Software Design and Development of Chang'E-1 Fault Diagnosis System

Ai-Wu Zheng [1,2,3]
1 School of Astronautics, Beihang University, Beijing 100191, P.R. China
2 Science and Technology on Aerospace Flight Dynamics Laboratory, Beijing 100094, P.R. China
3 Beijing Aerospace Control Center , Beijing 100094, P.R. China
Email:awzheng@163.com

Yu-Hui Gao[2] , Yong-Ping Ma[2] and Jian-Ping Zhou[1]
1 School of Astronautics, Beihang University, Beijing 100191, P.R. China
2 Beijing Aerospace Control Center , Beijing 100094, P.R. China

***Abstract*— During a mission, detecting failure accurately and implementing countermeasures in time may decide the success of a space mission. Chang'E-1 mission is the first lunar exploration mission of China. The satellite has several different telemetry formats and bit rates. Moreover, the accuracy requirement of orbit control is high and demands real-time handling during contingency or abort operations. In addition, the time duration of the mission is very long, lasting more than a year. All these cause some difficulties in fault detection, judgment and handling in the mission. Therefore, in order to detect faults accurately and duly, determine the failure modes in time, implement countermeasures effectively, and provide warning and trend analysis, software of fault diagnosis system is specifically developed for Chang'E-1 mission. The system adopts plug-in software structure and advanced Berkeley DB real-time database. In addition, ice middleware is exploited for message passing between processes. It is also the first time that the failure criterion of the mission is described by extensible markup language (XML), and this well solves the problem of separating failure criterion from program coding. The fault diagnosis system can not only well and truly judge failures with clear criterion, but can also provides trend warnings for failures without clear descriptions, and is very helpful for the personnel who are in charge of fault detection. The advanced technologies applied in the system make it expansible. It can also be used in other space missions.**

***Index Terms*—Chang'E-1 mission, fault diagnosis, plug-in, real-time database, middleware**

I. Introduction

Because of the complexity of the space environment and testing limitations of a spacecraft, sometimes there will be contingency or system failure during the space missions. The four most serious accidents in manned space flight history where astronauts were killed are Apollo 4A in January 1967, Soyuz-1 in April 1967, Soyuz-11in June 1971 and Challenger in January 1986[1]. Serious accidents are also happened in Hubble Space Telescope of the U.S. in 1990, Mars Climate Observer and Mars Polar Lander in 1999. In November 2000, the Indian satellite INSA-2B lost the function of pointing to the Earth, and the accident also caused huge losses [2].These accidents make countries begin to attach importance to the research of spacecraft fault diagnosis techniques, in order to detect failures in time, and to avoid and reduce casualties and equipment losses as much as possible, thus save spacecraft launch and operation costs [3, 4].

Chang'E-1 mission was the first lunar exploration mission of China. The satellite has several different telemetry formats and bit rates. The orbit control requires high accuracy and demands real-time fault handling. In addition, the time duration of the mission is very long, lasting more than a year. All these cause some difficulties in fault detection, judgment and handling. According to the orbit design scheme of the mission, the satellite needs two weeks' time from launch to finally going into the target orbit, and stays there for a year [5].If fault diagnosis entirely relies on manual monitoring, the personnel will be exhausted, and it is still very easy to miss some failures. According to the analysis during mission preparation, there would be more than eighty kinds of possible failure modes, and some failure would happen in a very concentrated time. For example, before and after the first brake maneuver near the moon, nearly twenty types of fault may occur intensively, and most of them are emergent requiring urgent disposal, or the satellite may be dangerous or it will fly out of the gravitational field of the moon.

Therefore, in order to monitor satellite status and detect fault accurately and duly in Chang'E-1 mission, decide prepared failure mode in real-time, implement countermeasures effectively, and provide warning and trend analysis, software of fault diagnosis system is specifically developed as an assistant to improve the

Manuscript received August 1, 2011; revised September 15, 2011; accepted September 20, 2011.
Copyright credit. Project 11173005 Supported by National Natural Science Foundation of China

doi:10.4304/jsw.7.12.2687-2694

timeliness, accuracy and efficiency of fault diagnosis, and to guarantee the success of the mission.

II. SPACECRAFT FAULT DIAGNOSSIS TECHNOLOGY

TABLE I
A BRIEF SUMMARY OF FAULT DIAGNOSIS TECHNOLOGY

Type	Advantage	Disadvantage
Signal processing-based fault diagnosis method	Threshold model is adopted and is the basis of other diagnostic methods.	
Rule-based expert system diagnosis method	1) Simple, intuitive, image and convenience; 2) Rapid diagnosis; 3) Require a relatively small data storage space; 4) Easy to program and implement.	1) Covers limited failure mode; 2) Misdiagnosed or diagnosis failure.
Fault tree based fault diagnosis method	1) Rapid diagnosis; 2)Easy to dynamically modify knowledge library and maintain consistency; 3) Domain-independent, as long as the corresponding fault tree given, the diagnosis can be achieved.	1)Can not diagnose unexpected failure; 2) Diagnosis results rely heavily on the information of the full extent of the fault tree.
Neural networks based fault diagnosis method	1) Highly nonlinear; 2) Highly fault tolerance; 3) Associative memory.	1)Can not reveal potential relationships within the system, can not give a clear explanation of the diagnostic process; 2) Can not diagnose failures that never appeared in the training sample, or even draw wrong diagnosis conclusion.
Model-based fault diagnosis method	1) High accuracy; 2) Can diagnose unforeseen failures, no need for experience and knowledge.	1) The diagnosis is slow; 2) Strong dependence on the accuracy of the model, any uncertainty of model can lead to false alarms.
Petri net-based fault diagnosis method.	Can dynamically describe the generation and propagation of fault phenomena, making it easy to diagnose from the change of the system	1) Totally dependent on the Petri net model; 2) Difficult to locate the source of the fault, problem-solving process is prone to conflict.

According to the domestic and foreign development history of fault diagnosis technology, a brief summary of the main methods is listed in table I [6-9]:

Currently fault diagnosis system has developed from fault diagnosis expert system of single subsystems (such as power system or thermal control system) to integrated spacecraft health management system integrating system status monitoring, fault diagnosis and fault repair as one[10]. Fault diagnosis method has developed from single diagnostic method (such as rule-based diagnosis method, fault tree based diagnosis method, etc.) to the combination of various methods. With the rapid development of computer technology, many new technologies is used in fault diagnosis system, such as network technology, information fusion technology, distribution theory, agent and multi-agent system technology, and excellent man-machine interface. These technologies provide strong support to the development and maintenance of fault diagnosis system.

Domestic research of spacecraft fault diagnosis technology started late. In recent years, domestic aerospace experts have gradually recognized the importance and urgency of the work in this area, and have done some research theoretically. But most diagnosis systems developed are demonstration systems.

Chang'E-1fault diagnosis system is the first fault diagnosis system applied to an actual mission. The system is a rule-based expert system. It successfully separated specific failure criterion from program coding by using XML-based markup language description. It also uses advanced real-time Berkeley DB database, ice middleware and plug-in software architecture. The system provides successful experience of fault diagnosis for future missions.

III. CHANG'E-1FAULT DIAGNOSIS SYSTEM DESIGN

A. System Functional Requirements

The task of Chang'E-1 fault diagnosis system is to provide technical support of centralized satellite status monitoring, fault diagnosis and emergency disposal for commanders and flight control operators. The system can improve the ground ability of centralized monitoring and real-time analysis of important satellite status, and can provide early warning and trend alarms, help the mission decisions.

The function of the system requirements include:

1) Telemetry monitoring, alerting and analysis.

The system receives real-time telemetry data package from the monitor display network, checks the correctness of the data formats and explains telemetry frames for associated judgments. When there is a telemetry interruption or overrun, alarm messages are given and corresponding parameters are displayed.

2) Real-time monitoring and alarm of satellite important parameters and events.

During the telemetry monitoring, alarm and analysis, the system also receive mission control plans, station tracking forecasting, telecommand sending sequence, orbit elements and other information to determine satellite important status and mission critical events.

3) Real-time fault inspection, diagnosis and alarm

According to different phase of the mission, one or more failure modes that are most possible to occur are selected, and the corresponding telemetry parameters of failure criteria are centralized displayed and monitored.

4) Fault handling support

Once a fault is conformed to occur, the system will prompt corresponding disposal measures step by step. During the fault handling, parameters involved and status changes are displayed in real-time to determine the implementation effect.

5) Fault trends warning and analysis

Some parameters or state change is a gradual process, such as temperature, pressure and voltage, etc., the system will monitor and analyze the trends. If the trend is close to the failure threshold, it will give early warnings.

B. System Flow

The fault diagnosis system receive telemetry packages, control plans, station tracking forecasting, telecommand sending sequence, orbit elements and other information to determine satellite important status and mission critical events. The outputs are alarms of failure or other unusual events, decision support information, all kinds of status messages, and data analysis and statistics. The system block diagram is shown in Figure 1.

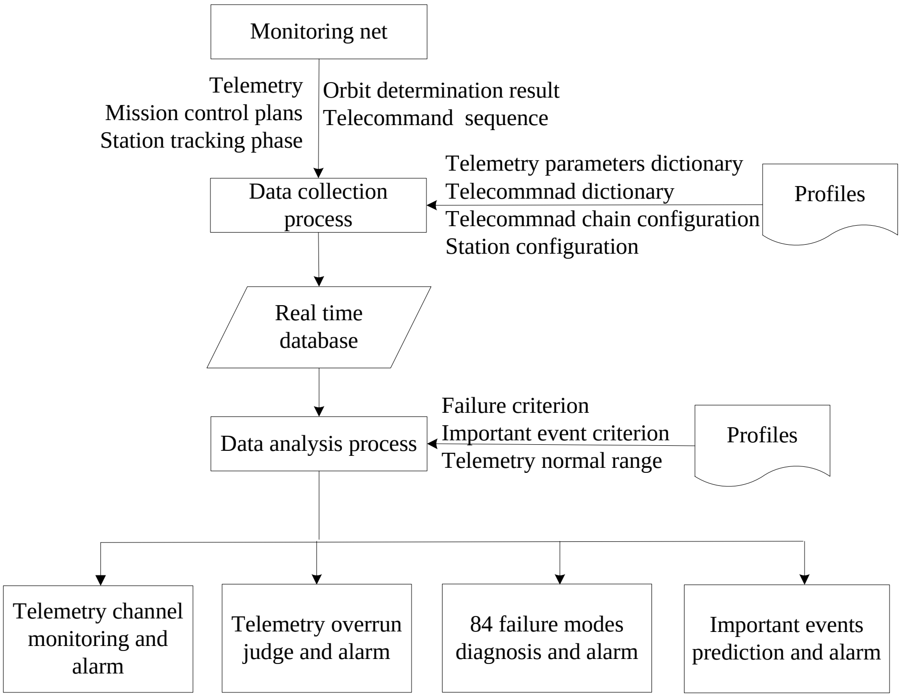

Figure 1. Block diagram of fault diagnosis system

C. System Hardware Structure

Fault diagnosis system is independent of Chang'E-1 mission control system. It is connected with monitor display network as shown in Figure 2. It receives the original data and results issued from the monitor display server directly as same as monitor display terminals.

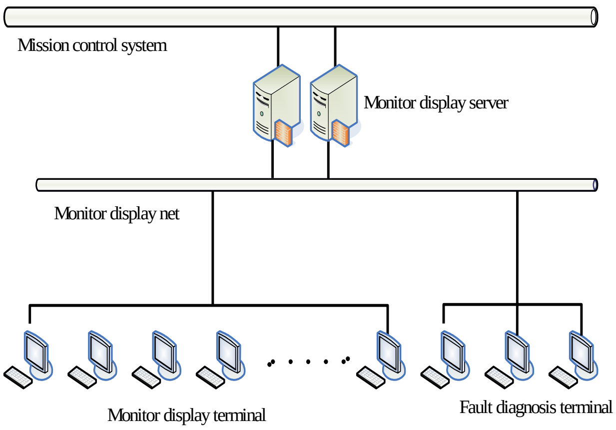

Figure 2. Network connection diagram of fault diagnosis system.

Each fault diagnosis terminal is equal, fault diagnosis system can run in any one of them, and can maintain consistency of monitoring, early warning and analysis by performing information and version synchronization operations in the process of fault diagnosis.

D. Software Hierarchy

The software hierarchy model of the fault diagnosis is shown as Figure 3.

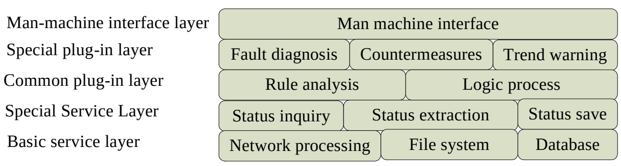

Figure 3. Fault diagnosis system hierarchy.

There are five layers in the system:

1) Basic service layer

Basic service layer is the foundation of the fault diagnosis system including network processing, file system and real-time database. It provides the ability of obtaining various mission information, spacecraft status and configuration information from the network, files and database for special service and higher layers.

2) Special service layer

Special service layer provides inquiry, extraction and storage of spacecraft status, configuration information and fault rules for fault analysis and diagnosis.

3) Common plug-in layer

Common plug-in layer includes rule explanation and logic process etc. Rule explanation plug-in provides explanation of labels, methods and logic operations defined in XML files, which describe failure modes. Logic process plug-in provides functions listed in table II to process logic operations of logical expressions.

4) Special plug-in layer

Special plug-in layer includes plug-ins of fault diagnosis, countermeasures and trend warning. They are based on common plug-in layer, and provide the ability of fault diagnosis, countermeasures and trend warning by rule explanation and logic process.

5) Man-machine interface layer

Man-machine interface layer provides interface for operations, configuration and management.

E. XML-Based Fault Rule Design

In order to determine the failure modes of the mission quickly and accurately, the spacecraft status and its change must be translated in a language that a computer can recognize. This requires the language easy to understand, easy to write and can give a clear description of failures. Because XML has the ability to self-describe data in a structured way and can be defined and used according to the needs of application, so it is selected for rule description of spacecraft fault diagnosis.

First, mathematical and logic operators are defined. Criterions of failure modes and significant events are expressed in the language to generate formulas of a single or several telemetry parameters, or a trend of change, or even a combination of several judgments. The formulas generate rules and are written into files. Finally, files are stored into the rule library as fault diagnosis criterions. When data analysis process reads the rules from the files, it will parse the rules to computer.

Rules are independent of program, so program needs no modification when rules change. The rules can be added or modified at any time, achieving the goal of separating a specific fault from program. When the criterion of an important event or a failure mode changes, it is the corresponding rule file that only needs to be modified. The rule library requires regular maintenance to be consistent with the mission.

F. Design of Formulas

Rule description of status is the key problem that is needed to solve. Rule formulas can use common math and logical operators directly, or introduce external functions to extend operation capabilities.

External functions are divided into several types including citing, extraction, judgment, computation and operation, etc. Citing is the type in the form of “FromXXX”, which gets related status information directly from telemetry parameters, orbit forecasting and control strategies. Extraction is the type in the form of “GetXXX”, which can obtain auxiliary information of other parameters through appropriate treatment. For example, function “GetPrev” can obtain the previous value of a specified parameter. “GetBit” can obtain a particular bit of a specified parameter. Judgment is the type in the form of “JudgeXXX”, which determine the trend of a parameter. For example, function “JudgeDecrease” is used to determine whether a specific parameter is in a increasing trend during a certain period. “JudgeChange” is used to determine whether a specific parameter remains unchanged during a certain period. Computation type is only one function named “Compute”, which gives the result of the logical expressions. Operation type is used to call a specified function to

TABLE II
FUNCTIONS FOR FAULT RULES

Function name	Function contents	Parameter description
FromGlobal	Direct from external global parameters	[parameter identification]
FromTM	Direct from external telemetry library	[telemetry library, parameter identification]
FromPlan	Direct from external plan	[plan identification]
FromPredict	Direct from external orbit forecasting	[forecasting identification]
GetFree	Read a parameter from a user’s input.	[parameter type, default value]
GetBit	Get bit from a byte.	[parameter code, starting bit, total bits]
GetPrev	Get the previous value of a telemetry parameter.	[parameter code]
GetAverage	Get the average value of a parameter during a certain period.	[parameter code, time(s)]
GetUpdateTime	Get the updated value of a parameter.	[parameter code]
GetLastTrueTime	Get the latest time of an event happened.	[sub-event code]
GetLastTrueBeginTime	Get the earliest time of an event happened.	[sub-event code]
GetHistory	Get the history value of a parameter.	[parameter code, numbers]
GetDecVal	Get the decrease amount of parameter during a certain period.	[parameter code, time(s)]
GetIncVal	Get the increase amount of parameter during a certain period.	[parameter code, time(s)]
GetAdd	Get the sum of several parameters.	[parameter code1, parameter code2,…]
GetMulti	Get the product of several parameters.	[parameter code1, parameter code2,…]
GetMax	Get the maximum of a parameter during a certain period.	[parameter code, time(s)]
GetMin	Get the minimum of a parameter during a certain period.	[parameter code, time(s)]
GetTmUpdateTime	Get the time of telemetry updating.	-
JudgeIncrease	Judge the change of a parameter if it is in a increase trend during a certain period.	[parameter code, time(s)]
JudgeExceed	Judge the parameter if it has exceeded.	[parameter code, lower limit, upper limit]
JudgeChange	Judge the parameter if it has changed.	[parameter code, time(s)]
Compute	Compute the value of a formula.	[logical formula]
OperateEvaluate	Assign a value to a variable	[function expression]
OperateLog	Operation log recording	-

operate such as assign a value. Another example, “OperateLog” can perform log record operation. Some of functions are listed in table II.

G. Real-time Database Technology

Berkeley DB real-time database, also known as an embedded database, is applied in the system. It is embedded in the application, and it is suitable for managing vast amounts of simple data up to 256TB. Because the application and database management system are running in the same process space, thus cumbersome inter-process communications can be avoided in data operations. So it is more efficient than relational database.

The advantages of embedded database are:

Firstly, cumbersome inter-process communication such as establishment of socket connections is avoided, so the overhead in communication cost is greatly reduced.

Secondly, Simple function call instead of frequently used SQL language is used to complete all database operations. Thus it saves time to parse and process query language.

In Chang’E-1 fault diagnosis system, real-time database is used to save the initial satellite status obtained from mission monitor and display network. Quick and accurate fault diagnosis needs to know real-time satellite status, as well as history satellite status for trend warning. Berkeley DB database can meet the demands.

H. Middleware Technology

Ice middleware was exploited for message passing and status synchronization between processes in the system.

Ice middleware technology is an object-oriented middleware platform. Basically, Ice middleware provides tools, API and library support for object-oriented client-server applications. Ice applications are suitable for use in a heterogeneous environment, where client and server can use different programming languages, different operating systems, different machine architectures and a variety of network technologies. So the fault diagnosis system can use different environment from the monitor display server.

IV. PAGE DESIGN AND SOFTWARE IMPLEMENTATION

A. Design of Functional Modules

Fault diagnosis system uses multiple processes. It is mainly composed of two processes. One of them is data collection process. It collects data from the monitor display net, explains the data frame, and stores the data into the real-time database. Another process is data analysis process. It extracts data from the real-time database for data analysis. Then it judges if a failure mode has happened based on the analysis. Finally, it gives warnings or alarms. The Real-time database provides storage and access of data. Data collection process functions as a write user of the database while the data analysis process functions as a read-only user. The relations have shown in Figure 1.

At the same time, each process uses a plug-in design. Plug-in is a kind of flexible component software architecture. Functional modules are implemented by plug-ins instead of conventional single program execution. The plug-in can be independent of program module, it can be developed solely, then loaded into the system when running, and can be deleted or replaced at any time. Thus, the extensibility and flexibility of the software are improved [11].

B. Fault Diagnosis Process

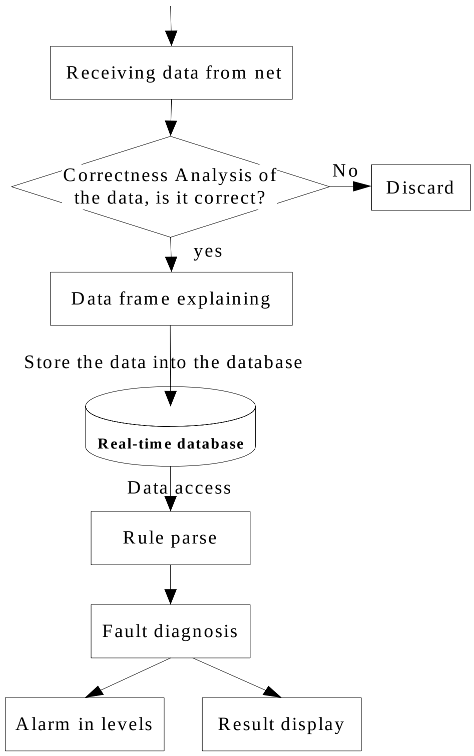

Figure 4. Fault diagnosis process.

Fault diagnosis process is shown in Figure 4. On one hand, the software stores the real-time online data from monitoring net into the database. On the other hand, it reads fault rules in the library, calls interpreter to parses the rules, determines the status of the satellite and fault conditions, then gives alarms. Diagnostic results are displayed in pages.

C. Implementation of Fault Rules

The fault rules includes three levels of description: status, event, and parameter.

Parameter rule is the basic unit of representation. In parameter configuration, satellite telemetry parameters can be used directly or new parameters can be created by formula. A parameter configuration is defined as following:

```
< Parameter Configuration>
  < Parameter Code ="xxx" Description="xxx" Formula ="xxx"/>
  < Parameter Code ="xxx" Description="xxx" Formula ="xxx"/>
  ……
</ Parameter Configuration >
```

Event rule is expressed by logic operations of one or more parameters. The code and description of the event are needed, which is shown as follows:

```
<Event Configuration Created time =" ">
<Event Code=”A” Description=”A”>
<Parameter Code = "xxx" Description = "xxx" Formula = "xxx"/>
<Parameter Code = "xxx" Description = "xxx" Formula = "xxx"/>
……
</Event>
<Event Code=”B” Description=”B”>
<Parameter Code = "xxx" Description = "xxx" Formula = "xxx"/>
<Parameter Code = "xxx" Description = "xxx" Formula = "xxx"/>
……
</Event>
……
</ Event Configuration >
```

Status rule is used for satellite status description. A status can be expressed by one or more events. The code, description and type of the status can be defined. And the level of the event can also be defined, shown as follows:

```
<Status Configuration >
  < Status Code ="xxx" Description ="xxx" Type="xxx">
    < Event Code ="xxx" Level="xxx"/>
    < Event Code ="xxx" Level="xxx"/>
    ……
    </Status>
</ Status Configuration >
```

D. Fault Checking Periods

Some failure modes can only occur in special periods, such as at the moment when satellite separates from the rocket, before or during orbit maneuvers etc. Some failure modes may occur throughout the mission. As for the disposal methods, the same fault occurred at different times may need different disposal. So period is introduced in fault diagnosis. The failures are checked according to the period that it may possible occur to improve the efficiency of the system. At the same time corresponding countermeasure is given according to the different period.

E. Alarm Mode

Alarms are notified in following ways:

1) Sound alarm

Give voice prompts according to the fault content, play the corresponding sound files.

2) Color Alarm

The color is distinguished by yellow and red, directly colored in the parameter or the corresponding button. Yellow is a warning color, suggesting that failure may occur. Red is an alarm color, suggesting that failure has occurred.

3) Warning dialog box

Pop-up a warning dialog box according to the failure contents, giving a specific cause of the failure occurred.

4) Alarm log

Write all abnormal information system monitored and all alarms into alarm logs. Alarm color remains for convenience for users to query.

F. Pages Designs

Many different pages have been designed for fault diagnosis system. One of which is the page displaying all prepared failure modes, including alarm diagnosis, alarm and monitoring, shown in figure 5.

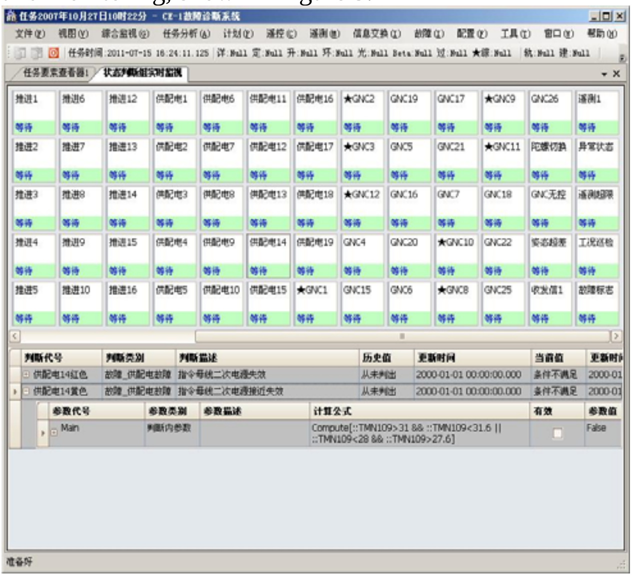

Figure 5. Judgments and alarm of prepared failure mode.

The comprehensive monitoring of each failure can be inquired in detail from the page shown in Figure 6.

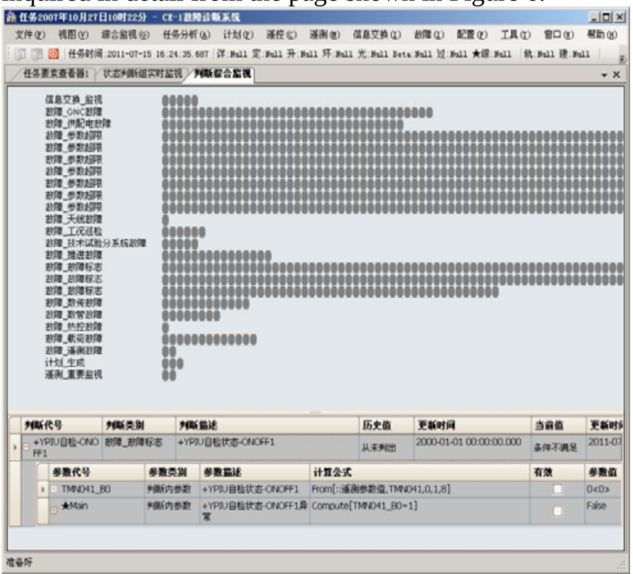

Figure 6. Comprehensive monitoring of failure mode

Figure 7 is the display of several tab pages, including real-time display of state judgments, parameter monitoring, alarm log, mission process viewer and telecommand sending monitoring.

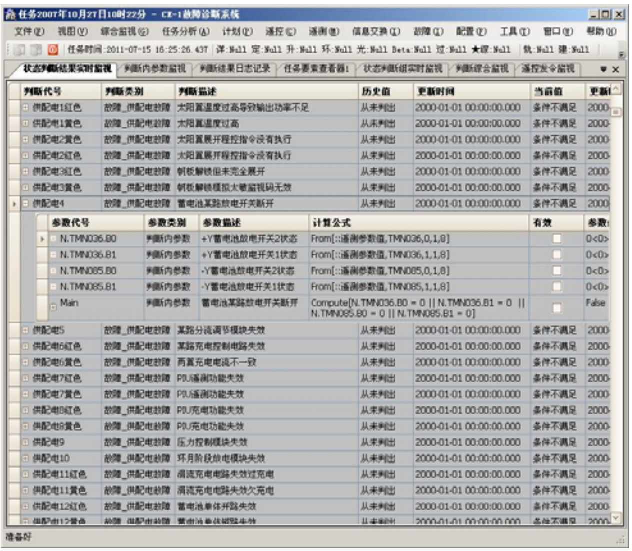

Figure 7. Tab pages display

G. System Testing

After the completion of program design and coding of Chang'E-1 Fault diagnosis system, the software is tested many times in failure drillings of mission preparation. And the fault rules are modified according to the diagnosis results. For a clearly described failure mode, only 3 telemetry frames, or in another word 6 seconds, are needed to judge the failure mode (one telemetry frame is equivalent to 2 seconds when the bit rate of telemetry is 512bps. The software uses 3 sentence 2 decision mechanisms).

However, some failures can not be quantitatively described themselves such as attitude abnormal, so it is difficult to judge whether the failure has happened, but the system still can provide trend analysis according to the changing trend of some parameters.

Because the rules are independent of program, so once the rules are modified and reloaded, the system will automatically diagnose according to the new rules. This makes the system very easy to upgrade.

IV. CONCULSION

Chang'E-1 fault diagnosis system is a multi-process program. The data collection process collects data from the monitor display net, processes data format and then store the data into the real-time database. At the same time, several data analysis processes are designed to extract data from the database for data analysis, judge failure mode, and gives warning or alarm. Every process adopts plug-in software structure. Advanced BerkeleyDB real-time database of the system not only saves huge amounts of data and results, but also provides high efficiency of data query. The system also used Ice middleware for message passing between processes. Based on the experience of the flight control of satellites, a XML language is used for rule description of failure mode. It is a solution to separate failure criteria from program. The system can not only rapid judge the failures clearly described, but also can provide trend warning for modes that can not be quantitatively described, help the mission operators to analyze. Chang'E-1fault diagnosis system is very helpful for fault diagnosis in the mission.

The system is the first fault diagnosis system applied to an actual mission. It provides successful experience of fault diagnosis for future missions.

ACKNOWLEDGMENT

The authors wish to thank Dr. Song-Jie HU. This work is supported in part by a grant from National Natural Science Foundation of China (Grant No. 11173005).

REFERENCES

[1] Zuojun Shen, "Statistical analysis of Manned space flight failure and system research of the safety requirements," *Aerospace China*, No.11, pp. 21-24, 1997.

[2] Newman J. Steven, "Failure-Space: a systems engineering look at 50 space system failures," *Acta Astronautica*, 48 (5-12), 2001, pp.517-527.

[3] William S. Morris, Timothy Hill, et al, "Advanced fault management for the space station external active thermal control system," in: *22nd International Conference on Environmental Systems*, Seattle, WA, July 13-16, 1992.

[4] A.S. Aljabri, D.E. Bernard, D.L. Dvorak, G.K. Man, B. Pell, T.W. Starbird, "Infusion of autonomy technology into space missions-DS 1 lessons learned", in: *Proc. IEEE Aerospace Conference*, Snowmass, CO, 1998.

[5] Yang Wei-lian, Zhou Wen-yan."Orbit Design for Lunar Exploration Satellite CE-1", *Spacecraft Engineering*, 16 (8), 2007, pp.16-24.

[6] J. Taylor, "An algorithm for fault-tree construction," *IEEE Transactions on Reliability*, 31(2), pp. 219-254, June 1982.

[7] M. G. Madden, P. J. Nolan, "Monitoring and Diagnosis of Multiple Incipient Faults Using Fault Tree Induction," *IEEE Proceeding on Control Theory and Applications*, 146(2), March 1999.

[8] Long Bing, Song Lihui, Jing Wuxing, Jiang Xingwei, "A Review and Propect of Fault Diagnosis Technique of Spacecrafts," *Missiles and Space Vehicles*, (3), pp. 31-37, 2003.

[9] F. N. Pirmoradi, F. Sassani, and C. W. deSilva, "Fault Detection and Diagnosis in a Spacecraft Attitude Determination System", *Acta Astronautica*, 65, pp.710-729, March. 2009.

[10] Fox Jack J , et al. "Informed maintenance for next generation reusable launch systems," *Acta Astronautica*, 48, pp.439～449, 2001.

[11] Wolfinger, R., Dhungana, D., Prähofer, H., "A Component Plug-In Architecture For The .NET Platform," in: *7th Joint Modular Languages Conference (JMLC 2006)*, Oxford, UK, September 13-15, 2006.

BIOGRAPHY

Ai-wu Zheng born in 1974. She received her M.S. degree from the National University of Defense Technology in 1999. Now she is a doctor student of School of Astronautics of Beihang University majored in Spacecraft design and Engineering. Senior engineer. Her main research is mission design.

Yu-hui Gao born in 1978. He received his M.S. degree from Beihang University in 2010. Engineer. His main research area is computer architecture.

Yong-ping Ma born in 1964. He received his Ph.D. degree in spacecraft design from Beijing University of Aeronautics and Astronautics in 2010. Researcher. His main research area is spacecraft flight control.

Jian-ping Zhou received his Ph.D. degree in Solid Mechanics from the National University of Defense Technology in 1989. Ph.D. Supervisor, Professor, Chief Design of China manned space flight program.