UDC 681 . 3 : 169 . 162. 26

The Application of Linux and Database System for Blast Furnace

Shinichiro SHIMOI*[1] Hiroshi MIYAZAKI*[1]
Nobuo SUMIDA*[1]

Abstract

We have developed the process computer system with Linux operating system and relational database, which leads to the drastic decrease of the costs for the hardware and the software compared to the conventional systems. The system has been applied to the blast furnace process and working effectively without any troubles.

1. Introduction

A blast furnace, which produces molten pig iron, constitutes one of the most important processes of a steel works. For this reason, a Level-II computer system composed of highly reliable hardware and OS tailor-made for the purpose has been used for the process computer supporting the operation of a blast furnace.

Over the last 15 years, which corresponds nearly to one campaign life of a blast furnace, remarkable advances have been made in the field of computer technologies, especially in open technologies. As a consequence, personal computers (PCs), which were initially applicable only to personal uses, have come to be used in the business and factory automation. In the meantime, the reliability and performance of PCs have been improved such that they are now perfectly applicable to the control of iron- and steel-making processes where around-the-clock operation is a natural prerequisite.

In this background, a process computer system using Linux *[1)], one of the most advanced open technologies, and a database system has been developed and applied to No. 3 Blast Furnace (BF) of Nippon Steel's Kimitsu Works as the first case of such a system used for the process control of a blast furnace. The outline of the developed process computer system is reported hereafter.

2. BF Process and Roles of Process Computer

2.1 Outline of BF process

A blast furnace is a shaft furnace having a slightly larger diameter at the middle as shown in **Fig. 1**. Iron ore (sintered ore), which is the raw material, and coke, which is the heat source, are charged into the furnace one after the other from its top, and air heated to approximately 1,200℃ in a hot stove is blown into the furnace from its lower part to have the coke burn and melt and reduce the iron ore into molten iron. The molten iron drips down to the hearth and is then tapped through a tap hole. The above processes are sustained without interruption during the period of a furnace campaign, which lasts for 20 years or more.

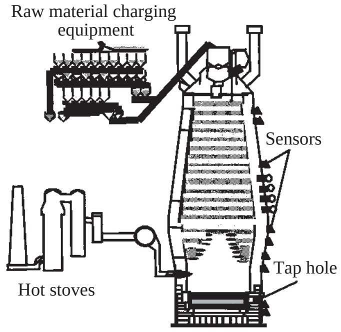

Fig. 1 Outline of blast furnace

2.2 Roles of process computer for blast furnace

Since a blast furnace is a very simple reaction vessel, it is difficult to control the conditions inside it directly and delicately. In addition, because the processes change very slowly, once stable operation of the furnace is perturbed, it is very difficult to recover a normal state. It is therefore necessary for maintaining stable operation to monitor very small changes in the condition inside the furnace and take appropriateoperation. For this end, a process computer collects and calculates data from sensors provided at various positions of the furnace, and thus monitors the condition of the furnace in real time. When it detects any change that may adversely affect the stable operation of the furnace, it outputs action guidance

*[1] Environment & Process Technology Center

for furnace operators.

More specifically, the monitoring and controlling functions of a process computer of a blast furnace are as follows.

1) Charging control of ore and coke: calculation of the charging ratio of iron ore and coke and the sequential order of their charging.
2) Charging operation control: setting of the operation mode of the rotating charging chute such that an adequate circumferential distribution of ore and coke is realized.
3) Hot stove Control: control of the combustion in hot stoves such that hot blast is supplied stably.
4) Furnace control: estimation of the condition inside the furnace based on information from various sensors.
5) Control of the tapping of molten iron and slag: control of the amount and quality of tapped molten iron.

As listed above, a process computer has very important roles in the operation of a blast furnace.

The following section outlines the developed process computer system applying open system technologies.

3. Outline of Process Computer System for Kimitsu No. 3 BF

3.1 Basic philosophy of system construction

(1) Cost reduction and high reliability

In order to realize a drastic reduction in the costs of the computer system, it was decided as the basic philosophy of the development to construct a system combining hardware of PC with economical de-fact standard basic software.

From past experiences, the reliability of open system hardware was judged to be sufficiently high. With regard to basic software, on the other hand, the MS-Windows system *2) had been used for the control of processing lines and the like. However, it was considered difficult with a Windows OS to maintain high reliability of a system for a blast furnace, which would be used for a long period of 15 to 20 years, because the source programs of a Windows OS are not available. For this reason, a Linux OS, which is open to the public and composed of highly transparent source programs, was selected [1,2]. Our evaluations of Linux and Windows-NT *2) are compared in **Table 1**.

(2) Improvement in software development efficiency

A process computer system of a blast furnace handles a great amount of process data, and for this reason, the ease in attaining data and the efficiency in the engineering of programs to process and edit data significantly influence the efficiency of daily system operation and maintenance. In view of this fact, it was decided that a database should be applied to file systems and a tool that could automatically generate programs to process and edit data be developed.

Table 2 Outline of evaluation tests

	Evaluation item	Content of test
1	Basic functions	·Basic structure: Linux + Oracle*4) ·Roughly 100 data items
2	Operation stability	·AP having a capacity about 1/3 that of a commercial system ·Long-run test for 40 h
3	Overall evaluation	·Parallel-run test using actual operation data (roughly 6 consecutive days)

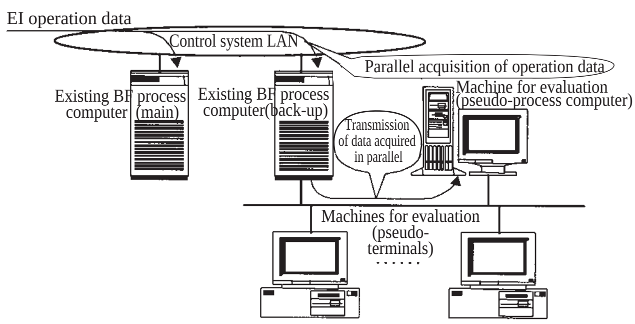

Fig. 2 Configuration for system evaluation

A prototype system was constructed in accordance with the above basic policy, subjected to evaluation tests for approximately 1 year and its applicability to a process computer system for a blast furnace was confirmed. Thus, it was decided that a commercial on-line system be developed using the open technologies. **Table 2** outlines the evaluation tests and **Fig. 2** shows the configuration of the prototype system.

3.2 Outline of on-line system

Fig. 3 shows the configuration of the process computer system for Kimitsu No. 3 BF. The characteristics of the system are as follows:

1) A PC server (Pentium-III *5) of 500 MHz, main memory of 1 GB) in which Linux is installed is selected as the control server.
2) An operation database server to store the process data of one furnace campaign (roughly 20 years) is provided.

3.3 Software structure

Fig. 4 shows the software structure of the developed system. In view of the use of the open technologies in order to help the engineering of application software, which comprises as many as 600 k steps, a middleware system (NS SEMI System *6)) was developed

Table 1 Comparison of Linux and Windows-NT

	Linux	Windows-NT
Origin	Unix *3) on PCs	Windows OS for servers
Procurement	Plural distributors; free of charge	Sold exclusively by Microsoft
Openness of OS source program	Entirely open to public Users can investigate and modify freely	Closed except for some vendors Modifications by users are not allowed
Function	Sufficient server function High reliability	Sufficient server function Superior to Unix in GUI
Performance	Compact owing to small GUI Highly responsive	Sufficient for some applications

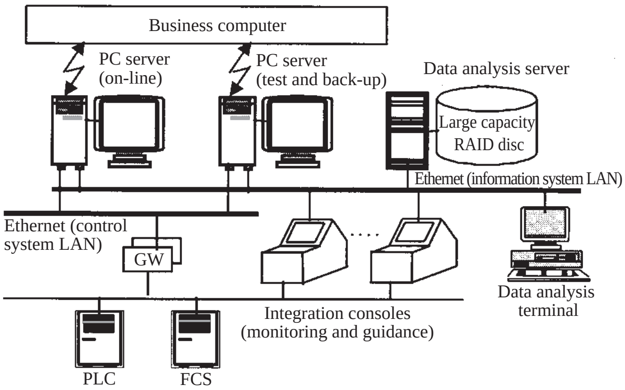

Fig. 3 Configuration of process computer system for Kimitsu No. 3 BF

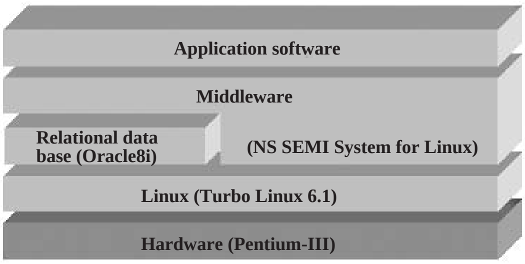

Fig. 4 Software structure

in-house. The engineering of application software can be done on the basis of this middleware, and as a consequence, the reliability of the whole system has been enhanced, the development of application software free from platform-consciousness has been made possible [3], and thus, high-quality and portability of developed large-scale software have been secured.

4. Performance Enhancement Measures

A problem was identified at the stage of an integrated test of application software in that procedures that had to be repeated periodically in every 1 min (required response 10 s) took 30 s or more. An analysis of the procedures made it clear that the access time to the database formed a bottleneck in most cases, and the following measures were taken to solve the problem:

(1) Improvement of SQL cache-hit ratio

It was found out through an analysis of the database access procedures that re-analysis of SQL was occurring too frequently and the SQL cache-hit ratio was as low as 69% or so. The following steps were taken to improve it:

1) Expansion of Oracle common memory

12 MB was initially assigned to the Oracle common memory, but SQL statements issued were not cached sufficiently. At this, the common memory was increased gradually to find an optimum size, and it was finally optimized to 128 MB.

2) Binding of SQL parameter

At the issuance of an SQL, the value of the process data item in question was used initially as the parameter of the SQL. However, since the value of a process data item varies from time to time, two or more SQL statements for one data item having different parameter values were recognized at SQL analysis as having different compositions. This resulted in the low cache-hit ratio. As a countermeasure, a fixed parameter was selected for SQL statements for one data item and the data item was link to the fixed parameter.

This resulted in an improved cache-hit ratio of 94% through the above two measures.

(2) Reduction of overhead of database login/logout

The logging in and logging out of a database have to be done program by program. This results in delays caused by overhead times if there is a large number of programs. In view of this, to reduce the login/logout overhead, program modules were integrated into a smaller number insofar as such integration did not adversely affect the ease of operation and the functions of individual tasks.

As a result of the above (1) and (2) and other measures, the processing time of the procedures that had to be repeated in every 1 min was reduced to approximately 7 s. **Fig. 5** shows the history of the improvement brought about by each of the countermeasures.

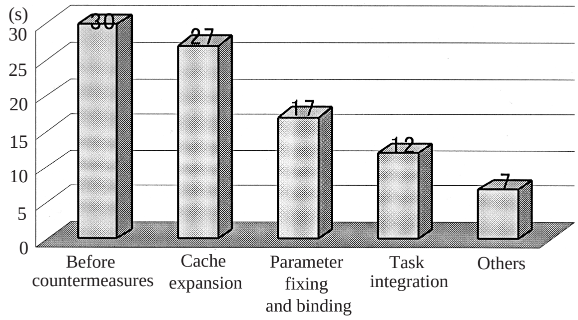

Fig. 5 History of countermeasures and improvements

5. Typical System Troubles and Countermeasures

All the main system troubles encountered during the development of the new system were related to the database, and the Linux OS operated very stably, free from any trouble. **Table 3** shows typical system troubles during the development stage.

6. Closing

The developed commercial system began on-line operation in May, 2001 and has since been working satisfactorily. A remarkable cost reduction, to about one half that of a conventional process computer system, was realized through the development and thus a foundation was established for expanding the application of open tech-

Table 3 Typical system troubles

Problem	Cause	Countermeasure
Too large portions of memory resources are used	Copying of common library to individual APs	· Oracle upgrade · Linux upgrade
System hanging at registration of stored procedures	Bug in Oracle	Oracle upgrade
Data acquisition error of Oracle common memory	Fragmentation in common memory	Insertion of periodical flushing
System hanging at data insertion between servers	Memory leakage at use of DB-Link	A countermeasure patch was acquired from Oracle.

nologies to the process computer systems for other iron- and steel-making processes.

*1) Linux is a trademark of Linus Torvalds in the United States and other countries.
*2) Windows and Windows-NT are trademarks in the United States and other countries of Microsoft Corporation of the United States.
*3) Unix is a trademark in the United States and other countries licensed exclusively from X/Open Company Limited.
*4) Oracle is a trademark of ORACLE Corporation.
*5) Pentium is a trademark of Intel Corporation.
*6) NS SEMI System is a trademark in Japan of Nippon Steel Corporation.

References

1) Shimoi: The Application of Linux and Database System for Blast Furnace, Inst. Elec. Engrs. of Japan - Metals Industry Committee, MID-01-33, Dec. 2001
2) Shimoi, Miyazaki, Sumida: The Application of Linux and Database System for Blast Furnace, Proc. 143rd Spring Conference of ISIJ, Mar. 2002
3) Kawahara, et al.: Application of Rightsizing and Open System to the Process Automation in Iron-and Steel-making., Shinnittetsu Giho. (363), 37 (1997)