Researching into Human-computer Interface for the Railway Luggage Application System

Xu Weixiang[1], Xu Ting[2], Liu Xumin[3]

[1]School of Traffic and Transportation, Beijing Jiaotong University, 100044, Beijing, China
[2]School of Humanities and Social Sciences, Tsinghua University, 100084, Beijing, China
[3]School of Information Engineering, Capital Normal University, 100037, Beijing, China

Abstract

In this paper, authors have studied the application of remote condition monitor to the system used in railway luggage management information system (RLMIS). The human-computer interface for the railway luggage remote maintenance system (RLRMS) is intended to supervise the running state of RLMIS of China, to control general maintenance. This paper mainly focuses on the human-computer interface of RLRMS. Firstly, the framework of RLRMS is introduced simply. Then, the user interface design is explained in detailed, included user requirements, types of model, monitor states, threshold values and alarm. Finally, authors put forward an algorithm to monitor system load problems and propose a Kalman filter for the data transmission problem.

Keywords: Human-computer; Interface; Remote; Supervision; Maintenance.

1. Introduction

Modern remote monitor and maintenance is a result of the rapid growth of communication and information technique. With the advanced technique, remote monitor and maintenance takes advantage of the huge resource on the Internet, which realizes the resource sharing. Remote human-computer interaction interface is one of the advanced parts of the modern remote monitor and maintenance system [1,7].

The railway luggage management information system (RLMIS) is one of importance information subsystems of China railway. The railway luggage remote maintenance system (RLRMS) is design to supervise the running state of RLMIS and to control general maintenance. The system structure of RLRMS is Client/Server mode. The clients are objects of supervision, which are located at the large railway stations of China, such as Jinan, Xuzhou, Tianjin, Huhohaote, Zhengzhou, Guangzhou, Shanghai etc. The server is the center of supervision for the purpose of maintenance, which is located at the capital, Beijing.

Its main mission are (1) to obtain successful information about course and load of the system, cpu_used, mem_used, disk_used, and so on, (2) to alarm the operator when the load exceed the threshold value, (3) to adjust the load turn away the system breakdown. The supervision of such experiments needs efficient tools to control and visualize the different stations.

After the network engineering campaign of 2002, it has been decided to design the human-computer interaction interface for the supervision center, to enhance its accessibility for remote participation and to build a modular and distributed architecture adapted to control systems development. All the components of the architecture interact with each other in a very simple and standard way. The actual implementation of a human-computer interface is a question of user-ergonomics. Hence, a user-directed study in 2003 produced a specification for the interface. The information is treated with a hierarchical order.

This paper mainly focuses on the human-computer interface. The rest of the paper is organized as the following. Section 2 introduces a framework of the railway luggage remote maintenance system. Section 3 explains the user interface design in detailed, included user requirements, types of model, monitor states, threshold values and alarm. Section 4 discusses Kalman filter approach to process of transmitting. Section 5 is concluding remarks.

2. Architecture design

Over the past few years, there have been many advances in remote monitor and maintenance. Individually and together, these factors are resulting in the need for higher standards of remote check and measure quality [2]. At the same time, railway undertakings have to minimize the expenditure on providing redundancy to cope with exceptional circumstances. Therefore, the most important performance characteristics required are greater reliability and reduced maintenance cost, while consolidating safety improvements.

The supervision utility of RLRMS is mainly used on monitor server of client system, which distribute in networks of China railway. We used the intranet based on Ethernet that had connected the center of luggage supervision of China railway to all large luggage stations. The network operating system is Turbo Linux 9.0, the database server is Sybase 12.5, and developing tool is PowerBuilder 8.0. Fig.1 is data flow chart of system monitor in architecture structure.

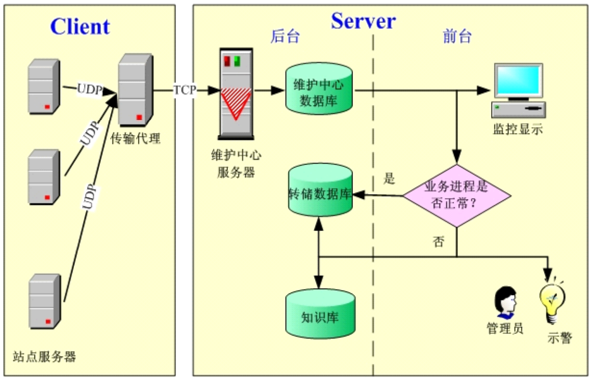


Fig. 1: Data flow chart of system monitor.

3. User interface design

3.1. User requirements

Requirements capture is an important part of all software engineering methodologies but often this activity focuses primarily on the system functional requirements - what the system must be able to do - with less emphasis on non-functional human issues such as usability and acceptability. Even where such matters are considered, they may reflect only the management's view of the user's needs rather then gathering information from the users themselves. User requirements model can redresses this balance [3].

3.2. Models

There are a number of models that can be used to capture a broader view of system requirements.

Socio-technical models are concerned with technical, social, organizational and human aspects of design. They recognize the fact that technology is not developed in isolation but as part of a wider organizational environment. It is therefore important to consider social and technical issues side by side.

Soft systems methodology takes an even broader view: that of the organization as a system of which the technology and people are components. It helps designers reach an understanding of the context of technological developments: the emphasis is therefore on understanding the situation rather than on devising a solution.

Participatory design is a philosophy, which encompasses the whole design cycle. The method is design in the workplace, incorporating the user not only as an experimental subject but also as a member of the design team. Users are therefore active collaborators in the design process, rather than passive participants whose involvement is entirely governed by the designer.

3.3. Users

The human-computer interface is designed for different users:

- the operations supervisor who keeps watch in the supervision center of remote maintenance,
- the computer maintenance man who controls data acquisition, supervision itself, system and network at the center,
- the computer administrators who diagnoses and adjusts the server of station.

So, the interface must be adapted to specialists of the different domains users.

3.4. Monitor states

The operator requires a fast and simple overview of the state of RLMIS, i.e. the whole state of the diagnostics systems (load, control, network and database). Information must be clear enough to diagnose any problem in any domain. That is why it is treated with a hierarchical order.

At the top level, the panel shows the general state of the main parameters of the system. If a problem occurs, the supervised parameter appears in red and the administrators has access to the detailed state by a simple double-click.

At the next level, the panel is specific to the supervised parameter, such as system course, system load, cpu_used, mem_used, disk_used, and so on. Indeed, the detailed state of a diagnostic is quite different from the detailed state of the system or the network. This state must help the operator to locate the origin of the fault and can suggest a corrective action if possible.

The lowest level corresponds to the maintenance and real-time information. This level is not essential for all plant systems since many of the specialists have their own tools to monitor and maintain the supervised elements.

So, at any time, the operator knows the functional state of RLMIS: first with a general approach and secondly for a detailed analysis of a fault situation.

3.5. Two states

The real state of a supervised parameter is very specific and constitutes part data rather than whole information.

The operator only needs to know if the state of the supervised parameter keeps in the field of using range, i.e. compliant with the expected state. As this state depends on the context. Showing this state on a panel allows to

- readily know if the parameter is OK or not,
- view a minimum of information: only a functional state is visualised, instead of the whole context,
- cater for non-specialists who do not know all the operational rules necessary to interpret a real state.

For any parameter, there are two way of shown. One is the histogram the other is the graph. The memory of system histogram is shown Fig.2 and Fig.3 shows the load of station system graph.

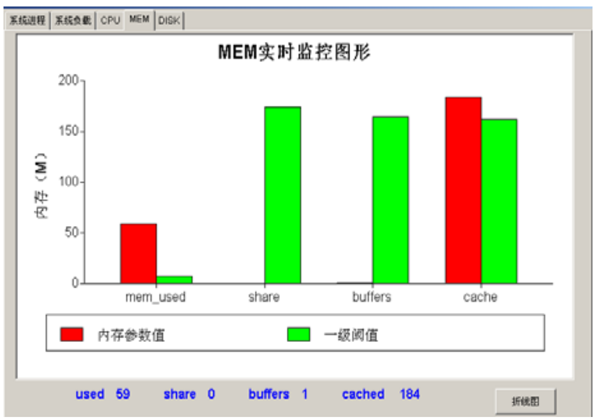


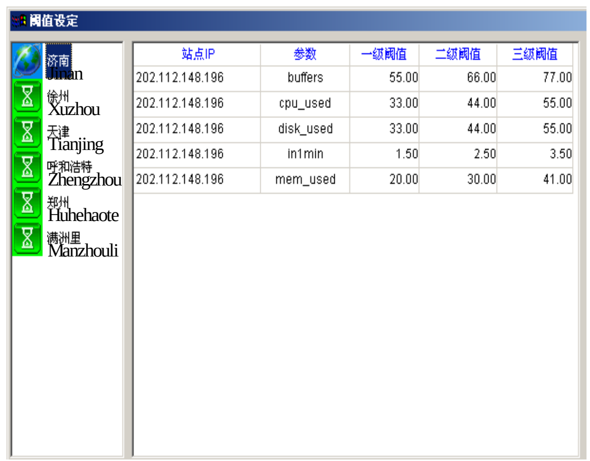
Fig. 2: Histogram of real-time monitor for the memory

Fig. 3: Graph of real-time monitor for the load of station system

When the data is sent, one state indicates if the diagnostic is 'OK'. This state is important because, during the operations, the administrator is only interested in the diagnostics involved with the parameter of system.

3.6. Threshold values

An empirical method is described, based on the concept of a permutation test, for estimating parameter's threshold values that are tailored to the experimental data at hand. The method is demonstrated using real data sets derived from experiment of system maintenance. In RLRMS, the supervised parameter threshold value is divided into three levels. For example Jinan station, which IP is 202.112.148.196, the threshold values of cpu_used are 33%, 44%and 55% respectively. Its defined is shown in Fig. 4.



站点IP	参数	一级阈值	二级阈值	三级阈值
202.112.148.196	buffers	55.00	66.00	77.00
202.112.148.196	cpu_used	33.00	44.00	55.00
202.112.148.196	disk_used	33.00	44.00	55.00
202.112.148.196	in1min	1.50	2.50	3.50
202.112.148.196	mem_used	20.00	30.00	41.00

Fig. 4: Threshold values of supervised parameters.

3.7. Alarms

The top and lower level panels show a whole snapshot state of the system. However, when an over loading occurs, it is also interesting to access the event chronology. Hence each detailed state contains an event log with priority, date and time event, parameter of the system involved, brief description and corrective action if any. The event log is a simple and powerful tool for the operator to

- locate an uncertain fault (the error remains in the event log until the operator deals with the problem his corrective action, if any),
- analyse dependant faults,
- deal with a deteriorating situation,
- carry the state and history information over to a new shift operator.

The description of an event also has a free format area where the specialist of the supervised parameter can insert some technical data.

Each state has a corresponding color: yellow's smiling face for 'OK', red's flare light for 'alarm'. The system course general graphical user interface are shown as Fig.4. It shows Jinan station general states, in which system load and disk_used are 'OK'; cpu_used and mem_used are 'alarm'.

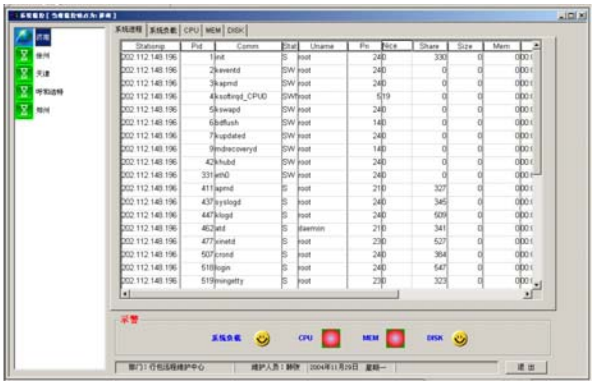


Fig. 4: The system course general graphical user interface.

On the GUI, the information is thus comprehensive and meaningful. Hence the design adapts to the different end users, specialists or not.

4. Kalman filter approach to process of transmitting

The Kalman filter approach is based on a set of mathematical equations that provide an efficient computational solution using the least squares method. The Kalman filter addresses the general problem of trying to estimate the state x of a discrete time-controlled process that is governed by a linear stochastic difference equation and where signals are of a Gaussian nature [4]. The objective in using Kalman filtering in this study was to increase the reliability of the model presented to the actual monitor system.

The Kalman filter estimates a process by using a form of feedback control: the filter estimates the process state at some time and then obtains feedback in the form of measurements. The equations for the Kalman filter fall into two groups: time update equations and measurement update equations. The time update equations are needed for projecting forward the load state and error covariance estimates to obtain a prior estimates for the next time step. The measurement update equations are responsible for the feedback. The time update equations can also be thought of as predictor equations, while the measurement update equations can be thought of as corrector equations [5,6]. Indeed, the final estimation algorithm resembles that of a predictor-corrector algorithm for solving numerical problems.

5. Conclusions

In order to achieve savings in traipse cost and in the maintenance of RLMIS it has become necessary to improve the monitoring of the load conditions, system states and associated management.

In preparing this paper, we have studied the application of remote condition monitoring to the course and load of the system, the parameter of cpu_used, mem_used, disk_used, and so on used in RLMIS. Data was collected on the system. The application software has been developed.

The objective in using Kalman filtering in this study was to increase the reliability of the model presented to the actual monitor system. Employing a Kalman filter, the reliability of algorithm was shown to be increase about 10 percent. Analyzing the difference between the actual data and the reference data in the form of absolute values, we can thus monitor the majority of faults as they develop.

References

[1] A. Lo´pez, J. Vega, A. Montoro, E. Sa´nchez, "Software and hardware developments for remote participation in TJ-II operation. Proof of concept using the NPA diagnostic", Fusion Engineering and Design, 60 (2002), 487-492.

[2] H. Yazdi, C. Roberts, S. Fararooy, "Intelligent Condition Monitoring of Railway Signaling Equipment Using Simulation", in: Proceedings of the IEE Seminar on Condition Monitoring for Railway Transport, Birmingham, November 1998.

[3] N. Utzel, B. Guillerminet, M. Leluyer, D. Moulin, "Java graphical user interface for the supervision of Tore Supra", Fusion Engineering and Design, 60 (2002), 415-420.

[4] F. P. Garc´ıa Márquez, F. Schmid, J. C. Collado, "Wear assessment employing remote condition monitoring: a case study", Wear, 255 (2003), 1209-1220.

[5] A.H. Christer, W. Wang, J.M. Sharp, "A state space condition monitoring model for furnace erosion prediction and replacement", Europe Journal of Operational Research, 101 (1997) 1-14.

[6] O.L.R. Jacobs, Introduction to Control Theory, second ed., Oxford University Press, Oxford, 1993.

[7] Carsten Wittenberg, "A pictorial human-computer interface concept for supervisory control", Control Engineering Practice 12 (2004) 865-878.