

REPRINT

The views, conclusions, or recommendations expressed in this document do not necessarily reflect the official views or policies of agencies of the United States Government.

This document was produced by SDC in performance of contract U.S. GOVERNMENT CONTRACT

TM- 2313/000/00

AUTHOR *Paul Beach*

TECHNICAL

RELEASE *V. LaBolle*
V. LaBolle

D. L. Drukey

DATE 3-25-65 PAGE 1 OF 47 PAGES

5/26

TECH MEMO



a working paper

System Development Corporation / 2300 Colorado Ave. / Santa Monica, California

QUALITY CONTROL FOR SYSTEMS AND PROGRAMMING

A Survey of the Literature

Distribution of this Document is unlimited.

Abstract

This report, prepared as an aid to the Quality Control (QC) Project, summarizes the contents of some 60 documents* that were thought to have possible bearing on QC for computer systems, especially programming. Sixteen of the documents are selected as worthy of further study, either for valuable insights into the QC problem, or because they contain concrete suggestions or experimental data. A synthesis of the sixteen is planned for a future paper.

Table of Contents

Abstract	
General Discussion	2
Appendix A. Sixteen Selected Papers	4
B. Reviews of Remaining Papers Studied	22
C. Index by Authors	43

CLEARINGHOUSE
FOR FEDERAL SCIENTIFIC AND
TECHNICAL INFORMATION

Hardcopy Microfiche

3.008-65-49

PPA

/ ARCHIVE COPY

*Among the SDC documents cited, several have the designation N followed by a number. These are SDC Notes, internal documents not generally available for outside distribution. Some papers contain in their designation the symbol (L), which indicates that their distribution is limited, i.e., controlled by the author. However, it is sometimes possible to release such documents. Requests for all notes or limited documents should be addressed to Document Distribution, SDC.

AD 642302

NOV 25 1966
C

General Discussion

This document was prepared as part of a project to explore the possibilities of applying quality control techniques for improving the design and production of computer-based information processing systems, and more specifically for improving computer programs. Most people who have studied the subject agree that measures of merit, except at a primitive level, are lacking in the computer software field; we know when a program fails to work, but not what is needed to make it "work better"--nor, indeed, what the expression "work better" means in any operational sense. They also agree that if we are to produce "better" software, we must first find a meaning for the word "better" in this context; and, second, find some way of recognizing degrees of merit, so that effort can be guided in the right direction. Anyone undertaking to study these problems naturally wants to know whether any significant work has already been done; or, if not, at least whether other people have thought about the problem and arrived at any useful conclusions. This document consists of a set of reviews of some 60 papers, reports, articles, and books, all selected for their possible bearing on the subject of quality control in information processing.

A true "Review of The Literature" was obviously impossible, because of the wide dispersion of possible sources: these would include (among others) journals in Russian, Japanese, Chinese..., together with in-house papers and graduate theses almost without number. Availability was a primary criterion for selection. This led necessarily to a concentration on SDC and RAND reports. Such a concentration is perhaps not as objectionable as one might suppose. One correspondent in another organization to whom I wrote for possible leads replied, "I would expect SDC to be the best source of this kind of information." Apparently SDC has the reputation of being a pioneer in the software field. Many people at SDC have expressed interest in the quality control problem and have written reports presenting their ideas, and at least some of these authors have presumably been conversant with current progress elsewhere. But even within SDC, it has not been possible to review every one of the thousands of documents produced during the company's history. Papers were selected for review because (a) somebody recommended them; (b) the title contained suggestive words (like Quality Control, or Performance Measures); or (c) the author of the paper was known to be working on some allied problem. The same method of selection served also for non-SDC publications.

The total contribution of all these papers is small. Not one of them is indispensable, and all of them put together fail to provide a firm basis for further work. The reviews here presented have mainly a negative value: they may save a future researcher the labor of acquiring, reading, and discarding. If he wants to review some literature, he will at any rate not have to review this particular literature. If he decides to review some of this particular literature anyway, he may, perhaps, be able, by looking over the reviews, to pick out papers more likely to reward his efforts than papers selected at

Best Available Copy

random, or because they have attractive titles. Selection by title can be especially hazardous; some of the least valuable papers reviewed actually use the words "Quality Control" in their titles.

I have said that all the reviewed documents put together do not provide us with a firm basis for further work; by this I mean that they are of no help. Many of them are nearly or quite unrelated to our task. A few contain good discussions of the need for quality control, but these deal with the destination, not with ways for getting there. An even smaller number give charts of trails that have been tried and abandoned; but in no case is the information complete enough to allow us to assert confidently that a particular trail can be dismissed as impossible. Not one of the trails described was followed long enough, and persistently enough, to establish it as either promising or unpromising; in fact, one might almost say that in each instance the explorer selected a path, followed it for a short distance, found that it did not lead to a paved superhighway, and then turned back.

But I do not say that the collection of reports is valueless. Some of them explain, quite convincingly, why it is important for us to cross the mountains. They do not help us find a way across, but they do help us to choose among the four possible courses of action. Others tell about approaches that the authors thought might be worth trying; these are speculative, but the speculations of informed people are entitled to consideration. Among the documents we find a few charts of trails actually followed for a short distance. I have selected sixteen papers for separate presentation and discussion; these are reviewed in Appendix A. In my opinion, these sixteen contain everything of value to be found in the entire collection. This is not to say that the rejected papers contain no useful ideas, but only that these ideas are discussed adequately (and I think better) in the selected group. There is necessarily some repetition even within the sixteen; this is good, as it suggests some agreement among the authors.

Appendix B contains reviews of all the other documents examined. Most of these reviews are brief, but a few are longer, presenting as much of the original as seemed interesting or useful. The report ends with Appendix C, an index, alphabetical by authors, of all the papers reviewed.

Appendix A

The Sixteen Valuable Reports

The following is a listing of the sixteen valuable reports. In three cases, several reports are considered together as a group. The order of listing is chronological, beginning with the earliest date. For groups, the date assigned is the date of the earliest paper in the group.

- Page 6 Interviews with SDC Management, Corporate Management System Staff, FN-6860/000/00, 7 September 1962.
- 7 Programming Languages and Standardization in Command and Control, J. P. Haverty and R. L. Patrick, RAND Document RM-3447-PR, January 1963.
- 8 Management Aspects of Computer Programming for Command and Control Systems, V. LaBolle, SP-1000/000/02, 5 February 1963; Management of Computer Programming for Command and Control Systems: A Survey, K. Heinze, N. Claussen, and V. LaBolle, TM-903/000/02, 8 May 1963; Quality Control in Computer Program Development, V. LaBolle, an undated paper which appears to be a draft.
- 11 A Summary of Lessons Learned from Air Force Management Surveys, ABSCP 375-2, 1 June 1963.
- 11 Program and System Testing (Chapter 16 of a projected book), N. E. Willmorth, N-19789/370/00, 28 April 1964 (the first section is titled "Program System Quality Control").
- 14 An Approach Toward Quality Control, P. V. McIsaac and F. D. O'Connor, N-LX(L)-621/000/00, 4 June 1964.
- 18 Factors that Affect the Cost of Computer Programming: A Quantitative Analysis, L. Farr and H. J. Zagorski, TM-1447/001/00, 31 August 1964.
- 18 A Technique for Improving the Management of a Computer Installation, R. L. Patrick, RAND Memorandum RM-4232-PR, September 1964.
- 19 Training and Job Performance Validities of Programmer Trainee Selection Variables, TM-2172, and Validity of Programmer Trainee Selection Variables, TM-2173, both by Dallis K. Perry, both dated 9 December 1964.

25 March 1965

5

TM-2313/000/00

- Page 20 Computer Status Quo, R. L. Patrick, SP-1947, 15 December 1964.
- 21 The Correlation of Computer Programming Quality with Testing Effort,
A. E. Tucker, TM-2219/000/00, 26 January 1965.

Interviews with SDC Management, Corporate Management System Staff,
FN-6860/000/00, 7 September 1962.

This document presented management personnel with a series of questions, and on page 4, we read Question 3, part D, "How would you try to convey to management the quality of the work?" This question is clearly relevant to our subject, since any attempt to convey the quality must imply some sort of estimate of quality. And on page 9, we have some answers; I quote now, the bottom paragraph on page 9: "Question 3-D asked for ways of conveying quality to management. In slight variance with the intent of the question, most interviewees were first of all concerned with ways of measuring quality. Many expressed concern over the difficulty of even determining the quality of software. Suggestions for improving quality control at SDC also appeared. All responses to this subquestion were treated as belonging in a single major information category."

On page 11, there is another question: "Have corporate goals been clearly established against which the contract can be weighed?" And on page 12, two more questions: "Is the proper skill mix available within SDC?" and then, two lines further down: "Do we have available the proper skill mix?" which seems to be the same question. On page 13: "Can quality be reasonably measured and reported?" On page 14: "Are reports available to insure that the most effective control can be exercised over the project?" Farther down: "Are adequate control measures being employed?" and still further: "Have proper criteria been established to judge the adequacy of the product?" In the body of the paper, percentages are given of the number of management personnel that showed some concern with the question. On one table we have from 88 to 100 percent of the management personnel concerned with quality. In another table we have from 0 to 38 percent. On page 21 we read, about the middle of the page, "A summary comment reflecting frequent stated opinion is, 'Generally, we at SDC produce too many reports, and they are too complex. We need simple summary reports which pinpoint key facts and serve as a basis for special requests for detailed reports.' Eight comments specifically supported the position that SDC needs to eliminate unnecessary reports or parts thereof." Then another quote, farther down the same page: "One significant observation voiced by several cautions that regular reports in their present form are not the general source of decision premises-- that information for decision making generally stems from personal contact or ad hoc information reporting. The regularly written report in standard format becomes a useful historical document but does not serve to keep the manager informed of the true status of the project at the working level."

On page 51, there is a selection of statements from interview data, and

I quote a few of these. "Project reports get to be bothersome without management support or feedback." Again, "When lower management feels that the information is actually being used, the quality of the information will improve." And on page 50, more interview comments: "If you're on schedule, on budget, and your customer is happy, this may be the best measure of quality." And again, "Management should be more concerned with quality and subordinates would find ways to report on quality if they thought management were interested." On the last page, "I can recall no other instance of an attempt to obtain systematically from within SDC at levels lower than the management council suggestions for the improvement of anything." Further along: "Somebody should come up with a measurement of quality and quantity other than money spent."

It seems fairly clear that a good many of the people interviewed here were concerned with the quality problem in the sense in which we are talking about it, that is, the control of quality during production rather than the repair or the correction of errors already produced. Two or three of the interviewee comments are to the effect that if top management displayed a really lively concern about quality, the production personnel would find some way to report it. This has certainly been the case in other establishments; as long as the general feeling was that top management was not interested particularly in some aspect of the work, that aspect was likely to be neglected by production personnel. I think perhaps this observation may be the most valuable pointer we can get from this particular document.

Programming Languages and Standardization in Command and Control,
J. P. Haverty and R. L. Patrick, RAND Document RM-3447-PR, January 1963.

This document belongs to the small group that propose definite measures, and moreover, has the additional characteristic that it includes some experimental data. On page 24, Table I, "Comparison of Compile and Execute Times," we have some data tending to show that JOVIAL programs take about twice as long to compile as FORTRAN programs, and that the JOVIAL execution time at best is equal to the FORTRAN time, and at worst takes four times as long. On page 57, Table II, "Comparison of IBM 650 Programming Systems," we have a comparison among four languages, with compile or assembly times, time required to load program in computer, number of chi-square solutions per minute, using the computer output; and a similar problem (chi-square solutions per minute) without read or punch instructions.

On page 31, we read, "The most glaring deficiency in the software area is in performance parameters. This deficiency will remain until we develop a cost and data collection endeavor and rigorously define each process and subprocess in the programming area. In the absence of these definitions, already complicated interrelationships become indescribable."

We must be able, at some point, to analyze multiple criteria and complex performance trade-offs."

In my opinion, this document definitely belongs to the small class of documents that must be reserved for close reading by anyone interested in applying quality control to systems.

The next three papers are treated as a unit because of a common authorship. These are Management Aspects of Computer Programming for Command and Control Systems, V. LaBolle, SP-1000/000/02, 5 February 1963; Management of Computer Programming for Command and Control Systems: A Survey, K. Heinze, N. Claussen, and V. LaBolle, TM-903/000/02, 3 May 1963; and finally, an undated paper which appears to be a draft; it does not bear an author's name, but it is by V. LaBolle and has the title Quality Control in Computer Program Development.

The first paper, SP-1000, deals mainly with costing, but since (in my opinion) cost control and quality control are very nearly opposite sides of the same coin, it is appropriately included in this review. LaBolle measures program output mainly by number of instructions, but on page 15 says as follows, "Although these charts provide some insight into programming costs and their relationship to product size, the charts have limited use for planning a large programming effort. Even if these data were highly reliable, and a sufficient number of cases were available for high confidence, the variables are missing that would permit a manager to make accurate cost estimates for programming. Data are needed to establish the relation between requirements and the size of the basic product, i.e., number of instructions. Ways must be sought to assign measures to requirements, such as complexity of the data-processing tasks, size of the data base, and expected response time. These measures, in turn, may be correlated with number of instructions in order to find which of these can be confidently used as predictors.

"A caveat is in order with respect to the use of the variable, number of instructions or program size, as an exclusive measure of the product. The popularity of size as a measure is really based almost entirely on its availability. Other very important measures are needed to indicate the value of a program: measures of quality, e.g., freedom from error; performance, e.g., program operation time; and convenience or 'usability' are important to the customer. An obvious shortcoming of program size as a measure of value is that clever, experienced programmers can generate the same logic with fewer instructions than those needed by inexperienced programmers. Even if programmer capability were a constant, the variation in logical power of various machines and their order codes may distort comparisons between efforts in terms of program size. Often, in developing large program systems, computer storage is at a premium and programming effort is used to reduce the number of instructions. Therefore, despite the availability of program size as a basis for

comparing programs, dangers exist when it is used without qualification." Further on, on page 17, LaBolle enumerates, "Some other attributes of program design that might be defined more precisely and possibly measured," and lists modularity, versatility, flexibility, coupling capability (the ease of adding new logic to the program and/or ease of integrating it into a program system) and usability (the ease with which personnel other than the designer can use the program).

Although these comments relate primarily to program costing, yet obviously, a costing program is meaningless unless we have a measure of output; if the output is twice as great, then a cost of twice as much would not be out of line. This measurement of output is the crux of the quality control problem, since it depends on the identification of output criteria.

The second document, TM-903, begins with an abstract, which states in part, "Managers have difficulty in controlling and planning programming efforts without precise and detailed cost data, standard performance measures, and definitions of tasks and products. Knowledge of managing and developing computer programming systems must be extended and detailed, and programming must be formalized." On page 1 of the document, there is a list of papers which the authors regard as relevant, and on page 2, a remark that is especially pertinent: "Project members had difficulty in gathering accurate and complete numerical data." I think this difficulty is going to hamper all attempts to improve understanding of cost relationships and quality control until management decides that these are prime questions on which substantial research effort should be expended. Generally speaking, we should not expect program supervisors and machine operators to take time out to supply data for research unless the effort has some sanction at the top management level.

On page 10, we read, "Opportunities are plentiful for personnel to secure jobs with increased salary and/or challenge, making it difficult for programming managers to keep their experienced people." Farther down the page, "High personnel turnover impairs work continuity." This point, it seems to me, has not been sufficiently recognized by other authors. Many qualified observers have stated that the level of technical writing in the United States is poor. If we have a high rate of turnover among programmers, and if they are not very skillful at documenting their work, the inevitable outcome is that much of their work is lost. I have in mind in particular the documentation on the COMMAND Model, which was produced in the ARPA project from January 1963 to November 1964. Everything considered, I think the documentation on this project was rather better than average; and yet, I believe it is not nearly good enough to enable later workers to pick up the project where it was laid down. If my conjecture is true, then all the work that went into the COMMAND Model is in effect lost; most of the personnel have been terminated or transferred and future personnel attempting to perform

a similar task will, in my opinion, prefer to do it over rather than try to reconstruct it from the documentation. In this connection, it is pertinent to refer to pages 13 and 14 of the document now under consideration (TM-903), in particular Section C, which has the headings "Improved Documentation, Production Control, Quality Control, and Information Retrieval."

TM-903 contains a number of tables and other presentations of data gathered in connection with earlier contracts, and so belongs to the group of papers that present facts rather than those that are chiefly speculative. Moreover, the section beginning on page 13 titled "Opinions and Recommendations," contains some fairly specific proposals.

The comments on the third paper will be brief, partly because most of the ideas presented there are included in TM-903, and partly because the paper has not been published and so is not available. Nevertheless, a few questions and comments seem to be in order.

On page 14, "The development of improved quality control may be regarded as an iterative process. Feedback from attempts to institute improvements will lead to recognition of (1) what can and cannot be controlled, (2) what techniques are best and (3) whether the effort is paying off.... With respect to organizational structure, the clear-cut assignment of responsibility and authority for various aspects of quality control is desirable. On the other hand, quality control goals and methods must be universally understood and agreed with. Improved quality control is almost impossible without mass participation.... Simple techniques used consistently can improve quality control."

In this document, LaBolle makes a proposal of special interest, considering the importance of good documentation to the preservation of work in systems and programming. On page 18, "Documents could also be rated for understandability, by using a scheme such as that developed by Flesch (The Art of Readable Writing, R. Flesch, Collier Books, 1963, paperback). Using sampling or 100 percent inspection, Flesch shows how to measure readability in terms of interest and ease of reading. Interest, for example, is a function of short sentences and the use of many syllables." An alternate to the Flesch index is that proposed by Robert Gunning in a booklet, "How to Take the Fog out of Writing," published by the Dartnell Corporation, Chicago 40, Illinois, copyright 1959. This booklet of 64 pages seems a good kind of thing for the Corporation to supply to all personnel involved with documentation, even if the proposed index is not used.

25 March 1965

11

TM-2313/000/00

A Summary of Lessons Learned From Air Force Management Surveys, ABSCP 375-2, 1 June 1963.

This is an official publication of the Air Force. On page 57, we have a subject: Quality Assurance Audits not Effective. Among symptoms of basic deficiencies we find listed, "quality procedures not followed," "Production frequently failed to meet contract specifications," "lack of communication between quality, reliability, engineering, manufacturing and test personnel," "inadequate manning of quality assurance organization," "incomplete support of top management," "quality audit results not enforced," "lack of follow-up to evaluate effectiveness of corrective action," and among desirable actions by contractors, "increased staffing of quality audit department," "QA trend charts and trend levels established," "quality review board established," "continuous surveillance" and "provide for Air Force feedback of malfunction data."

This document appears to be applicable mainly to hardware procurement, but the type of difficulties listed under the quality assurance heading are those that would be associated with any quality problem. To my way of thinking, many of the points made in this document in connection with quality assurance could be carried over into the quality control of software, if we had some sort of basic approach to the evaluation problem. Obviously, the Air Force document does not provide us with this.

Program and System Testing (Chapter 16 of a projected book), N. E. Willmorth, W-19789/370/00, 28 April 1964 (the first section is titled "Program System Quality Control").

With respect to its coverage of the quality control problem, this document is the longest and the most comprehensive of the documents examined. From page 1, I quote the following: "Unless a comprehensive plan for insuring product quality is adopted that insures continuous product review throughout the process, adequate quality cannot be expected of an ultimate product that is often poorly defined and whose attainable performance is uncertain and involves considerable innovation or risk. Further, production of a large program system is so expensive and lengthy, that the only real alternative to a faulty product is either to forego the system entirely or to live with an inefficient and ineffective system until it can be replaced. Although the greatest interest in quality assurance lies in such ultimate products as programs and operator's manuals, quality is not just something to deliver to a customer, but is part and parcel of the pride in workmanship associated with a responsible and professional attitude. Therefore, a quality control program cannot be based upon a single acceptance test or even a series of program tests, but upon a continuing program of quality assurance from the initial statement of system requirements to the final shakedown of the program system in an operating environment. Quality must be assured, not just of programs, but of the morass of intermediate plans, designs, and documentation leading up to and supporting them.

"Although a good program system development team automatically does the best job it is capable of, it is the management's task to establish the criteria of quality for both products and performance and to set forth the procedures that will make sure that these goals are being attained. Not only must adherence to performance requirements and product characteristics set forth by the customer be observed, but to the conventions and standards adopted by the project to insure greater efficiency in production and maintenance of the product. Some standards are fixed by the constraints and limitations of the production system (computer and utility system) and some by the criteria of good programming practices and the state of the art. Finally, criteria and methods must be set for the detection of the legions of minor clerical and logical errors that almost invariably slip through to crop up from time to time during system operation. Management's main tasks, then, are making sure that precise and accurate quality specifications exist, that searching inspection, and review procedures are established and observed, and finally, that the importance of doing good work is stressed in the goals of the project and in the evaluation of performance."

The foregoing quotation is the entire opening section titled "Program System Quality Control." Willmorth goes on with sections titled "Adherence to Specifications," "Adherence to Programming Principles," "Adherence to Standards," "Constraints and Limitations," and "Logical and Clerical Errors." He then has a section titled "Integrating System Quality Evaluations": "Insuring the quality of such a monolithic product, composed as it is of largely conceptual and logical components, is not an easy task. Traditional statistical sampling techniques do not apply; each product must be reworked until it does meet prescribed standards or is acceptable to the customer. Further, it is much easier to apply quality control techniques to individual components than it is to the overall system, often suboptimizing these at the expense of the system....The steps that must be taken in setting up a comprehensive quality control program are (a) identify those products whose quality must be assured.... (b) identify quality inspection procedures and tests....(d) identify the points where inspections and tests are to be made....(e) assign firm responsibility for conducting quality reviews and for testifying that standards have or have not been observed....(f) specify performance and quality standards for quality reviews and reports....As for any other product, conventions and standards should be set forth for the format and content of review reports and forms."

In sections that immediately follow, we have a subsection titled "Developing Product Quality Criteria": "Quality control plans must specify procedures for reviewing specifications as they appear and either phrasing the specification statements such that required performance and structural criteria are evident or extracting from the specifications the levels of quality that must be observed....Quality criteria are in general more than a specification statement; both a desired state and allowable tolerances should be determined."

In the next section, under "Quality Assurance Procedures": "It is important that at least a minimum set of procedures be established as a guide and to encourage adequate reviews...Unless the test procedures and requirements are quite clear and well structured, much redundant testing may be done and testing is not likely to be as comprehensive as it should be." The next section, titled "Performance Evaluation," says, "A frequently neglected aspect of the quality control program is the establishment of appropriate performance standards for quality review personnel...Quality control programs should establish criteria for adequate reviews and procedures for review evaluations. Management must take note of such procedures, since performance evaluation is largely a management concern, and use them to encourage and enforce adequate product reviews."

This document (which contains 94 pages) continues in very much the same vein, presenting the reasons why quality control is necessary and the objectives that a quality control program ought to attain. There are numerous flow diagrams, and some examples of what Willmorth regards as possible procedures: for example, on page 48, we find "A Sample Assembly Test Plan," which details the purpose of the test, identifies the programs used, and gives the inputs and outputs. On page 66, Willmorth discusses "Organizing for Program Quality Control," and says, among other things, "If quality control is to be at all effective, it must have the whole-hearted support of the project management. It is top management who determines objectives sets policy and defines the scope of project activities. It provides the impetus in establishing product and performance standards, and generates the energy with which these are enforced. In top management lies the ultimate responsibility for quality and from there stems the authority for insuring that quality is enforced. If project management neglects its duties, efforts to establish and enforce goals of quality at a lower level are built upon a base of sand....It must be recognized that a quality assurance program is not just a way of checking up on subordinates, but a methodological approach to the organization's assuring itself that the desired quality does exist."

It does not seem practicable to quote at any greater length from this rather long document. The general flavor is, I think, fairly represented in the extracts given. I, for one, do not quarrel with the author's statements; they appear to present the problem and define the objectives as completely as need be. Nevertheless, the document still leaves open the question, "How, in practice, are these desirable objectives to be attained?" We need, in short, an engineering approach to the problem; one that carries out the steps called for by Willmorth. One of these steps, for example, is to set up quality criteria and devise procedures for insuring that these criteria are satisfied. What we need now is to perform this operation in a specific case. Willmorth points out that basic quality requirements should come from specifications, either

directly or by interpretation. We need to take an actual specification and extract from it the quality criteria, and see just how this is to be done.

The next three documents are tied together by a common concept and a common authorship. The first of these is An Approach Toward Quality Control, P. V. McIsaac and F. D. O'Connor, N-LX(L)-621/000/00, 4 June 1964. The main characteristic of these documents is that they propose a definite kind of device, namely, "system catalog" as an adjunct to quality control.

I quote from the earliest of the documents. The opening paragraph reads: "The establishment and conscientious employment of sound quality control techniques is as necessary to the production of a computer base system as to the more conventional applications of manufacturing. Quality control techniques, however, need not depend upon strict statistical sampling and analysis. Control may be achieved in a variety of ways, including both concrete tools (e.g., COMPOOL, PERT charts) and sound methodology (e.g., state-of-the-art techniques). The end must not be confused with the means. The end is to obtain quality through control. The means must only achieve this end within the bounds of practicality and efficiency.

"New systems, however, pose new problems which necessitate the development of new controls. Toward this end, the following device, which we term a system catalog, has been conceptualized. The system catalog will serve for the collection and description of system data. Several of its primary advantages are envisioned as follows: (1) provide an accurate and up-to-date description of all system inputs and outputs, (2) provide a crosscheck between operational design and the real world, (3) provide a means to interact with outside agencies, (4) provide a source of information for changes to the program, (5) provide for accurate inter-program communication, (6) provide guidance and control for the efficient structuring of data, (7) provide a description of the relationship between operational design and program design, (8) provide a convenient source for the rapid updating of crucial changes....It is hoped that a realistic attitude will be assumed toward the catalog in order that it may become an efficient and fruitful tool rather than an awkward and useless relic."

There follow some diagrams in which the system catalog is shown as an intermediate between the ops equipment requirements and program coding; and then there is a list of components that might be in the catalog. First we have inputs: "teletype inputs, data link inputs, console actions, keyboard actions, and processing." Then for outputs, we have "teletype outputs, category displays, tabular displays, special displays, alarms hard copy displays."

There is then at the close of the document what the writer calls "a sample

of a single 425L system input message and its resultant outputs as it would appear had it been extracted from a complete catalog." The remaining seven pages of the document are taken up by this attachment and this shows, in tabular form, the various types of inputs and outputs. The table contains 12 columns and each has a heading. There are cross-indexing and various abbreviations in the table so that it's necessary to consult some sort of dictionary or glossary to know what the entries mean.

The second paper is Initial Insights Toward System Quality Control,
P. V. McIsaac, N-LX-621/100/00, 15 July 1964.

Again, quoting from the document, "Quality control, in its broadest sense, refers to the systematic control of those variables which affect the excellence of an end product. Quality, however, is an abstract word, unless related to definable and measurable characteristics of the product involved, in this case, a programming system. But how does one define the quality of a system? Obviously, one good measure would be the degree to which it meets consumer requirements....Such a definition is, of course, much too general, for what criterion measures does one use to measure adequacy and how and when are these administered? Such are the real problems of system quality control. But even at a general level, we can recognize two distinct quality control requirements: one is the need for standards and specifications that establish the quality objectives to be measured or evaluated, and the second is a more dynamic need to provide the devices and techniques by which such quality objectives are reached and subsequently maintained....Control measures should first be aimed at eliminating those assignable variables as opposed to chance variables which might be system availability, maintainability, computing efficiency or reliability. They might also include operating time, system size, expansion capability, flexibility to changes, or simply a criterion that the system be operational in x months....One of the greatest stumbling blocks toward effective quality control in the past has been a poor recognition of just what was to be controlled.

The
P. V

"One major bulwark standing squarely in the path toward more effective quality control is the very conception of the term itself. The development of a system does not currently consist of a single process but is more a series of large complicated processes involving a collection of professionals whose needs for quality control and thus their interpretation thereof are as diverse as their backgrounds and duties. System quality control cannot be limited solely to inspection or measurement, since a system, by its very nature, must be kept under continuous ongoing control. There is no scrap pile for a system which 'fails to measure up to inspection criteria.' Inspection is but one necessary and admittedly neglected area among a number of devices, techniques, and methodologies which enhance the successful development of the system and which must be relegated both legally and properly to the domain of quality control.

"We can break these control needs into an inspection or measurement function and a control or production function.

"One must readily accept the fact that any one set of quality control techniques will necessarily fail to satisfy an entire system's needs. It seems we must rather concentrate on control requirements according to area and function....We might attempt to develop techniques to assist management in controlling the production of a system or design a tool to assist the technician to better check the logic of his code.

"A narrow definition of the term quality control must not serve to restrict the means by which system quality is achieved. All means effective in developing higher quality must be given consideration. First consideration, of course, should be aimed at improving our current methods of operation....It may become necessary to revamp or even discard old methods....This seems to be the current plight of the system development process. Although we admit to inefficiency and lack of quality, we nevertheless are reluctant to adopt any device or methodology which fails to use the current basic approach to the process....Perhaps we should be examining the process itself rather than attempting to plug elusive holes created by the process and tagged with the label 'lack of quality control.'"

The third document is Model of a System Catalog for Quality Control,
V. McIsaac and F. D. O'Connor, N-LX-(L)-621/200/00, 5 August 1964.

The first paragraph reads as follows, "This Note is the third in a series of documents produced on the subject of quality control techniques for computer systems. It represents a further development of the initial concept of designing an input-output catalog to be used as the controlling source for operational and program system design." The bulk of this paper consists of something like 24 fold-out sheets printed out from a computer and showing an actual form that in the authors' opinion embodies the type of system catalogs they have in mind. This catalog apparently would contain quite a sizeable body of descriptive information, and on page 4, under subheading "Using Catalog Sections in Other Capacities," we read, "In addition to the conventional catalog application as a central agent for the collection and description of system data, each section of the catalog might also serve other purposes. For example, Section 2, Data Base, has been designed so as to retain the characteristics of the symbolic COMPOOL. Instead of modifying the COMPOOL structure, catalog control cards were added to each table description. With only trivial modifications to the COMPOOL assembly program, this section could easily serve the dual function of catalog section and symbolic COMPOOL. No additional manpower need then be expended to maintain a symbolic COMPOOL unique to catalog Section Two. Section Three might also be used in the capacity of a set/used listing, thereby reducing the need for such a function being produced independently of the catalog."

The next section is titled "Impact of a System Catalog on Existing Specifications." "The intent of the proposed system catalog is not to supplement the contents of operational and program design specifications but to constitute the primary and controlling source for system outputs and inputs. It should therefore be assumed that if a system catalog evolved from the onset of acquisition, the current contents of design specifications would probably differ. The existence of a system catalog would obviate the need for some of the detailed design information that currently exists in operational and program design specifications. In all likelihood, operational specifications would be subject to fewer changes (and inconsistencies) in addition to being a more meaningful document for customer review and concurrence. A system catalog should not be just another source of information, otherwise its primary purpose (to control) is defeated."

This is the end of the quotations from the three documents, all indexed under the primary designator of N-LX-621. It seems to me that in commenting upon these documents, it is necessary to regard them in two aspects. The first of these aspects, which is best illustrated in document number two and in some of the quotations from document number one, deals with the writer's concept of the quality control problem--its difficulties and some of the requirements of a valid quality control effort. The second aspect deals with the proposal for a system catalog. In this, the authors get down to a concrete recommendation, and supplement it with a rather extensive sample catalog, presumably intended to show what such catalogs ought to be like.

I can only agree wholeheartedly with McIsaac's interpretation of the quality control problem, with his philosophy of quality control, and with his broad ideas of how the problem must be approached. I agree that a doctrinaire limitation to some particular method or control device would be a mistake. I agree that various aspects of system production and checkout will require different methods. I agree that though system testing is important and probably could use much additional effort, yet it must be distinguished from process control. I think it perfectly possible that in order to achieve this control, it may be necessary for us at least to investigate quite revolutionary new methods of system production. In fact, I might almost say that McIsaac has expressed my opinions about the broad quality control problem quite as well as I could have expressed them myself.

With respect to the second aspect of these papers, the "system catalog" device, I feel much more hesitant. I do not see that McIsaac and O'Connor make it clear how this system catalog is to be used in practical quality control. In fact, I do not see how this listing of inputs and outputs would help a manager in any way. The samples given in documents 1 and 3 are not legible to managers without special training; not only would it

be necessary to understand the tabular arrangement, but the person using the catalog printout would also have to be familiar with a considerable vocabulary of abbreviations and computer jargon. The catalogs themselves are quite voluminous. They do not have the property of, say, a quality control chart, that the observer may assimilate a large amount of information at a single glance; one gets the impression, rather, that the catalog would at best be a source of information which would then need further processing before it could be presented as a display to the quality control supervisor. In short, it is not at all clear to me how this system catalog could be used, nor that any substantial area of quality control is covered by the system catalog.

I acquired so high a respect for the authors through the general philosophy of quality control as expressed in document two, that I prefer to close these comments with reservations. It may well be that the system catalog can be used as the basis for certain important control data that would solve part of the quality control problem. It may even be that the method of its use is implicit in the papers, and that it is clear to more preceptive minds than mine. The fact remains that I do not see how it is to be used, and I suggest that the method of its application will need to be made more explicit and described in simpler terms before it can be generally understood.

Factors that Affect the Cost of Computer Programming: A Quantitative Analysis,
L. Parr and H. J. Zagorski, TM-1447/001/00, 31 August 1964

As the title implies, this document is concerned rather with the cost of computer programming than with the task of maintaining or assuring program quality. However, on page 11, the authors point out, "Cost estimates are used for evaluation. Equally important to the direct uses of improved cost predictors are the indirect uses. For example, predictors can be sought that relate requirements and resources to the methods used to control costs."

Although TM-1447 does not address itself particularly to the quality or performance problem, yet in my opinion, it should be part of the library of anyone working on system quality and performance.

A Technique for Improving the Management of a Computer Installation,
R. L. Patrick, RAND Memorandum RM-4232-PR, September 1964.

This, like an earlier RAND document by Patrick and Haverty, has the merit of recommending specific proposals and presenting some data. On page 1, in the Introduction, we read, "The conclusions of a previous Memorandum (reference here to the Patrick-Haverty paper) stressed the fact that there are no generally accepted measures of performance for computer-based systems. In the discussions that followed publication of that

Memorandum, some concluded that any measure of performance is sufficient to improve the present state of affairs. An informal technique exists and is being employed by some managers of computer installations, but to date, none of them has formalized and documented the technique. It was decided worthwhile to formalize the existing technique and to make it available for possible use by others."

The method proposed by Patrick is, however, only indirectly a method of quality control; it is much more closely related to cost control, and, in fact, is described by Patrick as a type of cost accounting. Patrick divides the activities of a computing center into 24 categories, and proposes that separate costs should be assessed in each category. He presents specimen forms that he thinks might be used, together with samples of how they should be filled in.

Although the Patrick paper seems to deal with cost rather than with quality control, it may be as well to bear in mind that these two aspects of management are very closely related. Any system that would give us a better understanding of costs would almost automatically give a better understanding of quality. Although the methods of assessment produced by Patrick do not relate directly to quality, yet this document must, I believe, be selected for careful reading by anyone who proposes to continue with quality control in systems.

Two papers, Training and Job Performance Validities of Programmer Trainee Selection Variables, TM-2172, and Validity of Programmer Trainee Selection Variables, TM-2173, both by Dallis K. Perry, both dated 9 December 1964.

These documents report essentially the same data, TM-2173 being more in the nature of a summary and more informal.

These documents are of interest to the present survey only because they include performance measure, which would necessarily be closely related to, if not identical with, measures of the quality of a programmer's output. Data were collected on 452 trainees, including age, sex, level of education, and scores on the PMA (primary mental ability) test and the OTIS general intelligence test. Also available were grades awarded in training courses, first in Q-7 programming and later in SAGE programming. On-the-job performance was measured by several criterion variables, including length of employment, salary progress, salary review rating, and classification progress. Curiously enough, the largest correlation was between education and salary progress for trainees who took the SAGE course, and this correlation coefficient was negative and significant at the 1 percent level. Unfortunately, the relationship was not confirmed with trainees who did not take the SAGE course; the correlation coefficient here was again negative, but equal only to $-.01$. If the negative correlation had been confirmed for both groups, it might be interesting

to try the performance of persons with only a fourth-grade education. Several of the positive correlation coefficients attained significance at the 1 percent level, though this fact must be discounted when we remember that 80 such coefficients were computed for Table 8 in TM-2172. The predictor that gave most consistent positive correlations with the four criterion variables was number of dependents, and even here, the largest correlation coefficient was .23, suggesting that 95 percent of the variability is still unaccounted for.

Though the correlation coefficients obtained in this study are not exceptionally small in comparison with those obtained in most validation studies, yet they are much too small to form the basis for an operational quality control system. To put the same statement differently, we do not know whether the predictor variables are in fact related to performance by any strong relationship, nor are we sure that the criterion variables such as salary progress and salary review rating are valid measures of on-the-job performance. The conclusions of these two papers, that the tests proposed have some value in aiding in programmer selection, are not disputed; but if a quality control procedure is to support day-to-day actions, a performance criterion is needed with a much smaller standard deviation than is indicated for the criteria used here. Ferry obtains better correlations, of course, by using a weighted combination of predictors; but this improvement would itself have to be validated by a repetition of the experiment. In any case, it seems unlikely that the performance measures used by Ferry are sufficient for routine quality control purposes.

Computer Status Quo, R. L. Patrick, SP-1947, 15 December 1964.

Unlike the other document by Patrick, this one does not contain any data. It does contain, beginning on page 21, a suggested sequence of actions for producing a program. Patrick divides the total effort into seven acts, each of which is further divided into scenes (Act VII consists of a single scene, Act V contains 12). The tone of much of the writing is somewhat tongue in cheek, but it may well be that a discussion of this kind could be valuable.

On page 34, Patrick begins an epilogue, in which he says in part, "The author is not aware of any concentrated effort to solve the debugging problem by understanding the debugging process so that proper tools may be produced to alleviate it. In the past, we have spent vast amounts of money without a clear-cut objective or approved plan." Patrick concludes his paper with the following sentences: "We may design systems that are so demanding that we can never make them work. Such is the life of the computer professional. What an exciting time to live."

Although I do not believe that this paper contains any important

suggestions for operational quality control or improvement, yet it is readable and perhaps should be included in the quality control analyst's small shelf of books.

The Correlation of Computer Programming Quality with Testing Effort,
A. E. Tucker, TM-2219/000/00, 26 January 1965.

Among all the papers reviewed, there are very few that propose an operational scheme for evaluating any part of program production, but this paper is one of those few. Tucker has collected data on the number of discrepancy report forms as a function of time; the time, that is, expended in searching for and finding programming errors. There is a modest body of theory, applicable to problems of proofreading or search, that might be invoked in this kind of effort. Tucker presents some graphs showing how the number of errors discovered changes with progressive time, and in the light of proofreading theory, it seems quite possible that some mode of estimation could be devised that would give an estimate of the number of errors undiscovered after a particular time, using as input the number of errors already found. Tucker says in part, on page 32, "A qualitative feeling or indication of the quality of an individual program can be obtained from the slope of the error accumulation versus testing effort data plot. This feeling or indication can be reinforced by the use of an estimated total error population for the program....The relative quality of different computer programs can be established by a data normalizing procedure in which the total error population for each program is established." Tucker goes on to recommend that a more extended and systematic effort be made to collect data on numbers of errors and testing effort, and to analyze these data. "Until sufficient data is available to justify other standards, the quality of SPD computer products should be defined in terms of a nominal model based upon the normalized data presented in this report."

It is so refreshing to find a document with a specific operational procedure to propose, that perhaps my enthusiasm runs away with me; nevertheless, the number of errors in a computer program is certainly one measure of its quality, and there does exist some statistical theory that would allow its estimation, following the lines suggested by Tucker. I think this document must be placed very high in the stack of documents to be considered seriously in deciding how to continue with the problem of quality control.

Appendix B

This Appendix contains reviews of all documents considered but not included in the selected 16.

Computer Program System Development Milestones, no author, SSD Exhibit 61-47A, no date.

This document "defines the content of the computer program development milestones products to be delivered to the Air Force." It does not present specific methods of quality evaluation, but it does indicate that the Air Force expects some sort of quality standards to be observed. For example, on page 21, it speaks of acceptance criteria, "Outputs checked and any particular results to be noted are checked. Possible data to be preserved for other tests should be identified." This on Milestone 5. There is another requirement, under Milestone 11, that calls for quality recording and reporting, though again with no suggested measure of performance.

The document is of no value for suggesting specific quality control measures; its only interest lies in the witness it bears to the Air Force's interest in quality.

SAGE Program Implementation History No. 2--How to Set up a Field Site (1957), N. H. Cannell, W-10642, 22 February 1960.

This document is included only because the title suggests it might have material relevant to quality control; in fact, it contains no such material. It is chiefly an outline, apparently intended for training instructors, showing how they might proceed at a field site to break in new personnel.

The subject of testing is listed, with such headings as "Test Philosophy," "Standard Test Concept," "Additional Testing," and "Assembly Test Procedures." Operational testing is described only in a very general way. In my opinion, this document has nothing of interest for our case.

"How Do You Measure Useful Computer Time?" Automatic Data Processing Digest, Volume 6, Number 5, March 1960, pages 14-17.

The abstract begins with the sentences, "Measurement of good and bad computer time is not easy. Some activities may be considered good (i.e., productive) and others--bad (or nonproductive), while some may be open to doubt or regarded as neither. The activities are grouped under these headings: preventive maintenance, actual production runs, development of programs, time lost due to computer

faults, repair time of computer faults, time lost due to operator or programmer errors, idle time, miscellaneous occurrences." Further along, the writer points out that there is no unique scale for the measurement of goodness or badness in computer utilization and that widely different figures for the same set of circumstances might be obtained merely by applying different yardsticks. He concludes with the recommendation, "It is suggested that manufacturers get together, possibly under the auspices of the British Computer Society, and 'settle a formula that could be used for all installations and produce strictly comparable figures.'"

It seems clear that here again we have a pointing out that the problem exists, with not much specific in the way of a proposed solution.

Quality Control, T. J. Snyder, N-18476, 22 June 1962.

Mr. Snyder begins his paper with the following sentence, "Perhaps the first problem to be addressed in a working paper of this kind is that of a cautious definition of the term quality control. At first blush, of course, this term does not appear to be much more vague or difficult than some others we have lived with and used, but it is. Quality control is a dreadful term made worse by increasingly general use and misuse." After some preliminary discussion, Snyder resolves the problem into recognizing two general classes of errors which he calls system errors and nonsystem errors, and the objective of quality control appears to be the control of the frequency with which these errors occur. Both system and nonsystem errors can occur as either random or what Snyder calls methodic; I quote now, "Nonsystem errors probably comprise the bulk of any error package. These run the gamut, from simple, foolish mistakes to great, irrevocable blunders. They are discouraging to find and worse to be responsible for. Of all the types of errors we should most want to control, we are least likely to control these. But perhaps we can find the means to minimize them.

"Random nonsystem errors are very bad things. They result from keypunch failures, dull pencils, tired eyes, etc. These random errors can crop up almost anywhere in the production cycle, and in any medium employed. They are bad things." Further along in the paper, we read, "Off line error checking bears the brunt of most so-called quality control operations. It is here that the most sophisticated planning and creative thinking is needed, and has always been lacking. Normally, in a highly complex system, this operation exists as an ex post facto product measure. In fact, off line checking should be largely incorporated into the preproduction phase; that is, throughout the input specification and preparation stages. Further, it may be more feasible to plan its application in small portions--not only to detect errors as soon as possible, but in order to distribute the operation in time and

space." Finally, near the end, Snyder says, "We must determine whether we want an error-checking or an error-correction provision, or both, incorporated into the program system. This will be a difficult task. By adopting, for example, a set of programmed measures that will accomplish sophisticated error correction in conjunction with error detection, the quality control feature may become unwieldy. The feature can become so complex itself, in fact, that it too may begin generating sufficient errors to come under the jurisdiction of yet another quality control feature."

It does not appear that in this paper, Snyder has done more than point out the need for quality control, and the dichotomy between measures used after the product has been produced (which I prefer to call acceptance inspection) and measures applied to the production process, which I regard as true quality control. The paper contains interesting discussions of the kinds of errors that can adversely affect quality, but does not really suggest anything specific in the way of procedures.

On Measuring the Validity and Effectiveness of the Response of a Military Command Control System, C. K. Gordon, Jr., N-18651/000/00, 25 July 1962.

This document is not directly addressed to the problem of quality control, but since it deals with a measure of effectiveness, it is worth considering in any survey of supposedly quantitative approaches to a system problem.

I quote from the first paragraph, "Of the many kinds of responses open to a military command control system, possibly the most obvious is the attack. Such a response has a number of different properties, two of which will be considered in this paper, viz., validity and effectivenessThe validity of a response may be defined as the degree to which the response conforms to the intent of the commander. The effectiveness of a response may be defined as the extent to which the intent has been realized." The rest of the paper is a somewhat mathematical treatment based on some rather abstract concepts, and it is not at all clear that the parameters needed for the mathematical expression are operationally ascertainable. In any case, the document contains no suggestion at an operational level that might provide us with a real measure of either validity or effectiveness. Gordon even says on page 8, "The attempt to define validity or effectiveness in some simple, universal way is fraught with difficulties." The purpose of the paper seems more to emphasize these difficulties, rather than propose any method for overcoming them. As such, it seems to have little value for our present study and so I think can be disregarded.

The following two papers are by different authors, but on the same subject, so they are discussed together. These are Information Processing System Design: General Principles, Part I, Jack Jaffe, TM-743/001/00, 9 August 1962, and Principles of Information Processing System Design: Part II, George Masters, TM-743/002/00, 19 February 1963.

In the first paper, Jaffe says, on page 41, "Evaluation of the Design. This is the evaluation of the design product itself, as a design, before the system is constructed or before the design process is reiterated, whichever approach has been selected by management.... Errors of omission and internal consistency are searched out and when found, fed back to the design production effort for correction. This is a hard and lengthy process and one that a great deal of time must be allowed for....A very important element of the evaluation process is the establishment of design quality criteria. This means the discovery of the important features of the system and eventually the setting of levels of acceptability. Particularly with computer based systems, there is a need for defining the relevant functional areas. A programmer may need to know access speed for core memory, but a designer needs to know how long an operator will have to wait for a new display that he has requested."

In the second document, beginning on page 34, under the heading "System Analysis," Masters says, in part, "In discussing system analysis, we will not cover analysis involving mathematics or other sophisticated analytic or rigorous techniques although these techniques are certainly important aids to the designer when they are applicable. The concentration here will be on what will be called rational analysis." Masters goes on to discuss the subject in very much the same way that it is discussed in the book by Goode and Machol, using very much the same terminology. No specific procedure is suggested.

Although the discussions in these two papers are interesting and would be useful to someone approaching the problem of system design for the first time, I see no helpfulness for the task of creating a methodology for quality evaluation or control.

The next two documents are closely related to each other, and they are of special interest because both of them contain the expression quality control in the title. In point of time, the earlier of these documents is SPCR 1576-- Recommendation, Real Time Quality Control, W. L. Thomson and H. P. Dowst, TM-5353/576/00, 24 August 1962.

The opening paragraph reads as follows, "One of the principal criteria for successful operation of the SAGE system is that each subsystem be maintained at design level. That is, performance of each subsystem must be checked and verified to insure the reliability of that subsystem

as an integral part of the SAGE system. Present techniques for accomplishing this are inefficient and quite often ineffective. This document outlines modifications in SAGE system design to provide a real-time quality control capability."

A little down the page we have, under the heading, "Recommended Solution," the following: "The recommended solution to the quality control (QC) problem is a computer program vehicle which is time-shared with the real-time simulation program. This vehicle consists of several new DCA programs which, together with modifications to existing DCA programs, will perform the following functions:

- a. Real-time monitoring of LRR and GFR test messages.
- b. Real-time analysis of SIF codes reported by MK space x input equipment.
- c. Real-time analysis and correction of registration and collimation errors.
- d. Periodic calibration of height finder RHI consoles.
- e. Real-time monitoring of the height communications loop.
- f. Real-time analysis of negative height replies.
- g. Recording of information preparation of subsystem performance summaries.

The implementation of some of these features requires equipment modifications; additionally, the programming of the time-shared solution can best be implemented so as to be concurrent with other SPC activity. For these reasons, an interim package which does not contain monitoring of GFR test messages or any of the height features will be provided. This interim package will provide an early operational quality control capability, temporarily using existing spare drum allocations which will be freed by the release of the final time-shared package."

If we look through the remainder of this document, we find that it consists essentially of proposals for a number of standard test problems or messages intended to be presented to the system, with monitoring of the resulting output. This is conceptually identical with industrial testing, where the product is checked from time to time to verify its performance; it is distinct from process quality control, which is intended to monitor the process of production. In the subject document, a number of tests are described, but these are all quite specific to the particular system; they have no general application, and could not be transferred to computerized systems of all kinds. Necessary as this type of inspection and control is to the successful operation of a large system, it is not the total task as that visualized in the present approach to quality control.

Closely allied to this is Interim Production list of SPCR 1576 (the reference is to the previous document), R. W. Green, N-17264/040/02, 15 October 1962.

N-17264 appears to be an attempt by the author to translate the requirements of SPCR 1576 into computer jargon. Thus, on page 29, we read under the heading, "Alternate Quality Control/Real-Time Simulation

Taken by ASO, AST
 C ϕ Activate Button is Used
 C ϕ 2 if QRTI is ϕ , PSMI must not be 'ON' (3).
 R ϕ 1 Complement ARTI
 R ϕ 2 set QMOD to 1"

and so on.' This is then a set of instructions to the programmer intended to enable him to carry out the ideas of SPCR 1576, and as such, is of no general interest to us.

Operational Measures of Information Characteristics, R. M. Longuire and W. J. Erickson, SP-930, 1 September 1962.

The title of this document is very encouraging, but I do not believe the contents quite come up to the expectations generated. The authors recommend a device which they call a STAMIC chart, which turns out to be simply a double dichotomy in which information statements are classified as true or false, relevant or irrelevant. This proposal is followed by a symbolic mathematical treatment, which, in my opinion, is not very operational, in spite of the title. I am, in fact, not able to see how the proposals of the authors could actually be followed out in any practical case. Thus, I believe the document may be set aside.

System Simulation for Evaluating Product Quality, J. R. Crawford, SP-992, 12 October 1962 (written for presentation at the 17th Midwest Quality Control Conference in Denver, October 26 and 27, 1962).

The title of this document is appealing, but upon examination, it turns out to be a proposal for a queuing model to simulate transactions in a warehouse. I do not believe it has any relevance to our problem.

The Elusive Criteria in Command and Control, Gordon M. Becker, SP-198, November 1962, General Electric (TEMPO), Santa Barbara, California.

This paper was presented to the human factors working group at the Tenth Military Operations Research Symposium in October 1962. Becker starts out by saying, "One of the oldest problems we face in military research, and one of the most difficult to resolve involves the selection of a criterion to evaluate a military system, subsystem, or component. The problem is especially important, and difficult, in command control

systems research. It is important because these systems vitally affect the nation. It is difficult because these systems must be acceptable to many diverse groups in the nation prior to being procured, and because they must satisfy the requirements of a multitude of users after they are procured." A little later, we read, "Given this structure within which we must operate, what can we do to increase the efficiency of our efforts in the design of command control systems? Two recommendations to the Director of Defense Research and Engineering, which appeared in a report of the Institute for Defense Analysis are pertinent to this problem. These recommendations were (1) mechanisms whereby technical and functional compatibility efforts are coordinated within the Department of Defense should be strengthened, (2) the military users of a command control system should actively participate in the design development and the evolution as well as the use of the system.

"We can help to implement both of these recommendations. We can help if we restrict the assumptions we make in our studies to those that are the direct responsibility of the agency that funds our study, i.e., we should not act as though we know the input, requirements or utilities of agencies other than the particular agency and its subordinate agencies, that are directly responsible for our study." Again on the next page, "We can also help by making more use of simulation during our design work. This simulation should be so structured that the contracting agency can participate in the test runs. Moreover, the design group can help the contracting agency make use of other government and military groups, including the alternate users of the system both above and below the level of the system assigned to the agency to participate in the simulation." On the last page, "In summary, I have indicated that the selection of a command control system depends upon the criteria of many groups. Since all of the groups in the evaluation chain must be satisfied if a system is to be adopted by the operating forces, none of the systems accepted will be judged best by any group if they apply different criteria. Moreover, the designer cannot expect to obtain an analytic expression of the real utilities used to evaluate military systems. Despite the lack of an explicit analytic utility expression, we can, by simulating our systems in the design phase and by including other DOD agencies as subjects and experimenters in the simulation of the system, design systems which satisfy the criteria of the various groups in the chain and increase their cooperativeness. Thus, we can satisfy that elusive command control criteria through simulation studies even if we cannot measure it." This is the final paragraph of the paper.

It seems to me plain that although Mr. Becker has spoken at length about the need for reconciling the criteria of various users, he has at no time got specific about any of the criteria, nor about methods for assuring that any criterion is satisfied. I think the paper has no value from our point of view.

Stages in the Design and Development of an Information Processing System,
Lorraine G. Gay, SP-1023, 9 November 1962.

This paper is included because the title suggests that it might contain something about evaluation; in fact, it does not.

The next two papers are treated as a unit because of their common authorship. First is Implementing Computer Systems and Evaluation of Operational Systems, SP-918/002/00, 10 January 1963. The second is Implementing and Evaluating Information Processing Systems, SP-1294, 6 November 1963, both by Robert E. Adamson.

These papers are not identical, but they include essentially the same material, and the following quotations and comments are from the latter one, SP-1294.

On page 2, we read, "Now, the introduction of data processing devices, such as computer, implies that, for one reason or another, the existing system is unsatisfactory....The reasons for adopting automation are expressed in such general terms as 'increased efficiency.' It is true that one should expect greater efficiency of operation, but his expectations should be geared to specific instances if he is to properly assess the potential benefits to be derived from automation. These specific instances furnish the criteria by which a system may be evaluated.

"There is no single set of criteria against which all systems can be measured and which, at the same time, would be sufficient for the evaluation of a given system. Accordingly, although this paper discusses certain general standards, it will emphasize the process of developing particular criteria for individual systems." Developing this idea, Adamson begins, on page 4, a list of questions, "which the system designer must ask himself about his observations of user activity and questions which he must also direct to the user." Among these questions, we have one regarding user objectives, another on operational functions, another on operational tasks, still another on information requirements, and so on, ending with the question "Who uses the information, and for what purposes?"

On page 19, we read, "The operational requirements furnish criteria by which the user evaluates the finished product. They also furnish criteria by which the designer evaluates the design specifications, which in turn furnish the basis for the designer's evaluation of the production effort. There are other more general ways of evaluating systems...one of these is costs.

"There is one final criterion which demands close study by the user: ...is the equipment being used to capacity?"

This is about as specific as Adamson gets in his discussion of performance criteria, and clearly not specific enough to support a quality control activity.

Managing the Development of Complex Systems, M. G. Holmen, W-19760,
5 February 1963.

In the Introduction, Holmen says, "There are at least eight major steps to be performed in managing the development of a complex command and control system. This article is concerned with those steps, the sequence in which they are taken, and the way in which they are arranged."

Further along the eight major steps are described. Step 8 includes system testing, but this paper, again, does not present any method for quality control of program production or for the evaluation of a produced program.

Detailed Evaluation Plan 0100 by Field Design Branch, TM-CN-003/000/00,
30 April 1963 (no author given).

This paper was included because its title suggests that it deals with the evaluation problem. Actually, its scope is quite narrow. It refers to an experiment on system 425L, in which the experimenters sought to determine the effect upon system response time of inputs, additional consoles, and recording programs. The conclusions from the experiment are not very precise: "It would be extremely tenuous to attempt to extrapolate these exact results to changes in recording parameters, increases in the number of consoles and simultaneous actions, or change of inputs....Further investigation is currently being planned along these lines." Since the experimenters did not seem to feel that they had established anything very firmly and since at best the experiment has limited scope, there seems to be no likelihood of further gain from this paper.

The next three papers are treated as a unit because of their common authorship. They are Command and Control Management Decision Making, SP-1174, 20 May 1963; Development of an Operational Management System, SP-1175, 16 April 1963, and Social Value and Technological Design, SP-1506, 6 January 1964, all by R. J. Rhine.

A quotation from SP-1174, on page 18: "A major difficulty of status monitoring is the selection of appropriate data. Busy managers cannot be expected to absorb data describing everything about the company any more than an admiral or a general can be expected to spend night and day before the displays of a command and control system. The data for status monitoring should be carefully screened and selected so that they are of interest to the group which will see them....The danger of status monitoring is the well known frustration caused by masses and masses of

25 March 1965

31

TM-2313/000/00

reports being sent to managers who do not have time to read them." Very nearly the same paragraph occurs in SP-1175, page 8. In SP-1506, page 4, under the heading "Values and Criteria," we read, "The values toward which a technological design is directed are those characteristics of the ensuing product that are considered desirable. A criterion is the degree of the valued characteristic which must be obtained before the product is acceptable." (This is not, I believe, the definition of criterion that is usually accepted.)

N-19789/311/00, G. F. Weinwurm, 22 May 1963.

This short document is a preliminary outline for a chapter with the title "Operational Design." It has a section under the heading "What is a good operational design?" and this, I think, comes close to our subject. Weinwurm lists as good qualities that an operational design should have accuracy, consistency, completeness and logical organization. Probably none of us would quarrel with these objectives, but Weinwurm does not attack the problem of how these qualities are to be measured, and that, I believe, is at the core of the quality control problem.

JOVIAL Compiler/OASIS Maintenance Procedures, W. M. Mineart, TM-WD-78, 27 May 1963.

The opening sentences of this document are, "The purpose of this document is to describe the procedures used by SDC in maintaining the Phase I JOVIAL compiler and OASIS systems at the NAVCOSSACT computer facility. Further, the document provides guidance to the users of these systems in the reporting of discrepancies and malfunctions of the systems and provides for the dissemination of information concerning the disposition of errors." This paper is specific in that it presents definite report forms that have actually been used to report discrepancies, which could, of course, be regarded as quality failures. However, it does not at all address the problem of overall system performance testing, nor of the establishment of performance criteria.

"Prediction of Creativity in a Sample of Research Scientists," Cecil J. Mullins, in IEEE Transactions of Engineering Management, June 1963, page 52.

I will quote the summary which appears at the beginning: "In an attempt to identify test predictors of scientific creativity, two criteria of creativity were used: supervisor's ratings and number of publications. An interest questionnaire, a vocabulary test and nine tests of the Guilford Creativity Battery were administered to 131 research physical scientists. Of 42 test scores derived from the battery, 4 were significantly related to the rating criterion and 7 to the publications criterion. The two criteria were not significantly related to each

other and none of the predictor scores correlated significantly with both criteria. A composite predictor gave promise of increasing effective prediction of the ratings criterion, but not of the publications criterion."

Although this article is not aimed at quality control, it was examined because of the possibility that a breakthrough in measuring scientific creativity would give some clue for methods to measure program productivity or program excellence. As appears from the summary, no such hope can be entertained; we are still unable to predict creativity in research scientists, and in fact, one might even argue whether either supervisor ratings or numbers of publications is a valid measure of creativity. The paper is accordingly laid aside as of no interest to our project.

Is Relevance an Adequate Criterion in Retrieval System Evaluation?

Lauren B. Doyle, SP-1262, 1 July 1963.

This paper was included in the survey because the title suggests that criteria and evaluation might be discussed, as indeed they are; but not in a quantitative way suitable for quality control uses.

An Updated Plan for a Corporate Management System, R. J. Rhine, 24 July 1963.

This document is concerned mainly with describing the various activities that have to be undertaken in developing a specific management system. Some attention is given to timing and use of the PERT system, but nothing here apparently relates to quality control.

Management Standards for Data Processing, Dick H. Brandon, Van Nostrand, August 1963.

The problem of quality control in programming is still far from a solution, and thus one does not look for much help in books, which tend to be a few years behind the current journal articles. Still, Brandon has two chapters labeled "Methods Standards: Programming," and clearly these must be examined, if only because of their title.

The first of the subject chapters (numbered IV in the book, pp. 69-109) deals with the standardization of procedures: flow charting conventions, character writing conventions, and the like. The second chapter has headings such as "Testing and Program Validation" (page 110), "Testing Standards" (page 115), and "Program Change Administration" (page 143); but none of this material seems specific enough for use in quality control. There are also sections on "Performance Standards," which propose such comparisons as "Production Time/Compiling Time" (page 213) or "Average test set-up time/Average test time" (page 220). Some of the

suggestions are concrete enough (see table on page 236) but I get the impression that they were derived from a contemplation of what ought to be, rather than from experience with a working crew.

In any case, the methods described have clearly not been validated by actual trial and iterative modification. One would expect that papers of more recent date would be more valuable for our purposes. I think we have to accept the fact that we cannot get much help from books.

Command Control Software System Development During the Conceptual Program Definition and Acquisition Phases, TM(1)-IX-74/000/000 Draft, 14 August 1963.

This document weighs five pounds three ounces, including cover, and so by dint of sheer size might be supposed to have something for everybody. When it is published in its final form the pages doubtless will be consecutively numbered, but in the draft most pages are not numbered at all. Instead, they bear designators such as D1-50-29 or AS-49-1, and explanations of these designators appear at the beginning of the sections into which the document is divided. At the back of the book is a set of fold-out charts, and these show the designators and their relationship to the general scheme of classification.

Designator C-25 means, Preliminary System Analysis and includes seven numbered subheads, including No. 2, Prepare Evaluation Framework; No. 6, Evaluate Cost-Performance of Alternatives; No. 7, Select Most Promising Configuration and Adjust System Performance Requirement.

We have also C-25A, System Performance and Design Documents.

Under C-25.2 we read, "This marks the first step in a continuing concern of the system design engineer with the problem of design verification and validation.

"Based upon a review of interface consideration, command organization, mission and operational environment information and the structure of a threat, the system design establishes a set of general performance criteria which will be used as a measure of system concept effectivity."

C-25.7 begins with the following paragraph: "With concurrence with the user, the Systems Division will select the most promising configuration for a preliminary system design sub-phase. The configuration is selected as a result of the previous cost/performance evaluation activity."

C-25.6, Evaluate Cost Performance of Alternatives, says, "Cost performance studies involve the determination of the cost of a certain specifiable complex of machines, devices and facilities which will provide the capability of implementing a specific system idea or concept (mission concept or concept of operation). This cost is compared with other

costs developed from studies of other configurations supporting a different concept of operations (system concept) or other configurations supporting the same system concept.

"Included in the items for cost performance analysis is the traditional principle of system configuration; which in command control is the geographic dispersal of men, equipment and facilities.

"We must include in an analysis the software or computer programming element with an implied examination of trade-off between human and automatic information processing operations and decision making.

"For instance, the greater the automatic processing requirement, the greater is the need for equipment and programming capabilities and the less control exercised directly by the human decision maker. From a cost point of view, by reducing the number of decision makers and increasing the size of your electronic magnetic information processing complex, one could be playing operations dollars off against development dollars.

"From an operations point of view, the question is one of control and reliability. Once the automatic information processing complex malfunctions or ceases to work, the entire system comes to a halt. This, in the case of the death or incapacity of a CINC or senior officer, is not true; the proper subordinate takes over the role of decision making and battle management. The designer has to examine and measure the cost/worth value of greater speed and accuracy and capability for the automatic processing complex against its relative vulnerability and lack of strength-in-depth.

"The most promising configuration of men, machines, facilities, automatic information processing complex are selected and prepared for the next design step."

The above quotation includes the whole of the entry under C-25.6.

These quotations appear to be the ones most closely related to the subject of quality control. About all one can say of them is that they concede, at least by implication, that quality control is an important objective. Nothing is said anywhere about methods or concepts of approach to the quality control problem. In fact, about all the document really says is that we have an obligation to deliver a reliable product of good quality. How we distinguish good quality from bad, or how we are to set about making a product reliable remains to be determined; I find nothing in the present document that sheds light on these questions.

Research in the Management of Computer Programs, V. LaBolle, TM-1626/001/00,
4 December 1963.

On page 17, there is the beginning of a section under the heading "Quality Control Techniques." "In this task, we would undertake to clarify, define, and help determine measurements for the quality of computer programs and computer program documentation. A part of this area of work that overlaps the cost work described above would seek to build a classification system for programs in terms of what they do. Such a taxonomy is vital to the cost analysis and is implied by many of the cost factors already identified.

"A deeper investigation of quality would consider: (1) what programs are supposed to do and how they are intended to be used as reflected in requirements and design specifications, (2) what programs actually do as determined by test, exercise and operational use, (3) ways in which desired quality, including performance characteristics, can be expressed unambiguously and preferably quantitatively, and how the products, both documents and programs, can be inspected during each programming activity to insure that the quality standards can be met.

"Clearly, the study of quality can not only contribute to insight into a cost/value relationship in program development but also will point to changes in the methods for performing the programming job."

This document is essentially a research proposal and the section under quality control says in essence that because quality control is an important problem we should be devoting some effort to its solution. It recognizes that operational methods can be developed, if at all, only by long research, and thus we do not find in this document anything more than a recognition of the gravity of the quality problem; there is nothing suggested in the way of specific methods.

Organization Decision Making, Julian Feldman and Herschel E. Kanter, SP-1357,
30 December 1963.

The authors consider the decision-making process as one of selecting a particular path among a number of alternate paths, so as eventually to reach a desired goal. Although the methods proposed are relatively close to the standard approaches of matrix theory, game theory, and operations research generally, yet the presentation here appears to be too abstract for immediate value in the quality control problem. No attention is paid, for example, to the question of selecting among alternate goals; it is assumed, in the Feldman-Kanter treatment, that a goal has been selected.

I think we may set this paper aside.

A Comparative Evaluation of JOVIAL and FORTRAN IV, C. J. Shaw, N-21169,
7 January 1964.

This paper was included for examination because its title suggested that there might be some sort of verifiable criterion used in the comparative evaluation. It turns out that this is not quite the case. In the paper, we have (in parallel columns) discussions of various features of JOVIAL and FORTRAN, and at the conclusion of the paper, statements such as, for example, on page 23, "FORTRAN IV is better for compiling large programs out of many small subroutines.

*FORTRAN IV compilers provide for the manual inclusion, into the programs they process, of previously compiled subroutines," and so on. In short, the two systems are described in qualitative language; there seems to be nothing here that would lend itself to quantitative assessment.

A User's Experience with Three Simulation Languages (GPSS, SIMSCRIPT, and SIMPAC), Karen Young, TM-1755/000/00, 17 February 1964.

This paper was reviewed because its title implies a comparison and comparison in turn implies evaluation. The author considered method of modeling, programming flexibility, outputs provided, programming time required, and programming ability required; but all these quantities were assessed subjectively, and I do not believe they give us much hint for quality control methods.

The Design and Production of Operational Procedures, L. A. Friedman,
N(L)-21357/000/00, 3 March 1964.

On the cover, this is identified as a chapter to be contributed to a book, The Development of Computer-Based Information Systems, sponsored by SDC. Near the end of his chapter, on page 106, of a total of 114 pages, Friedman has a section titled "Testing the Operational Procedures." He makes an interesting point, on page 107, "One of the major and most important reasons for procedure testing is that procedures which are developed from design documents are only nominal, that is, what should be. Experience has always shown that what actually is done in an operational environment does not always resemble what should be done. Thus, delivering the procedures to a potential user without testing them can lead to troubled operations. Hence, a test must be designed to check the validity of the procedures, i.e., do the operational procedures prescribed actually operate and control the computer programs for which they are written?" There is a little more along this line, but here, again, the emphasis is obviously on testing a completed program or perhaps monitoring it over a period of continuous operation; nothing here related to the control of the program production process itself.

Automated Computer Efficiency: the ACE Method for Efficient Computer Programming, M. I. Bolsky and S. L. Feingold, SP-1292/000/01, 5 March 1964.

This document's title suggests that it might be relevant to our problem, but it is in fact not so. On page 1, in the Abstract, we read, "This paper documents ACE, a method for the systemization of the innumerable details involved in digital computer programming and checkout, specifying these details in step-by-step form on a series of checkout charts. This systemization insures that programmer actually perform all of these details, and in correct order. The charts indicate the specific actions to be taken to prevent errors and to track down the causes of errors that do occur."

We have here, then, essentially a procedural or checkout list, whose purpose is to insure that no essential step is omitted. It has, I think, no relevance for our task.

System Programming Management, N. E. Willmorth, TM-1578/000/00, 13 March 1964.

This document is in process of revision; the following quotation on quality control is from a preliminary version (page 178):

"Control of document and program quality is one of the toughest problems facing the programming manager. Making sure that documents are accurate and programs are debugged takes so much of the manager's efforts that other aspects of document and program quality are often ignored.

"Quality control of documentation and programs presents some unique problems. High levels of reliability are demanded, but exacting methods of determining reliability and validity are not well established. Since the products produced are one-of-a-kind items, sampling methods are not applicable. A document or program must pass all tests in order to be acceptable. Fortunately, quality criteria for most programs are concerned with whether or not the programs perform the required function and occasionally with processing speed, but seldom with elegance or optimum efficiency. Once in a while flexibility and modularity are mentioned, but criteria of flexibility and generality are seldom established or enforced.

"On the other hand, program and program testing seems an interminable job. Getting a complex system completely debugged seems an impossible task. Many programming managers would like to establish some means of stopping testing activities short of complete perfection without being called to account for every bug discovered in the future. The determination of some method of establishing of reliability and quality better than debugging for obscure cases is desirable.

It
and
N-2

Stu
Eva

"In addition to product excellence quality control is used for the evaluation and improvement of the processing system. Keeping statistics on the 'scrap rate' of designers, programmers and computers is something that is infrequently done. Computer reruns are sometimes accounted for when the rerun is due to machine, system or operator error, but seldom for program errors. The amount of 'bad code' that is produced and scrapped in the course of producing a program is almost never accounted for, although rates ranging from 50 to 500 per cent have been alleged for particular programs. Very little can be done to improve the quality of performance in program and document production unless such scrap rates and other performance measures are collected and evaluated for normal performance measures.

"Strong control over quality is an asset to successful performance in production planning and control. Most of the products turned out in a program system production phase are intermediate items used by the next phase of the process. Poor quality in items is not only very irksome to the personnel who must deal with them but create errors, inefficiencies and work delays in the later phases. Having to rework a program or a document can disrupt work projections and other work may be stalled awaiting the results of the re-do."

This quotation includes all of this section, and speaks for itself.

It is of some interest to consider the subject paper in connection with another paper by the same author (Managing the Software Development Effort, N-21395/000/00, 12 March 1964).

This chapter is directed mainly at problems of costing and scheduling, but contains some material relevant to the quality problem. For example, on page 18, we find a graph, "Instructions per Man Month as a Function of Program Length." According to this graph, programmer productivity falls off quite sharply when the total number of instructions in a program exceeds about 200,000. The following page has another graph, "Computer Hours Used as a Function of Program Length," which suggests that the number of computer hours required rises almost exponentially as the total number of instructions increases. It seems reasonable to guess that the cause of this increase in computer time, and lower productivity of programmers, arises from the difficulties of maintaining quality in the production and assembly of very large systems; at any rate, the quality problem can certainly not be divorced from the production problem.

Study Proposal for the Development of Measurement Variables for Air Defense Evaluation, R. E. Hillis and M. S. Sheldon, N-21735, 2 July 1964.

This paper is the last of a series of four dealing with proposed measures

for the effectiveness of air defense crews. It is included because such measures might very possibly be the basis of quality control variables. In fact, the measures proposed by Hillis and Sheldon do seem to hold some promise of being useful for quality control, though the exact way in which they would be used is not discussed by the authors.

The proposed measures include such things as detection (yes or no), detection latency (time), false positives (number), detection adequacy (yes or no), maximum tracking error (nautical miles), number of override actions (number), and so on.

It seems perfectly possible that the list proposed by Hillis and Sheldon might be used for reporting quality of performance of a system, though it is not very clear how they could be applied to the measurement of program quality. Nevertheless, as a specific proposal naming definite variables, the paper seems to merit consideration.

The Administration of Research, Joseph Fink, SP-1684, 15 July 1964.

The title suggests that the paper might discuss evaluation of the results of research, and so the paper was examined. No discussion of evaluation is in fact presented, and although the paper is interesting, with a number of cogent comments on how research should be administered, it does not seem to have relevance for our problem.

Trip Report dated 31 July 1964 from F. B. Tierney to J. W. Singleton.

Tierney says, at the end of his second paragraph and the beginning of his third, "Is there a manageable number of variables which can be used in evaluating given computer facilities?...Economic evaluations of computing systems, large and small, began in the areas of strict business applications." On page 4, he continues, "There is, in the persons of both managers and consultants, an awareness of the fluid state of affairs in the measurement of performance concerning large systems. The advent of new and dramatically changed computers will affect the historically utilized comparative measures. It is the purpose of these individuals to monitor these relative and absolute measures and to determine whether or not they are deficient or superfluous or whether rerankings and/or additions are required."

Similar comments cover the first five pages of the memorandum. The next 24 pages are devoted to a rather detailed description of the computer equipment available at SDC Santa Monica; this includes not merely a description of the computers themselves, with such details as memory capacity, cycle time, and special features, but also a listing of peripheral equipment available with each computer. There is, however, no attempt in these pages to set up criteria of value or performance.

Th
no

Cle
Com
002
Cor

Sys
Sep

25 March 1965

40

TM-2313/000/00

The importance of the document lies principally in the fact that it has a bibliography at the end, containing 98 titles, and this list of references would presumably be of considerable use to anyone undertaking a further study of evaluation; however, Tierney at no time mentions program quality or quality control, so we may take it that the documents he lists are not highly specific to our problem, but at best shed a marginal light.

The Results of the ESD/MITRE/SDC Work on Management Control of Software, no author, MITRE TM-3551, August 1964.

This document is included because of its title. It has a section with the heading, "The AFSC Technical Requirements and Standards Program," but does not contain or suggest specific quality standards for software. In fact, it reads in part, "It is specifically recommended that a follow-on group be established to further design and to implement, install, train for and maintain staff surveillance of the process of software subsystem acquisition." It is not very clear that such a group would be concerned with product quality, though on the following page, the writers recognize the relevance of such characteristics as reliability and maintainability, and call for monitoring of tests, acceptance standards and standards application.

The document does not appear to contain any material useful for the present survey, unless we so regard the expressed concern of ESD over standards and testing.

Closely related documents with overlapping authorship are SDC Experience in Computer Program Implementation: Costs and Cost Factors, L. Farr, W(L)-18931/002/00, 30 June 1964, and Cost Aspects of Computer Programming for Command and Control, L. Farr and B. Manus, SP-1372/000/01, 13 January 1964.

These documents all have about the same orientation, but it will probably be sufficient for anyone interested to consult TM-1447/001/00.

System Installation and Testing, Frank W. Hopkins, MITRE Publication SR-123, September 1964.

On page 4, Hopkins says, "Implementation testing should be designed and conducted to answer the following questions:

- Does the system as installed meet the specification?
- Does the system, meeting the specifications, allow the job to be done?
- How well does the system do this job?"

There can be no doubt that these are cogent questions, but Hopkins does not propose any operational method of answering them. On page 9, for

example, he says, "Insure that the information is interpreted and used properly at the receiving site. The best example here is, again, that of data quality. Presumably at the receiving site there are some conditions under which we wish to incorporate cross-told information of good quality tracks, other conditions under which we wish to incorporate cross-told data of poor quality tracks, and, perhaps, some conditions under which we wish to use only part of the information received." Here, the definition of good quality and poor quality seems to be taken pretty much for granted and our real problem is to make these definitions operational.

Hopkins does get a little more specific later on in his paper. On page 18, under the heading "Tests to Evaluate Performance," he says, "Our third general objective is to determine how well the system does the job for which it was bought....Let us assume as a desired criteria for tracking continuity that the track should be within 5 percent of aircraft position 90 percent of time that the aircraft is within radar coverage of the environment, while maintaining the same track number and identification through out. Our measures may actually indicate that for 95 percent of the time the track heading is within 2 percent, the track speed within 10 knots and the track position within $1\frac{1}{2}$ miles of the corresponding parameters of the aircraft. This information is useful to know. It can be obtained as a direct by-product of testing for the second general objective and can be collected repeatedly after system operations begin." It is clear that Hopkins has in mind the reporting of such quantities as the proportion of time that an estimation error exceeds some assigned percentage, and this may be acceptable for system operation, but the proposal leaves undetermined the question whether these particular deviations are in fact the best ones to report, or how we should arrive at critical percentages that would be taken as having special inferential value. The paper, in short, does not offer a specific proposal for the operation of quality control or performance evaluation.

In-Plant Testing of a Computer Program System, J. E. Crnkovich and G. Neil,
N(L)-21707, 7 October 1964.

This deals with parameter testing and points out some of the problems to be solved in connection with parameter testing but does not deal with program production.

In connection with the preceding document, we have M-21585 from H. E. Wilmorth, 19 October 1964.

This Memo is a set of comments on N(L)-21707 and records some differences of opinion with the authors of that Note; however, it, like the Note, is concerned with testing a program after it is produced, and not with any evaluation of the programming process.

25 March 1965

42

TM-2313/000/00

Data Processing Task Items, Dallis K. Perry, Memo M-20963, October 9, 1964.

The first paragraph of this Memo reads as follows, "Attached is a list of data-processing task statements which have been prepared for use in a study aimed at developing better understanding of the data-processing jobs in SDC. Individual programmers will be asked to describe their jobs by sorting the task statements according to the importance of the tasks in their jobs." The detailed list continues on the following pages, and includes a number of steps such as desk checking and parameter testing, but is merely a checklist similar to the one produced by Bolsky; it does not suggest specific criteria.

I think the document is not helpful for discovering quality control procedures.

Quality Control of the BUIC Operational Program: Passive Tracking, H. S. Stone, WTRR Publication W-07051/0104/00/0/00, Confidential (Title Unclassified), 16 October 1964.

The contents of this paper are classified, and so are not reported here. However, they are very specific to the BUIC program, and have no application to quality control in a general sense.

Controlling Time and Cost Factors in Programming, Robert Bohrer, CSC Report Volume 11, No. 5, a publication of Computer Sciences Corporation, December 1964.

This document is general in character; its main purpose seems to be to serve as a sort of sales talk to prospective clients. It describes in qualitative terms some of the things that CSC personnel do in approaching systems problems, and even has a facsimile of a CSC form; but I don't believe the paper has any particular value.

Appendix C

Index by Authors

- Page 29 Adamson, Robert E. Implementing and Evaluating Information Processing Systems, SP-1294, 6 November 1963.
- 29 _____ . Implementing Computer Systems and Evaluation of Operational Systems, SP-918/002/00, 10 January 1963.
- 33 Anonymous. Command Control Software System Development During the Conceptual, Program Definition and Acquisition Phases, TM(L)-LX-74/000/000 Draft, 14 August 1963.
- 22 _____ . Computer Program System Development Milestones, SSD Exhibit 61-47A
- 30 _____ . Detailed Evaluation Plan 0100 by Field Design Branch, TM-CN-003/000/00, 30 April 1963.
- 22 _____ . "How Do You Measure Useful Computer Time?" Automatic Data Processing Digest, Volume 6, No. 5, March 1960.
- 40 _____ . The Results of the ESD/MITRE/SDC Work on Management Control of Software, MITRE TM-3551, August 1964.
- 11 _____ . A Summary of Lessons Learned from Air Force Management Surveys, ABSCP 375-2, 1 June 1963.
- 27 Becker, Gordon M. The Elusive Criteria in Command and Control, SP-198, November 1962.
- 42 Bohrer, Robert. Controlling Time and Cost Factors in Programming, CSC Report Volume 11, No. 5, a publication of Computer Sciences Corporation, December 1964.
- 37 Bolsky, M. I., and S. L. Feingold. Automated Computer Efficiency: The Ace Method for Efficient Computer Programming, SP-1292/000/01, 5 March 1964.
- 32 Brandon, Dick H. Management Standards for Data Processing, August 1963.
- 22 Cannell, M. H. SAGE Program Implementation History No. 2--How to Set Up a Field Site (1957), N-10642, 22 February 1960.
- Claussen, N. See Heinze, K.

- Page 6 Corporate Management System Staff. Interviews with SDC Management,
FW-6860/000/00, 7 September 1962.
- 41 Crnkovich, J. E., and G. Neil. In-Plant Testing of a Computer
Program System, N(L)-21787, 7 October 1964.
- 27 Crawford, J. R. System Simulation for Evaluating Product Quality,
SP-992, 12 October 1962.
- Dovst, H. See Thomson, W. L.
- 32 Doyle, Lauren B. Is Relevance an Adequate Criterion in Retrieval
System Evaluation? SP-1262, 1 July 1963.
- Erickson, W. See Longmire, R.
- 40 Farr, L. SDC Experience in Computer Program Implementation: Costs
and Cost Factors, N(L)-1893/002/00, 30 June 1964.
- 40 _____, and B. Namus. Cost Aspects of Computer Programming for
Command and Control, SP-1372/000/01, 13 January 1964.
- 18 _____, and E. J. Zagorski. Factors that Affect the Cost of
Computer Programming: A Quantitative Analysis, TM-1447/001/00,
31 August 1964
- Feingold, S. L. See Bolsky, M. I.
- 35 Feldman, Julian, and Herschel E. Kanter. Organization Decision
Making, SP-1357, 30 December 1963.
- 39 Fink, Joseph. The Administration of Research, SP-1684,
15 July 1964.
- 36 Friedman, L. A. The Design and Production of Operational
Procedures, N(L)-21357/000/00, 3 March 1964.
- 29 Gay, Lorraine G. Stages in the Design and Development of an
Information Processing System, SP-1023, 9 November 1962.
- 24 Gordon, C. K. On Measuring the Validity and Effectiveness of the
Response of a Military Command Control System, N-18551/000/00,
25 July 1962.
- 27 Green, R. W. Interim Production List of SPCR-1576, N-17264/040/02,
15 October 1962.

- Page 7 Haverty, J. P., and R. L. Patrick. Programming Languages and Standardization in Command and Control, RAND Document RM-3447-PR, January 1963.
- 8 Heinze, K., N. Claussen, and V. LaBolle. Management of Computer Programming for Command and Control Systems: A Survey, TM-903/000/02, 8 May 1963.
- 38 Hillis, R. E., and M. S. Sheldon. Study Proposal for the Development of Measurement Variables for Air Defense Evaluation, N-21735, 2 July 1964.
- 30 Holmen, M. G. Managing the Development of Complex Systems, N-19760, 5 February 1963.
- 40 Hopkins, Frank W. System Installation and Testing, MITRE Publication SR-123, September 1964.
- 25 Jaffe, Jack. Information Processing System Design: General Principles, Part I, TM-743/001/00, 9 August 1962.
- Kanter, Herschel E. See Feldman, Julian.
- 8 LaBolle, V. Management Aspects of Computer Programming for Command and Control Systems, SP-1000/000/02, 5 February 1963.
- 8 _____. Quality Control in Computer Program Development.
- 35 _____. Research in the Management of Computer Programs, TM-1626/001/00, 4 December 1963.
- _____. See Heinze K.
- 27 Longuire, R. M., and W. J. Erickson. Operational Measures of Information Characteristics, SP-930, 1 September 1962.
- 25 Masters, George M. Principles of Information Processing System Design: Part II, TM-743/002/00, 19 February 1963.
- 15 McIsaac, P. V. Initial Insights Toward System Quality Control, N-LX-621/100/00, 15 July 1964.
- 14 _____, and F. D. O'Connor. An Approach Toward Quality Control, N-LX-621/000/00, 4 June 1964.
- 16 _____, and F. D. O'Connor. Model of a System Catalog for Quality Control, N-LX-(L)-621/200/00, 5 August 1964.

- Page 31 Mineart, W. M. JOVIAL Compiler/OASIS Maintenance Procedures,
TM-WD-78, 27 May 1963.
- 31 Mullins, Cecil J. "Prediction of Creativity in a Sample of
Research Scientists, IEEE Transactions of Engineering
Management, June 1963.
- Manus, B. See Farr, L., 13 January 1964.
- Neil, G. See Crnkovich, J. E.
- O'Connor, F. D. See McIsaac, P. V., 4 June 1964.
- O'Connor, F. D. See McIsaac, P. V., 5 August 1964.
- 20 Patrick, R. L. Computer Status Quo, SP-1947, 15 December 1964.
- 18 _____ . A Technique for Improving the Management of a Computer
Installation, RAND Memorandum RM-4232-PR, September 1964.
- _____ . See Haverty, J. P.
- 42 Perry, Dallis K. Data Processing Task Items, Memo M-20963,
October 9, 1964.
- 19 _____ . Training and Job Performance Validities of Programmer
Trainee Selection Variables, TM-2172, 9 December 1964.
- 19 _____ . Validity of Programmer Trainee Selection Variables,
TM-2173, 9 December 1964.
- 30 Rhine, R. J. Command and Control Management Decision Making,
SP-1174, 20 May 1963.
- 30 _____ . Development of an Operational Management System,
SP-1175, 16 April 1963.
- 30 _____ . Social Value and Technological Design, SP-1506,
6 January 1964.
- 32 _____ . An Updated Plan for a Corporate Management System,
24 July 1963.
- 36 Shaw, C. J. A Comparative Evaluation of JOVIAL and FORTRAN IV,
M-21169, 7 January 1964.
- Sheldon, M. S. See Hillis, R. E.

- Page 23 Snyder, T. J. On Quality Control, N-18476, 22 June 1962.
- 42 Stone, H. S. Quality Control of the EUTC Operational Program: Passive Tracking, MITRE Publication W-07051/0104/00/0/00, Confidential (Title Unclassified), 16 October 1964.
- 25 Thomson, W. L., and H. P. Dowd. SPCR-1576--Recommendation, Real Time Quality Control, FN-5263/576/00, 24 August 1962.
- 39 Tierney, F. B. Trip Report, 31 July 1964.
- 21 Tucker, A. E. The Correlation of Computer Programming Quality with Testing Effort, TM-2219/000/00, 26 January 1965.
- 31 Weinwurm, G. F. Preliminary Outline for Chapter 10, Operational Design, N-19789/311/00, 22 May 1963.
- 41 Willmorth, H. E. Comments Regarding N(L)-21787, "In-Plant Testing of a Computer Program System", M-22585, 19 October 1964.
- 11 _____ . Program System Testing, N-19789/370/00, 28 April 1964.
- 37 _____ . System Programming Management, TM-1578/000/00, 13 March 1964.
- 36 Young, Karen. A User's Experience with Three Simulation Languages (GPSS, SIMSCRIPT, and SIMPAC), TM-1755/000/00, 17 February 1964.
- Zagorski, H. J. See Farr, L., 31 August 1964.

Unclassified

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) System Development Corporation Santa Monica, California		2a. REPORT SECURITY CLASSIFICATION Unclassified	
		2b. GROUP	
3. REPORT TITLE Quality Control for Systems and Programming a Survey of the Literature			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates)			
5. AUTHOR(S) (Last name, first name, initial) Peach, Paul			
6. REPORT DATE 25 March 1965		7a. TOTAL NO. OF PAGES 47	7b. NO. OF REFS
8a. CONTRACT OR GRANT NO. U.S. Government Contracts		8a. ORIGINATOR'S REPORT NUMBER(S) TM-2313/000/00	
8. PROJECT NO. c. d.		8b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
10. AVAILABILITY/LIMITATION NOTICES Distribution of this document is unlimited			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY	
13. ABSTRACT This report, prepared as an aid to the Quality Control (QC) Project, summarizes the contents of some 60 documents that were thought to have possible bearing on QC for computer systems, especially programming. Sixteen of the documents are selected as worthy of further study, either for valuable insights into the QC problem, or because they contain concrete suggestions or experimental data. A synthesis of the sixteen is planned for a future paper.			

