CheckContent;
```

```
                iNode.Tag = TreeIdxToTag(i, j, k, -1);
                iNode.SelectedImageIndex = iNode.ImageIndex =
SetupImgIdx(2, gIdxPost, i, j, k);
                dNode.Nodes.Add(iNode); }}}}
```

If the devices fail, the system will build a Fault table which shows device tree, fault phenomenon and fault reason. It’s realized as follows:

```
if (classDatas.pDataPerson.Posts[gIdxPost].Areas[gIdxArea].Devices
[gIdxDev].Items != null && gIdxItem >= 0 && gIdxItem < classDatas.
pDataPerson.Posts[gIdxPost].Areas[gIdxArea].Devices[gIdxDev].Items.Count)
{ FaultCode = classDatas.pDataPerson.Posts[gIdxPost].Areas
[gIdxArea].Devices[gIdxDev].Items[gIdxItem].FaultCode; }
```

The communication of Handhold Terminal and server is finished simultaneously by RDA (Remote Data Access). The communication is connected by USB or WIFI. Computer name must be input into the Dialog box of interface server when using USE and server IP address and port number should be input when using WIFI. System.Data.SqlServerCe is used to name Space. Pull is used to download data and SubmitSql is used to submit command to the database of server-side to add, updates, download data. It’s realized as follows:

```
// updates
public bool DoPull(string strSql, string strTableName) {
bool blRet = true;
SqlCeRemoteDataAccess rda = new SqlCeRemoteDataAccess();
rda.InternetUrl = this.InternetUrl;
rda.LocalConnectionString = Settings.AppSettings.ConnectionString;
rda.InternetLogin = "";
rda.InternetPassword = "";
try {
rda.Pull(strTableName, strSql, this.OleDbConnectionString,
RdaTrackOption.TrackingOff); }
// download
public bool DoSubmitSql(string strSql) {
bool blRet = true;
SqlCeRemoteDataAccess rda = new SqlCeRemoteDataAccess();
rda.InternetUrl = this.InternetUrl;
rda.LocalConnectionString = Settings.AppSettings.ConnectionString;
rda.InternetLogin = "";
rda.InternetPassword = "";
try {
rda.SubmitSql(strSql, OleDbConnectionString); }
```

IV. Conclusion

To realize the automation, networking and intelligence of enterprise equipment management, the system of equipment management was developed using Cloud calculating and Internet of things, which can reduce the equipment failure rate and downtime. The system realizes wireless connection and synchronization technology for instrument and equipment management system. It can also improve equipment reliability and reduce repair costs.

References

[1]W.D. Rustenburg,G.J. van Houtum,W.H.M. Zijm. Spare parts management for technical systems: resupply of spare parts under limited budgets. IIE Transactions, 2000,32(10)

[2] Mei Hong,Zhang Lu,Yang Fuqing. A component-based software configuration management

model and its supporting system.Journal of Computer Science and Technology.2002,17(4)：432-441.

[3] An-Hsiang Wang. Effects of Palm and WinCE menu-design for PDA on users operating performance and subjective preference .Displays, 2005, 97 (102) :97-102.

[4] Tan Zhang-lu, Zhang Chang-lu, YU Jin-zhi. Research on Construction of Equipment Management System in Coal Mines Based on Application of IOT. Coal Mine Machinery.2013.6, Vol.34 No.06

[5] Song Gui-rong，Cao Hong. Design of Mining Machinery Fault Diagnosis Integration Platform. Coal Mine Machinery. 2013.9,Vol.34 No.09