

A Design of Test Telemetry Data Management System with Embedded Quality Bus

Wei Wang^{1, a}, Ruipeng Luan¹

¹ The Chinese People's Liberation Army Unit 91550, Dalian, 116023, China

^aemail: weiWang_DL@163.com

Keywords: Telemetry Data; Quality Management; Quality Bus

Abstract: In view of the existing problems in the telemetry data management of aircraft test, the design scheme of the test data management system with embedded quality bus is proposed. The design idea of quality bus, architecture and key technologies of the system are described, and the superiority of the system in quality control, data sharing and system security are analyzed.

1. Introduction

With the development of data acquisition, processing and transmission technology, the parameters of the parameters of the telemetry data, the acquisition channel and the total amount of data processing have been greatly improved[1]. At the same time, the requirements for the quality of the data of the aircraft test also mention an unprecedented height. How to do a good job in quality management in the era of large data Work is a problem to be solved urgently. This paper presents an experimental telemetry data management system with embedded mass bus. It describes the system from the design idea, architecture and key technology, and analyzes the advantages of the system in quality control, data sharing, system security.

2. The Problems in the Management of Test Telemetry Data

The narrow sense of test is the process of obtaining data [2]. The data processing is the key link in the whole process of testing. Due to the complexity of the test task, organization, equipment and environment, the experimental telemetry data has its own characteristics, which is mainly shown as the following four points:

1) The task has specific requirements. It is necessary to organize different measurement and control equipment and obtain various parameters accordingly. In addition, the experimental environment is different and the test target is different, and the storage and organization of each data shows great variability [3].

2) There are a wide variety of equipment parameters, including fast change, slow change and so on. Each parameter has different data formats. The original data are diverse, not only binary string, but also images and video, and static data (environment parameters, earth results) and dynamic data (the process data collected in the test) coexist.

3) With the improvement of the aircraft assessment requirements, the test time is getting longer and longer, the number of channels for telemetry data acquisition is more and more, the process of aircraft testing is becoming more and more complex, and the amount of data produced by the test task is increasing geometrically.

4) The quality of the telemetry data of the aircraft test is related to the evaluation of the test results of the aircraft and the assessment of the state of the data acquisition equipment. The value and function are very important and the quality is very high.

3. The Design Idea and Key Technologies of The System

In view of the problems in the management of the test telemetry data and the special requirements for the quality, the design idea of embedded quality bus is put forward in the design.

Four design principles of object modeling, role binding, mass storage and hierarchical structure are proposed in order to combine modern management with quality management. Build a unified management platform with quality management, security, flexibility, integration and efficiency.

3.1. The Design Idea of Quality Bus

The design idea of the quality bus is the focus and entry point of quality management, which is refined to each batch of data processing in the telemetry data processing process through information technology. The idea of quality bus design proposed in this paper is to design a set of standardized naming rules first, and use this rule to standardize each batch of data in the process of telemetry data processing. The ‘test name - quality bus information table ’ is set up to record ‘serial number, process number, system name, parameter name, use software and version, calibration data file and version, calibrated data file, check date check, data consistency check, etc.’.

The data form of generating parameter A for a processing stage is telemetering, named "processing parameter A" (Table 1), and the test name quality bus information table (Table 2) and the mapping relation between the two are shown in Figure 1.

Tab.1. Telemetry data processing stage, sub system, data form for generating parameter *n*

Serial number	time code	parameter n
1	12:22:25.000	Value 1
2	12:22:25.250	Value 2
...

Tab.2. Test name, quality bus information table

Serial number	Process parameter	subsystem	parameter	software and version	calibration data source	data validation period inspection	data consistency check
1	01	control system	height	TelePostProceNew.exe	Product specification	yes/ name	yes/ name
2	01	electrical system	voltage	TelePostProceNew.exe	Product specification	yes/ name	yes/ name
...

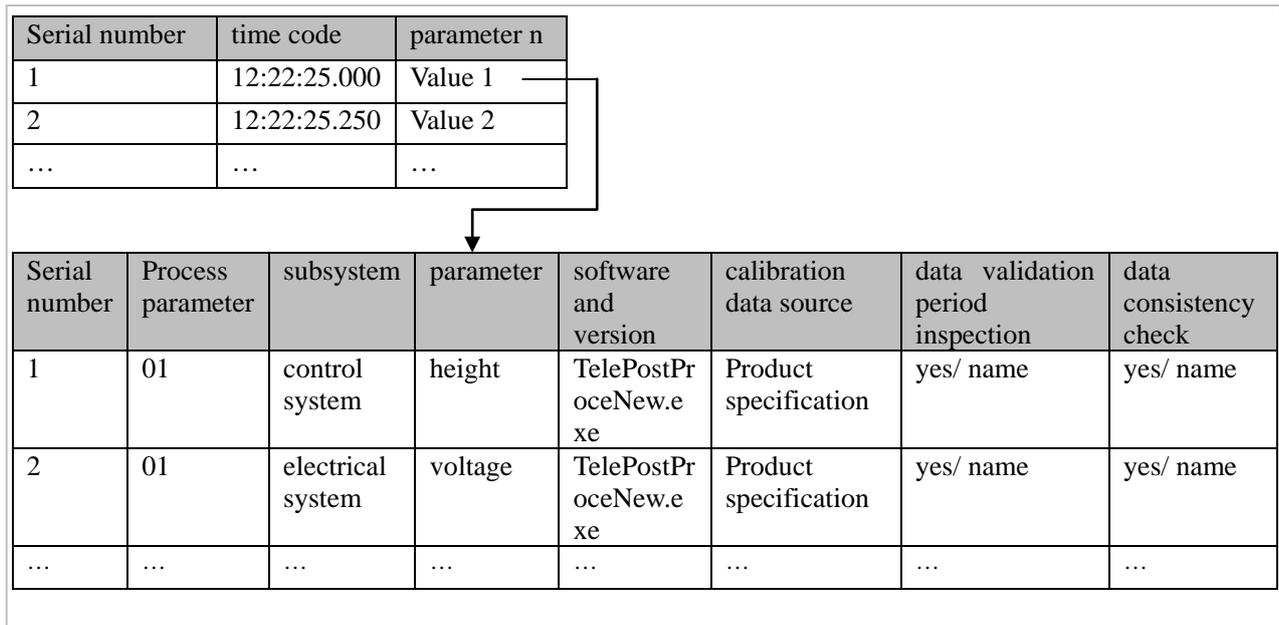


Fig.1. Mapping relation between forms

After the bottom modeling is completed, the different processing batches, different types of parameters, and different test tasks are turned upwards, and the quality tables or information tables are set up for the generation of equipment and static data (time origin, environment parameters,

earth results, and the parameters of the subject object). The forms are linked by the flow path and combined with the macro quality system. Quality records, work instructions, program files, quality manuals and other quality management auxiliary files are used to form a top-down quality management bus - the upper level can be traced to the result report, the bottom layer can be finely positioned to each data point, the whole operation process and the data are covered in the middle, and all of them can be examined horizontally.

3.2 Object Modeling

System technology runs through the idea of object modeling. Data types, quality forms, database bottom operations are encapsulated in the form of objects, allowing role users to customize their own database models through data objects, including data tables, data views, data check rules and constraints [4]. To meet the needs of users, such as adjusting data structure, organizing and searching test data, increasing the conditions and rules of data verification, and so on, to adapt to the changing requirements of the user's test task.

3.3 Role Binding

Role based access control (RBAC) is the most mature security strategy in today's large-scale information systems. The user's permission to the system operation is not directly granted to the user. Instead, a role set [5] is set up between the user set and the permission set (Figure 2). The user assigns the appropriate role to have the relevant permissions of the role, changing the one to one relationship of the direct authorization into the multi to multi relation of role binding. When the factors, such as personnel, equipment, post, process, and task, are easy to decouple, it is more flexible and robust and avoids the problem of authority. By mistake deletion, misoperation and leakage, the security and flexibility are greatly improved.

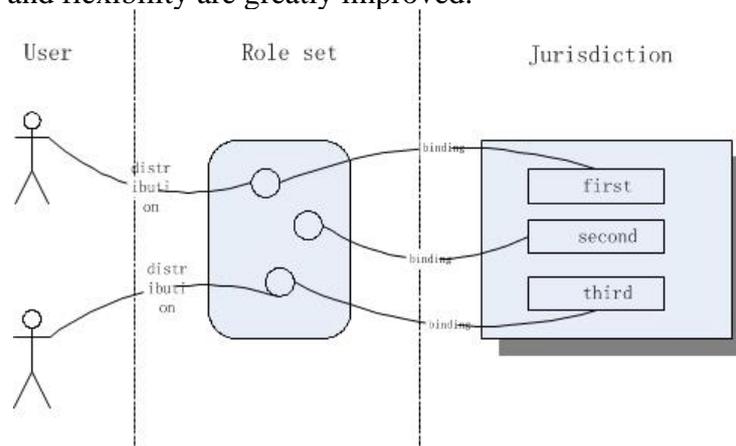


Fig.2. Role binding diagram

3.4 Mass Storage

In view of the characteristics of the variety of data and the large amount of data, the database management system Oracle 9i, which is mature (avoiding the existing version of the new technology), is used to implement all data types (external, telemetry, GPS, earth results, environmental parameters, equipment parameters, test information, etc.) and all data formats (Massive storage of binary strings, character text, pictures, videos, word documents, etc. Because of the large amount of data in the test task, the storage performance is a key index, and it should be above 3G/ seconds in theory. The Oracle 9i database system uses caching fusion (Cache Fusion) technology to exchange information [6], the peak of communication can reach 12G/ seconds, which is fully qualified for system requirements. Oracle 9i also provides powerful online analytical processing (OLAP) capabilities, providing strong technical reserves for online data mining and knowledge discovery.

3.5. Hierarchical Structure

The system uses three layers of architecture, which divides the whole business application into presentation layer, business logic layer and data access layer. The hierarchy structure realizes the concept of software engineering design, divides the problems and solves each problem, which is beneficial to the integration, expansion, maintenance and deployment of the system. The .Net framework can achieve a better implementation of the three layer architecture, with component assembly functions. The background code uses its own defined components through namespace. It can use C#, C++, VB and other programming language coding and encapsulation classes to implement the business logic [7], and provide a large number of database connection components to help implement the data access layer function. The presentation layer can embed business logic and data layer operation classes into ASPX pages through ASP.net, and realize the automatic publishing of Web portal. It is very suitable for the construction of unified integrated test information management platform.

4. The Physical Structure of the System

The physical structure of the system is mixed with three layers of B/S (Browser / server) and C/S (client / server). The data layer and service layer service are distributed on different servers, and they are physically isolated from each other, thus improving the security of disaster tolerance. The database management system runs on a dedicated server and provides powerful basic data management functions. Considering the "mass" characteristics of the test data, the C/S model is designed, using the reading server and the client data interface to read and write the mass data, and improve the speed and reliability of the data storage. All functions except data reading and writing are published by Web portal, and any computer connected to the network can use a browser landing system to enable the system to be flexible and extensibility. The physical structure of the system is shown in Figure 3.

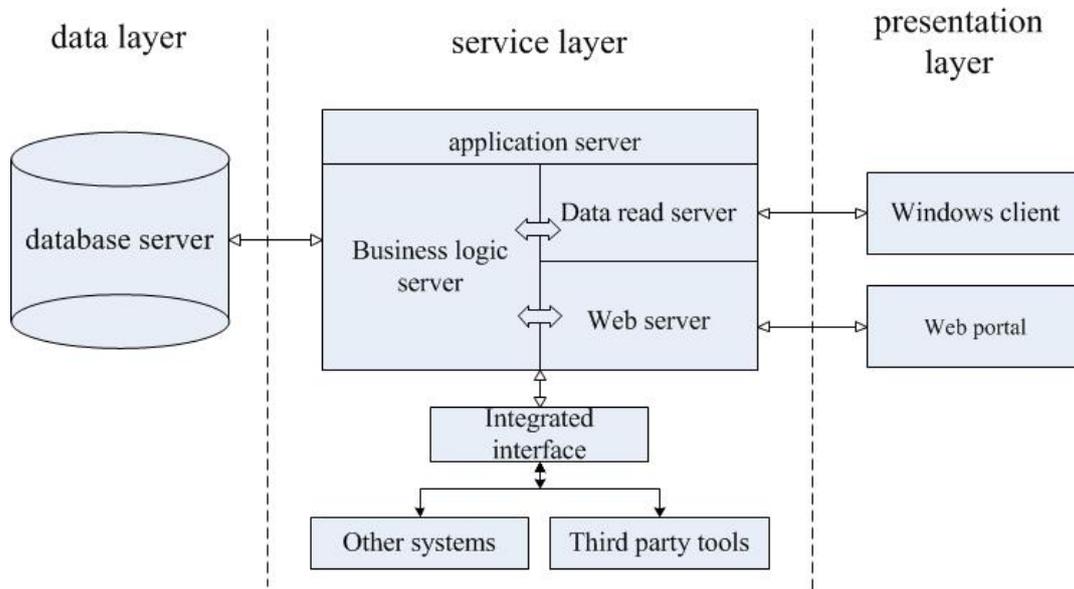


Fig.3. System physical structure diagram

5. Conclusion

This paper presents a design of an experimental telemetry data management system with embedded mass bus. This system can improve data management level and utilization efficiency, ensure the security of experimental telemetry data, and maximize the sharing and coordination of data between multi departments under the condition of safe and controllable data, so as to make information resources available. Sharing and comprehensive utilization is possible. At the same time, it is beneficial to improve and optimize the design of the product and improve the quality of

the design by fully excavating the value of the test task data and for the analysis, comparison, identification and evaluation of the aircraft test.

References

- [1] Liu Jiaying. *Spacecraft TT&C and Information Transmission Technology* [M]. Beijing: National Defense Industry Press, 2016, 24.
- [2] Zhao Hailong. Key Techniques of Next Generation Wireless Communication and Their Applications in Telemetry [J]. *Information and Electronic Engineering*, 2016, 10 (1):1-6.
- [3] Chris Potter, Kurt Kosbar. Hardware Discussion of a MIMO Wireless Communication System Using Orthogonal Space Time Block Codes [C]. ITC 2014.
- [4] Adam Panagos, Dynetics, Inc. Chris Potter, Kurt Kosbar. New Results in Unitary Space-Time Code Construction and Comparison to Upper Bounds [C]. ITC 2015.
- [5] Huang rh, Riloff e. Inducing Domain-specific Semantic Class Taggers from Nothing[C]. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, USA: Association for Computational Linguistics*, 2015: 275-285.
- [6] Li Yanhua, Lu Manhong, Wang Xiaobo. The Development of Aerospace Telemetry Technology [J]. *Journal of Spacecraft Technology*. 2017, 26 (5):18-23.
- [7] Erik Perrins, Michael Rice. Reduced-Complexity Approach to Iterative Detection of Coded SOQPSK [J]. *IEEE Transactions on Communications*, 2015, 55 (7):1354-1362.