A REAL-TIME TELEMETRY DATA PROCESSING SYSTEM WITH OPEN SYSTEM ARCHITECTURE

Zhang Jun, Feng MeiPing, Zhu Yanbo, He Bin & Zhang Qishan
Beijing University of Aeronautics & Astronautics

ABSTRACT

In face of the characteristics of multiple data streams, high bit rate, variable data formats, complicated frame structure and changeable application environment, the programmable PCM telemetry system needs a new data processing system with advanced telemetry system architecture.
This paper fully considers the characteristics of real-time telemetry data processing, analyzes the design of open system architecture for real-time telemetry data processing system(TDPS), presents an open system architecture scheme and design of real-time TDPS, gives the structure model of distributed network system, and develops the interface between network database and telemetry database, as well as telemetry processing software with man-machine interface.
Finally, a practical and multi-functional real-time TDPS with open system architecture has been built, which based on UNIX operating system, supporting TCP/IP protocol and using Oracle relational database management system. This scheme and design have already proved to be efficient for real-time processing, high speed, mass storage and multi-user operation.

KEYWORDS

Real-time data processing, Open system architecture, Man-machine interface

INTRODUCTION

Along with the increase of the flight tests and measurement parameters and the application of multiple data streams telemetry system, the data type and data capacity to be processed become more and more enormous. Especially, as to telemetry of aircraft and manned space vehicle, which need real-time monitor, output and analyze the telemetry data in a long time, ordinary telemetry system made up of telemetry devices and general computers are difficult to meet the demand. A new type of real-time TDPS based on advanced system architecture must be studied to meet the need

of multiple data streams, real-time data processing, mass storage and multi-screen display.
This paper fully analyzes the characteristics of telemetry data processing, researches the design of open system architecture for real-time TDPS, describes an application TDPS under open system environments. Moreover, an application software package is developed for this system, the interface between telemetry database and distributed network database is realized.

SYSTEM DESCRIPTION

As telemetry system evolved, it is necessary and efficient to process the telemetry data distributedly. When we design and develop a TDPS, it becomes more important to agree upon industry standards and to be compatible with new technologies. Then the hardware and software presented by different vendors, as long as meet the industry standards, can exchange each other. This is a typical open system environment. The most highlighted feature of the open system is interoperability and portability. The TDPS based on open system architecture adopts the platform abided to international standard and industrial standard (including hardware, operation system and database management system), supports open operating system, open communication protocol, open database, open user interface. The architecture diagram of the open system application environment is shown in figure 1.

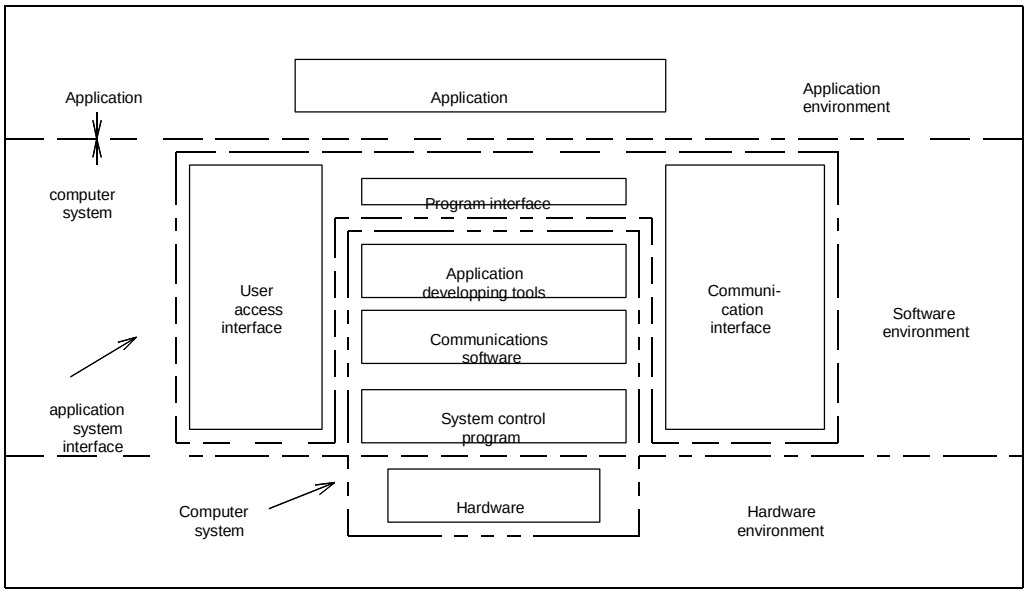

Figure 1. The application environments architecture of the open system

To achieve this kind of application environment, we should observe the following aspects:

1. Use computers with CPU of 386, 486, X86 in future or RISC chips.
2. Use UNIX operating system.
3. Use relational database management system with standard SQL language.

Telemetry front-ends play the part of acquisition and preprocessing telemetry data, it synchronizes and decommutates the telemetry data according to time sequence, and stores the data into a data table existing in the storage media such as tape or hard disk, the data can also interface with ethernet by a special component.
Through the quick-look function provided by TDPS, we can display the unchanged date formats and the parameters after E.U. conversion . Via building a data dictionary corresponding to the data variables, i.e., setting up the parameter database, we can define E.U. conversion algorithm. Through this method, we have realized the inter-connection between raw telemetry database and network database of TDPS. So, by developing the interface between network database and telemetry database, the telemetry processing system can access the data with data dictionary.
This kind of open TDPS supports multi-user operation and X Windows, likewise, a user can open multiple windows, run multiple programs, and display multiple processing results in one time on one screen.

SYSTEM DESIGN AND REALIZATION

1. System architecture

On the basis of ethernet, the TDPS is a LAN using bus architecture, as shown in figure 2. It is reliable, flexible and easy to expand. The system stores the data in centralization mode and processes the data in distributed mode. The telemetry data after format conversion is stored in storage media, it is managed by Oracle RDBMS, and can be accessed by SQL language.
Figure 2 shows that the system is server/client architecture, the server is Sun SPARC Station 10 (workstation from Sun Microsystems Inc.), the client is workstation from Sun, HP or IBM. The design of central database effectively guarantees that the telemetry data can be updated consistently in multi-user environments, while processing the data distributedly. So, the system takes advantage of local resource, and reduces the network data transportation expenditure.

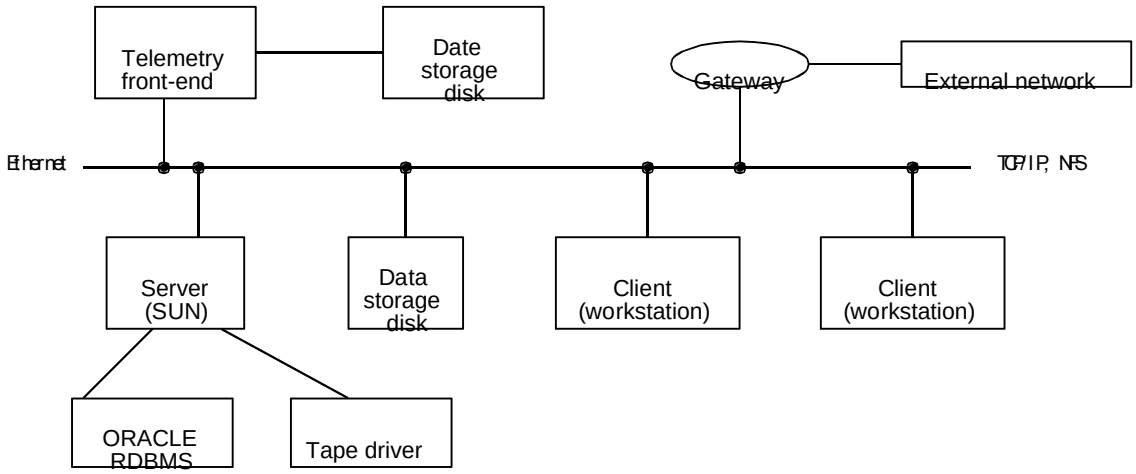

Figure 2. The hardware architecture of the distributed TDPS

In the procedure of processing distributed telemetry data, the server computer can fulfill the data processing routing such as data query, data retrieving, data sorting, and transmit the processed data to corresponding clients by the network. The client computer can complete special functions related with the telemetry tasks, such as engine parameter measurement, flight statue parameter measurement, etc. Thus, by assign the task to the server and client distributively, the burden of network data transportation can be reduced. In this LAN, the network communication protocol is TCP/IP, it is easy to transmit the telemetry data and file stored in media. Also, if we use NFS, the files in the system can be shared by any workstation, and do not consider the location of the data or file. The combination and share of information resource is realized.

2. Network database management

It is inevitable to link with database when carrying on data management and data exchanging services in TDPS. Database management systems around industry standards have been provided by major vendors including Oracle, Ingress, Sybase, etc. Conveniently, Let us analyze how to design, process, and display the telemetry data using Oracle database management system.

In open TDPS, database management system servers any workstation in the network. The client/server architecture system is based on database server. The server computer can control columns and rows of the table in the database, and provide better concurrency control, so as to guarantee multiple users access to the same data simultaneously. This scheme suits the application environments which operate telemetry data frequently in TDPS.

To solve the data link problem between telemetry database and network database, remote-copy, remote-visit or specific routings that have the same function can do it. We can use API (application programming interface) provided by database

management system to developing the routings, for examples, we can write program use C language with SQL inserted into it (for instance, PRO*C in Oracle), or in Oracle, we can employ OCI (Oracle call interface) to achieve it. We use Oracle call subroutine to access the database in OCI. The OCI programs are completely written by C, Oracle calls are C functions provided by Oracle. Users are able to access to the data in server, where the data is more integrated, secret and can be recovered easily after trouble, meantime, users can develop application modules in clients with API. But it is a non-standard SQL call, so in the open system, this kind of program is less compatible with others.

Because SQL is a non-procedure language, this makes it independent to execute each sentence. It is difficult to exchange information among sentences. Thus, SQL is limited in procedure program. When we use PRO*C, we can insert SQL sentences into C, in this way, the programs have the ability to access database, and also have powerful function to process data. When using another type of database, we can replace SQL sentences with corresponding commands, that is, it is easy to transplant.

3. User software

The TDPS based on open system architecture adopts UNIX operating system, supports multi-user and multi-task operation, and executes lots of software products including database management, network communication, graphics processing, and file management, etc.

When developing the user application software, we can use C language to write special algorithm and routing, use X Window (Motif or OpenLook graphics user interface) to develop man-machine interface.

The application software includes telemetry front-ends set-up, parameter databases definition, format conversion and system parameters set-up, telemetry data display, playback, and quick-look query processing, etc., is easy to use because of graphics user interface and menu management.

CONCLUSION

From above, the hardware and software environments of a ream-time TDPS based on open system architecture are built. it includes user software, system software, network database and hardware architecture, supports industry standards widely. Regardless of hardware upgrading, function expanding, software transplanting and system interoperating, we can use the open system products provided by different vendors. In table 1, we compare the open telemetry information processing system with ordinary one.

Table 1. The performance comparison between open system and ordinary system

	open system	ordinary system
adaptability	wide	normal
real-time ability	good	good
interoperability	good	bad
scalability	good	bad
function expansion	easy	difficult
software portability	easy	difficult
user application	easy	normal
industry standard	support	not support
ratio of performance to price	high	low

The TDPS described in this paper is a typical open system. By experiment, it shows that this kind of scheme and design is effective and adaptive. Because of its open architecture , we can expand its scale and performance to use it in a more wide field.

REFERENCE

(1). Mark D. Macmillan & Douglas Ice, "Open Architecture Telemetry Processing System", ITC/USA/1993, Vol. XXVIV

(2). Sloman, M & Kramer. J, "Distributed System and Computer Networks", Prentice-Hall, 1987

(3). Daniel Dawson, "An Open Software Architecture for UNIX Based Data Acquisition/Telemetry Systems", ITC/USA/1992, Vol. XXVIII