Evaluation Approach for Stable System Operation

● Yuichiro Kuroki ● Yoshifumi Jonishi

(Manuscript received July 10, 2007)

The extensive and widespread trend toward open systems has resulted in increasingly complex and diversified IT systems. IT vendors test their products using various methods to confirm compliance with their quality standards prior to shipment. At the same time, customers expect the total quality of IT products to contribute to the stability of their systems. As a total IT system vendor, Fujitsu has engineering teams with top-class evaluation skills that thoroughly test our products to ensure compliance with the criteria of mission-critical systems. These teams also test products to ensure sufficient connectivity for use in multi-vendor system environments. Moreover, they incorporate customer applications into their test beds and work on preventing potential problems from occurring. As a result, the products we ship offer proven high quality for our customers. This paper describes the activities and evaluation tools of Fujitsu's engineering teams.

1. Introduction

Fujitsu has warranted quality by the product development department for each of its products. At the same time, customers request a mechanism for warranting the quality of an entire system against the background of increasingly complex information systems. This prompted us to conduct full-fledged, comprehensive system evaluations since 1993 for quality assurance by combining products from the customer's viewpoint. Starting in 2001, we also began verifying connectivity to the products of other companies in view of an expanding multi-vendor environment on the customers' side due to the trends of open systems. And in 2003, we established a system to verify the applications and operations of systems of individual customers in order to stabilize those systems at an early stage and improve customer satisfaction.

This paper describes our efforts to achieve quality assurance for these systems.

2. Comprehensive evaluation of our products

In evaluating a system, we comprehensively assess the entire system, assuming customer operations performed by combining products that have been evaluated by development departments based on product specifications (**Figure 1**).

This method is used primarily to evaluate configurations and connections focusing on consistencies between hardware and software, system recovery in case of trouble, stability in high-load environments, and the entire system.

2.1 Evaluation of configurations and connections

Customer systems are configured with many different hardware and software products. Therefore, we evaluate the products cyclopaedically by using a method based on the Design of Experiments. This method allows us to further enhance sufficiency by combining not only products but also connection conditions including those of intermediate equipment and forwarding modes. We also enhance the evaluation environment by developing evaluation system modeling for financial systems, contribution systems, and manufacturing systems operated by customers for evaluating those systems according to specific operations performed by the customers.

We also provide customers with reliable optimum systems patterned based on the TRIOLE concept in which servers, storage, networks, and middleware are integrated and incorporated as an evaluation environment.

2.2 System recovery evaluation

In the field of mission-critical social systems, such systems need to be operated continuously even in case of any abnormal event. Consequently, we evaluate the continuity of a customer's business in case of component trouble in an environment having hardware redundancy and a cluster configuration.

We have been evaluating recovery in case of individual hardware parts encountering trouble by generating quasi-failure in the memories and control circuits of hardware. This method of generating quasi-hardware failure is based on the circuit information of hardware. We have established a new method of generating failure where only logic signals are searched without using hardware circuit information for parts whose information needed to generate quasi-failure cannot be obtained due to changes in materials used for constructing individual products. This method allows us to verify the recovery of an entire hardware unit.

We verify the recovery of software failure, on the other hand, by using such operating systems as Solaris, Linux, and Windows for existing open systems. This makes it difficult to generate software trouble. To deal with this situation, we evaluate generated failure (i.e., false failure) in operating software by employing a method of generating such failure as a hangup, panic, and timeout that occurs in software from outside, and which is not dependent on the type of software. We have also established a new method of generating failure in drivers to evaluate the recovery of drivers that control connections to peripheral devices.

As described above, in our system recovery evaluations we verify hardware as well as

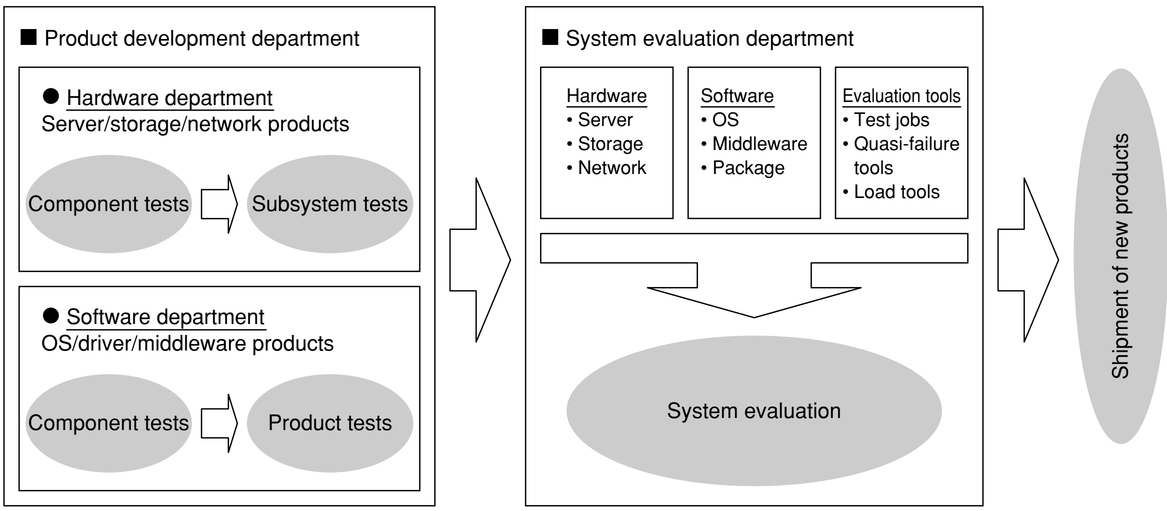

Figure 1
Positioning of system evaluation.

software, including operating systems, drivers, and middleware (**Figure 2**).

2.3 Load and stability evaluation

The workloads of customer systems are likely to change, particularly on weekends and at the end of the month. The system loads are also increasing due to the expansion of customer businesses. We are developing tools to simulate thousands of client processes and test tools targeting individual components in order to generate these load conditions. We verify stable operation under changing loads and, at the same time, evaluate maximum system throughput in accordance with scenarios assuming daily and monthly processes.

3. Connectivity to other vendor products

In today's era of open platforms, customer systems are being configured with the products of multiple vendors. In this situation, we need to verify beforehand that such systems operate stably, even if connected to the products of other companies so that customers can safely use Fujitsu products. When products constituting systems are stratified (**Figure 3**), the hardware includes servers, storage, connection configurations (i.e., switches), and interface cards, while software includes operating systems, multipaths, storage management, clusters, middleware, and databases for a total of ten components. The combinations of connecting these components can be obtained by multiplying the number of products belonging to each of these ten component categories. In an environment marked by an expanding market for the storage area network (SAN), individual companies are releasing new products every year. Consequently, the number of hardware and software components constituting systems increases annually. There are as many as several million possible combinations of logically connectable devices. For these reasons, we research global industry trends, clarify all targeted products, and then address challenges based on information about customers obtained from service depots.

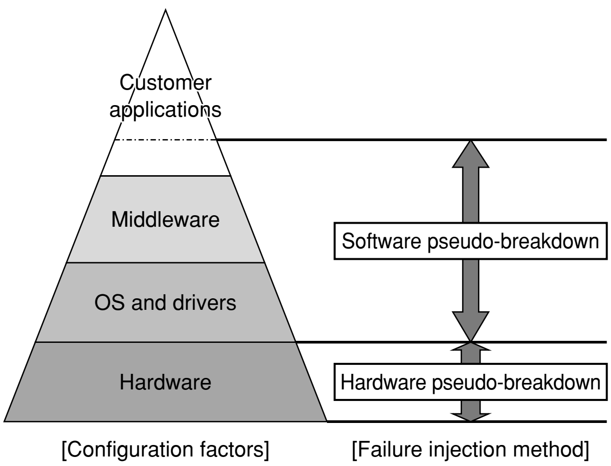

Figure 2
Scope and method of system recovery evaluation.

3.1 Connection recovery verification

We verify the faults in connections to the products of other companies by using self-developed tools for verifying false faults, assuming interface faults between the interface cards mounted in servers and the storage that are connected via switches. We also verify multipath and cluster redundancy, and to ensure quality, confirm that no failure occurs in operating systems, storage management, middleware, and databases in case of a connection fault.

3.2 Promoting verification efficiency

We consider it very important to promptly provide information about connection configurations as requested by customers. We intend to standardize verification items and shorten the time needed to verify each combination by introducing tools that allow unattended, around-the-clock operations. We also review the grouping of products and verification levels based on past verification results in order to cover a larger number of combinations and introduce verification methods applying the Design of

Experiments for promoting more streamlining.

4. Verification of customer systems

As described above, evaluating Fujitsu products in a comprehensive manner and verifying connectivity to the products of other companies ensures the quality of individual hardware and software products. Customer systems consist of these products as well as special customer applications. There are also many kinds of system operations involved. Therefore, it is important to verify these applications and operations in advance so that customer systems operate in a stable manner.

To that end, we provide verification settings for the system development department when they introduce new systems, and install or modify more systems in order to support verification. In this way, we can adequately verify systems in advance to prevent problems from occurring in customer systems.

Configurations of the verification environment should basically be the same as those of the customer environments. We construct hardware and install operating software (such as the OS) and middleware in advance for providing environments to reduce verification procedures. We have also developed security rooms for protecting the data of customers and teleoperating environments for improving work efficiency.

We support verification through false-fault supports for confirming the degree of effects of hardware failure on systems and recovery after failure occurs. We also support the verification of loads applied to online systems with pseudo-terminal tools by considering the environments of actual devices. Moreover, we have reduced the time needed to address trouble in cooperation with the product development department.

5. Evaluation tools

We have described comprehensive evalu-

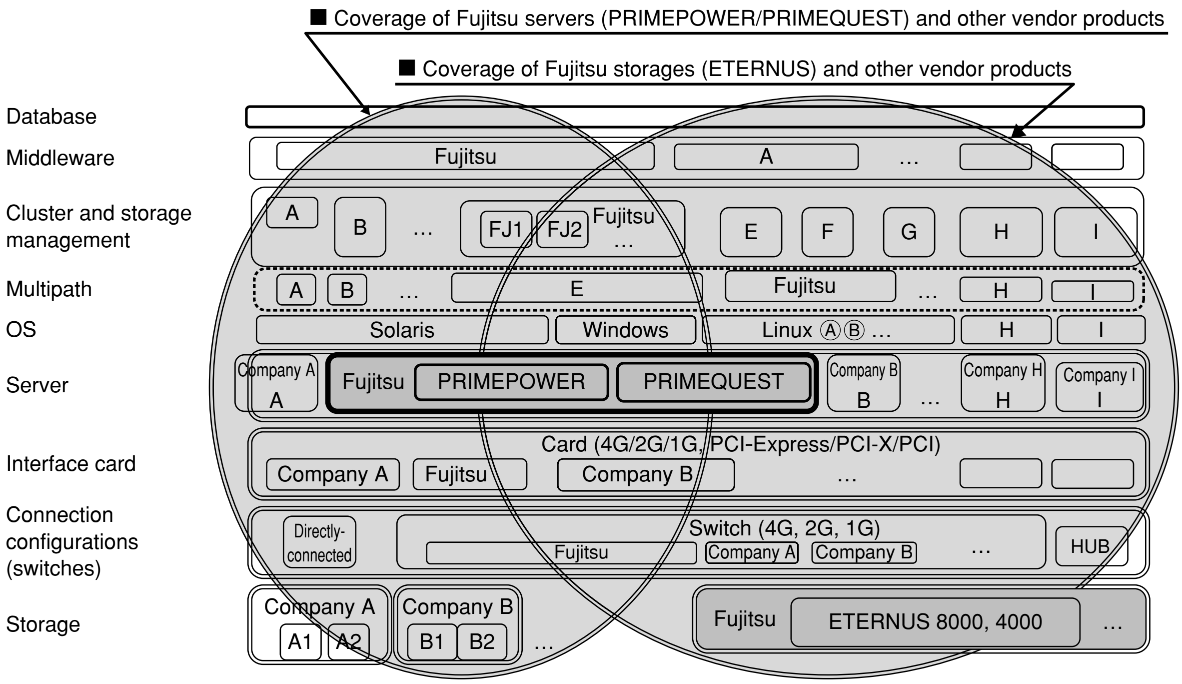

Figure 3
Stratified system configuration including other vendor products.

ations, the evaluation of connectivity to ensure the stable operation of customer systems, and the verification of customer systems. Customer systems are becoming increasingly easier to interconnect with mainframes, UNIX servers, Windows servers, Linux servers, storage systems, and PCs thanks to the spread of the broadband Internet. Consequently, customers can now select equipment by themselves.[1] The trend toward open platforms has entailed more trouble in devices provided by multi-vendors and which are used in multi-platforms. Therefore, we test connections among various devices by operating software to prevent such trouble. Since the test patterns increase by matrix, we have self-developed highly exhaustive evaluation tools to solve these problems.

5.1 Automatization and work saving

Automatization and work-saving tools are typified by PCs or dedicated jigs used for operations previously performed by humans based on scenarios that describe the work originally done by humans. Some devices serve as the terminal emulators of PCs via networks to perform such operations as changing settings and issuing instructions to start an evaluation. Since a certain pattern for these procedures may be developed, we create scenarios of equivalent procedures and automatize operations by using automatized tools operated from consoles to replace the hands and eyes of humans, and enhance the reproducibility of failure.

The hard disk drives (HDDs) mounted in storage systems can also be replaced during system operation (through hot-swapping). The various timings at which to install or replace parts (which vary among individuals) must be verified cyclopaedically to confirm that a hot-swap is enabled in any cases. We have developed a pseudo hot-swap tool that can electrically connect and disconnect the connectors of HDD interfaces at the specified timing. With this tool, we can set up any environment for reproducing all cases by changing timings in consideration of actual work to be done.

5.2 Loads and performance

In line with Moore's Law, the CPU throughput, memory capacity, and hard disk capacity of information equipment continue to increase yearly. And in line with Gilder's Law, network speed is sharply increasing at an accelerated pace. Thus, systems that process data and require such high throughput should operate continuously and stably at a maximum level of performance, even if loads are applied to any part of such systems. This is why we need to verify performance by preparing the same conditions beforehand. For that purpose we have developed a network load application device to verify performance. The device can simulate loads corresponding to transmission bandwidths at multiple ports and terminal connections equivalent to 100 000 units. This device is used in evaluating traffic involving http, https, pop, and ftp.

5.3 Recovery

Customer systems in a mission-critical field should operate 24 hours a day, 365 days a year without any interruptions. Hardware failures are also considered unexceptional. We ask customers to introduce storage systems having cluster systems or redundant functions for this purpose. In verifying these systems in advance, we need to verify that the systems do not stop operating even if failure occurs in hardware constituting the systems. It is actually difficult, however, to prepare hardware breakdown. Therefore, we have developed a pseudo-breakdown test robot that generates false hardware failures artificially in a positive way (**Figure 4**).

This device deactivates signals at individual points for all patterned wirings on the printed circuit boards of equipment. The elements connected to the patterned wiring stop functioning immediately upon signal deactivation,

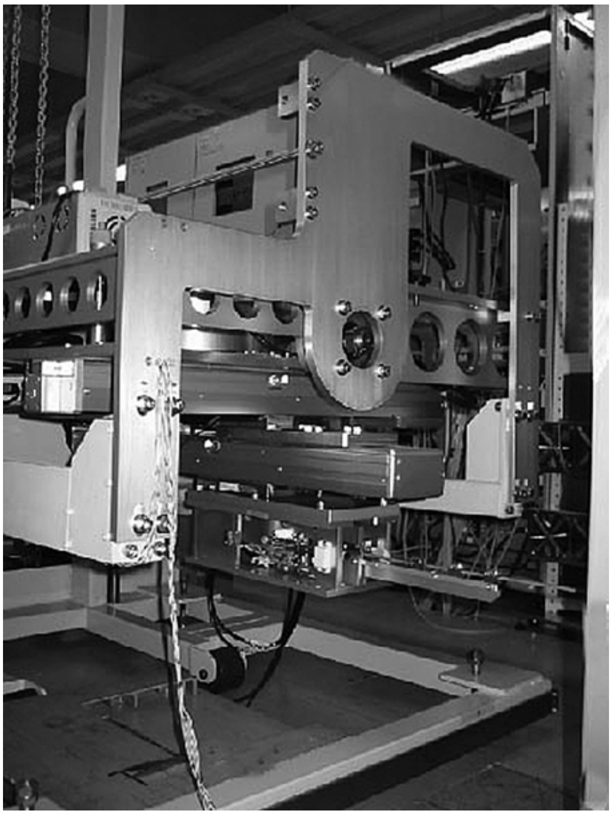

Figure 4
Pseudo-breakdown test robot.

resulting in the same condition equivalent to a hardware fault. We also use another pseudo-breakdown test robot for memory systems and expansion cards for which the timings when high-precision false faults occur are requested. The equipment monitors the conditions of individual interfaces and deactivates signals at timings corresponding to actual conditions.

6. Conclusion

This paper described our activities for conducting comprehensive evaluations and verifications of customer systems in order to ensure proper operation.

The needs of customers will become increasingly more sophisticated in the future. Information systems will also become increasingly complex along with the development of new products. In response to these situations, there will be a growing demand for higher reliability. For these reasons, we must make our evaluations more sophisticated as a total system in the future from the viewpoint of customers.

Reference

1) T. Abe: UNIX server software: PRIMECLUSTER. (in Japanese). *FUJITSU*, **53**, 6, p.456-462 (2002).

Yuichiro Kuroki, *Fujitsu Ltd.*
Mr. Kuroki received the B.S. degree in Industrial Administration from Tokyo University of Science, Noda, Japan in 1979. He joined Fujitsu Ltd., Kawasaki, Japan in 1979. Since 2006, he has been engaged in system evaluation.

Yoshifumi Jonishi, *Fujitsu Ltd.*
Mr. Jonishi graduated from Asahikawa National College of Technology, Asahikawa, Japan in 1974. He joined Fujitsu Ltd., Kawasaki, Japan in 1974, where he has been engaged in quality assurance of global server systems. Since 1996, he has also been engaged in system evaluation.