

## Improvements to NICE-ELWISE Test Efficiency

*Noriko Fukuda, Kenji Murai<sup>†</sup>, Hideki Kawabe, and Katsuaki Miyabo*

### Abstract

We have developed an automatic testing tool that can increase the efficiency of testing work for NICE-ELWISE, which is an information distribution platform based on IC (integrated circuit) cards composed of two completely different types of software: a web-server-based operations and management system called NICE and embedded software for ELWISE cards. We investigated the features of both types of software and determined which tests are suitable for automation. This article describes our investigation and presents evaluation results.

### 1. Effectiveness of automating testing

Automated software testing can generally reduce the amount of testing work, and it can also help maintain a fixed level of software quality. However, in some cases, the cost of creating and maintaining the automatic test code can be high, so the desired effect is not achieved, even if all test items are automated. Consequently, it is important to take a strategic approach to test automation [1], [2].

### 2. NICE-ELWISE architecture

NTT Laboratories has developed an information distribution platform based on IC (integrated circuit) cards called NICE (network-based IC card environment) [3] and an extremely multipurpose IC card conforming to ISO 14443 Type B standards, called ELWISE [4], and has been promoting business based on these products [5]. The architecture of NICE-ELWISE is shown in **Fig. 1**. The NICE-ELWISE software can be broadly divided into the NICE web-server-based operations and management system and the IC card embedded software, including the JIC-

SAP2 infrastructure, the card manager, and various libraries for functions such as encryption (JICSAP: Japan IC Card System Application Council).

### 3. NICE-ELWISE testing

#### 3.1 Testing overview

At NTT Laboratories, whenever a product is passed from the developers to the shipping stage for version upgrades or bug fixes, or when there is a change to its external environment such as a security patch to the operating system or middleware used by the product, the product's individual functions as well as its overall quality are tested before it is delivered to businesses to ensure that no problems will result. Test items include operational tests done by the shipping personnel and items that provide an overall assessment, including items from test documents created by the developers. The latter include tests of functions that are used infrequently, such as when the batteries expire or there is a memory fault in the card.

#### 3.2 Test features

The features of the NICE-ELWISE tests are listed in **Table 1**. Integration testing for web software is in the form of user operations from the application screen and entails frequent patches of the operating system and middleware as well as many structural

<sup>†</sup> NTT Service Integration Laboratories  
Musashino-shi, 180-8585 Japan

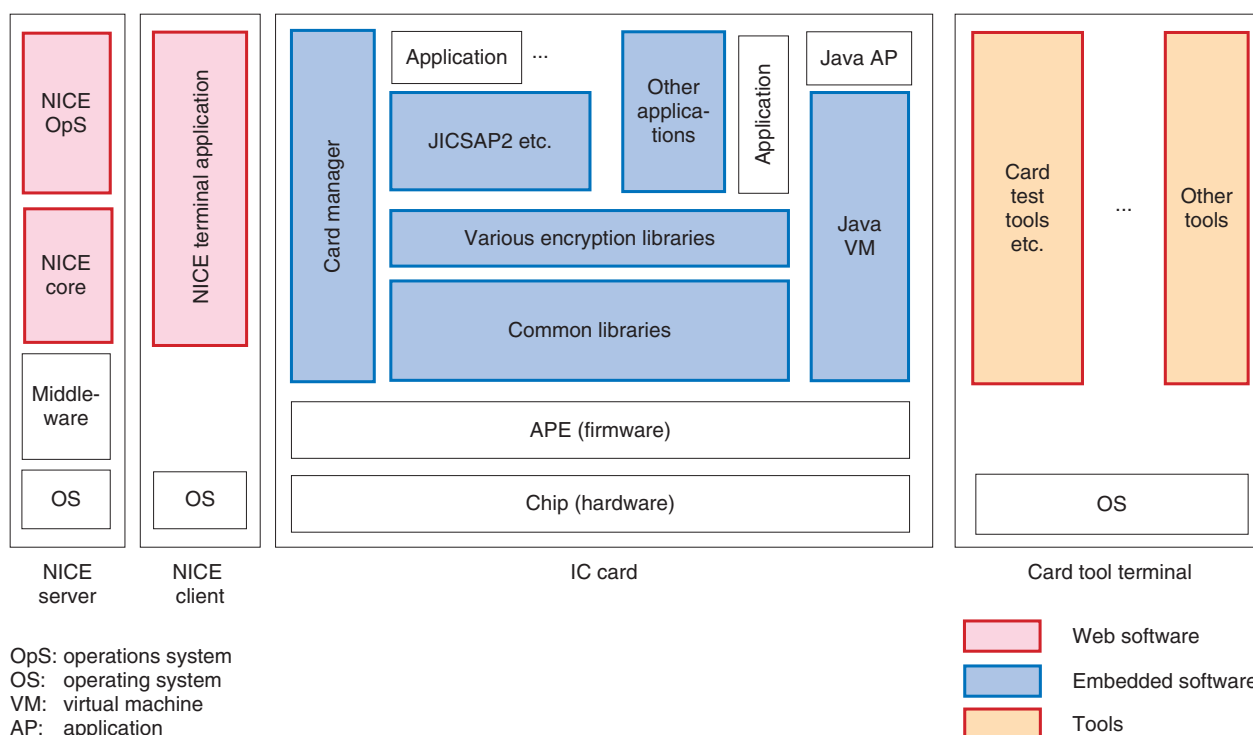


Fig. 1. NICE-ELWISE architecture diagram.

Table 1. NICE-ELWISE test features.

Type	Web software	Embedded software
Product	NICE etc.	JICSAP2 infrastructure Card manager Various libraries etc.
Integration test format	User operation tests from the GUI	Input/output interface tests
Configuration variations	Multiple	Specific
External environment	(OS, middleware) Patches are frequent. Product life averages approx. three years.	(chip, APE) Changes are infrequent. Life averages ten years or more (reader/writer). Compatibility testing is required.

variations. For embedded software, input/output interface tests are used and updates to IC chips or card firmware (APE: application execution environment) are infrequent, but tests for compatibility with reader/writer equipment are necessary.

### 3.3 Testing issues

Since there are many products and versions, the cost of compatibility testing is very high and requires knowledge of the implementation and overall know-how about each product. The fact that it is difficult for

someone unfamiliar with a product to perform the tests has been an issue in efforts to complete the testing work. For these reasons, we studied ways in which overall product quality testing can be done in a short amount of time to a consistent level by persons that have not had the opportunity to accumulate detailed implementation knowledge.

## 4. Testing suited to automation

In general, tests that are suited to automation are

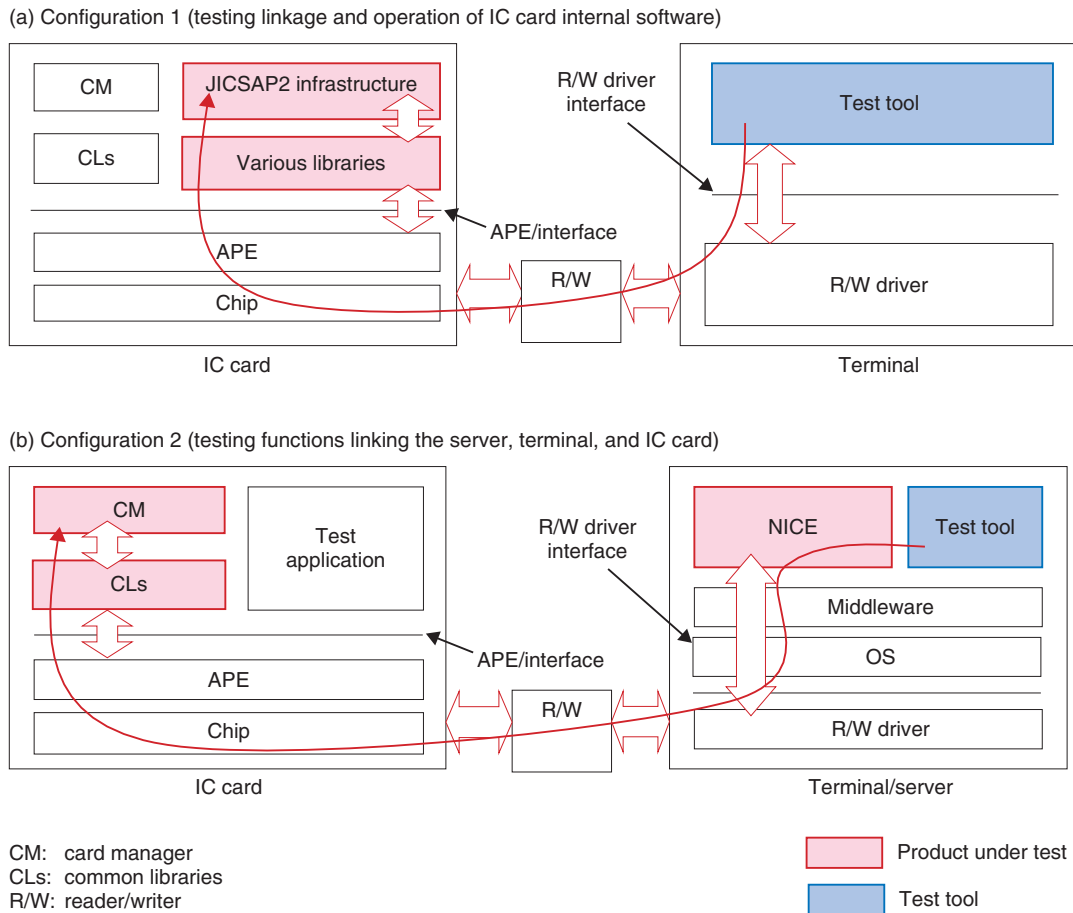


Fig. 2. Test organization.

ones for which little work to update the test automation code is required. Examples of suitable tests include tests of functions for which changes to the application programming interface are infrequent, tests where the code is used many times, such as load testing, or simple test scenarios where the code can be created easily. In contrast, tests that are not well suited to automation include tests for functions that are changed frequently, such as the graphical user interface (GUI), regression testing that requires changes or additions to the test code, testing of configuration variations such as environment settings, and tests where the results from various input values must be checked.

## 5. Application of automation

Because research and development of the basic functions of NICE-ELWISE was complete, no further sig-

nificant expansion in functionality was expected, business deployment was proceeding, and usage patterns had become well-known, we decided that it had become easier to automate many tests that had previously been done manually by shipping personnel. For example, GUI testing is generally not considered suitable for automation because of the frequent changes, but no further significant changes were expected to the GUI for NICE. Furthermore, while configuration variation tests in multiple environments are not usually suitable for automation, the conditions for embedded systems are fixed; in particular, those for NICE are fixed. Finally, periodic checks related to security patches are also required, so test code for this purpose can be used many times over. For these reasons, we decided that the software was suitable for test automation.

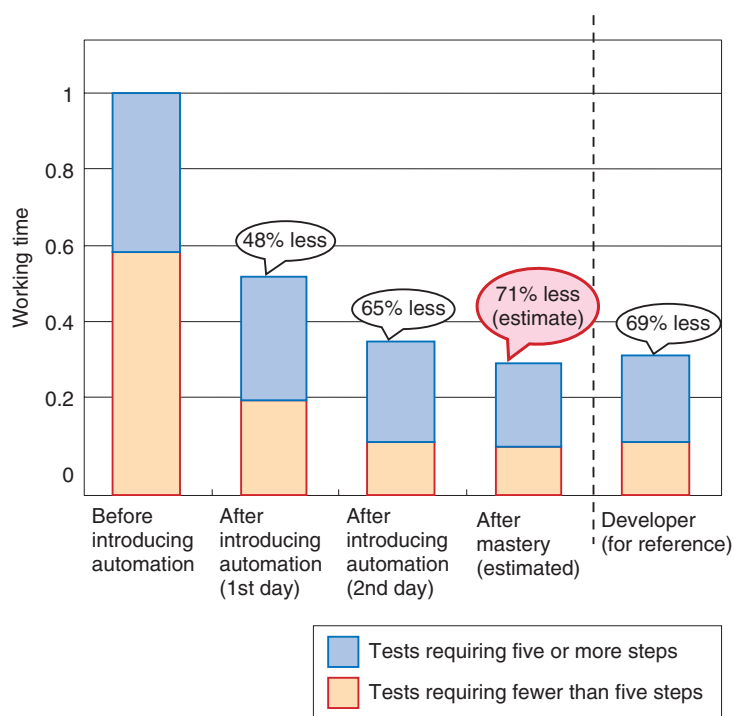


Fig. 3. Time required for inexperienced personnel to perform NICE test operations.

## 6. Clarification of test viewpoints

To ensure uniformity in the level of test work, we first established clear test viewpoints to be used by the shipping personnel.

- (1) To check the functionality at the individual product level, we took the user viewpoint and performed system operations correctly under normal conditions and in normal environments.
- (2) To check the overall product quality, it is necessary to perform checks at all levels, from the upper application level to the lower libraries, ensuring that they are operating together correctly.

## 7. Test organization

By using the test viewpoints instead of testing the products individually, we studied ways of testing that encompass test items from the upper application levels all the way down to lower-level libraries. For example, we selected upper application-level commands that could operate all normally used functions by studying all of the interfaces between each product and between the products and hardware in detail. In this way, the tests to be performed by the shipping personnel could be

designed to cover about 70% of the NICE test items, 85% of the JICSAP2 infrastructure items, and 90% of the library items relative to the entire test item set. As shown in **Fig. 2**, the test organization comprises two configurations: configuration 1 tests operation of functions linking internal IC card software and configuration 2 tests functions linking the server, terminal, and IC card.

## 8. Evaluation of test optimizations

We studied automation suitability and established clear test viewpoints as described above and then developed test automation tools.

### (1) Efficiency of configuration 1

We evaluated the efficiency of configuration 1 by measuring the time taken to perform the tests manually and by using the test tools. Automation reduced the testing time by 98%, and testing for each version could be accomplished in approximately 30 minutes.

### (2) Efficiency of configuration 2

For configuration 2, we not only examined the increase in efficiency after test tool introduction, but also considered that a person with limited NICE testing experience could now perform the tests to the same level

as a NICE developer. The testing time required by an inexperienced NICE tester is shown in **Fig. 3**. For test procedures using the test tool and requiring fewer than five operations, a significant reduction in test times was seen after the first attempt. Since tool operation is very simple, no further effect was observed from the second attempt onwards, regardless of the tester's experience level.

For test items requiring five or more procedural steps, such as those requiring preparation before running the test tool, additional time was required for inexperienced testers on the initial attempt, but from the second attempt, once they had completed the entire test once and understood the procedure, a significant reduction in required time was observed because test settings and conditions were fixed.

By performing the tests many times and mastering the procedures, inexperienced testers could perform them to the same level as a developer (shown for reference), so a time saving of about 71% can be expected.

The results shown that tests can now be performed to the same level as an experienced tester, even when test personnel change, once the new personnel have mastered the test operations.

## 9. Future developments

By studying the suitability of automatic testing taking

into consideration the particular characteristics of the software, we can learn how various approaches—such as building product linkage test environments that clarify a product's interfaces down to the hardware level and maintaining the quality and level of testing by establishing environmental conditions from the user viewpoint—can be applied to general software development. We are advancing the application of these methods as one way to increase software development productivity in the future.

## References

- [1] J. Takahashi and Y. Kakuda, "Effective Automated Testing for Graphical Objects," Transactions of Information Processing Society of Japan, Vol. 44, No. 7, pp. 1695–1708, 2003.
- [2] R. D. Craig and S. P. Jaskiel, "Systematic Software Testing," Artech House Publishers, 2002.
- [3] R. Toji, Y. Wada, S. Hirata, and K. Suzuki, "NICE—A Network-based Platform for Multi-application Smart Cards," NTT REVIEW, Vol. 14, No. 1, pp. 13–19, 2002.
- [4] M. Yoshizawa, H. Unno, T. Fukunaga, and H. Ban, "ELWISE—A Super Multi-purpose Smart Card," NTT REVIEW, Vol. 14, No. 1, pp. 23–27, 2002.
- [5] S. Ijuin, T. Yamamoto, S. Hirata, K. Suzuki, Y. Wada, T. Kashiwagi, and N. Kaku, "Development of a NICE-based Smart Card System Conforming to the GlobalPlatform Specifications," NTT Technical Review, Vol. 2, No. 4, pp. 66–69, 2004.  
<https://www.ntt-review.jp/archive/ntttechnical.php?contents=ntr200404066.pdf>



**Noriko Fukuda**

Member, Smartcard Service Promotion Project, NTT Service Integration Laboratories.

She received the B.E. degree in information engineering from Muroran Institute of Technology, Hokkaido, in 1995. She joined NTT in 1995 and is currently engaged in the development of an ID management platform.



**Hideki Kawabe**

Senior Research Engineer, Smart Card Service Promotion Project, NTT Service Integration Laboratories.

He received the B.S. degree in mathematics from Waseda University, Tokyo, in 1986. He joined NTT Communications and Information Processing Laboratories in 1986.



**Kenji Murai**

Member, Smartcard Service Promotion Project, NTT Service Integration Laboratories.

He received the B.A. degree from the Faculty of Environmental Information, Keio University, Kanagawa, in 1995. He joined NTT in 1995 and is currently engaged in the development of an ID management platform.



**Katsuaki Miyabo**

Senior Research Engineer, Smart Card Service Promotion Project, NTT Service Integration Laboratories.

He received the B.E. and M.E. degrees in electrical engineering from Kanazawa University, Ishikawa, in 1980 and 1982, respectively. He joined the Electrical Communication Laboratories, Nippon Telegraph and Telephone Public Corporation (now NTT) in 1982. He is currently engaged in the development of an ID management platform.