

SOFTWARE FORMAL INSPECTIONS GUIDEBOOK

National Aeronautics and
Space Administration
Washington, DC 20546

Approved: August 1993

FOREWORD

The Software Formal Inspections Guidebook is designed to support the inspection process of software developed by and for NASA. This document provides information on how to implement a recommended and proven method for conducting formal inspections of NASA software.

This Guidebook is a companion document to NASA Standard 2202-93, Software Formal Inspections Standard, approved April 1993, which provides the rules, procedures, and specific requirements for conducting software formal inspections. Application of the Formal Inspections Standard is optional to NASA program or project management. In cases where program or project management decide to use the formal inspections method, this Guidebook provides additional information on how to establish and implement the process.

The goal of the formal inspections process as documented in the above-mentioned Standard and this Guidebook is to provide a framework and model for an inspection process that will enable the detection and elimination of defects as early as possible in the software life cycle. An ancillary aspect of the formal inspection process incorporates the collection and analysis of inspection data to effect continual improvement in the inspection process and the quality of the software subjected to the process.

The software formal inspection process supports NASA Management Instruction (NMI) 2410.10B, NASA Software Management, Assurance, and Engineering Policy, effective April 20, 1993.

This document has been prepared under the direction of the personnel shown on the following page. Specific questions concerning this publication should be referred to one of them, while general questions should be referred to the Office of Safety and Mission Assurance, NASA Headquarters, Washington, D.C. 20546.

~

Original signed by

Charles W. Mertz
Acting Associate Administrator for Safety and Mission Assurance

I. GENERAL

Software formal inspections are in-process technical reviews of a product of the software life cycle conducted for the purpose of finding and eliminating defects. Formal inspections may be applied to any product or partial product of the software development process, including requirements, design, and code. Formal inspections are embedded in the process of developing products and are done in the early stages of each product's development.

Formal inspections were developed by Michael Fagan at IBM to improve software quality and increase programmer productivity. As such, the formal inspection process involves the interaction of the following five elements:

- o Well-defined inspection steps
- o Well-defined roles for participants (moderator, recorder, reader author, inspector)
- o Formal collection of process and product data
- o The product being inspected
- o A supporting infrastructure

The formal inspection process was designed to help the developing organization produce a better product. The process also provides other advantages. As defects are found and fixed the quality of the product increases. A more technically correct base is available for the each new phase of development. The overall software life cycle cost is lower since defects are found early and are easier and less expensive to fix. The effectiveness of the test activity is increased and less time may have to be devoted to testing the product. Another important benefit of formal inspections is the immediate evaluation and feedback to the author from his/her peers which will bring about improvements in the quality of future products.

This guidebook describes the formal inspection process, shows where formal inspections fit during each life cycle phase, and provides some suggestions on how a formal inspection program may be started and improved. It is intended as a tutorial introduction to the inspection process. The Software Formal Inspection Process Standard, NASA-STD-2202-93 should be cited and used where formal definition of the process is needed, such as in process requirements and contracts.

II. CONCEPTS AND DEFINITIONS

A software formal inspection is a defect detection, defect elimination, and correction verification process carried out by a small group during the development life cycle. A defect is any error, nonconformance, or failure to satisfy a requirement in a product. The goal of formal inspections is to ensure that defects

are fixed early in the life cycle rather than late, when they are harder to find and more costly to fix. Formal inspections are held throughout the software development life cycle phases to correct the products and to provide a short feedback loop for authors.

Formal inspections are distinguished from other types of peer reviews in that they follow a well-defined, tried, and proven process for evaluating, in detail, the actual product or partial product with the intent of finding defects (see Section III). They are conducted by individuals from development, test, and assurance, and may include users. Formal inspections are more rigorous and well-defined than other peer review processes such as walkthroughs, and significantly more effective. Inspections do not take the place of milestone reviews, status reviews, or testing.

Table 1 provides a list of candidate work products for formal inspections. The first category, Typical Work Products, indicates the most likely candidates for inspections. The designations in parentheses are commonly used by Fagan and others and are included for reference. The second category in Table 1, Additional Candidate Work Products, shows that virtually any product that has requirements, constraints, and guidelines can be examined by using the formal inspection process.

Inspections should be used to judge the quality of the software work product, not the quality of the people who create the product. For this reason, managers should neither participate in nor attend the inspection meeting. In addition, the results of inspections should be presented to management either statistically or as results for groups of products. This grouping of results will show managers the value of the inspection process without singling out any author. Using the inspection process to judge the capability of authors may result in less than honest and thorough results; that is, co-workers may be reluctant to identify defects if finding them will result in a poor performance review for a colleague.

Formal inspections are performed by a team that may be comprised of a moderator, reader, recorder, author, and other inspectors. The actual team size should consist of four to seven persons, although a minimum of three is acceptable. Each team member has a specific, defined role. In addition, it is the responsibility of the entire inspection team to find and report defects; therefore, all members of the team are considered inspectors. The moderator's primary responsibility is to accomplish a good inspection. In addition, the moderator selects team members, prepares the team for the inspection, conducts the inspection meeting, and reports on the results. The reader's responsibility is to guide the team through the work product. The recorder's responsibility is to accurately report each defect during the inspection meeting. The author's responsibility is to answer inspectors' questions and, after the inspection meeting, correct found defects. In addition, support may be provided to the inspection process by the project library, which may assist the moderator in keeping the status and statistics of formal

inspections. Refer to Section IV for more extensive information on each role.

Entrance and exit criteria are used to determine if a product is ready to be inspected (entrance) and has successfully completed the process (exit). Entrance criteria to be satisfied depend on the product, but generally involve a determination that the product is mature enough to be used as is after defects are removed. For example, entrance criteria for a code inspection are usually that the code has been compiled successfully and run through any static analyzers used by the project, such as an automated standards checker. It should not, however, have been tested. Exit criteria are used mainly to assure that major defects found during the inspection process have been corrected. Correction of minor defects, those that will have no impact on the use of the product, may not be required by exit criteria. All work on the product to be inspected should cease once it has been submitted for inspection, otherwise the purpose of the inspection process will be defeated. Thus, the inspection must be scheduled in a timely manner.

The formal inspection process and its results are supported and documented by a set of forms. Some of the forms that may be used are: Inspection Announcement, completed by the moderator, that notifies participants of the inspection date, time, location, and other important information; Preparation Logs, completed by each inspector, that list defects found and time spent in preparation; and an Inspection Defect List, completed by the recorder, to provide information on each defect. In addition, the moderator should complete a Detailed Inspection Report at the end of each inspection. Other useful forms include checklists that provide guidance for inspectors in finding possible defects, and the Inspection Summary Report that summarizes data found during the inspection. Refer to Appendix A for sample checklists, and Appendix B for sample inspection forms.

III. THE FORMAL INSPECTION PROCESS

In order to use the formal inspection technique to its fullest extent, i.e., to increase product quality while maintaining cost effectiveness, it is important to follow the procedures that have been tested and refined. A formal inspection is accomplished through a series of steps or stages using a trained inspection team. Team members include the moderator, reader, recorder, author, and other inspectors. Figure 1 is a chart of the formal inspection process and team members.

The seven stages that comprise the inspection process are described briefly below. In addition, if reinspection is required, several stages may be repeated if a high number of defects were found during the inspection or if defects required complex corrections. Determination of the need for reinspection is done by the inspection team at the end of the inspection meeting.

Planning	The period of time used to determine whether the product to be inspected meets the entry criteria, plan the inspection, select the
----------	--

team, assign roles, schedule the inspection meeting, and prepare and distribute the inspection forms materials. In addition, a determination is made on whether to hold an overview.

Overview	The optional stage in which the inspection team is provided with background information for the inspection. This stage may not be necessary if the team is already familiar with the product being inspected.
Preparation	Key stage in which members of the inspection team prepare individually for the inspection by reviewing and finding potential defects in the product being inspected. Potential defects found by individuals are discussed during the actual inspection meeting.
Inspection Meeting	Meeting where team members as a group review the product to find, categorize, and record, but not resolve, defects.
Third Hour	Optional additional time, apart from the inspection meeting, that can be used for discussion, possible solutions, or closure of open issues raised in the inspection meeting.
Rework	Stage when the author corrects defects found during the inspection.
Follow-up	Short meeting between the moderator and author to determine if defects found during the inspection meeting have been corrected and to ensure that no additional defects have been introduced.

Each stage of the formal inspection process is addressed more completely in the following sections. The responsibilities of each member of the inspection team are described in Section IV.

A.Planning

Activities in the planning stage are performed primarily by the moderator, who assures that the product is ready for inspection, selects the inspection team and assigns roles, schedules the meeting place and time, and assures the distribution of inspection materials. Inspection materials include the product to be inspected, the inspection announcement, the preparation log, background information, and checklists.

Entrance criteria are used to screen out products that are not ready for inspection. For example, the entrance criteria for inspection of code should require that a clean compilation has been achieved. Entrance criteria should also specify that available automatic tool checking (using such tools as static analyzers, spell checkers, CASE tools, compilers, etc.) should be performed prior to distributing material to the inspection team.

The moderator determines whether the product to be inspected is of reasonable size so that the inspection can be completed in one meeting. If not, the moderator divides the product into manageable pieces. Sample size criteria are given in Table 2. The moderator then selects members for the inspection team and assigns them roles (See Section IV). An inspection package containing the product to be inspected, checklists, references to relevant higher level documents, and blank preparation logs should be distributed to all inspection team members. Sample checklists and inspection forms are provided in Appendix A and Appendix B, respectively.

The moderator also must decide whether the inspection team members are adequately familiar with the material to be inspected or whether an overview must be held. The team should know the background material from which the product was derived (e.g., design and requirements for source code). The team also should know how the material fits into the overall system being developed. In most cases, team members will be adequately familiar from their own work on the project. If they are not, an overview by the author should be scheduled.

If there is a project library, it may support the moderator during the planning stage by keeping records of the items to be inspected and those that have been completed. Once the moderator selects the inspection team, the library may provide each inspector with the inspection package. The project library can provide copies of reference and backup material to the inspection team. If there is no project library, or if the library cannot provide the needed support, the moderator must perform these functions.

B. Overview

The overview is an optional stage that is scheduled if the inspection team is not familiar with the material being inspected and its background. At the overview, the author presents the rationale for the product, its relationship to the rest of the products being developed, its function and intended use, and the approach used in developing it. This information is necessary to the inspection team so that it can perform a successful inspection. Interested persons aside from the inspection team may be invited to attend; however, all team members must attend.

An overview is scheduled if one or more of the following circumstances apply:

- o The inspection team is not familiar with the product.
- o The product is new or is being inspected for the first time.
- o Inspections are new to the project.
- o Novel techniques have been used in the product.

C. Preparation

Preparation is the key part of the inspection process. During this stage, inspection team members individually prepare for their roles in the inspection meeting. Each inspector reviews the product line by line. The inspector looks at the product for general problems as well as those related to his/her specific area of expertise. Checklists should be used during this stage for guidance on typical types of defects to be found in the type of product being inspected. In addition, the product being inspected is checked against higher level work products, standards, and interface documents to assure compliance and correctness. While familiarizing themselves with the product, inspectors record on the preparation log the defects found and the time spent during preparation. Completed preparation logs are submitted to the moderator prior to the inspection meeting. Sample checklists are shown in Appendix A; sample inspection forms are included in Appendix B.

Prior to the inspection meeting, the moderator reviews the logs submitted by each inspector to determine whether the team is adequately prepared. The moderator also checks for trouble spots that may need extra attention during the inspection, common defects that can be categorized quickly, and areas of major concern. If the logs indicate that the team is not adequately prepared, the moderator should reschedule the inspection meeting, as a team not fully prepared will waste time and will not be effective in finding defects. Preparation Log forms are returned to the inspectors at the inspection meeting for their use in pointing out defects.

D. Inspection Meeting

During the inspection meeting, the inspectors review the product as a team. Again, the focus of the meeting is finding defects. Briefly, the activities of the inspection meeting are: the reader provides a logical reading and interpretation of the product, the author provides clarifying information as needed, and the team identifies defects that are classified and recorded.

The moderator calls the meeting to order and notes the product being inspected. If the team is new or contains new members, the moderator may begin the inspection meeting by introducing team members, briefly describing their roles, and restating the purpose of the inspection and the product.

The reader then begins a logical and orderly interpretation of the product. This description should note the function of items (e.g., paragraphs, code blocks) and their relationship to the product and higher level documents. The inspection meeting is structured so that any inspector may interrupt the reader at any time when an item containing a possible defect is read. If a short discussion of the possible defect is needed, the reading is halted temporarily. The reading is resumed when the defect is noted and categorized by the recorder on the inspection defect list (see Appendix B). The moderator should try to limit discussions that appear to be consuming too much of the inspection meeting time. Imposing a time limit on discussions may be useful. If discussions are not completed within the time limit, the moderator will declare the issue unresolved and proceed with the

meeting. The recorder should note unresolved issues as open items to be resolved during the third hour (see section E.).

The team should reach a consensus on whether each possible defect raised is an actual defect. Sometimes, what seems to be a potential defect may be a mistake on the part of the inspector or is only a misunderstanding that may be clarified by the author. If consensus cannot be reached, the potential defect should be recorded as an open issue to be dispositioned during the third hour. This ensures the right of every inspector to have each possible defect recorded and resolved. The recorder will itemize each agreed upon defect by recording its location, a brief description, its classification, and the inspector who found it.

The question of whether a potential defect is a real defect may be resolved by reference to parent documents, which should be available to the inspectors. If the discussion identifies the parent document as potentially in error, an open issue should be noted and the inspection should continue. Resolution of the issue (whether the defect is in the product being inspected or in the parent document) can be done during the third hour. Open issues are logged on the Inspection Report form.

To determine the priority for fixing the defects in the product, the inspection team or moderator should classify them by severity (major or minor). For example, a defect that would cause the system to fail to satisfy a requirement would be classified as major; all others would be classified as minor (e.g., typographical errors, minor standards violation,). Additional data collected from each inspection is the classification of defects by type such as data error, requirements compliance, standards compliance, logic, interface, data, performance, and readability. If using the Inspection Defect List sample form in Appendix B, this information goes in the "Type" field for each defect.

At the end of the meeting, the number of defects are summarized, and the moderator and/or the team determines whether reinspection will be needed. At this time, the moderator also determines whether a third-hour activity is needed. If a third hour is needed, action items are assigned to individual inspectors at this time.

The team must focus on finding defects and should not be concerned with other activities such as problem solving. It is the responsibility of the moderator to control and focus the meeting. To avoid fatigue and thus missing defects, the inspection meeting should be limited to 2 hours. If the inspection is not completed in the original meeting, a second meeting must be scheduled.

After the meeting, the author and moderator meet briefly to estimate rework time and schedule the follow-up meeting. The author is given a copy of the defect list for reference during rework. Note that formal discrepancy reports (DRs) and change requests (CRs) are not written against defects found in the document or code being inspected; however, discrepancy reports

should be written against any defects found in higher level documents. The reason that DRs and CRs are not required is that inspections take place before the product is under configuration control and that closure is assured by the defect list, the follow up stage, and by reinspections.

E.Third Hour

The third hour is time that is used for discussion or for closing open issues. A third hour should be scheduled when the author wishes to discuss corrections of defects found, or when open issues, such as a potential major defects in parent (higher level reference) documents, need to be dispositioned. The third hour may take the form of an additional meeting or of time for team members to perform and report on action items. It does not have to take place immediately after the inspection meeting and it does not have to be limited to 1 hour.

When used as meeting time, the third hour provides an opportunity to discuss solutions and resolve disagreements. Attendees may include any subset or superset of the inspection team including relevant managers (present for technical reasons only), outside technical experts, and other people who could be affected by the way the defect is fixed or the issue resolved. In many cases, only those inspectors who have a specific interest need attend. During a third hour meeting, the author is provided with information that would make rework more effective, major defects found in parent documents are reported, and any open issues remaining from the inspection are resolved.

When used as an opportunity for individual inspectors to perform action items, the third hour usually is spent researching and dispositioning open issues, finding information to resolve a discrepancy, or writing discrepancy reports or change requests for major defects found in parent documents under configuration management.

F.Rework

The purpose of rework is to correct defects found during the inspection. The author is responsible for correcting all major defects noted in the inspection defect list. Minor defects should be resolved if cost and schedule permit. The moderator should make sure that information generated from any open issues or action items are communicated to and addressed by the author .

G.Reinspection

Reinspection may be required when there are a large number of defects in the product or when one or more defects require extensive or complicated corrections. Reinspection allows the changes to the product to be reviewed by the entire team instead of just the moderator. The moderator and the team decide the necessity for reinspection at the end of the inspection meeting.

H.Follow-Up

Follow-up is a short meeting between the moderator and the author to ensure that all major defects found during the inspection have been corrected and no secondary defects have been introduced. The author reviews the measures taken to correct each major defect with the moderator. The moderator ensures that all open issues have been resolved and that changes due to the resolution, if any, have been incorporated in the product. Corrections to minor defects also may be reviewed, but with less emphasis. The moderator ensures that the exit criteria for the type of inspection have been met. The moderator may include additional reviewers in the follow-up meeting if extra technical expertise is required.

If all of the major defects have been corrected, all open issues have been resolved, and the product has satisfied the exit criteria, then the moderator "Passes" the product by recording the completion of inspection on the Inspection Summary Report. If these conditions are not met, the author returns to the rework stage to make the necessary changes.

IV. THE INSPECTION TEAM

The inspection team is a small group of peer staff members with a vested interest in the product. The minimum team size is three persons, although a typical team varies from four to seven. Larger teams are generally used for high level documents, while smaller teams are used at detailed technical levels. Members are added when their point of view is needed. A good mix of inspectors representing various areas of expertise is important to the inspection process. The knowledge and experience of such a group, each looking at the product from his/her own perspective, helps bring many subtle defects to light. "Synergy," where an idea from one team member often leads to another idea from a different team member, is one indication of a healthy inspection process. The role of each individual is explained in the following sections.

A. Inspectors

Every member of the inspection team is considered an inspector in addition to any other assigned role. Inspectors are responsible for finding defects in the work during preparation and during the inspection meeting. In addition to functioning as an inspector, some members of the inspection team will carry out roles as moderator, author, reader, and recorder, as appropriate.

Primary candidates for inspectors are personnel involved with the product in immediately preceding, current, and following life cycle phases. For example, in a design inspection, good inspectors may be selected from the individuals who wrote the requirements, people who will code the design, and designers of interfacing segments of the system. An exception to this rule is that the author of a unit of code should not serve as an inspector for test procedures that are to be used to test that code, because the code author may want changes made to the test procedures so they will work with the code as it exists. Any group with an interest in the product should be considered for potential team members including Systems Engineering, Testing,

Software Assurance, Systems Administrators and Operators, and Users. Sources for inspectors are not limited to the staff of the software development organization. Outside inspectors should be brought in when they have a particular expertise that would add to the effectiveness of the inspection.

B.Moderator

The moderator leads the inspection team and is responsible for ensuring that a good inspection is achieved. Because this role is critical to the formal inspection process, training for moderators is more important and extensive than that of other inspectors. The moderator is directly active in all stages of the inspection process except rework. Since acting as a moderator is time consuming and requires specific skills, moderators often are selected and trained by the development organization and then assigned to a specific development project. Primary duties of the moderator include coordinating the selection of the inspection team, assigning team roles, and leading the team throughout the process. A major function of the moderator is to ensure that the team keeps its emotions in check and that the inspection meeting is not used to find faults with the author. The moderator is also responsible for assuring inspection data are collected on the inspection report forms.

C.Reader

The reader is responsible for guiding the inspection team through the product during the inspection meeting by reading or paraphrasing the product. An individual who will use the product during the next life cycle phase is an excellent candidate for reader as the process of reading and paraphrasing it will cause this potential user to become very familiar with the product before it is delivered. The reader also performs the duties of a regular inspector.

D.Recorder

The recorder is responsible for accurately recording information during the meeting about each defect found on the inspection defect list. The list should include the location of the defect, a brief description of it, its classification, and the identity of the inspector who found it. The recorder also must record information on any issues raised and not resolved, and any defects that are found in parent documents. The recorder also performs the duties of a regular inspector.

E.Author

The author is the originator of the product being inspected. As such, she/he is primarily responsible for providing information on the product being inspected and for answering inspectors' questions to ensure that simple misunderstandings are not classified as defects. In addition, the author should also perform the duties of a regular inspector. The author is required to correct all major defects found during the formal inspection, and minor defects as time and resources allow.

V.FORMAL INSPECTIONS DURING THE SOFTWARE LIFE CYCLE

Formal inspections are in-process peer reviews conducted within the phase of the life cycle in which the product is developed. The following describes the life cycle phases of software development and suggests products that may be inspected during each phase. The software life cycle used is the NASA standard waterfall model; the material may be adapted to other life cycles if needed.

A. Software Concept and Initiation Phase

During this phase, the software concept is developed, the feasibility of the software system is evaluated, and the acquisition strategy is developed. Much of the documentation for the project is started or a draft provided within this phase. The most important document to inspect is the portion of the system requirements document that applies to software. This inspection has the shorthand designation of R0 (see table 1). Other candidates for inspection include system specifications and plans such as the software management and assurance plans. Potential inspectors are the users and assurers of this documentation and the system to be developed.

B. Software Requirements Phase

During this phase, the software concept and allocated system requirements are analyzed and documented as detailed software requirements. Test planning is started, with a preliminary method for verifying each requirement identified and included in a first version of a test plan. Risks are identified and planned for, and the size and scope of the remainder of the project is re-estimated. Methods, standards, and procedures are detailed and put in place.

The requirements document is the product that is most typically inspected in this phase. This is known as the R1 inspection. The requirements document should be inspected for completeness and accuracy, for traceability to higher level documents, and to assure that a sufficient base is provided for the software design. Other documents that are produced during this life cycle phase, such as the draft acceptance test plan, can also be inspected (Inspection IT1). Candidates for inspectors include the assurers and potential users of the documents, including designers, coders, and testers.

C. Software Architectural (Preliminary) Design Phase

During this phase, the overall design for the software is developed, allocating all of the requirements to software components. The architectural design inspection is designated I0.

The design should be inspected for satisfaction of and traceability to the requirements, correctness, clarity, codability, testability, and consistency.

D. Software Detailed Design Phase

During this phase, the architectural design is expanded to the unit level. Interface control documents are completed and test plans updated. Constraints and system resource limits are re-estimated and analyzed, and staffing and test resources are validated.

The detailed design should follow exactly the base-lined higher level design, and should be inspected for the same characteristics. As a secondary condition, the design should be inspected for satisfaction of software quality engineering standards such as information hiding, use of simple structures, etc. The detailed design inspection is designated I1.

Candidates for the inspection team may be selected from participants in the design, code, and test phases.

E. Software Implementation Phase

During this phase, the software is coded and unit tested. All documentation is produced in quasi-final form, including internal code documentation.

Code and all new documentation are the candidates for inspections during this phase. Code inspections (designated I2) should check for technical accuracy and completeness of the code, verify that it implements the planned design, and ensure good coding practices and standards are used. Code inspections should be done after the code has been compiled and all syntax errors removed, but before it has been unit tested. Other candidates are the integration and test plan and procedures, and other documents that have been produced. Documents should be inspected for accuracy, completeness, and traceability to higher level documents. The inspection team may be selected from participants in the detailed design, code, test, verification and validation, or from software quality assurance.

F. Software Integration and Test Phase

During this phase the software units are integrated into a completed system; nonconformances are detected, documented, and corrected; and it is demonstrated that the system meets its requirements. The integration and test plan is executed, the software documentation is updated and completed, and the products are finalized for delivery.

The final version of the Acceptance Test Plan should be inspected to detect defects in the definition of test cases and to verify that each test case will verify the requirements with which it is associated. This is the IT1 inspection. Test case and test procedure inspections should verify that they are in accord with one another and with the Acceptance Test Plan. These inspections should verify that the test cases and procedures will execute properly and correctly, and that all needed data are available. Potential inspectors are representatives from any of the life cycle phases before or after this one.

While the products listed above are used in the Integration and Test Phase, they may have been developed in prior phases. Inspections should be integrated into the development process, and these products inspected when they are developed. If so, few or no inspections may actually be done during this phase; inspections are needed only if new test cases and procedures are developed.

G. Software Acceptance and Delivery Phase

The formal acceptance procedure is carried out during the acceptance and delivery phase. As a minimum, there is a requirements-driven demonstration of the software to show that it meets its requirements. The phase also may include acquirer tests, field usage, or other arrangements that are intended to assure that the software will function correctly in its intended environment.

There is little or no inspection activity during this phase.

H. Software Sustaining Engineering and Operations Phase

During this phase, the software is used to achieve the objectives for which it was developed and acquired. Corrections and modifications are made to the software to sustain its operational capabilities and to upgrade its capacity to support its users. Software changes may range in scope from simple corrective action up to major modifications that require a full life cycle process.

Formal inspections should be scheduled in response to the degree of new development activity involved. Significant new material to be incorporated into any product should be inspected. A useful technique is to have the need for inspections evaluated as part of the change control and configuration management process.

VI. STARTING A FORMAL INSPECTIONS PROGRAM

Formal inspections have been proven to be effective in detecting and removing defects, and to be cost effective when compared to the cost of finding defects by testing. However, they represent an up-front cost and a diversion from more traditional uses of software development resources. Since there may well be skepticism about inspections, beginning a program may find some resistance. Educating management in the advantages of formal inspections, particularly stressing how the up-front costs are likely to be made up by reduced testing costs may help to overcome the skepticism.

A critical first step in initiating an inspection program is to select the class of material to first be inspected. It is advisable to begin with material in which everyone will agree that defects have to be found and removed. Based on their own experiences in starting inspection programs, both JPL and LaRC recommend starting with requirements inspections, as the benefits of formal inspections are shown early in the software life cycle.

Defects in requirements also have more impact than those in other products. For example, for each defect found and corrected in the

requirements, many "defects" would have been present in the design and code. If these resultant defects were not found until testing, they would cause a great deal of rework to many products. Such early results will be popular with management, and should raise enthusiasm for starting the program. Defects in requirements, especially, and in design, are more expensive to correct after the system has been implemented than are code defects.

NASA experience has shown that inspection of requirements and design will significantly reduce code "errors;" some projects conduct formal inspections of all of their requirements and design, but only inspect critical code. Although code inspections may be the easiest type of inspections to perform, they may not be the most productive.

Once a starting point for inspections has been selected, needed resources must be budgeted and roles and responsibilities decided upon. Resources will be needed for start-up costs, overhead costs, and operational costs. Start-up costs include training of moderators and other inspectors, development of forms and report formats, and acquisition of data processing resources for the recording and trending of inspection data. Overhead costs associated with the formal inspection program consist of moderator time for arranging and scheduling inspections and follow-up, and the moderator's or project librarian's role in making copies of materials available and keeping track of the status of items as they progress through the inspection process. In addition, while the collection and trending of data is important, it will consume some resources. Operational costs consist of the time spent by project members preparing for and participating in inspection meetings.

The chief moderator is key to the whole formal inspection program and should be selected as part of the start-up. The chief moderator oversees and directs the inspection program; analyzes the effectiveness of the inspection process; and coordinates the evolution of the program, forms, and checklists. The chief moderator should be the moderator for the inspections on the initial project. When sufficient inspectors have been trained and become experienced, additional moderators may be selected from this pool and trained for new projects to which formal inspections are to be applied.

If it is possible to arrange, project libraries can make both the start-up and the inspection process run more smoothly; this support will allow the moderator to concentrate on the success of the inspection program. The project library helps the moderator to maintain lists of what is to be inspected and assists with some of the mechanics of the inspection process such as delivering materials, setting up meeting rooms, etc. The library should provide reference documentation for the inspectors.

Checklists to guide the inspectors may be developed from the samples provided in Appendix A, obtained from JPL, or developed specifically according to the needs of the program. In the long run, checklists will be needed for each type of material to be

inspected and each major language used. For example, there should be checklists for requirements, design, Ada code, C code, user's guides, etc. At the start-up, only checklists for the limited items to be inspected are needed.

Forms to record results and collect inspection data should be defined. Example forms are shown in Appendix B. They should be tailored as needed to reflect working conditions and to capture the specific data desired. The standard forms will evolve over time.

Once the centralized resources are in place (e.g., chief moderator, librarian, forms, checklists, and data processing resources), project individuals who are to participate in the inspection process must be identified. Project technical people are the key to the program since they are the readers, inspectors, and the ones who have to use the inspection results to improve the product. Once participants have been identified, they should be trained. JPL has developed a formal inspection training class for NASA that is workshop-oriented and very effective. Alternatively, outside organizations have inspection training available.

Projects introducing inspections must plan to accommodate them. In addition to identifying inspectors for training, managers should plan time in schedules for inspections, analysis, and rework. Experience shows that the resources used for inspections are more than made up for in shortened test time and the costs of finding and repairing defects that are imbedded in the system, but resources are needed at different points in the life cycle. Project or other management must also provide sufficient and appropriate space in which the inspection process can take place.

Once resources are available and the moderator and an adequate number of inspectors have been trained, the inspection program can begin. One important factor to have in operation from the beginning is a data collection program. Formal inspection data is easy to collect because the process is very structured. The data can also be used to improve the processes that produce the products that are inspected. The subject of data collection and evaluation and improvement of the inspection process is discussed in Section VII.

Once the inspection program is underway and there are several moderators, moderators should meet regularly to discuss problems and successes with the inspection program and suggest ways to improve it. This meeting should be chaired by the chief moderator who is responsible for carrying out or recommending improvements and evaluating whether the level of training and experience is being maintained.

VII. PROCESS EVALUATION AND IMPROVEMENT

Formal inspections have been demonstrated by many organizations to be an effective method for finding and removing defects in software products. However, just putting a formal inspections program in place does not guarantee that the program will operate at maximum efficiency. It is important to evaluate the implementation of the formal inspections process and to improve

it by fine tuning the procedures that are followed. The items that most need to be tailored are the inspection rates and the checklists. If too large an amount of a software product is to be inspected at a meeting, the meeting will have to rush along too rapidly to be effective, or meetings will routinely have to be continued with resultant inefficiencies and schedule delays. If too little of the product is allocated to one inspection, the program will also not work at peak efficiency. The inspection rates must be tailored to the complexity of the product and the ability of the inspectors. Checklists should be tailored to ensure that inspectors pay attention to the types of errors that actually occur in the products being inspected. Fine tuning of the checklists will make more efficient use of preparation time and meeting time, and should help ensure that more of the defects are found.

In order to evaluate the effectiveness of the inspection program, the data collected from inspections should be routinely analyzed in order to reveal trends. The trends that should be evaluated are the amount of product inspected at a meeting, the time taken in preparation and in the inspection meeting, total defects found per inspection, the types of defects found, and the phases in the development life cycle where defects were found. The data for trending is normally collected by the moderator, using forms provided for this purpose (see appendix B).

In the case where a trend points to a decline in efficiency in, for example, the time spent preparing for and conducting inspections, action can be taken to analyze the procedures and correct the problem. The analysis might lead to changes in the amount of product scheduled for each meeting, or to the checklists provided to the participants, or to the training for the participants. If the trend data shows fewer defects are being found in inspections late in the life cycle, such as code inspections, an analysis might show that the inspectors were not adequately preparing, or it might show that the organization has become so effective in performing requirements and design inspections that there are few code defects to be found. In this case, steps may be taken to modify procedures to inspect only critical code.

The data from inspections may also be used in another manner. If, for example, the inspection data show that during code inspections a high percentage of code defects are due to defects introduced during the design phase of the life cycle, then steps could be taken to attempt to improve the effectiveness of both the design process and the design inspections. This ability to point out the life cycle step in which defects were introduced is dependent on careful data gathering, but could pay high dividends. Any attempt to change the life cycle processes used in an organization should be done with great care, and information from other sources than just inspections should be used, but the inspection data could be of great assistance.

The evaluation process should be continuous, that is, the trend data should be kept up to date, it should be examined regularly, and the trends should be available to all participants in the

inspections program. Only continuous monitoring can ensure the maximum cost effectiveness in the resources used for inspections.

The data gathered could be used to modify the inspection process itself. The third hour is one such modification, introduced by JPL to the process defined by Michael Fagan based on their analysis of their inspection program. Modification of the inspection process should be done only after very careful analysis and testing of the proposed changes. The formal inspection process is effective because it is well defined, well tested, and done in exactly the same manner time after time.

VIII. SUMMARY

The following summarizes the essentials of the formal inspection process and provides a quick reference:

1. Inspections are carried out on products that have been completed by their author but not yet tested, reviewed, or otherwise approved or baselined.
2. The objective of the inspection process is to detect and remove defects. Typical defects are errors of documentation, logic, and function.
3. Inspections are carried out by peers of the author. Participants in the inspection process should represent organizations that will use or will be affected by the material being inspected.
4. Inspections should not be used as a tool to evaluate workers. Management is not to be present during inspections. When a management official has technical expertise which is not available from other sources, that individual may be brought into the third hour.
5. A trained moderator leads inspections, and all participants should have training in the process.
6. Inspectors are assigned to and prepare for specific roles (e.g., reader, recorder, author).
7. Inspections are carried out in a prescribed series of steps from planning through follow-up.
8. Inspection meetings are limited to two hours.
9. Checklists of questions and typical defects are used to stimulate defect finding. Project-tailored entrance and exit criteria should be developed for each type of product to be inspected.
10. The product being inspected should be of an appropriate size that it can be inspected during a two hour meeting.
11. Correction of defects is the responsibility of the author,

and is verified by the moderator. The inspection team must refrain from suggesting methods for correction during the inspection meeting.

12. Data and trends on the number of defects, the types of defects, and the time expended on inspections should be maintained. This information should be used to evaluate and improve the effectiveness of the inspection program.