DOT/FAA/AR-01/116

Office of Aviation Research
Washington, D.C. 20591

Software Service History Handbook

January 2002

Final Report

This document is available to the U.S. public through the National Technical Information Service (NTIS), Springfield, Virginia 22161.



U.S. Department of Transportation
Federal Aviation Administration

NOTICE

This document is disseminated under the sponsorship of the U.S. Department of Transportation in the interest of information exchange. The United States Government assumes no liability for the contents or use thereof. The United States Government does not endorse products or manufacturers. Trade or manufacturer's names appear herein solely because they are considered essential to the objective of this report. This document does not constitute FAA certification policy. Consult your local FAA aircraft certification office as to its use.

This report is available at the Federal Aviation Administration William J. Hughes Technical Center's Full-Text Technical Reports page: actlibrary.tc.faa.gov in Adobe Acrobat portable document format (PDF).

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
DOT/FAA/AR-01/116		

4. Title and Subtitle	5. Report Date
SOFTWARE SERVICE HISTORY HANDBOOK	January 2002
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Uma D. Ferrell and Thomas K. Ferrell	

9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Ferrell and Associates Consulting, Inc. 1261 Cobble Pond Way Vienna, VA 22182	11. Contract or Grant No. DTFA0300P10138

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
U.S. Department of Transportation Federal Aviation Administration Office of Aviation Research Washington, DC 20591	Final Report
	14. Sponsoring Agency Code AIR-130

15. Supplementary Notes

This handbook is one of two deliverables under this contract. It is intended for wide dissemination while the accompanying Software Service History Report is primarily intended for FAA use. The FAA William J. Hughes COTR is Charles Kilgore.

16. Abstract

The safe and reliable operation of software within civil aviation systems and equipment has historically been assured through the application of rigorous design assurance applied during the software development process. Increasingly, manufacturers are seeking ways to use software that has been previously developed for other domains or that has been previously certified for use in lower criticality aviation applications. Product service history is one method for demonstrating that such software is acceptable for use in the new application domain. In theory, product service history would seem to be a fairly simple concept, both to understand and to apply. However, in practice, such use has proved extremely problematic, as questions of how to measure the historic performance and the relevance of the provided data have surfaced. This handbook is intended to aid industry and the Federal Aviation Administration in the formulation and evaluation of product service history data for certification credit. It provides a discussion of the major issues associated with product service history and provides an approach for methodically evaluating service history data.

17. Key Words	18. Distribution Statement
DO-178B, SC-190, DO-248, Product service history, Software reliability, Airborne systems and equipment, Error rates	This document is available to the public through the National Technical Information Service (NTIS) Springfield, Virginia 22161.

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	44	

TABLE OF CONTENTS

		Page
EXECUTIVE SUMMARY		vii
1.	INTRODUCTION	1
1.1	Purpose	1
1.2	Scope	1
1.3	Background	1
1.4	Related Activities/Documents	2
1.5	Document Structure	3
1.6	Using the Handbook	3
2.	DO-178B FRAMEWORK	4
2.1	The Definition of Product Service History	4
2.2	Analysis of Product Service History in DO-178B	5
2.3	Relationship With Previously Developed Software	14
2.4	Product Service History Versus Software Reliability	14
3.	THE ELEMENTS OF PRODUCT SERVICE HISTORY	15
3.1	Questions of Problem Reporting	15
3.2	Questions of Operation	17
3.3	Questions of Environment	19
3.4	Questions of Time	21
4.	ADEQUACY OF DEVELOPMENT PROCESS	23
5.	ESTABLISHMENT OF “EQUIVALENT SAFETY”	25
6.	SUMMARY	26
7.	BIBLIOGRAPHY	27
APPENDIX A—EVALUATION WORKSHEETS		

LIST OF FIGURES

Figure		Page
1	Operation	17
2	Environment	19
3	Timeline	21
4	Calculation of Service History Duration	23

LIST OF TABLES

Table		Page
1	Analysis of DO-178B, Section 12.3.5	6

LIST OF ABBREVIATIONS

AC	Advisory Circular
ACO	Aircraft Certification Office
ATM	Air Traffic Management
CAST	Certification Authorities Software Team
CNS	Communication, Navigation, Surveillance
COTR	Contracts Technical Representative
COTS	Commercial Off-The-Shelf
DO	Document
DOT	Department of Transportation
ED	European Document
EUROCAE	European Organization for Civil Aviation Equipment
FAA	Federal Aviation Administration
FAR	Federal Aviation Regulations
HBAW	Airworthiness Handbook
IEEE	Institute of Electrical and Electronic Engineers
JAA	Joint Airworthiness Authorities
NASA	National Aeronautics and Space Administration
OIS	Other Industry Sector
PDF	Portable Document Format
PSAC	Plan for Software Aspects of Certification
RTCA	formerly Radio Technical Commission for Aeronautics
SAS	Software Accomplishment Summary
SC	Special Committee
SOUP	Software of Unknown Pedigree
SSAC	Streamlining Software Aspects of Certification
TGL	Temporary Guidance Leaflet

EXECUTIVE SUMMARY

The safe and reliable operation of software within civil aviation systems and equipment has historically been assured through the application of rigorous design assurance applied during the software development process. Increasingly, manufacturers are seeking ways to use software that has been previously developed for other domains, has been previously certified for use in lower criticality aviation applications, or has been certified to earlier versions or different standards than those currently employed. Product service history is one method for demonstrating that such software is acceptable for use in a new application. In theory, product service history would seem to be a fairly simple concept, both to understand and to apply. However, in practice, such use has proved extremely problematic, as questions of how to measure the historic performance and the relevance of the provided data have surfaced. To date, no specific guidance has been produced to aid in the formulation of service history approaches beyond the limited discussion in DO-178B, "Software Considerations in Airborne Systems and Equipment Certification." This research effort was designed to collect, analyze, and synthesize what is known and understood about applying product service history and then to adapt this data into a handbook.

This technical report presents the results of this research effort in the form of a handbook intended for use by both the Federal Aviation Administration and industry in formulating and evaluating service history arguments. Using a taxonomy of questions derived from the definition of product service history in DO-178B, both quantitative and qualitative considerations are explored. This discussion is extended by inclusion of additional questions from other industries in which service history is used in evaluating software maturity. Finally, a set of worksheets are derived that can be used by anyone evaluating the relevance and sufficiency of service history data for possible certification credit. The handbook concludes with a discussion of process assurance and equivalent levels of safety for the purposes of determining when and what type of supplemental data may be required to fulfill the objectives of DO-178B in conjunction with the use of service history.

1. INTRODUCTION.

1.1 PURPOSE.

The purpose of this handbook is to aid industry and the Federal Aviation Administration (FAA) in the application of product service history for certification credit within the framework of DO-178B, "Software Considerations in Airborne Systems and Equipment," as invoked by FAA guidance, Advisory Circular (AC) 20-115B[1].

1.2 SCOPE.

The scope of this handbook covers the subject of product service history for software used for certification credit in approval of airborne systems. The content may be useful in other situations where product service history credit is being sought.

1.3 BACKGROUND.

During the creation of DO-178B, product service history was identified as a possible alternative method for demonstrating compliance to one or more of the objectives in DO-178B. To date, use of this method has been limited due to both the difficulty in demonstrating the relevance and sufficiency of the product service history, as well as a lack of any consistent guidance for approaching such a demonstration.

This handbook, the result of an FAA study, attempts to capture in one place what is known about the topic of product service history. Using the guidance provided in DO-178B as a starting point, other safety-critical industries were canvassed in an attempt to identify if service history was being used as part of system evaluations, and if so, in what manner. Similarly, other sources of guidance for the aviation industry were explored, most notably the work accomplished by RTCA committees (SC-180 and SC-190) and by the Certification Authorities Software Team (CAST).

The SC-180 committee produced DO-254, "Design Assurance Guidance for Airborne Electronic Hardware." It outlines how product service experience may be used "to substantiate design assurance for previously developed hardware and COTS components." DO-254 was released in April of 2000. DO-254's treatment of product service experience is contained in two separate sections, 11.3 and Appendix B, paragraph 3.2. The guidance in 11.3 closely parallels the guidance in DO-178B for product service history. However, the guidance in the appendix requires additional design assurance for levels A and B hardware if service experience is claimed. There is also a requirement to link any analysis of product service history experience into the functional failure path analysis for levels A and B. This is analogous to the tie back to system safety objectives required in 12.3.5 of DO-178B.

---

[1] DO-178B is published by RTCA, Inc. and is used widely in the United States. It's European counterpart, ED-12B, is published by EUROCAE and is used throughout EUROPE. The documents are technically identical. The Joint Airworthiness Authorities (JAA) invokes the use of DO-178B via Temporary Guidance Leaflet (TGL) No. 4 in a similar fashion to AC 20-115B.

SC-190 is still an active committee, although their final work products are currently moving through editorial review as of the publication of this handbook. Their outputs include DO-248B and a guidance document for nonairborne Communication, Navigation, Surveillance/Air Traffic Management (CNS/ATM) ground systems. Within DO-248B, frequently asked question No. 19 and discussion paper no. 4 specifically address the use of product service history. The content of these items have been reflected in this handbook. Considerable discussion occurred in the SC-190 CNS/ATM subgroup on the establishment of a tiered approach to minimum service history duration based on criticality levels. No consensus could be reached on the minimum duration since the proposals were derived from the software reliability field that, by some, is not viewed to be a mature field.

CAST is comprised of representatives of certification authorities from Europe, Canada, and the United States. CAST meets regularly to discuss technical and regulatory matters related to the uniform interpretation of DO-178B and related guidance. CAST produced a position paper on the subject of product service history. Both the final paper and a number of earlier drafts were considered in the course of completing this research effort.

In addition to the aviation sources mentioned above, numerous references were reviewed from the nuclear, military, consumer products, and medical devices domains, as well as general software literature. The most mature treatment of the topics were found in Europe, most notably in standards published by the United Kingdom Ministry of Defense (MoD) and by the safety-consulting firm, Adelard. In addition to the written materials that were reviewed, a series of interviews were conducted with practitioners in these other fields to further explore exactly how the subject of service history was addressed in practice.

The final activity prior to the actual creation of this handbook was the conduct of a detailed breakout session during the FAA's National Software Conference held in Boston in June 2001. Preliminary results of the study were shared and feedback was sought related to specific issues arising from the product service history definition. Both the interviews and the breakout session served to validate the findings from the literature review and contributed greatly to the final handbook contents.

1.4 RELATED ACTIVITIES/DOCUMENTS.

The following documents relate directly to the issues addressed herein and define the nature of the problem studied in this evaluation:

a. DO-178B/ED-12B
b. DO-254/ED-80
c. DO-248B/ED-94B
d. Federal Aviation Regulations (FARs) (more formally known as Title 14 Code of Federal Regulations (CFR) and associated ACs, most notably

- XX.1309
- XX.1301
- XX.901

where XX represents the aircraft type for example, CFR Parts 25.1309, 23.1309 etc.

In addition, FAA Notices 8110.85 and 8110.89 are relevant to this discussion. Finally, specific discussion papers presented at the RTCA SC-167 and SC-190 meetings were reviewed and considered. In many cases, these papers contained useful ideas that were not included in the final products of their associated committees due to a lack of consensus or the constraints placed on the committee.

1.5 DOCUMENT STRUCTURE.

This handbook is comprised of seven sections, and one appendix. A brief summary of the contents is provided here for reference.

- Section 1 provides introductory material including the purpose, scope, related documents, background, document structure, and use of the handbook.
- Section 2 discusses DO-178B's treatment of the product service history alternative method, as well as its definition.
- Section 3 provides a detailed discussion of the elements that comprise the product service history definition.
- Section 4 discusses the relationship of product service history to both process and product assurance.
- Section 5 draws the various aspects of a product service history argument together for the purposes of illustrating how an equivalent level of safety argument might be made using product service history data.
- Section 6 is the summary.
- Section 7 contains a bibliography listing the most relevant sources of information on service history and related topics.
- Appendix A provides a series of worksheets that may be used to evaluate product service history data.

Note: Throughout this handbook, the language and the philosophy of DO-178B are retained. For example, the vocabulary used in various domains of this research is different from that used in DO-178B. Words such as "in-service history," "field data," and "item history" are used for product service history. A translation has been performed to maintain commonality of terms with those used in DO-178B. Similarly, the terms product service history and software service history are used interchangeably.

1.6 USING THE HANDBOOK.

This handbook has been designed to capture industry's best practices used in evaluating product service history for possible certification credit. Practitioners are encouraged to review the

commentary in sections 1 through 3 when initially contemplating the use of product service history on a project. The worksheets contained in appendix A of this handbook can be used in performing an evaluation using the questions and ideas discussed in section 3.

Once the initial evaluation has been completed using sections 1-3 and appendix A of this document, sections 4 and 5 can be used for ideas on how to supplement the service history data if necessary. The need for such supplemental activities is a result of the inclusion of 12.3 in DO-178B which states that all alternative methods be shown to meet the objectives of DO-178B. Since product service history is often being considered because complete development data is unavailable, multiple alternative methods may be needed to satisfy all DO-178B objectives (more on this in section 5). Any use of service history should be discussed in the Plan for Software Aspects of Certification (PSAC) and coordinated with the appropriate Aircraft Certification Office (ACO).

Note: This handbook is the output of a research effort. It does not, by itself, constitute policy or guidance. The FAA may use this handbook in the creation of future policy or guidance.

2. DO-178B FRAMEWORK.

DO-178B outlines a total of 66 objectives that should be satisfied for software with the highest potential impact on safety in the event of its failure. Four additional levels of software are provided, each with a decreasing number of objectives that must be satisfied as the potential safety impact is reduced. All of the objectives are described in the context of the software development process and a series of integral processes that cut across the entire software development effort. In addition, DO-178B discusses a small number of alternative methods for demonstrating compliance to one or more of the 66 objectives. Product service history is one such alternative method.

2.1 THE DEFINITION OF PRODUCT SERVICE HISTORY.

DO-178B defines product service history as "a contiguous period of time during which the software is operated within a known environment, and during which successive failures are recorded." This definition has three major components:

- Problem reporting
- Environment
- Time

For the purposes of this handbook, the environment component has been subdivided into two pieces. The first one is focused on external operations such as operating modes, people and procedures, and the second is focused on computing (hardware environment) aspects of the software. Likewise, the problem reporting component has been broadened to include all facets of configuration management as they relate to the use of product service history.

DO-178B, Section 12.3.5, states that the acceptability of any argument predicated on the use of product service history depends on six items:

- configuration management of the software
- effectiveness of problem reporting activity
- stability and maturity of the software
- relevance of product service history environment
- actual error rates and product service history
- impact of modifications

The next section provides a detailed look at the 11 guidance statements in Section 12.3.5 of DO-178B as they relate to demonstrating the above six items.

2.2 ANALYSIS OF PRODUCT SERVICE HISTORY IN DO-178B.

Table 1 was constructed as a result of analysis of current guidance given in Section 12.3.5 of DO-178B. Each item in this section was studied to understand what questions must be asked to get pertinent information and what additional considerations are not discussed directly in DO-178B. The components of time, environment, operations, and problem reporting have been included to categorize each of the guidance statements from DO-178B. This taxonomy will be explored in detail in section 3.

Column 1, DO-178B Section 12.3.5 reference, contains each of the 11 guidance statements concerning product service history as it appears in DO-178B.

Column 2, observations on DO-178B Section 12.3.5, provides a brief commentary on the guidance statement discussing how that item may be met and in what way.

Column 3, software service history questions, provides a short list of questions that can be directly derived from the DO-178B statement. Note that these questions are expanded in the worksheets found in appendix A using best practices taken from other domains that employ service history.

Column 4, question category, places the DO-178B guidance statement in one or more of the four categories used throughout this handbook to discuss the various aspects of software service history. These categories are further explored in section 3.

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
a. The applicant should show that the software and associated evidence used to comply with system safety objectives have been under configuration management throughout the product service history.	If this evidence for safety objectives is missing or noncompliant, there is no work-around. The service history data that is associated with software that is not configuration controlled should not be used as evidence since there is no confidence in the data or software. There is no basis for computing error rates.	1. Are the software versions tracked during the service history duration? 2. Are problem reports tracked with respect to particular versions of software? 3. Are problem reports associated with the solutions/patches and an analysis of change impact? 4. Is revision/change history maintained for different versions of the software? 5. Have change impact analyses been performed for changes?	Question of Problem Reporting
b. The applicant should show that the problem reporting during the product service history provides assurance that representative data is available and that in-service problems were reported and recorded and are retrievable.	If this evidence is unsatisfactory, the significance of available service history data is reduced. There are a number of issues regarding the integrity of the problem reporting system: the vendors' self interest in not publishing all of the problem data, not all of the data may be reported, and not all of the data may be recorded. In case there are doubts as to whether "representative" data is available, the available data may be used for finding out if there were safety critical problems during the service history duration without taking credit for certification. Although this item is describing the elements of configuration control, the guidance does not directly state that the problem reports need to be under configuration control.	1. Were in-service problems reported? 2. Were all reported problems recorded? 3. Were these problem reports stored in a repository from which they can be retrieved? 4. Were in-service problems thoroughly analyzed and are those analyses included or appropriately referenced in the problem reports?	Question of Problem Reporting

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
c. Configuration changes during the product service history should be identified and the effect analyzed to confirm the stability and maturity of the software. Uncontrolled changes to the Executable Object Code during the product service history may invalidate the use of product service history.	This point also deals with the integrity of the problem data, in terms of whether the software was under configuration control. Without configuration control, the problem reports cannot be associated with specific versions of the software. Uncontrolled changes to software also indicate deficiencies in the development process.	1. Is each problem report tracked with its status of whether it is fixed or open? 2. If a problem was fixed, is there a record of how the problem was fixed? 3. Is there a record of a new version of software with the new release after the problem was fixed? 4. Are there problems with no corresponding record of change in software version? 5. Does the change history show that the software is currently stable and mature?	Question of Problem Reporting
d. The intended software usage should be analyzed to show the relevance of the product service history.	If there are fundamental differences in the usage, the product cannot be used in the proposed environment and there will not be any need to use the service history. In some cases, additional code, in the form of a "wrapper" or "glue code," may be needed to integrate the existing software into the new operating environment. In this case, the effect of this additional code on the relevance of the service history should also be considered.	1. Is the intended software operation similar to the usage during the service history? (Its interface with the external "world," people, and procedures) 2. Have the differences between service usage and proposed usage been analyzed?	Questions of Operation and Environment

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
e. If the operating environments of the existing and proposed applications differ, additional software verification should confirm compliance with the system safety objectives.	There is no guidance on what elements of the two operating environments need to be compared. There exists a dilemma in the case of multiple copies of the product being used in many diverse operating environments, service history in a dissimilar environment may actually give a larger coverage of input/output space and thus resulting in problem reports, which may not have been written up if the product was executed only in similar environment. This circumstance has a larger probability of weeding out latent errors although the service history in a dissimilar environment may not be used for error rate calculations.	1. If the input/output domains differ between service history duration and intended use, has there been an analysis of what functions are covered by the service history? 2. Is the hardware environment of service history and the target environment similar? 3. Is the product compatible with the target computer without making modifications to the product software? 4. If the computer environments are different, are the differences verified (through analysis and/or testing)? 5. Are there differences in the operating modes in the new usage?	Question of Operation and Environment
f. The analysis of configuration changes and product service history environment may require the use of software requirements and design data to confirm the applicability of the product service history environment.	This item is a companion to items c and e. The text in DO-178B does not claim that software requirements and design data will be needed for this analysis nor does it claim that these are the only sources of needed information. In fact, the required information may be present from looking at the problem reports, version dates, patch descriptions, and perhaps details of how the problem was fixed which might be present in the problem report logs/database. If this additional information is insufficient, then the requirements and design data may need to be reviewed.	1. Is the data needed to analyze similarity of environment available? Such data are not usually a part of problem data). 2. If not, have the software requirements and design data been reviewed to support the service history claim?	Questions of operation, environment, and problem reporting

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
g. If the software is a subset of the software that was active during the service period, then analysis should confirm the equivalency of the new environment with the previous environment and determine those software components that were not executed during normal operation. Note: Additional verification may be needed to confirm compliance with the system safety objectives for those components.	This item does not discuss what happens to functions that are not going to be used in the proposed application. Assurance requirements for dead code or deactivated code in DO-178B may apply, depending upon the technique used to prevent the usage of these functions.	1. Are only some of the functions of the proposed application used in service usage? 2. Is there a gap analysis of functions that are needed in the proposed application but have not been used in the service duration?	Questions of Operation and Environment
h. The problem report history should be analyzed to determine how safety-related problems occurred and which problems were corrected.	If safety-related problems are still pending, the analysis must show that the problem is not an issue for the intended application. This is, generally, a difficult issue since, if the operational environment is similar, the safety critical problem is of significance; if the operational environment is not similar, then the service history is not of consequence.	1. Are all problems within the problem report repository classified? 2. Are safety-related problems identified as such? 3. Can safety-related problems be retrieved? 4. Is there a record of which problems are fixed and which problems are left open? 5. Is there enough data after the last fix of safety-related problems to assess that the problem is solved and that no new safety-related problems have surfaced? 6. Do open problem reports have any safety impact?	Question of Problem Reporting

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
i. Those problems that are indicative of an inadequate process, such as design or code errors, should be indicated separately from those whose cause are outside the scope of this document such as hardware or system requirements errors.	The level of detail that is required here for logging problem reports with their solutions and the source of errors may itself indicate good engineering practice. A lack of sufficient information may be indicative of an inadequate process.	1. Are the problem reports and their solutions classified to indicate how a fix was implemented? 2. Is it possible to separate those problems that caused a design or code correction? 3. Is it possible to separate the problem reports that were fixed in the hardware or change of requirements?	Questions of Problem Reporting and Environment
The data described above and these items should be specified in the Plan for Software Aspects of Certification: j. (1) Analysis of the relevance of the product service history environment	Analysis of the relevance of the product service history environment is discussed in items d, e, f, g, and i.	All questions on similarity of usage and operational environment associated with these items apply.	Questions of Operation and Environment

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
The data described above and these items should be specified in the Plan for Software Aspects of Certification: j. (2) Length of service period and rationale for calculating the number of hours of service including factors such as operational modes, the number of independently operating copies in the installation and in service, and the definition of "normal operation" and "normal operation time"	Item j (2) has a requirement for calculating the length of service history duration by taking into consideration the number of hours of service, similar operation, and the number of independently operating copies. However, the independence here is not fully applied. It is inferred that the independence in operation should result in statistically independent error rates. The definition of "normal operation" is also left up to the applicant. This definition should be consistent with the proposed use of the software.	1. What is the definition of service period? 2. Is the service period defined appropriate to the nature of software in question? 3. How many copies of the software are in use and being tracked for problems? 4. Are the input/output domains the same? 5. What was the criterion for evaluating service period duration? 6. What is the definition of normal operation time? 7. Does service period include normal and abnormal operating conditions? 8. Is the definition of "normal operation" and "normal operation time" appropriate to the product?	Question of Time, Environment, and Operation
The data described above and these items should be specified in the Plan for Software Aspects of Certification: j. (3) Definition of what was counted as an error and rationale for that definition.	Item j (3) is not referenced directly in any of the other items; there is a reference to classification of problems in item h where the classification does not go beyond whether the problem was safety related. Definition of errors here refers to how errors were defined in the service history data. It is important that this definition have an acceptable rationale, and also, that the definition should have been applied to all of the problems consistently throughout the service history duration.	1. Was there a procedure used to log the problem reports as errors? 2. What was the reasoning behind the contents of the procedure? 3. Is there evidence that use of this procedure was enforced and used consistently throughout the service history period?	Question of Problem Reporting

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
The data described above and these items should be specified in the Plan for Software Aspects of Certification: j. (4) Proposed acceptable error rates and rationale for the product service history period in relation to the system safety and proposed error rates.	Acceptable methods for computing error rates are not stated in DO-178B. The requirement that this has to be forecast in relationship with the system safety is in direct conflict with the idea that software reliability measures cannot be used to justify safety, nor can these numbers be used in system safety analysis. Duration that is used in the equation for computing error rates can be varied. If data exists from many application environments, but only a few of these environments are similar to the target environment, not all available errors or the duration can be used in the error rate computation. In that case, if there were noted safety problems in the applications that are not considered in the error rate, should we be concerned? Common sense dictates that we should be. However, if we do not use this data in the error rate, how can we consider these problems?	1. Do you have a proposed error rate, justifiable and appropriate for the level of safety of proposed usage (before analyzing the service history data)? 2. How do you propose that this error rate be calculated (before analyzing the service history data)? 3. Is the error rate computation appropriate to the application in question? Note: (Error rates may be computed in a number of different ways such as total errors divided: • by time duration, • by number of execution cycles, • by number of events such as landing, • by flight hours, • by flight distance, • by total population operating time, and • by number of hours during which the product is expected to perform; Other definitions for error rates may be used, depending upon the particular usage.)	Questions of Time, Operation, Environment, and Problem Reporting

TABLE 1. ANALYSIS OF DO-178B, SECTION 12.3.5 (Continued)

DO-178B Section 12.3.5 Reference	Observations on DO-178B Section 12.3.5	Software Service History Questions	Question Category
k. If the error rate is greater than that identified in the plan, these errors should be analyzed and the analyses reviewed with the certification authority.	Same issues as in item j (4) above exist for this item. Unless a method for using error rates in system safety assessment is established, this item cannot be resolved objectively. Also, there is no mention here of whether the errors were safety related or not. High error rates of nonsafety problems cannot have the same weight as the low error rates of safety problems. Although, item h deals with an analysis of safety problems and the problems that may not have been fixed, the analysis is not required to be used in the review of error rates. Error rates during "normal operation" within the service history duration are usually large soon after deployment of major corrections and releases. It is not clear whether the referenced error rates are over the span of the entire period of service history or a selected section such as in the last year of usage. Although impact of modifications is cited as one of the considerations in the acceptability of service history for certification credit (in the lead in paragraph of section 12.3.5), it is not referenced in the determination of error rates. For example, error rates can be computed for every version of the software product or it can be computed for all versions combined. This relates to the problem of what are the rules for computing the time duration i.e., the rules for resetting the clock. This also brings up the point of whether the applicant can choose to use only the last 2 years of contiguous service history during which the software may be more stable compared to the very first fielding of the product. The guidance is silent on how long is long enough for a particular level, whether the length of service history is important, definition of error rates, definition of stability, and how to use error rates in system safety assessment to arrive at what rates are acceptable for a given criticality level.	1. What is the actual error rate computed using the service history? 2. Is this error rate greater than the proposed acceptable error rate defined in PSAC according to J(4)? 3. If the error rate is greater, was analysis conducted to reassess the error rates?	Question of Problem Reporting

2.3 RELATIONSHIP WITH PREVIOUSLY DEVELOPED SOFTWARE.

DO-178B uses the term previously developed software (PDS) to describe software that falls in one of three categories:

- Commercial Off-The-Shelf (COTS) software,
- Software developed to a standard other than DO-178B, and
- Software developed prior to DO-178B.

By this definition, it is hard to imagine an instance when a product service history argument will be made on software other than PDS. DO-178B provides specific guidance for PDS that should be used in conjunction with the contents of this handbook when seeking certification credit. Combining alternative methods to meeting one or more objectives is best accomplished by conducting a gap analysis designed to determine where data may be insufficient to clearly demonstrate compliance with the objective. Such an approach is described in DO-248B, discussion paper no. 5.

2.4 PRODUCT SERVICE HISTORY VERSUS SOFTWARE RELIABILITY.

The DO-178B definition of product service history is very similar to the IEEE definition of reliability, which is "the ability of a product to perform a required function under stated conditions for a stated period of time". It should also be noted that DO-178B includes the following paragraphs regarding software reliability:

> Section 2.2.3: *"Development of software to a software level does not imply the assignment of a failure rate for the software. Thus, software levels or software reliability rates based on software levels cannot be used by the system safety assessment process as can hardware failure rates."*
>
> Section 12.3.4*: "During the preparation of this document [DO-178B], methods for estimating the post-verification probabilities of software errors were examined. The goal was to develop numerical requirements for such probabilities for software in computer-based airborne systems or equipment. The conclusion reached, however, was that currently available methods do not provide results in which confidence can be placed to the level required for this purpose. Hence, this document does not provide guidance for software error rates. If the applicant proposes to use software reliability models for certification credit, rationale for the model should be included in the Plan for Software Aspects of Certification, and agreed with by the certification authority."*

The effect of these two statements has been a virtual moratorium on the application or even exploration of software reliability as an alternative method for satisfying DO-178B.

This creates an inherent difficulty for the product service history approach as well, since service history arguments are largely predicated on the residual error rates or the probability of latent software errors remaining after verification. The authors of DO-178B side-stepped this issue by

allowing certification credit for service history based on qualitative assessments of the sufficiency and relevancy of the product service history.

3. THE ELEMENTS OF PRODUCT SERVICE HISTORY.

As noted in the previous section, the topic of product service history may be examined by looking at the various elements that comprise its definition. For the purposes of this handbook, four components were defined: problem reporting, operations, environment, and time. Considering each of these components separately results in different but interrelated sets of questions that must be asked when the use of product service history is being considered. The questions have been broken into these classes only to simplify the problem. Answers to these questions must satisfy both the relevancy and sufficiency criteria discussed in Section 12.3.5 of DO-178B.

This section provides a discussion of each set of questions arising from the product service history definition. One representation of these questions is provided in the form of worksheets (see appendix A). While these worksheets may be adapted or tailored to fit a particular project, users are cautioned to maintain an objective view when evaluating service history data. As illustrated in the sections below, even subtle changes in any one of the four areas can lead to unpredictable results when software is used in a new system or in a way not originally envisioned.

3.1 QUESTIONS OF PROBLEM REPORTING.

Questions of problem reporting are primarily the same as the ones faced in configuration control and management. All of the elements of DO-178B Section 11.4 apply. The problems have to be uniquely identified, they should be traceable to the version of software/product, a method of closing the problems must be defined, and closure of the problems must be accomplished with proper change control activity. Changes must be reviewed for priority of problems and change impact. Data on problems, corrections, and baselines must be kept under control to assure the integrity of the data.

All of these activities are a natural part of a well-defined process. However, in the case of previously developed software, it is assumed that these activities are not visible to the user of the product. The vendor who is in charge of problem collection may not have a robust process. The vendor may not have a robust policy for classifying and prioritizing the problems. Multiple users, employing the software in ways to which the vendor has no visibility, further exacerbate this issue.

When patches are installed, some users may install the patch, whereas many others may not. This means that some users are using the uncorrected version and some are using the corrected version. This results in service history that cannot be treated as a monolithic block; rather it must be distributed across the different versions. Only those versions with a clear similarity to the intended use may be used to arrive at the total product service history. There are numerous reasons affecting problem report classification and accuracy including:

- Not all users may be using the software product per the user's manual.

- Vendors may not have a complete, consistent, or accurate way of identifying those problems attributable to software.

- Not all users may be reporting the problems they encounter.

- Users may find work-around procedures and thus stop reporting all occurrences.

- Vendors may not require subsequent occurrences of a problem to be reported.

- Vendors may treat problems found internally differently than those found by their customers, thus underreporting the total number of problems experienced.

Problems may also be introduced while fixing other problems. Such problems should also be logged once the product is fielded. If some of the problems are unique to a particular small sector of users, the vendor may not fix the problem or may selectively provide a patch. Attention must be paid to the number and type of open problems. A vendor's policy for choosing which errors are to be fixed should also be noted in the qualitative assessment. A vendor may place priority on non-safety-critical problems reported by a large sector of users over safety-critical problems reported by a small sector of users.

Assignment of version numbers and tracking the operating versions of the product to be traced to the problems is a difficult task. If vendors provide patches for their software or frequently introduce revisions to the field, this must be taken into account in arriving at the total number of versions for which service history is valid and for which the total service periods can be combined.

Visibility into how problems were fixed may be of use when the solutions affect the usage of the product in a safety-critical application (whether requirements were compromised, new assumptions were made, new requirements were added, new design features were added, change impact analysis was conducted, list of affected requirements/assumptions are provided to the user, or any effect on hardware is noted, etc.).

Some vendors may be following certain regulations or policy regarding configuration control of the problem reporting process. Such policies may help in determining if service history data is clean. Some problems may also be corrected in periodic upgrades of the product. It is important to understand the vendor's policy for dissemination of patches, warnings, work-arounds, and upgrades. Spikes in error rates after a new version is disseminated need to be traced to assess the complexity of changes, the quality of change impact analysis, the quality of the vendor's verification process, and the diversity of the product usage.

The key questions to be addressed in the area of problem reporting and configuration management for the purpose of establishing the minimum objective criteria for using service history data include a good consistent definition of problems, classification of problems, tracking with respect to software versions, and tracking with respect to solutions.

Appendix A, table A-2 provides a set of questions in the form of a worksheet that may be used to evaluate the relevance and sufficiency of problem report/configuration management using the data available from the product service history.

The following considerations, based on Section 12.3.5 of DO-178B, were used in formulating the questions in appendix A, table A-2:

- Data available on problems
- Data derivable from the problem reports
- Analysis to be performed
- Indications of supplemental verification

3.2 QUESTIONS OF OPERATION.

The concept of operation is to define the usage characteristics of the software within the previous domain as compared with the target domain. These characteristics include people and procedures and the modes in which the service history was documented against the same items within the target domain. Figure 1 illustrates the type of comparisons that are needed.

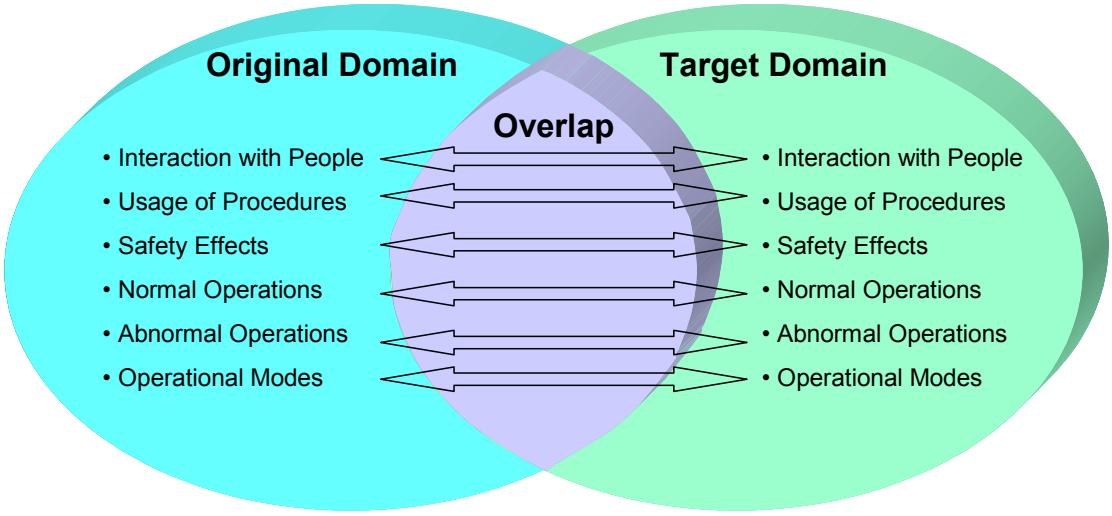


FIGURE 1. OPERATION

There are many concerns in evaluating the similarity of operations that may not be so obvious. Where people and procedures are concerned, the training and qualifications of the people in the service history domain have to be noted so that the proposed domain of usage can account for this by requiring similar training and qualification requirements.

Similarity in operational modes and the subset of software functions between the service history domain and the target domain are the main focus in this area. In general, it is expected that the functions to be employed in the target domain are a subset of those from the service history domain. Input and output domains may be evaluated in normal and abnormal operations to assess the completeness of coverage of all of the functions that are to be reused in the target

domain. This is the most fundamental basis for getting credit for service history by assessing that the software has a tried and tested history of pertinent functions.

Consider the case of ARIANE 5.[2] The self-destruction of the launcher was caused by the failure to mitigate the environmental differences between ARIANE 5 and ARIANE 4. Software reused in ARIANE 5 from ARIANE 4 included a function for the alignment of the strap-down inertial platform to be operative for 50 seconds. This requirement was based on a particular operational launch sequence that is no longer used. The software exception generated from this portion of the code caused a chain of events that eventually led to; the backup processor shutting off, errors in the primary processor causing an angle of attack of more than 20 degrees, separation of booster on the main stage, and self-destruction of the launcher. The reused function should have been operative only before liftoff or there should have been a thorough analysis of abnormal operating modes, differences in flight operations, and nominal range and value of parameters. There should have been a discussion of software exceptions and differences in actions taken to resolve these exceptions. Questions of operation and environment (discussed next) are highly interrelated. In this example, a study of target operations could have found the fault just as easily as a study of the target environment.

The total availability of service history data may be much longer than what is considered similar operation. For example, there may be a total of 10,000 copies of a particular software in use in the public domain, out of which only 10 copies may be in use in domains similar to the proposed usage. This would have a direct bearing on the ability to calculate error rates. This is discussed in more detail in the Section 3.4, Questions of Time, of this handbook.

Modifications to the product during the service history interval need to be studied to understand if these modifications were made for correcting errors in a dissimilar domain for which service history credit is not being sought. The question here would be to note if a change impact analysis has been performed to assure that the functions that are of consequence in the service history data have not been adversely affected. This is quite possible if the changes have affected either the assumptions or requirements in this area.

If the service history collection occurred when the software was being used at a lower criticality than the intended usage in the target domain, caution should be exercised in taking credit. The types and severity of errors, as well as open problem reports, must be examined to assure that the service history gives the proper level of assurance.

It must be noted that the service history duration should ideally include both normal and abnormal operations to cover features such as redundancy, backup, other fault tolerance techniques, and corner conditions. An analysis should be conducted to find which features were not exercised in the service history, so that supplemental verification can be performed.

If the product was used with different data/parameters (for example adaptation data, external data sets, internal parameters) in the service environment, these differences should be examined for possible risks in the target environment.

[2] "ARIANE 5 Flight 501 Failure," Reported by the Inquiry Board, the Chairman of the Board, Professor J. L. Lions, Paris, 19 July 1996.

The key question to be addressed in the area of operation for the purpose of establishing the minimum objective criteria for using service history data include an analysis for establishing similarity of operation between the service history and the proposed application. Service history data that reflect dissimilar operations cannot be used for computing service history duration.

Appendix A, table A-3 provides a set of questions in the form of a worksheet that may be used to evaluate the similarity of operation using the data available from the product service history.

The following considerations, based on Section 12.3.5 of DO-178B, were used in formulating the questions in appendix A, table A-3:

- Data pertinent to operation
- Derivable data associated with operations
- Analysis to be performed
- Indications of supplemental verification

3.3 QUESTIONS OF ENVIRONMENT.

Questions of environment were broken away from the questions of operation in order to distinguish the immediate computer environment in which the service history data was collected. This particular set of questions are designed to address and mitigate software errors, which have their origin in hardware errors, interface errors, or resource assumptions. It should be noted that the exception handling and fault tolerance of the product, whose service history is being tracked, should be separated from the larger system in which it is embedded so that assurance is gained on the robustness of the product. This knowledge allows for an appropriate reuse of the product in the new environment. Figure 2 illustrates the items that should be considered in this area.

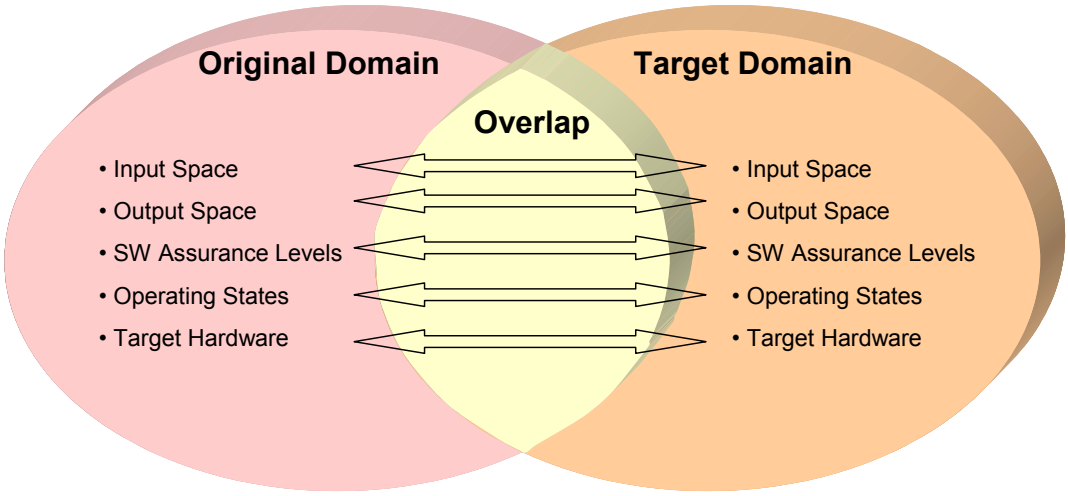


FIGURE 2. ENVIRONMENT

Similarity of environment may be assessed using the history of modifications to the product due to the particular hardware platform or because of resource requirements in the service environment or the similar types of modifications made to the product in the target environment.

Consider the example of the Patriot systems "failure" to intercept the El Hussein (Scud) missile in Deharan. Operational specifications for the system matched with the way the system behaved. However, the "problems" in the software were bugs only AFTER the operational environment had been redefined. The weapon was used, not in the detection and interception of aircraft, but rather in the detection and interception of land-launched missiles. In its new capacity, the software "failed" because (1) there were missiles in a speed range that could and should be attacked and (2) the Patriot system's primary mission would NOT be defending against hostile aircraft over a relatively short attack time, but rather, defending against a potential land-launched missile threat over many days. System performance degradation due to uncompensated clock drift crippled the weapon's defensive capability after the system had been continuously powered for days rather than the hours it was designed for[3]. Unlike the ARIANE case, it would have been difficult to detect the "problems" in this case since the system's failure was ultimately tied to the overall environment definition.

Service history credit should be counted strictly when the types of installations match the target environment; i.e., same or similar hardware platforms. Product literature may be reviewed to compare computer environments in terms of limitations and constraints such as resource usage.

If the problem reports identify problems because of usage in a particular computer environment differ from the target environment and the changes were made to fix these problems, the effect of these changes in the target environment should be considered.

If the product was used with different data/parameters (for example adaptation data, external data sets, internal parameters) in the service environment, these differences should be examined for possible risks in the target environment.

The key questions to be addressed in the area of environment include assessing the computing environment to assure that the environment in which the software was hosted during service history is similar to the proposed environment. This analysis must include not just object code compatibility, but time and memory utilization, accuracy, precision, communication services, built-in tests, fault tolerance, channels, ports, queuing models, priorities, error recovery actions, etc.

Appendix A, table A-4 provides a set of questions in the form of a worksheet that may be used to evaluate the similarity of environment using the data available from the product service history.

[3] Patriot Missile Defense Software Problems led to Systems failure at Dhahran, Saudi Arabia, GAO Report February, 1992, B-247094.

The following considerations, based on Section 12.3.5 of DO-178B, were used in formulating the questions in appendix A, table A-4:

- Data pertinent to the computer environment
- Derivable data associated with the computer environment
- Analysis to be performed
- Indications of supplemental verification

3.4 QUESTIONS OF TIME.

There are many different ways of measuring the service history duration; duration may be measured without considering the changes made to the software or the clock may be restarted at the time corrections are made to safety problems. This question is related to how problems are classified and the vendor's priority system for correcting the problems. Figure 3 illustrates one common approach to measuring service history duration.

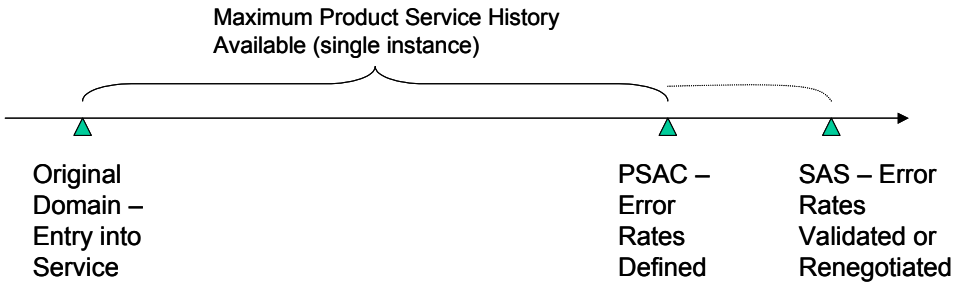


FIGURE 3. TIMELINE

The question of defining time relative to certification or continuing airworthiness has its parallels in other areas of the FAA. For example, following the Aloha Airlines incident in 1988, the National Transportation Safety Board noted, as part of their findings, that there appeared to be confusion in the terms flight cycle versus flight hour. The FAA released a Flight Standards Handbook Bulletin (HBAW 94-05B) to address this confusion as it related to aircraft maintenance.

The premise for using service history is based on the assumption that service history data gives evidence that all of the required functions have been repeatedly exercised and is correct. Strictly speaking, this assumption has no bearing on time at all. Time comes into the picture only because there is comfort in a statistical sense that the probability of exercising all of the needed functions is greater as more time passes.

Time is further modified within the definition by the need for its measurement to take place over a 'contiguous' period. This qualification is designed to eliminate the potential for periods of improper execution or dormancy to be suppressed, thus skewing any conclusions drawn about the software under consideration.

A close review of the DO-178B product service history guidance produces additional terms that directly relate to time, most notably "hours in-service" and "normal operation time." These sound suspiciously like terms used to arrive at reliability numbers for hardware. An applicant wishing to use product service history is asked to describe in their Plan for Software Aspects of Certification (PSAC) the rationale for choosing a particular number of hours in-service including how normal operation time is defined. Consider a software function that is only used during landing. It hardly seems reasonable to define in-service hours as flight time when the landing phase during which the software is being exercised accounts for only a small fraction of this overall time.

While DO-178B is silent on whether the contiguous time period varies with software level, all of the discussions within SC-190, SC-180, and CAST have tended to accept this as an axiom. Likewise, the assumption is always made that what is being discussed, in some way, is measurable using hours, minutes, days, etc. It is generally felt that attempting to categorize software execution in terms of clock cycles, frames, or states is generally something for which sufficient data would be impossible to directly measure and would ultimately rely on inference from a clock-based measurement.

DO-178B in Section 12.3.5 j. (2) and k refer to computation of error rates. DO-178B does not provide specific guidance as to how this computation should be performed or what specific data is to be used. This provides the applicant with a fair amount of flexibility in the application of the service history argument. Error rates could be defined as number of errors divided by the time duration. In some cases, time duration is not as relevant as using number of events such as takeoffs or landing, flight hours, flight distance, total population operating time, or only the number of times an operator queried the software for specific information. For use in this computation, the duration should be analyzed to be relevant. DO-178B does not specify whether all errors are considered to be of the same weight in these computations. It seems logical that even a few safety errors should be of higher consequence than a large number of nonsafety errors. Although there is a discussion of safety problems in Section 12.3.5 h, there is no indication of how these problems are used in error rate computations.

Note: Grounds for restarting the clock within the service history duration is not discussed in DO-178B. When a major software or hardware change occurs a key question must be answered. The key question to answer is whether service history duration should be measured before or after the implementation of the changes. The measurement of the error rates for the updated software or hardware is dependent upon the answer to this question. In a number of software reliability models, time is reset when major changes are made since the software that was tracked is no longer the software that is used. There are other models that compensate for changes in software. This gap is tied to whether software reliability models can be used, and if so, how do you assess that the correct assumptions are made in the use of a particular model in a particular circumstance. This is illustrated in figure 4.

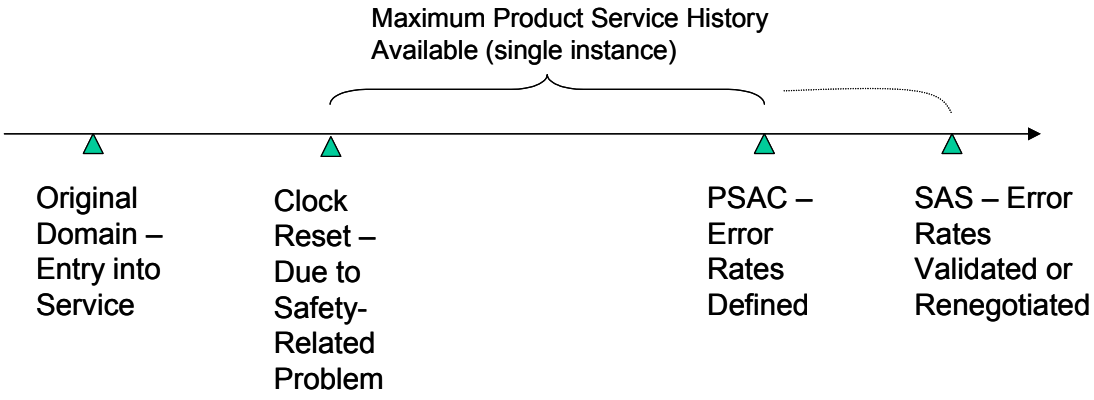


FIGURE 4. CALCULATION OF SERVICE HISTORY DURATION

The key questions to be addressed in the area of time for the purpose of establishing the minimum objective criteria for using service history data include units of measurement in the service history duration definition as appropriate to the usage of the software in question, reliability and consistency of measurement of this time, and justification for duration used in the calculation of error rates.

Appendix A, table A-5 provides a set of questions in the form of a worksheet that may be used to evaluate service history time duration and error rates using the data available in the product service history.

The following considerations, based on Section 12.3.5 of DO-178B, were used in formulating the questions in appendix A, table A-5:

- Pertinent data related to time
- Derivable data regarding time
- Error rate considerations
- Analysis to be performed
- Indications of supplemental verification

4. ADEQUACY OF DEVELOPMENT PROCESS.

DO-178B gives guidance for both process and product assurance. Product service history does not provide any direct objective evidence of the process used in creating the software. Applicants wishing to make use of product service history must determine a way of demonstrating compliance with the objectives of DO-178B. This generally involves complementing product service history with additional alternate methods.

Numerous attempts have been made to equate specific objectives for which product service history could be "traded with." Such attempts within SC-190 and CAST actually involved the creation of tables listing the objectives of DO-178B and stating for each objective whether

service history data that could fully or partially satisfy the objective. These attempts were reviewed as part of this research in hopes that something had been overlooked that prevented their acceptance by the broader community. The inherit problem is the unquantifiable nature of the processes used to create and assure software. DO-178B is based on the premise that a good development process yields a better product; one that is more likely to perform its intended function and less likely to perform unintended functions.

The 66 objectives of DO-178B are divided across six main areas: planning, development, verification, quality assurance, configuration management, and certification liaison. The definition of product service history really only addresses two of these. The first is fairly direct, namely, problem reporting and configuration management of the software the data describes. The second is verification of the code to some degree by virtue of its execution. In fact, a cogent argument can be made that service history represents a variety of testing techniques, including:

- Stress testing
- Random testing
- Scenario-based testing
- Regression testing
- Accelerated life testing
- Exhaustive testing
- Domain testing
- Error guessing

All of these techniques may be used to accomplish one or more of the verification objectives outlined in DO-178B. These techniques frequently are applied to the elements of blackbox testing in controlled environments, either laboratory or airplane, in typical DO-178B projects. The good news is that about 60% of the objectives in DO-178B are verification objectives. The bad news is that there would not seem to be any corollary to product service history for planning, development, quality assurance, and certification liaison during the original development of the software that the service history data describes.

With this in mind, it would seem most appropriate to focus specific attention on things that may be done to gain confidence in these other areas. If any development records are still available from the original vendor, these may be reviewed to gain confidence in the process that was followed. Such records could include the requirements documents, design data, quality assurance data, and test data that may supplement the service history data. In this last case, special attention should be paid to testing completed for error-handling routines, performance testing, and other testing focused on the robustness characteristics of the software. Remember that these are the parts of the code least likely to have been exercised as part of the service history.

Confidence in the supplier's development process may also be gained through careful analysis of the problem report data collected over the service history period. In addition to the items discussed at the beginning of section 3, trend data may be analyzed to determine how well the supplier accomplishes reverification and whether the software does, in fact, appear to be

maturing over time. This type of analysis is not directly discussed in the DO-178B but is generally accepted by the software community.

Note that each of the above approaches can be stated in the negative as well. Spikes in problem reports right after a major build or a patch may indicate that the software is not maintainable or the quality of updates is not quite high enough. Recurring or chronic problems that go unresolved may also indicate poor processes.

5. ESTABLISHMENT OF "EQUIVALENT SAFETY".

Within a month and a half of its publication, DO-178B was formally recognized via AC 20-115B as a means, but not the sole means, for securing FAA approval of software in airborne systems. For new projects started after this AC was published, most applicants have chosen to use DO-178B as the means of compliance for their airborne software. Those who have sought to use other approaches for securing FAA approval have generally been required to show how their approach met the intent behind the DO-178B objectives.

One of the most basic issues when discussing product service history is to understand what that service history is demonstrating. Since the service history data generally exists for a system, typically a line-replaceable unit on an aircraft, any claim made for the software is an extrapolation from the system's performance. Systems are required to comply with one or more CFR before being certificated for use on an aircraft. A careful reading of DO-178B along with the guidance governing certification of parts and equipment described in 14 CFR Part 21 shows that DO-178B is simply a means of satisfying the CFRs, specifically those elements describing intended function and absence of unintended function as noted earlier. The logical question that arises is whether service history can be used to demonstrate compliance directly with the CFRs.

While current guidance does not preclude such an argument, actually being able to demonstrate CFR compliance would be extremely difficult. Any attempt would need to overcome the basic issue of reliability applied to software. CFR XX.1309 uses terms such as extremely improbable to describe events that simply should not happen in the lifetime of a particular aircraft type. This has historically been translated to failure probabilities of 10^{-9} or better. There exists no commercially accepted model for software reliability that comes close to this number and that can be shown to be based on a realistic model.

A component of unknown pedigree within a system is a safety risk. Contribution to safety from the software components of the system has been under constant debate since software was first introduced in a Flight Management System in the early 1980s. For the time being, design assurance remains the only viable approach, with DO-178B serving as the most mature model for its application. There are, however, other ways of mitigating risk from an unknown software component. For example, architectural means may be employed to limit the effect of a software error leading to a system-level failure. Examples of architectural means include:

- Partitioning—preventing failures from noncritical functions affecting critical functions

- "Wrappers"—wrapper software prevents use of unneeded functionality, checks validity of parameters

- Software Performance and Safety monitors—credibility checks; redundant processors checking one another, fail-safe architectures

- External monitors—e.g., watchdog timers

Unfortunately, architectural means may not always be an option to protect against latent errors in software for which only service history data is available for certification credit. It may also be that the use of architectural means actually increases the system's complexity, thus potentially decreasing safety rather than increasing it. For higher criticality systems, service history may simply not be an appropriate or practical choice.

It is generally accepted that use of service history data becomes easier when systems are relatively simple and of lower criticality, the service history data is both abundant and of high quality, and the system's operating characteristics and external environment are close to the original systems.

6. SUMMARY.

Service history is a powerful approach, which when used correctly, can make it possible to demonstrate the maturity of software that has previously been fielded and for which good data regarding its operation is available. To accomplish this, careful attention must be paid to a number of questions concerning the application of service history. Section 3, The Elements of Product Service History, of this handbook, discussed these questions in detail. Sections 4, Adequacy of Development Process, and Section 5, Establishment of "Equivalent Safety," helped to place those questions in the context of design assurance and safety; two fundamental aspects of creating and assuring software for airborne systems and equipment. In appendix A, a detailed set of worksheets are provided to aid applicants in answering the questions relating to service history and to provide a framework for the necessary dialogue with the certification authority.

Service history, as an alternate method for demonstrating compliance to the objectives of DO-178B, is only one of many approaches that may be taken to demonstrate software maturity. When contemplating its use, one must be careful to consider the relationship between service history and software reliability. As noted in section 2 software reliability remains a controversial idea and cannot be used quantitatively in a safety assessment. Careful attention must be applied when defining error rates for a particular application and their definition should be discussed with the appropriate Aircraft Certification Office (ACO) at the onset of the certification project.

As more confidence is gained in the use of service history arguments for supporting certification efforts, the FAA may develop additional guidance. It is also expected that this subject will be revisited when DO-178B is revised in the future. In the interim, it is hoped that this report will help applicants apply service history arguments in a more thorough and consistent manner. Likewise, the use of this handbook by the FAA should allow for more consistent expectations

being placed on applicants, something that has generally been shown to help control costs associated with achieving certification.

7. BIBLIOGRAPHY.

Almost 200 references were examined for this project. The following is a subset chosen for presentation of ideas directly related to software service history, as well as representative of regulations in other domains in this regard.

"NASA Preferred Reliability Practices, PART 1 of 3: Design and Test Practices for Aerospace Systems," NASA/TM-4322, National Aeronautics and Space Administration, February 2000.

"Software Safety," NASA-STD-8719-13A, National Aeronautics and Space Administration, September 1997.

"Use of COTS/NDI in Safety-Critical Systems," Report of the Challenge 2000 Subcommittee of the FAA Research, Engineering, and Development Advisory Committee, February 1996.

C Jones, R.E. Bloomfield, P.K.D. Froome, and P.G Bishop, "Methods for Assessing the Safety Integrity of Safety-Related Software of Uncertain Pedigree (SOUP)," Adelard, contract research report 337/2001, 2001.

Carney, David, "Quotations From Chairman David: A Little Red Book of Truths to Enlighten and Guide in the Long March Toward the COTS," written for SEI Joint Program Office, Hanscom AFB, Carnegie-Mellon Software Engineering Institute, July 1998.

Coombes, A.C., "Report to Civil Aviation Authority Safety Regulation Group, Comparison of Standards for Safety-Related Software Development," YSE Reference: CF 17 1/3/53, 25 February 1999.

Hayhurst, K.J., C.A. Dorsey, J.C. Knight, N.G. Leveson, and F.G. McCormick, "Streamlining Software Aspects of Certification: Report on the SSAC Survey," NASA/TM-1999-209519, National Aeronautics and Space Administration, August 1999.

Hissam, S. and D. Carney, "Isolating Faults in Complex COTS-Based Systems," SEI Monographs on the use of Commercial Software in Government Systems, Carnegie-Mellon Software Engineering Institute, February 1998.

J.A. Scott, G.G. Preckshot, and J.M. Gallagher, "Using Commercial Off-The-Shelf (COTS) Software in High-Consequence Safety Systems," Fission Energy and Systems Safety Program, Lawrence Livermore National Laboratory sponsored by the U.S. Nuclear Regulatory Commission, UCRL-JC-122246, November 1995.

Jim Krodel, "Commercial Off-The-Shelf (COTS) Avionics Software Study," United Technologies Research Center on contract for the Federal Aviation Administration, DOT/FAA/AR-01/26, May 2001.

Judith A. Clapp and Audrey E. Taub, "A Management Guide to Software Maintenance in COTS-Based Systems," MP 98B0000069, MITRE Corporation, November 1998.

Ljerka Beus-Dukic, "Non-Functional Requirements for COTS Software Components," School of Computing and Mathematics, University of Northumbria at Newcastle.

Ministry of Defence, Defence standard 00-40 (PART 6)/Issue 1 (ARMP-6) Reliability and Maintainability, Part 6: In-Service R & M, December 1988.

Ministry of Defence, Defence standard 00-43 (Part 1)/Issue 1, Reliability and Maintainability Assurance Activity, Part 1: In-Service Reliability Demonstrations, January 1993.

Ministry of Defence, Defence standard 00-44 (Part 1)/Issue, Reliability and Maintainability, Data Collection and Classification, Part 1: Maintenance Data & Defect, Reporting in the Royal Navy, the Army and the Royal Air Force, June 1995.

Nuselbeh B., "ARIANE 5: Who Dunnit," IEEE Software, May/June 1997, pp 25-16.

P.G. Bishop, R.E. Bloomfield, and P.K.D. Froome, "Justifying the Use of Software of Uncertain Pedigree (SOUP) in Safety-Related Applications," Adelard, contract research report 336/2001, 2001.

Presentation Materials From the Safety-Critical Systems Club Seminar "COTS and SOUP: Current Thinking and Work in Progress," Newcastle University, UK, April 2001.

Prof. J.L. Lions, "ARIANE 5 Flight 501 Failure," report by the Inquiry Board, Paris, 19 July 1996.

RTCA SC 167 Papers on Service History, Previously Developed Software and Alternate Means.

RTCA SC 167, "Software Considerations in Airborne Systems and Equipment Certification," Document number RTCA/DO-178B, RTCA, Inc., December 1992.

RTCA SC 190 papers on Service History, Previously Developed Software and Alternate Means.

RTCA SC 190 Web discussions on Service History, Previously Developed Software and Alternate Means.

Shane Lunga and Meera Galoria, "Using COTS Components: Their Location, Qualification and Selection," DERA Portsdown West, UK.

Software Engineering Laboratory, National Research Council, Canada, July 1996.

Struck, W., "Guidance for Assessing the Software Aspects of Product Service History of Airborne Systems and Equipment," Discussion Paper P-17, Draft 2, Certification Authorities Software Team, Federal Aviation Administration.

United States Navy submarine electronic system acquisition program offices (Team Submarine), Strategy 2000, Commercial Off-The-Shelf (COTS) Acquisition Primer, 12 November 1996.

W.M. Gentleman, “If Software Quality Is a Perception, How Do We Measure It?,” National Research Council Canada, Institute of Information Technology, NRC No. 40149, July 1996.

Wallace, D.R. and R.D. Kuhn, *Converting System Failure Histories Into Future Win Situations,* Information Technology Laboratory, National Institute of Standards and Technology.

Wallace, D.R. and R.D. Kuhn, *Lessons From 342 Medical Device Failures,* Information Technology Laboratory, National Institute of Standards and Technology.

APPENDIX A—EVALUATION WORKSHEETS

The following worksheets are designed to provide a uniform and consistent mechanism for conducting an evaluation of product service history using the questions discussed in sections 3 through 5 of this handbook. Questions may need to be added or tailored, depending on a particular project or through discussions with the appropriate Aircraft Certification Office.

These worksheets contain the questions derived from Section 12.3.5 of DO-178B and as discussed in tables A-1 through A-4 of this handbook. Those questions without a DO-178B reference originated from other industry sectors (OIS) that make use of service history for the purposes of evaluation and approval and are indicated by OIS. Since these represent the best practices for the application of service history arguments, they have been included here for completeness.

TABLE A-1. WORKSHEET FOR QUESTIONS OF PROBLEM REPORTING

Area:	Problem Reporting/ Configuration Management	Software Being Evaluated:	
Project:		Evaluator:	
Date:			

	DO-178B Reference	Question	Response	Issues
1.	12.3.5 a and c	Are the software versions tracked during the service history duration?		
2.	12.3.5 a and c	Are problem reports tracked with respect to particular versions of software?		
3.	12.3.5 a	Are problem reports associated with the solutions/patches and an analysis of change impact?		
4.	12.3.5 a	Is revision/change history maintained for different versions of the software?		
5.	12.3.5 a	Have change impact analyses been performed for changes?		
6.	12.3.5 b	Were in-service problems reported?		
7.	12.3.5 b	Were all reported problems recorded?		
8.	12.3.5 b	Were these problem reports stored in a repository from which they can be retrieved?		
9.	12.3.5 b	Were in-service problems thoroughly analyzed, and/or those analyses included or appropriately referenced in the problem reports?		
10.	OIS	Are problems within problem report repository classified?		

TABLE A-1. WORKSHEET FOR QUESTIONS OF PROBLEM REPORTING (Continued)

	DO-178B Reference	Question	Response	Issues
11.	OIS	If the same type of problem was reported for multiple times, were there multiple entries or a single entry for a specific problem?		
12.	OIS	If problems were found in the lab in executing copies of operational versions of software during the service history period, were these problems included in the problem reporting system?		
13.	12.3.5 c	Is each problem report tracked with the status of whether it is fixed or open?		
14.	12.3.5 c	If the problem was fixed, is there a record of how the problem was fixed (in requirements, design, code) ?		
15.	12.3.5 c	Is there a record of the new version of software with a new release after the problem was fixed?		
16.	12.3.5 c	Are there problems with no corresponding record of change in software version?		
17.	12.3.5 c	Does the change history show that the software is currently stable and mature?		
18.	OIS	Does the product have the property of exhibiting the error with message to the user? (Some products may not have error-trapping facilities, so they may just continue executing with wrong results with no indication of failure.)		
19.	OIS	Has the vendor (or the problem report collecting agency) made it clear to all users that problems are being collected and corrected?		
20.	12.3.5 h	Are all problems within a problem report repository classified?		
21.	12.3.5 h	Are safety-related problems identified as such? Can safety-related problems be retrieved?		
22.	12.3.5 h	Is there a record of which safety problems are fixed and which problems remain open?		

TABLE A-1. WORKSHEET FOR QUESTIONS OF PROBLEM REPORTING (Continued)

	DO-178B Reference	Question	Response	Issues
23.	12.3.5 h	Is there enough data after the last fix of safety-related problems to assess that the problem has been corrected and that no new safety-related problems have surfaced?		
24.	12.3.5 h	Do open problem reports have any safety impact?		
25.	OIS	Is there enough data after the last fix of safety-related problems to assess that the problem is solved and that no new safety-related problems have surfaced?		
26.	12.3.5 i	Are the problem reports and their solutions classified to indicate how a fix was implemented?		
27.	12.3.5 i	Is it possible to trace particular patches to specific release versions and infer from design and code fixes that the new versions correspond to these fixes?		
28.	12.3.5 i	Is it possible to separate the problem reports that were fixed in the hardware or change of requirements?		
29.	OIS	Are problem reports associated with the solutions/patches and an analysis of change?		
30.	OIS	If the solutions indicated a change in the hardware or mode of usage or requirements, is there an analysis of whether these changes invalidate the service history data before that change?		
31.	OIS	Is there a fix to a problem with changes to software but with no record of change in the software version?		
32.	12.3.5 j(2)	Is the service period defined appropriate to the nature of the software in question?		
33.	12.3.5 j(2)	How many copies of the software are in use and being tracked for problems?		
34.	OIS	How many of these applications can be considered to be similar in operation and environment?		
35.	12.3.5 j(2)	Are the input/output domains the same between the service duration and the proposed usage?		

TABLE A-1. WORKSHEET FOR QUESTIONS OF PROBLEM REPORTING (Continued)

	DO-178B Reference	Question	Response	Issues
36.	OIS	If the input/output domains are different, can they be amended using glue code?		
37.	12.3.5 j(2)	Does the service period include normal and abnormal operating conditions?		
38.	OIS	Is there a record of the total number of service calls received during the period?		
39.	OIS	Were warnings and service interruptions a part of this problem-reporting system?		
40.	OIS	Were warnings analyzed to assure that they were or were not problems?		
41.	12.3.5 j(3)	Was there a procedure used to log the problem reports as errors?		
42.	12.3.5 j(3)	What was the reasoning behind the contents of the procedure?		
43.	OIS	Is there evidence that this procedure was enforced and used consistently throughout the service history period?		
44.	OIS	Does the history of warranty claims made on the product match with the kind of problems seen in the service history?		
45.	OIS	Have problem reports identified as a nonsafety problem in the original domain been reviewed to determine if they are safety-related in the target domain?		

TABLE A-2. WORKSHEET FOR QUESTIONS OF OPERATION

Area:	Operation	Software Being Evaluated:	
Project:		Evaluator:	
Date:			

	DO-178B Reference	Question	Response	Issues
1.	12.3.5 d	Is the intended software operation similar to the usage during the service history (its interface with the external "world," people, and procedures)?		
2.	12.3.5 e	Have the differences between service history usage and proposed usage been analyzed?		
3.	12.3.5 e	Are there differences in the operating modes in the new usage?		
4.	12.3.5 g	Are only some of the functions of the proposed application used in service usage?		
5.	12.3.5 j(1),g	Is there a gap analysis of functions that are needed in the proposed application but have not been used in the service duration?		
6.	12.3.5 j(2)	Is the definition of "normal operation" and "normal operation time" appropriate to the product?		
7.	OIS	Does service period include normal and abnormal operating conditions?		
8.	OIS	Is there a technology difference in the usage of product from service history duration (manual vs automatic, user intercept of errors, used within a network vs standalone, etc.)?		
9.	OIS	Was operator training on procedures required in the use of product during the recorded service history time period?		
10.	OIS	Is there a plan to provide the similar training in the new operation?		
11.	OIS	Will the software level for the new system be the same as it was in the old system?		

TABLE A-3. WORKSHEET FOR QUESTIONS OF ENVIRONMENT

Area:	Environment	Software Being Evaluated:	
Project:		Evaluator:	
Date:			

	DO-178B Reference	Question	Response	Issues
1.	12.3.5 e	Is the hardware environment of service history and the target environment similar?		
2.	OIS	Have the resource differences between the two computers been analyzed (time, memory, accuracy, precision, communication services, built-in tests, fault tolerance, channels and ports, queuing modes, priorities, error recovery actions, etc.)?		
3.	OIS	Are safety requirements encountered by the product the same in both environments?		
4.	OIS	Are exceptions encountered by the product the same in both environments?		
5.	12.3.5 f	Is the data needed to analyze the similarity of the environments available? (Such data are not usually a part of problem data.)		
6.	OIS	Does the analysis show which portions of the service history data are applicable to the proposed use?		
7.	OIS	How much service history credit can be assigned to the product, as opposed to the fault tolerant properties of the computer environment in the service history duration?		
8.	OIS	Is the product compatible with the target computer without making modifications to the product software?		
9.	12.3.5 e and j(2)	If the hardware environments are different, have the differences been analyzed?		
10.	OIS	Were there hardware modifications during the service history period?		
11.	OIS	If there were, is it still appropriate to consider the service history duration before the modifications?		
12.	OIS	Are software requirements and design data needed to analyze whether the configuration control of any hardware changes noted in the service history are acceptable?		

TABLE A-4. WORKSHEET FOR QUESTIONS OF TIME

Area:	Time	Software Being Evaluated:	
Project:		Evaluator:	
Date:			

	DO-178B Reference	Question	Response	Issues
1.	12.3.5 j(2)	What is the definition of service period?		
2.	12.3.5 j(2)	Is the service period defined appropriate to the nature of software in question?		
3.	12.3.5 j(2)	What is the definition of normal operation time?		
4.	12.3.5 j(2)	Does normal operation time used in the service period include normal and abnormal operating conditions?		
5.	Glossary	Can contiguous operation time be derived from service history data?		
6.	OIS	Is the “applicable service” portion recognized from the total service history data availability?		
7.	12.3.5 j(2)	What was the criterion for evaluating service period duration?		
8.	12.3.5 j(2)	How many copies of the software are in use and being tracked for problems?		
9.	OIS	What is the duration of applicable service?		
10.	OIS	Is the applicable service definition appropriate?		
11.	OIS	Is this the duration used for calculation of error rates?		
12.	OIS	How reliable was the means of measuring time?		
13.	OIS	How consistent was the means of measuring time throughout the service history duration?		
14.	12.3.5 j(4)	Do you have a proposed accepted error rate that is justifiable and appropriate for the level of safety of proposed usage, (before analyzing the service history data)?		
15.	12.3.5 j(4)	How do you propose that this error rate be calculated? (Before analyzing the service history data)		
16.	OIS	Is the error rate computation (total errors divided by time duration, number of execution cycles, number of events such as landing, flight hours, flight distance, or by total population operating time) appropriate to the application in question?		

TABLE A-4. WORKSHEET FOR QUESTIONS OF TIME (Continued)

	DO-178B Reference	Question	Response	Issues
17.	OIS	What was the total duration of time used for this computation? Has care been taken to consider only the appropriate durations?		
18.	12.3.5 k	What is the actual error rate computed after analyzing the service history data?		
19.	12.3.5 k	Is this error rate greater than the proposed acceptable error rate defined in PSAC according to j. (4)?		
20.	12.3.5 k	If the error rate is greater, was analysis conducted to reassess the error rates?		