An Operational Profile for the Cartridge Support Software

Ken Chruscielski
Lockheed Martin Tactical Aircraft Systems
1919 Lillian
Red Oak, Texas 75154, USA
ken@mps.lmtas.lmco.com

Jeff Tian
Southern Methodist University
Dept. of Computer Science and Engineering
Dallas, Texas 75275, USA
tian@seas.smu.edu

Abstract

This paper describes our experience and findings in constructing an operation profile for the Lockheed Martin Tactical Aircraft System's (LMTAS) Cartridge Support Software (CSS). The process is an adaptation of Musa's 5-step approach. The resulting operational profile was reviewed and evaluated by LMTAS's Software Product Manager, System Engineers, and Software Test Engineers. An account of the findings and conclusions from the independent review and evaluation is discussed. This operational profile allowed the LMTAS software engineering team to derive some clear insights about the usage rate of the CSS functions from the customer's perspective.

1. Introduction

Every software system must undergo various quality assurance activities, such as peer review, inspection, testing, and formal verification, before it is released to the market or delivered to its target customers. The primary purposes of these quality assurance activities include: 1) assuring that the customer requirements and product specifications are correctly implemented; and 2) detecting and removing defects before product release. For most software systems, because of their size and complexity, these quality assurance activities need to be carried out selectively rather than exhaustively. To achieve the optimal combination of high quality and low cost, appropriate information regarding the software usage by its target customers needs to be collected to guide quality assurance activities.

An operational profile is a quantitative characterization of how a software system will be used by its target customers [5, 6]. It consists of a collection of individual operations for a software product and the associated probabilities of occurrence. Operational profiles can be used to guide testing and other quality assurance activities, and provide the basis for us to analyze data collected from testing and other development phases to assess product quality.

The overall quality of software systems from a customer's perspective can be assessed through reliability measurement and modeling. A software system is said to be "reliable" if it performs its specified functions correctly over a long period of time or under a wide variety of usage environments and operations [2, 4]. Most software reliability models assume that the software is used in an environment that resembles its actual usage by its target customers to ensure validity of the modeling results [2]. Therefore, operational profiles play a very important role in reliability assessment and prediction. In addition, risk identification and reliability improvement techniques, such as in [7], also assume the availability and usage of operational profiles so that operational reliability can be maximized.

There are several types of operational profiles and corresponding ways to construct them. The approach by Musa, outlined in [5] and illustrated with various examples in [6], provides a systematic way to derive operational profiles in five easy-to-follow steps. In this paper, we present our experience using this approach in developing an operational profile for the Lockheed Martin Tactical Aircraft System's (LMTAS) Cartridge Support Software (CSS). The resulting operational profile was reviewed and evaluated by LMTAS's Software Product Manager, System Engineers, and Software Test Engineers, and was found to offer some valuable insight to the CSS.

2. Background and Problem Statement

The Cartridge Support Software (CSS) developed by Lockheed Martin Tactical Aircraft Systems (LMTAS) is used by aircraft personnel to load mission planning data to a read/writable media. The read/writable me-

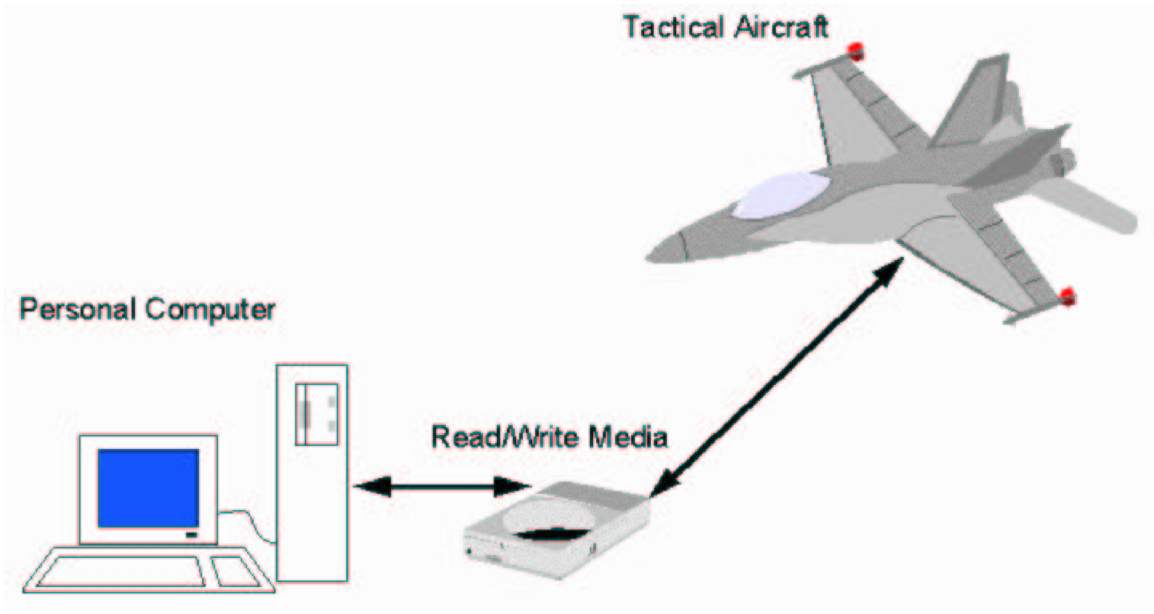

Figure 1. CSS data transfer

dia is used by pilots to upload data to the avionic computers residing on an aircraft. The ability to use a personal computer to load mission planning data is a significant convenience to aircraft personnel who would otherwise be relegated to keying in complex data settings via the cockpit interface. The CSS affords aircraft personnel an efficient and highly expedient means of performing the mission planning exercise from the Graphical User Interface (GUI) of a personal computer. CSS data transfer is illustrated in Figure 1.

The distribution of the CSS to many air forces throughout the world complicates the communication of any deficiencies identified by the user to the CSS software engineering team. In the interest of gaining an understanding of the CSS product from the perspective of the user, a goal was set for developing an operational profile. By re-focusing the software engineering testing efforts on high use functions, the reliability of the CSS could be improved.

3. Strategy

We next examine different approaches to operational profile development and outline our strategy for developing an operational profile for CSS.

3.1. Types of operations profiles

Two primary types of operational profiles include:

- Specifying the probability of operations which are mutually exclusive. Each operation usually represents a sequence of invocations of individual services provided by the software system. Musa's approach [5, 6] belongs in this category.
- Specifying the software states and state transitions in a Markov chain. Each state presents a unit of service provided by the software and linking them together forms the end-to-end operations. The operational profile derived in [1, 8] are examples of this approach.

In this paper, Musa's approach, outlined in [5] and illustrated with various examples in [6], was followed, because it provides a systematic way to derive operational profiles in five easy-to-follow steps. This choice is also influenced by the characteristics of the CSS, where multiple users and their normal usage patterns (e.g. the users may select one or more individual functions in any order and then terminate the mission, see Section 4.4) would be hard to model using Markov chains.

3.2. An overview of Musa's approach

In Musa's approach [5, 6], the operational profile is generated using these five basic steps:

1. *Find the customer profile:* A definition of a complete set of customer categories weighted per a usage factor.
2. *Establish the user profile:* A definition of user types and weighted in relation to the customer group and relative usage rates.

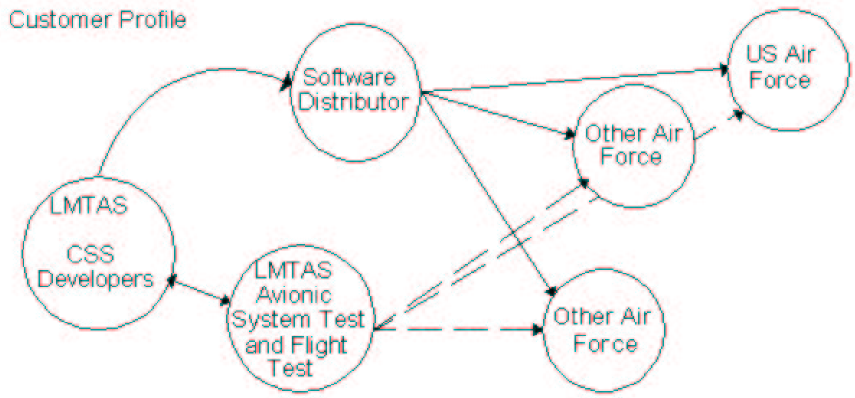

Figure 2. Customer profile

3. *Define the system modes:* A definition of the types of operations/modes of the software system.

4. *Determine the functional profile:* An analysis and abstraction of the system modes into individual functions and an analysis of environmental variables which affect the specified functions.

5. *Determine the operational profile:* The collection of functions the software item can perform with their probabilities of occurrence.

Musa suggested that the operational profile should be developed by a combination of system engineers, high-level designers, and test planners, with strong participation from product planning and marketing functions.

3.3. Participants

The primary personnel involved with the CSS Operational Profile development were:

1. *Software Product Manager (SPM):* Responsible for the product planning and marketing function of the CSS. The SPM was instrumental in determining communication paths for the solicitation of user inputs to the Operation Profile.

2. *Software Test Engineers:* Responsible for providing understanding into testable input states per a typical field operation. The test engineers have considerable discernment into pockets of knowledge regarding the customer perspective on certain software functions. The term "pockets of knowledge" describes the effect of a localized knowledge base to a relatively few amount of experts.

3. *System Engineers:* Responsible for specifying the system requirements and the high level design. The system engineers detail all of the deliverable functions which are then verified during the software testing phase of the lifecycle. The system requirements do not provide information regarding which functions will be used at a higher/lower rate in relation to other functions.

4. CSS Operational Profile Development

This section describes in detail each of the five steps that were followed in developing our CSS OP. The various problems that were encountered and their solutions are also discussed.

4.1. Customer profile

The customers of the CSS are the air forces which use the LMTAS tactical aircraft. A consultation with the SPM revealed the communication paths as shown in Figure 2.

The LMTAS CSS developers utilize a centralized agency in the Department of Defense which serves as the sole interface/distributor to the external customers; they are labeled in the above diagram as the "software distributor". The CSS external customers are labeled in the above diagram as "US Air Force" or "Other Air Force". These external customers each have a unique contractual agreement with LMTAS. The CSS also has internal users which have been labeled in the above diagram as "Avionic System Test and Flight Test." The internal customers are part of the LMTAS organization and are in direct communication with the CSS software development team.

User Group	Marketing Concerns	Frequency of Use	Total Weighting Factor
Air Force Pilot	0.85	0.05	0.45
Flight Test Support	0.10	0.80	0.45
Avionics System Test	0.05	0.15	0.1

Table 1. User profile

Musa suggested that if all of the customers employ the product in an equivalent manner a customer profile will not be of particular value. For the purposes of the CSS operational profile the investigation into the customer profile did not result in a "weighting" of the customers, as each customer had a similar use of the CSS. Unique applications of the CSS existed for each of the customers, however, an analysis of these special functions would require a more intensive effort with conclusions applicable for a specific customer base. Therefore, the CSS operational profile dealt with the common functions put into use by the generic Air Force customer. The exercise did serve as a means of establishing communication paths for the collection of user inputs.

4.2. User profile

The users of the CSS include the following types of flight personnel and engineering support:

1. *Air Force Pilots:* The primary users of the CSS. The pilots are directly involved with the mission planning exercise, however their use can be very infrequent.

2. *Flight Test Support:* These users have been found to be frequent users of the CSS and interface directly with test pilots during the mission planning exercise.

3. *Avionic System Test:* These users are involved in integrating specific, as well as, the entire suite of avionics residing in the aircraft.

4. *System Administrators:* This user group was found to be performed by an Air Force Pilot. The user group was combined into the Air Force Pilot classification.

The user groups were weighted as shown in Table 1. The pilots are the primary contractual customer of the CSS, and as such, were weighted very heavy due to marketing concerns. A key factor for the user profile was how often the user operates the CSS. The pilots were found to be very infrequent users. The Flight Test Support personnel who work directly with pilots were weighted very heavy due to their frequency of use (on a daily basis). The Flight Test Support users work directly with pilots so the marketing factor was significant. The avionic system test personnel's marketing factor was weighted extremely light relative to the importance of our primary customer base (Air Force pilots).

4.3. System modes

The system modes for the CSS were determined to fall into the following categories:

1. *Preflight Mission Planning:* The Pilot or Flight Test Support personnel plan a mission.

2. *Avionic System Test:* The system test engineers use the CSS to stimulate avionics as part of the verification process during system integration.

3. *System Administration:* The administrator uses the CSS to maintain a database of preflight mission files.

The personal computers located at the site of the CSS users are x486 or Pentium processor based machines. The operating system for the personal computers includes the following: Microsoft Disk Operating System (MS-DOS), Windows™ 3.x, Windows NT, and Windows 95. The users of the system rarely run a concurrent application, but if they do it might be a word processor or a read/write media support package. LMTAS software test engineers inquired as to whether a large number of applications are run concurrently, thus requiring large amount of system resources (memory). The results of the user survey determined that the respondents tended to execute the CSS without any accompanying software applications.

An analysis of the system operational behavior revealed that there is not an appreciable difference accounted for between the system modes. Therefore a categorization and weighting of identified system modes was not performed. The derivation of the CSS user profile and the associated weighting factors remains the most significant component for determining the CSS operational profile.

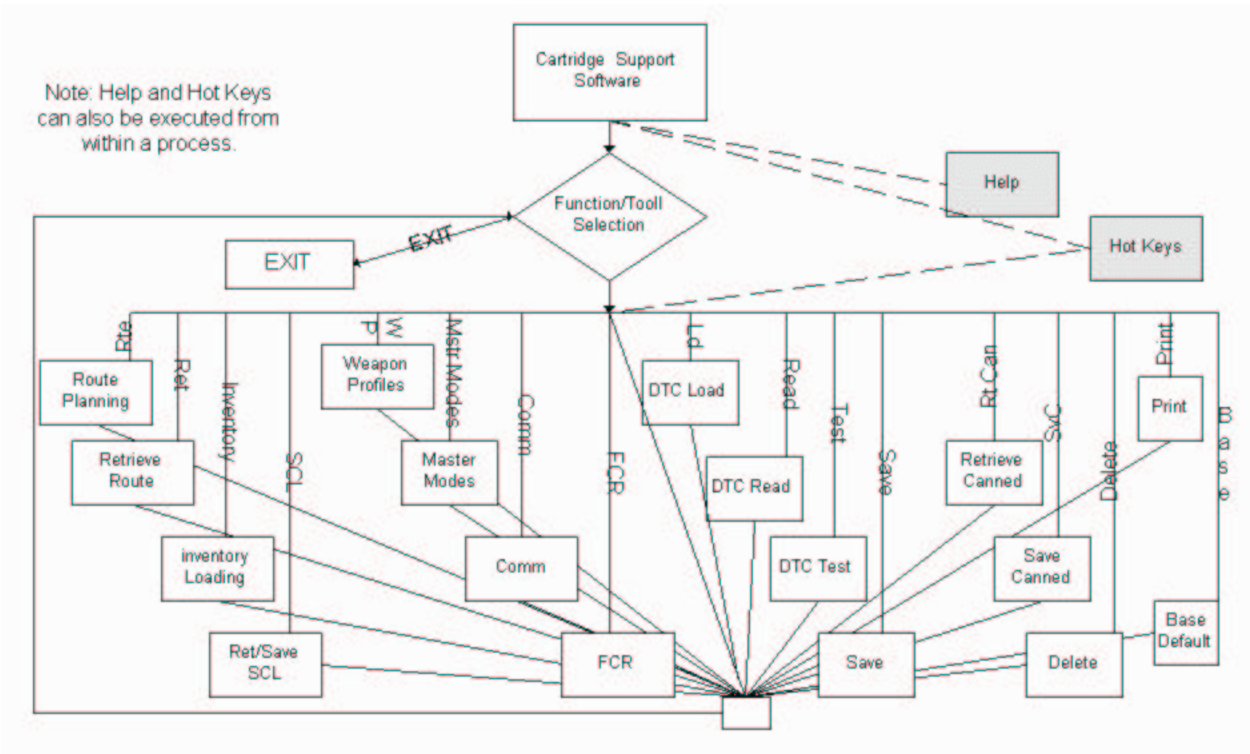

Figure 3. Control flow diagram for the CSS

4.4. Functional Profile

The functional profile of the CSS is broken down into distinct processes which are referred to in CSS documentation as "tools" or as separate functions. Each function of the CSS is typically associated with it's own dialog or window. A few of the functions of the CSS such as the "hot keys" function are simply a means of accelerating the mission planning exercise and are not associated with a dialog. A control flow diagram showing the actions involved in using the CSS is shown in Figure 3.

As can be seen from Figure 3, the user has the ability to enter one or more individual functions (processes) and then terminate the mission planning session. The user does not have a defined order in which the functions must be executed. Because of this, we chose to use an implicit functional profile [5], listing only the occurrence probability for each individual function, rather than the end-to-end functional sequences that define an explicit functional profile.

The classification of the CSS into easily divisible functions made for a straightforward selection of the more common functions for the operational profile. Each contractual agreement with a customer has a certain amount of unique functions which are developed for that particular application. An extensive part of the CSS, however, has functions which are largely replicated from one program to the next. It was these common functions which were analyzed in the generation of the operational profile.

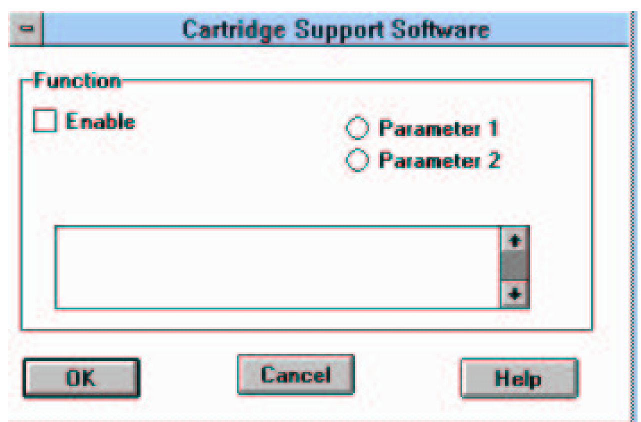

Figure 4. Typical CSS dialog

The primary tools or functions of the CSS are partitioned into dialogs. The dialogs are built with a various assortment of user selectable options and parameters much like what is shown in Figure 4.

Route Planning	Retrieve Route
Inventory Loading	Ret/Save SCL
Weapon Profiles	Master Modes
Communications	Fire Control Radar
DTC Load	DTC Test
DTC Read	Help
Save Canned Data	Ret Canned Data
Save	Delete
Hot Keys	Print
Base Default	

Table 2. CSS tools and functions

The containment of a given function per a dialog allowed for a convenient means of measuring usage rates. The user was questioned as to the rate of use for a given function - which was easily discernible due to the embodiment of the function on a given dialog. The exception to this rule was the use of the "hot keys" which were used without regard to tool use. (In the case of the hot keys the overall accelerator key function was surveyed as to frequency of use.)

The identification of certain existing CSS Tools and Functions were determined to be directly applicable to future projects. Table 2 lists those tools/functions which were analyzed for the operational profile. The acronym DTC in the table refers to a Data Transfer Cartridge which is the read/writable media discussed earlier in this paper.

4.5. Operational profile

An operational profile was created for each of the user groups (shown in Figures 5, 6, and 7). Notice that an implicit operational profile [5] is used here, i.e., the probabilities are for individual operations rather than end-to-end operation sequences (Section 4.2). A comprehensive operational profile (Figure 8) was created from the weighted factors shown in the user profile.

An indication of how each user group uses the CSS was found to be beneficial to the System Engineers and Test Engineers. The user groups all have a significant contribution during the lifecycle of a tactical aircraft and each user and their requirements have to be satisfied.

4.6. Metrics collection

The numbers for the operational profile were derived from electronic mail and fax copies of user surveys through the communication channels described in the customer profile section of this report. The survey was generated in the format of the Questionnaire for User Interface Satisfaction (QUIS). This format, as described by D. Hix and H. Hartson [3], is "the most extensive and most thoroughly validated questionnaire for determining subjective interface usability." The advantage of using electronic communications was that it allowed for a quick transmission of the surveys to remote locations. The desirable prospect of this approach was that one user would "forward' the survey to other users - thus creating a "chain letter" effect, and thereby increasing user participation in the survey. The disadvantage of this approach was that the status of the survey replies was a difficult factor to correctly determine. The projected response to the survey was 30-50 users. The actual response was 12 users. However, the participants who did engage in the survey were considered to be significantly reflective of their user groups. As Musa suggests [5], "...a sample of 30 or perhaps even fewer users may be satisfactory for generating an operational profile with acceptable accuracy."

5. Result Analysis and Validation

In the CSS developmental organization the software test effort is applied to most of the functions using a broad coverage type of testing strategy. A small minority of functions are considered to be of prime importance and receive a copious amount of emphasis in the development lifecycle. Those functions which are considered to be of very low importance are given a brief cursory test to determine their functionality. The formation of the CSS operational profile was intended to appraise this current approach that is used in software testing.

5.1. Classification of operations

The comprehensive operational profile has been grouped into High, Medium-High, Medium-Low, and Low use categories. These classifications correspond to the software management concern for the prioritization of defect resolution. The LMTAS software engineering team reviewed and evaluated the operational profile results and the usage probability classifications shown in Table 3.

5.2. Evaluation by software engineering team

A review of the CSS operational profile was conducted with the SPM, System Engineer, and Test engineers. The test engineers felt that the operational profile confirmed some of the expectations we had of

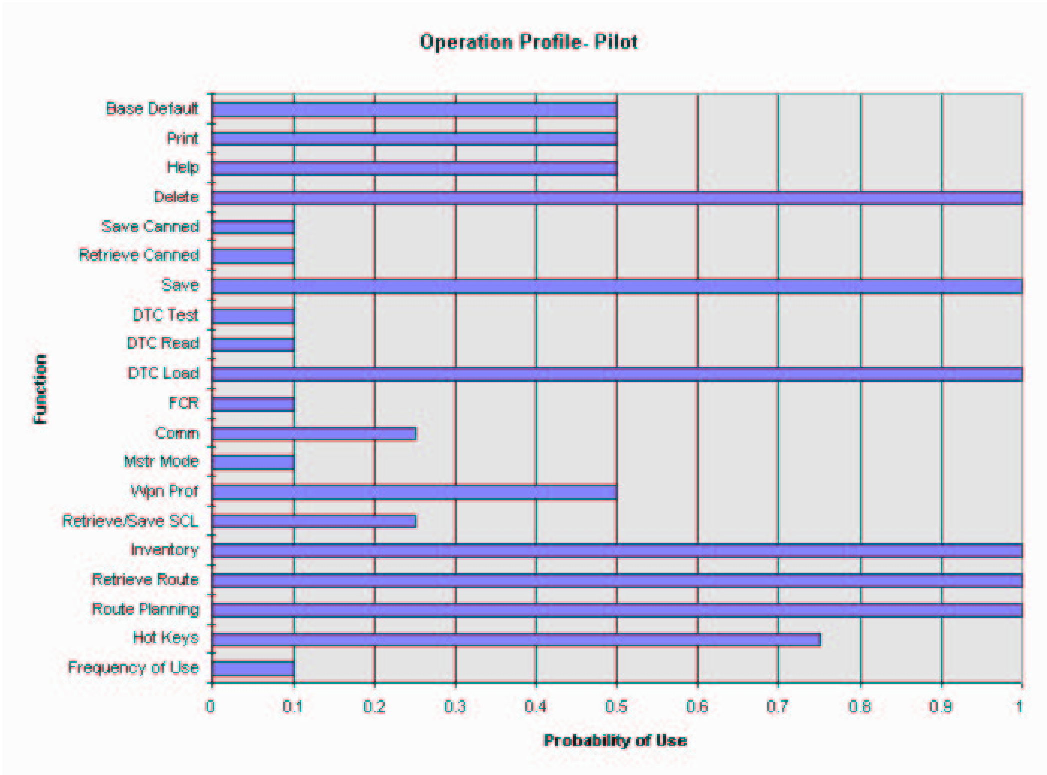

Figure 5. Operational profile: Pilot

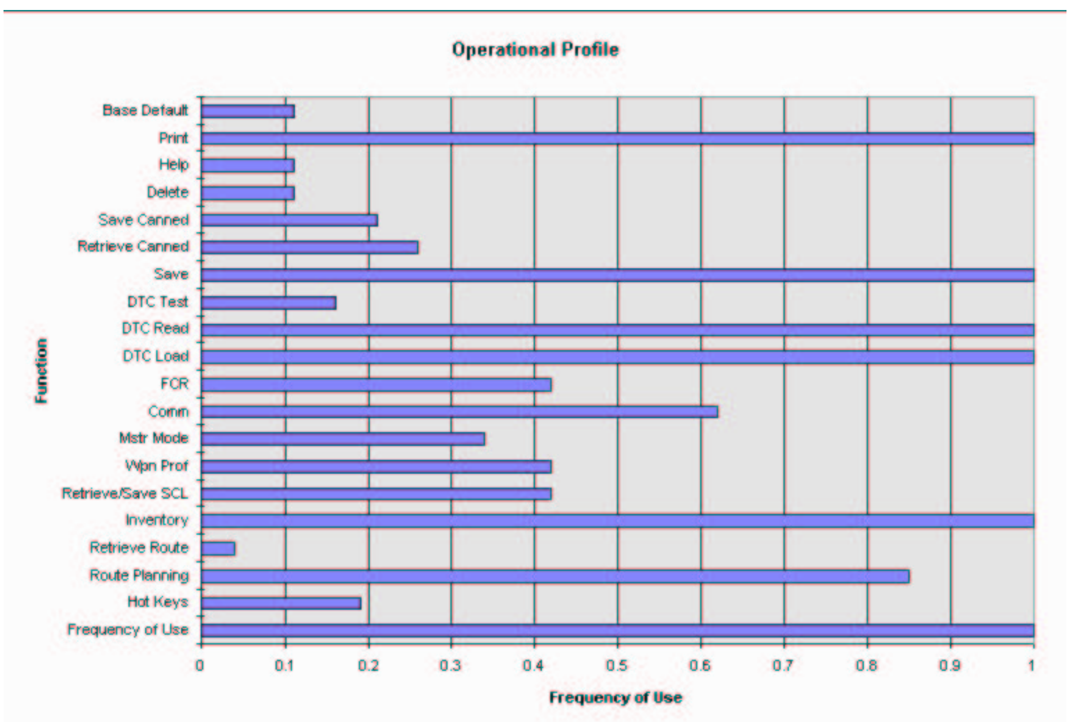

Figure 6. Operational profile: Flight Test Support

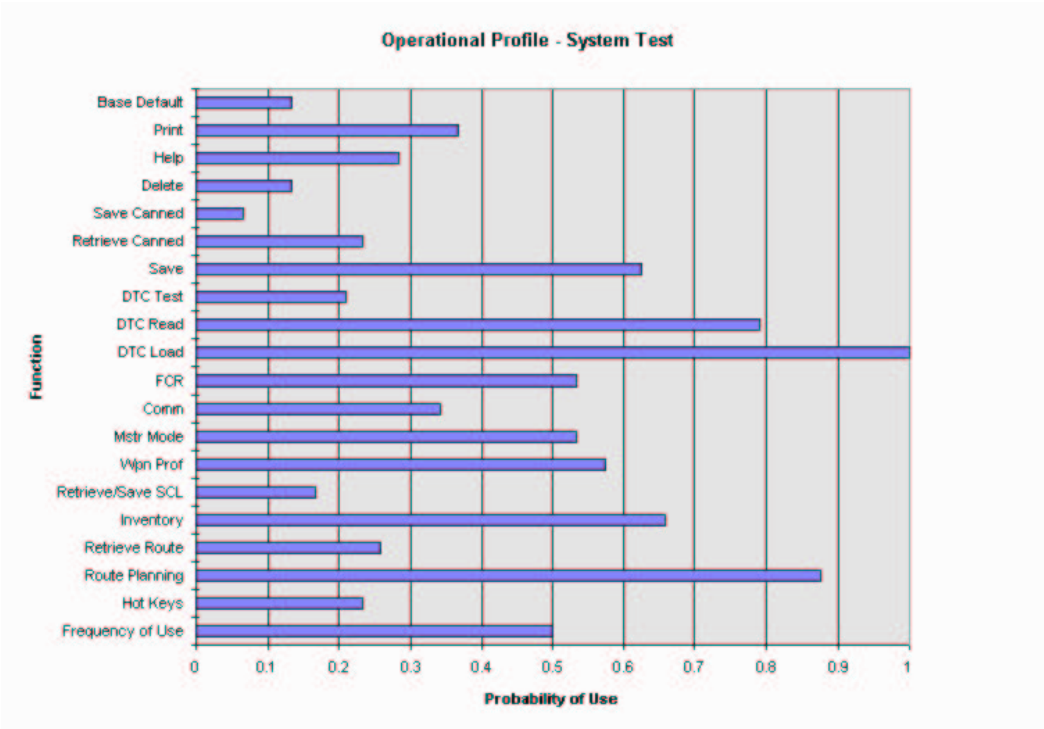

Figure 7. Operational profile: System Test

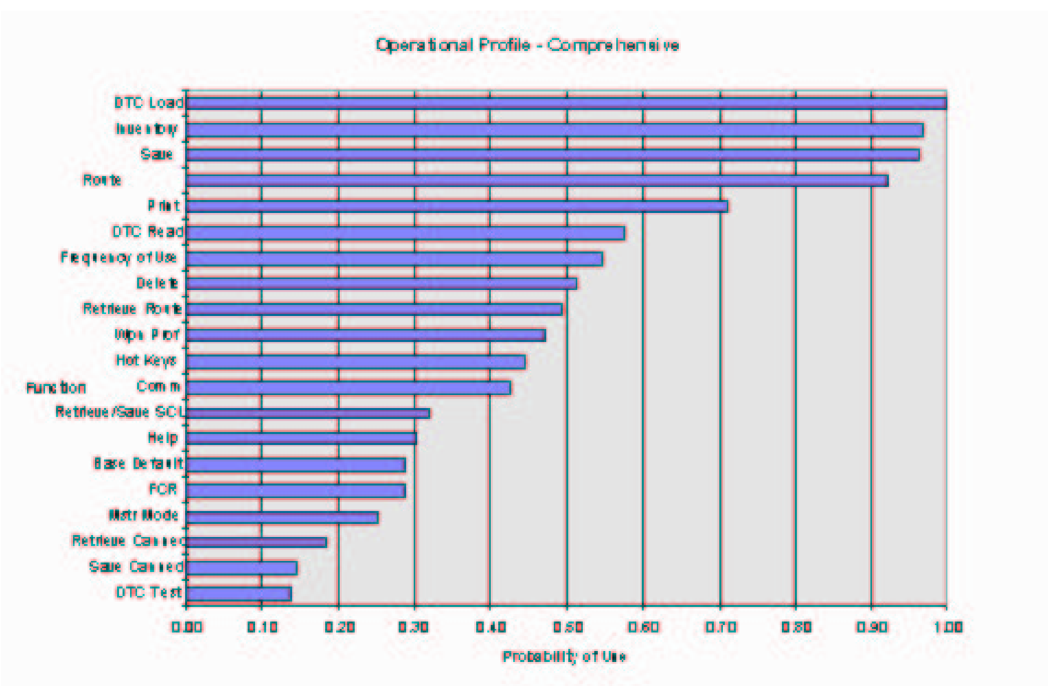

Figure 8. Operational profile: Comprehensive

High	Medium-High	Medium-Low	Low
DTC Load Inventory Save Route Planning Print	DTC Read Delete Retrieve Route	Wpn Prof Hot Keys Comm Retr/Save SCL Help Base Default FCR Mstr Mode	RetrCanned Save Canned DTC Test
High Usage = 100% – 75%	Medium High Usage = 74.9% – 50%	Medium-Low Usage = 49.9% – 25%	Low Usage = 24.9% – 0%

Table 3. CSS Classification according to probability of use

our customers. The comprehensive profile classifications for the "high" use functions indicated that existing testing efforts on the DTC Load, Route, and Inventory functions have been validated. The Route function in particular is a function which has received considerable attention from our software test function. Authenticating the current software engineering efforts towards these functions provides confirmation that LMTAS's efforts are on target towards the CSS high use functions.

The system engineer and test engineers evaluated the "medium to high" and "medium to low use" functions and found the (medium use) classification of the Hot Keys Function as a completely unexpected result. This function has been considered to be of low importance to the customer. The CSS has been re-designed to take to take full advantage of Windows GUI standards. The user's continued reliance on Hot Keys should require modifications of current testing strategies.

The classification of the Help function as a "medium to low" use function was another unexpected result. The CSS developers had believed that most users are familiar with the overall operation of the software functions, and as such, would require a minimal amount of help. The engineers on the review team hypothesized that the Help function may receive a significant measure of use because the current releases of software are new Windows products with a substantially different look and feel.

The Weapon Profiles, Communication, Fire Control Radar, and Master Mode tools may not be "high" use functions (as expected) due to the saving and retrieving of files from previous missions. The system engineer on the review board suggested that further inquiry on this point should be weighed from a peace time scenario versus a war time scenario. The saving and retrieving of this data from previous missions might become a important part of the software testing strategy.

The more critical findings of the evaluation have been listed, but information regarding each function's relative importance to the user has been beneficial to the software engineering team. Prior to the generation of the CSS operational profile there had never been a comprehensive review of the CSS product from the customer perspective. The findings of the operational profile provide a resource for all team members to attain a basic understanding of the strategy required to improve software reliability.

The review and evaluation of the CSS operational. provided a significant contrast to a testing strategy based primarily on a verification according to a requirements checklist. A testing approach supported by an operational profile is able to emphasize the verification of system functionality determined to be of high importance to the customer. The solicitation of input from the customer worked to establish increased confidence in the development of a product which would be delivered with a greater degree of reliability. A higher appreciation for communicating with the customer, as well as, an increased opportunity for improving the CSS testing strategy were a direct result of the operational profile.

6. Effort

The generation of the CSS operational profile required the participation of the SPM to outline the marketing aspects of the software product. Several short interviews with the SPM, over the span of a few weeks, identified key areas of the CSS and several communication paths to the users. Follow up discussions with the SPM helped to define the requirements for the user profile and functional profile.

Much of the existing system design of the CSS guided the generation of the functional profile. Consultations with system engineers and test engineers, during a two week period, were instrumental in the development of the survey form which was sent out to the CSS users. Ninety two percent of the responses to the survey were collected within two weeks of the survey distribution. Eight percent of the responses took as long as 3-4 weeks for collection (due to the delay involved in "forwarding" the email surveys to the appropriate people).

The final results evaluation required each member of the software engineering team to interpret the operation profile. Individual interviews with each member of the software engineering team was beneficial in capturing unique perspectives on the operational profile. Initially, team members were not aware of how the data could be used in practical situations. Over the course of the interview each member began to suggest possible explanations for the results of the data. This led the review team to contrast current testing strategies with the identified needs of the customers. Individuals then offered action plans to accommodate customer needs and improve software reliability.

7. Conclusions and Future Applications

The development and usage of operational profiles in software development and testing can help improve the testing process and product reliability. This quantified customer usage information for our software products can guide us to focus our testing effort on functions and operations that are important to our customers, and can lead to improved product reliability.

Among the various approaches proposed by researchers and practitioners to develop operational profiles, Musa's approach [5, 6] provides a systematic and easy to follow process. As demonstrated by this paper an operational profile can be developed for the LMTAS CSS with a reasonable amount of effort. Feedback from participants in this activity, including program managers, system engineers, and test engineers, also confirms the feasibility and benefit of this activity.

It should also be noted that the generation of an operational profile has quite a few hidden benefits besides improving the reliability of the software product through a more focused testing effort. The increased emphasis on the customer perspective also affected the system and high level design effort. Continual improvements to the electronic communication across the Internet, and a pervasive use of email, may increase the possibilities for appropriating user input to the software development effort.

With the advent of new software metric collection tools we expect to be able to estimate improvements in software reliability and further refine and augment the CSS operational profile. The customers of the CSS demand higher levels of reliability from their software product and the LMTAS software engineering effort will continue to apply new methodologies to meet or exceed customer expectations.

Acknowledgment

This effort could not have been possible without the support and expertise provided by the following LMTAS software engineering team: Ken Stogsdill, Scott Law, Richey Havenhill, and Martin Smith.

References

[1] A. Avritzer and E. J. Weyuker. The automatic generation of load test suites and the assessment of the resulting software. *IEEE Trans. on Software Engineering*, 21(9):705–716, Sept. 1995.

[2] A. L. Goel. Software reliability models: Assumptions, limitations, and applicability. *IEEE Trans. on Software Engineering*, 11(12):1411–1423, Dec. 1985.

[3] D. Hix and H. Hartson. *Developing User Interfaces: Ensuring Usability Through Product and Process.* John Wiley & Sons, 1993.

[4] M. R. Lyu, editor. *Handbook of Software Reliability Engineering.* McGraw-Hill, New York, 1995.

[5] J. D. Musa. Operational profiles in software reliability engineering. *IEEE Software*, 10(2):14–32, Mar. 1993.

[6] J. D. Musa, G. Fuoco, N. Irving, D. Kropfl, and B. Juhlin. The operational profile. In M. R. Lyu, editor, *Handbook of Software Reliability Engineering*, pages 167–216. McGraw-Hill, New York, 1995.

[7] J. Tian. Integrating time domain and input domain analyses of software reliability using tree-based models. *IEEE Trans. on Software Engineering*, 21(12):945–958, Dec. 1995.

[8] J. A. Whittaker and M. G. Thomason. A Markov chain model for statistical software testing. *IEEE Trans. on Software Engineering*, 20(10):812–824, Oct. 1994.