Lessons Learned Report

For

the "State-of-the-Art Software Inspections and Reading" Initiative

Authors:

Forrest Shull, Fraunhofer Center – Maryland

Judith Bachman, Computer Sciences Corporation

John Van Voorhis, Fraunhofer Center – Maryland

Patricia Larsen, Fraunhofer Center - Maryland

Acknowledgements

The authors of this report wish to thank Mike Stark of Goddard Space Flight Center and John Kelly of the Jet Propulsion Laboratory for their support and direction on this work, and their extensive efforts in introducing us to likely participants. Special thanks goes to Barb Pfarr and her branch (Code 584) at GSFC, which provided the highest participation rate of any of the groups that were contacted.

Executive Summary

Software defects have had negative effects on National Aeronautics and Space Administration (NASA) missions, ranging from increased expense due to corrective actions, to the loss of the mission. The reading of software products, in conjunction with inspection or walkthrough meetings, is a proven verification and validation technique. The objective of the "State-of-the-Art Software Inspections and Reading at NASA" Initiative, sponsored by the NASA Software Independent Verification and Validation (IV&V) Facility Center, is to pilot an integrated, full life cycle approach to readings and inspections and to assess whether new reading techniques, that have been validated under laboratory conditions, can be applied effectively within NASA.

This report describes a study undertaken as the first step of that initiative, to extract lessons that projects have learned in performing walkthroughs and inspections. These lessons learned will be used as the basis for identifying methods and procedures used by project teams, as well as problem areas that should be addressed in later phases of the initiative. The research performed under the initiative builds on existing work at the Jet Propulsion Laboratory (JPL) and within the Software Engineering Laboratory (SEL) at the Goddard Space Flight Center (GSFC).

For the purpose of this work, the term “inspection” is broadly defined as any technical examination process during which a product is examined for defects by personnel other than the author. This was a conscious decision made to facilitate the collection of all kinds of inspection-related experience, and to minimize the chance of excluding NASA personnel by providing a restrictive process definition.

Subjects of the Study

Participants were solicited, at several NASA centers, who were interested in sharing any experiences relating to inspections. An emphasis was placed on hearing from NASA personnel who had decided that inspections were not cost-effective for their projects, as well as those who had decided to apply them. In this way, a more balanced look at inspection effectiveness and an understanding of the suitability for different types of development environments were gained.

The 17 participants in this study were all personnel with experience either participating in or managing software inspections at NASA centers. All were highly experienced, averaging about 7.5 years at their current NASA center; 65% had performed 10 or more inspections at their site. Roughly three-quarters of the respondents had formal training in inspections; in seven of these cases the training was the formal inspection training developed at JPL.

Study Procedure

The study proceeded through three steps:

1) Review of existing inspection process documentation to understand how representative NASA centers recommend that inspections be used as part of development activities.

An analysis was performed of four inspection process standards identified as representative of inspections used on projects by both NASA staff and contract developers. These standards were:

- the NASA Formal Inspection Standard and Guidelines;

- the SEL Recommended Approach;
- Computer Sciences Corporation's (CSC) System Engineering Analysis Support (SEAS) System Development Methodology (SSDM);
- the JPL Inspection Process Training Materials.

These processes were analyzed and compared to provide a baseline for understanding the inspection processes actually being used in practice, as elicited directly from NASA personnel.

2) Interviews with project personnel to gather data on how the processes are used in practice.

Participants completed a questionnaire and were then interviewed. Whenever possible, the representative recommended processes were used to describe the work practices of participants, by allowing participants to describe when they followed the process and when they deviated. The interviews were conducted to elicit as much information as possible from the respondents concerning what process steps they undertook and in what manner.

3) Analysis of that data to summarize the state of the practice and identify problem areas for possible improvement.

Data from the questionnaires and interviews were coded (i.e. responses were examined and assigned to categories) and compared in aggregate to identify discrepancies and formulate lessons learned.

Analysis Results

Quantitative context results revealed that, in terms of important quality concerns, participants ranked issues pertaining to requirements above all others. Qualitative results revealed that while many respondents reported early resistance to inspections, participation in effective inspections almost invariably led to a highly supportive attitude toward further inspections. The results also revealed that most people use a standard, written or *de facto*, to guide their inspections. Where there is a written standard, it is often based on one of the processes discussed in this document, but may be heavily tailored to meet the project needs and constraints. In addition, almost all participants agreed that it is important to have a variety of people participate in an inspection and to ensure that as many different perspectives are represented as possible.

The analysis in general revealed that practitioners found benefits in inspections. Not only do inspections serve their intended purpose of identifying defects in the products under review, but the team gains benefits such as improving communication among the team members and external stakeholders. In the years since the NASA and JPL formal processes were introduced, resource limitations have led to reduced support for metrics collection and analysis, training, and other support infrastructure. Practitioners who have had that support and the discipline it encourages miss it and report that the process, while still viewed as "essential," is less effective. Those who have not had that support generally do not recognize the value that it could add.

Recommendations for Future Phases of this Initiative

In the later phases of this initiative, state-of-the-art inspection approaches will be introduced to NASA projects through a series of controlled studies during training sessions and pilot studies.

Based on the information extracted from the interviews and analysis, the following candidate inspection techniques have been selected for follow-up work:

- **Reading Techniques.** Reading techniques provide step-by-step procedures to guide individual inspectors when they review a software artifact. Reading techniques provide a systematic and well-defined way of inspecting a document, allowing feedback and improvement for both the inspector and the technique. Follow-up work would consist of providing training in reading techniques and collecting data to verify their effectiveness.
- **Formal inspections process.** The JPL inspection process is based on training materials that have been used at JPL and elsewhere within NASA. These materials were cited by a majority of our respondents, from several centers, as an important part of their inspection training. The impact of this training was quite far-reaching; it was consistently mentioned as a positive influence on inspection practices by our respondents. Follow-up work would consist of performing a pilot study of the process, providing long-term support and observation on a current or future project.
- **Remaining Defect Estimation Techniques (RDET).** Defect estimation techniques use the results of inspections to extrapolate the remaining defect density, in order to assess whether further inspections or re-inspections are necessary and cost-effective. Since inadequate resources (a lack of management support and time/schedule resources) were cited as a key frustration in performing inspections, RDET could provide support for projects by minimizing the cost-effectiveness of effort spent on inspections. Follow-up work would consist of further investigation of published RDET.

Recommendations for inspection application at NASA

Efforts should be made to give all NASA developers training in inspection techniques and encourage all development teams to make use of such techniques on their projects. Developers and managers are convinced of the cost-effectiveness of inspections after participation in their first effective inspection. (An additional benefit is that such training satisfies a Level 3 key process area of the CMMI software process improvement model.)

NASA centers should provide support to teams for metrics collection, including common forms and analysis of the data gathered. Respondents felt there was a cost-effective benefit gained from metrics collection but that teams were unable to provide the effort themselves because 1) the benefits were not immediately available to the team, so it was hard to correlate the benefits to the effort spent, and 2) the effort was required over an extended period of time, and schedule and time pressures for project deliverables made it difficult for development teams to make the sustained commitment.

The guidelines for 'selecting perspectives' from the JPL formal inspection training should be advocated for use by team managers and QA as a starting point for choosing useful perspectives for a product review. Of all activities associated with inspections, making an effort to get the right people to review the product was most clearly connected to the greatest list of benefits, including communication (effort is used to get more people, possibly from outside the team, familiar with the system being built) and defect reduction (the more unique perspectives that are brought to bear on a document, the more likely it is to discover any problems that may exist).

Table of Contents

Acknowledgements 2

Executive Summary 3

Table of Contents 6

Section 1 INTRODUCTION 7
- 1.1 Background 7
- 1.2 Objectives 7
- 1.3 Scope and Constraints 7

Section 2 STUDY/ANALYSIS METHODOLOGY 8
- 2.1 Subjects 8
- 2.2 Procedure 9
- 2.3 Review of existing process documentation 9
- 2.4 Interviews with project personnel 9
- 2.5 Analysis of data 10

Section 3 DESCRIPTION OF SOFTWARE INSPECTION PROCESSES 12
- 3.1 NASA Processes 12
- 3.2 Points of Comparison 12
- 3.3 The NASA Software Formal Inspection Standard and Guidebook 14
- 3.4 CSC SSDM Inspection Standard 15
- 3.5 SEL Recommended Approach 16
- 3.6 JPL Inspection Process 17
- 3.7 Comparison of the Processes 18

Section 4 STUDY/ANALYSIS RESULTS 19
- 4.1 State of the Practice - Quantitative 19
- 4.2 State of the Practice - Qualitative 22
- 4.3 Lessons Learned 23

Section 5 VALIDITY CONCERNS 26

Section 6 RECOMMENDATIONS 27

Section 7 REFERENCES 32

Appendix A – Fagan Inspection Process 34

Appendix B – Inspections Questionnaire 35

Appendix C – Predefined Interview Questions 40

Appendix D – Abbreviations and Acronyms Used in This Report 44

Section 1 INTRODUCTION

1.1 Background

Software defects have had negative effects on National Aeronautics and Space Administration (NASA) missions, ranging from the expense of corrective actions to the loss of the mission. Reading of software products, in conjunction with inspection or walkthrough meetings, are a proven verification and validation technique. The objective of the "State-of-the-Art Software Inspections and Reading at NASA" Initiative, sponsored by the NASA Software Independent Verification and Validation (IV&V) Facility Center, is to pilot an integrated, full life cycle approach to readings and inspections and to assess whether new reading techniques, that have been validated under laboratory conditions, can be applied effectively within NASA. The initiative started on November 15, 2000 and is scheduled to complete in November of 2002.

This report describes a study undertaken as the first step of that initiative, to extract lessons that projects have learned in performing walkthroughs and inspections. These lessons learned will be used as the basis for identifying methods and procedures used by project teams, as well as problem areas that should be addressed in later phases of this Initiative. The research performed under this initiative builds on existing work at the Jet Propulsion Laboratory (JPL) and within the Software Engineering Laboratory (SEL) at the Goddard Space Flight Center (GSFC).

1.2 Objectives

The objective of the activities leading to this report is to extract lessons that projects have learned in performing walkthroughs and inspections. These lessons learned are reported here and will be used as the basis for identifying problem areas that should be addressed and techniques, methods and procedures that project teams have developed and that may be of general use. The output of this phase serves as input to planning experimentation and pilot use of new reading techniques.

1.3 Scope and Constraints

The aim of this report is to produce a set of lessons learned concerning effective inspection practices, across key system domains and development environments that represent important types of NASA software development projects. This initiative does not provide a comprehensive description of how inspections are performed throughout NASA. The development domains of the participants are described in Section 2.1.1; the limitations to the scope of the conclusions are described in Section 5.

For the purpose of this work, the term “inspection” is defined broadly as any technical examination process during which a product is examined for defects by personnel other than the author. Participants were requested who would be interested in sharing any experiences relating to inspections. An emphasis was placed on hearing from NASA personnel who had decided that inspections were not cost-effective for their projects, as well as those who had decided to apply them. The goal was to acquire a more balanced look at inspection efficacy and some understanding of whether inspections were more or less suitable for different types of development environments.

Part of the scope of this report is to review the training and recommended practices that participants received while at NASA, as a first step toward understanding the inspection practices they used during actual development. A detailed analysis of all the training received by the participants is not feasible, but this report focuses on representative processes from the NASA Centers from which the majority of participants came. Section 3 describes the analysis of these processes. Subsequent discussions revealed that these representative processes did in fact provide the training for a majority of participants.

Section 2 STUDY/ANALYSIS METHODOLOGY

2.1 Subjects

The 17 participants in this study were all personnel with some experience either participating in or managing software inspections at NASA centers. Table 1 shows how subjects were distributed across NASA Centers (all acronyms are explained in Appendix D), and Table 2 identifies the system domains in which subjects had expertise.

Table 1: Centers Represented	
GSFC*	9
JPL	4
GRC	2
LRC	1
JSC	1
Total	*17*

*includes 2 respondents from CSC, a GSFC contractor

Table 2: System Domains Covered	
Control Center	4
Data Capture	6
Facilities	1
Flight Software	5
Infrastructure	1
Mission Planning	3
Science Processing	1
System Management	1
*Total**	*22*

*includes multiple categories per respondent

Participants were solicited by approaching key personnel at each center and asking them to participate in an interview directly or to recommend contact people at their center who have direct experience performing or managing inspections. (These key personnel included John Kelly, Program Element Manager of the Software Assurance Research Program at JPL; Martha Wetherholt, Software Assurance Lead at Glenn Research Center; and the managers of the Information Services Center at Goddard Space Flight Center. Dr. Kelly also recommended personnel at LaRC and JSC.) The key personnel were asked to recommend interview participants who work on projects that are “typical” of important classes of projects for the center or whose inspections were considered extraordinarily successful by their peers and managers. Although the total number of interviews is too small for a comprehensive report of inspection practices within NASA, this study has explored the use of inspections in several different development paradigms of importance to NASA, and has resulted in several important lessons learned.

The participants were all highly experienced personnel. On average, they have spent about 7.5 years at their current NASA center; 65% have performed 10 or more inspections at that site. All of the interviewees were experienced developers and team leads. This overall high level of experience is a good indicator of the quality of the derived lessons learned.

Roughly three-quarters of the respondents (13/17) had formal training in inspections; in seven of these cases, the training was the formal inspection training developed at JPL. Even with the training, three-quarters of the respondents felt that on-the-job training played a significant part in their inspection knowledge.

2.2 Procedure

This study proceeded through three steps: 1) review of existing inspection process documentation to understand how representative organizations recommend that inspections be used as part of development activities; 2) interviews with project personnel to gather data on how the processes are used in practice; and 3) analysis of that data to summarize the state of the practice, identify process variations introduced on certain projects and their impact on process effectiveness, and identify problem areas for possible improvement.

2.3 Review of existing process documentation

The first step of this study was an analysis of four inspection process standards:

- the NASA Formal Inspection Standard and Guidelines;
- the SEL Recommended Approach;
- Computer Science Corporation's (CSC) System Engineering Analysis Support (SEAS) System Development Methodology (SSDM);
- the JPL Inspection Process Training Materials.

These processes were chosen because of the expectation that they would be known directly by a majority of the participants, and representative of the processes applied by a large majority of NASA personnel. These expectations were borne out during the study, in which almost half (7) of the participants had trained on the JPL materials; three more knew the SSDM, and another two the Recommended Approach. As described in Section 3, the NASA Standard, while not used directly by any of the participants, formed a significant basis of the other approaches. Only one participant cited training in an inspection process (Fusion) other than those previously listed.

In order to better understand these processes and the differences between them, the important dimensions along which they vary were identified, and the processes were compared along those dimensions. The resulting summary of the recommended processes is included in Section 3 of this document. This summary provides a baseline reference for the inspection processes actually being used in practice, as elicited from the interviews. The existing documentation described the processes with which respondents were expected to be familiar.

2.4 Interviews with project personnel

Participants were first asked to fill out a short questionnaire. The questionnaire contained mostly multiple-choice and yes/no questions to characterize the organization where the participant worked, the participant's experience with inspections, and the type and size of projects on which inspections had been used.

The interviews themselves were semi-structured, addressing a set of predefined questions (facilitating the comparison of results from different interviews) but leaving time for the interviewers to pursue interesting leads as they arose. Some interviews began with the interviewers asking the participant to clarify or further explain some information from the questionnaire. The predefined questions were aimed at eliciting the following types of information:

- *State of the practice for applying inspections.* Participants were asked to describe for what lifecycle phases they felt inspections were most effective, and for what types of projects they felt inspections were most appropriate. Participants were also asked to describe what kinds of benefits were being seen on their projects due to inspections, and whether they felt the current level of investment in inspections was worth the benefits seen.

- *Problem areas.* Participants were asked if the existing process support was adequate for their projects, if inspections were performed but not felt to be cost-effective for certain types of projects or work products, and what difficulties they encountered in performing inspections on their products.

- *Local techniques and methods that could be of general use.* Participants were asked if they had tailored the processes in some way for their projects, had developed effective guidelines or process aides, or had identified useful results to consult when performing or planning inspections.

- *Preparation activities.* Participants were asked what activities were part of preparing for inspections, and how much time they usually spent on these tasks.

Whenever possible, the representative recommended processes described in Section 3 were used to describe the work practices of participants, by allowing participants to identify when they followed the process and when they deviated. In most cases, however, the interviews were conducted to elicit as much information as possible from the respondents concerning what process steps they undertook and in what manner.

All interviews were conducted by at least two people, one taking the lead on asking the questions and both taking notes to ensure that the information was properly captured. Both interviewers were free to interject follow-up questions as they felt appropriate. The research team typed and reviewed their notes for accuracy and clarity. The notes were then returned to the interviewee, who was asked to review them for omissions or incorrectly captured information.

Immediately after the interview, the interviewee was assigned a numerical reference number. Interviewees were identified only by that number throughout the analysis and reporting, in order to preserve the anonymity of respondents.

The questionnaire and the list of predefined interview questions are included as appendices.

2.5 Analysis of data

The first phase of the analysis featured the compilation of the notes taken during the interviews followed by the “coding” of these interview answers. In coding, the researchers examine the

interview responses and assign categories to the topics being discussed at a particular point in the text. Initially, the codes corresponded to the predefined interview questions (i.e., the interviewee's statements were grouped according to the question where they best fit as an answer, regardless of when during the interview they occurred). During the analysis, new codes were introduced to further refine the information. For example, the code "inspection benefits" was used to label any text where the interviewee mentioned any positive results of using inspections on his or her projects. Later, the code "team issues" was introduced to specifically differentiate a common type of response dealing with issues such as team cohesiveness and the training of new members.

Coding is an important step in qualitative analysis because it provides the key to organizing the data and grouping it into categories so that it can be analyzed more rigorously. Coding in this study was first done independently by the researcher who served as the note-taker during the interview. The coding was then examined by the remainder of the research team and disagreements were resolved by discussion between the researchers.

Once coding was completed, simple descriptive statistics were used to better understand factors affecting how inspections were applied. The codes were analyzed to determine correlations between the interview answers and the characterization data from the questionnaire (e.g. Did participants who performed similar types of inspections tend to report similar problems?), as well as within the characterization data itself (e.g. Did participants developing similar types of systems tend to find inspections on the same types of work products cost-effective?).

The resulting descriptive statistics are found in Section 4.

In the main part of the analysis, the data from multiple interviews were compared to determine whether the interviewees' statements in each of the coding categories tended to agree or disagree. Where there were differences of opinion between interviewees, the other information gathered from the questionnaires and interviews was examined to look for possible explanations for the discrepancy. The results from this analysis were used to formulate the lessons learned, which are included in Section 4 of this report.

Although the approach outlined above is not a common method of analysis in computer science, it is a recommended approach for social sciences and other fields that require the analysis of human behavior (for example; Eisenhardt, 1989, and Miles, 1979). In software engineering, it has been used to elicit processes from software developers, for example to understand processes by which developers use an Object Oriented framework to develop a new system, and evaluate the effectiveness of those processes (Shull, 2000; Basili, 1998). It has also been used in previous studies at NASA, for example to understand how time is spent during inspection meetings and factors influencing their effectiveness (Seaman, 1998). This form of analysis is well suited for the purpose of this initiative because the variables of interest are heavily influenced by human behavior and because the goal is not to definitively test a set of heuristics or guidelines, but rather to elicit the hard-won experience of people who have been using inspections in practice.

Section 3 DESCRIPTION OF SOFTWARE INSPECTION PROCESSES

3.1 NASA Processes

This section describes four of the process definitions that are used by NASA personnel as a basis for inspections. These documents are representative of inspections used on projects by both NASA staff and contract developers.

- The NASA Software Formal Inspection Standard and NASA Software Formal Inspection Guidebook (NASA, 1993a, and NASA, 1993b) were intended to provide a common foundation for software inspections throughout NASA and its contractors.

- The current JPL standard (JPL, 1999) is an updated version of a process that formed part of the background for the NASA formal inspections guidebook and standard.

- The SEL Recommended Approach (NASA, 1992) was developed in parallel with the NASA standard and provided baseline procedures for work occurring at the Flight Dynamics Division of GSFC and at CSC.

- CSC's SSDM (CSC, 1999) was developed, concurrently with the SEL RA, as part of CSC's SEAS project but has been modified and updated over time. It forms a baseline for work done by CSC.

All of these processes are ultimately derived from the formal inspection process initially described by Fagan and further refined by Gilb (Fagan, 1976; Fagan, 1986; Gilb, 1993). The specific details of a "Fagan inspection" are described in Appendix A. The next section describes the questions used to compare the process standards and also provides an explanation of the Fagan inspection process. The following subsections analyze the representative NASA process documentation using the common questions.

3.2 Points of Comparison

Four questions were used to structure the analysis of the different standards. These questions were selected in order to provide enough information to serve as a starting point for others who wish to learn about or use these procedures in their own work. These questions summarize the key points of any process: What is the process applied to? Who is expected to apply the process? When can the process be applied? How can the effectiveness of the process be measured?

3.2.1 What kinds of work products can be inspected?

This question asks whether an inspection process is meant to be used with only certain types of work products (e.g., a process should be applied only for code inspections, but does not apply to requirements). A subsidiary question asks if all parts of a given artifact are inspected or only certain parts. If inspections are performed selectively, what determines what products or parts of products are inspected?

Fagan's initial paper described inspection as "a *formal*, *efficient*, and *economical* method of finding errors in design and code" (Fagan, 1976). While his process was initially developed for

reviewing program design and code, he pointed out in a later paper (Fagan, 1986) that it could be applied to any document in the software development process.

3.2.2 Who are the participants in an inspection?

Before an inspection process can be applied, it is necessary to know how many people are required, and whether those people must possess certain types of expertise. Subsidiary questions include: Are there constraints on the specific roles individuals adopt? Are there roles that can be played by the same person? Roles that must be fulfilled by different inspectors? Which staff provide good candidates for inspectors?

In a Fagan inspection, inspectors should be selected from among members of the project team and staff of similar but unrelated projects. Several roles are identified, with specific responsibilities:

1. **Author**. The author of the product to be inspected by the group.
2. **Moderator**. The coordinator of the inspection process as a whole.
3. **Inspector**. A knowledgeable person who will review the product and attend the inspection meeting.
4. **Reader**. The leader of the inspection meeting.
5. **Recorder**. The recorder of the defects and open issues found at the meeting.

3.2.3 When is it appropriate to have an inspection?

This question is associated with a number of subsidiary questions. When is an artifact ready? Who determines that and what criteria is the decision based on? When is it clear that an artifact or product needs an inspection? Does the process divide projects and/or products into inspectable and non-inspectable types? Are there instances where time or budget constraints prevent inspections from happening?

Within a waterfall project process, Fagan describes several points where inspections should be performed (Fagan, 1976). He also presents an example of an entry criterion that determines whether a particular design module is ready for inspection. Such entry criteria need to be developed or tailored by each organization or project.

3.2.4 How is the inspection process itself evaluated?

Does the procedure describe how to collect metrics concerning the quality of the document being inspected? Are those metrics fed back to the development process? Does the procedure describe how to collect metrics to measure the cost-effectiveness of the inspection process itself? Are there guidelines concerning how to use the metrics to determine whether the process needs to be changed? Are the metrics saved and, if so, maintained in accessible form?

Fagan suggested using module defect densities to help drive inspection and reinspection decisions at subsequent stages of the development process. In addition, error and effort metrics can be used to manage the inspection process itself by indicating inspectors or projects that have metrics that deviate from historical or other comparative data.

3.3 The NASA Software Formal Inspection Standard and Guidebook

The NASA Software Formal Inspection Standard (NASA, April 1993) is reference standard for projects that want to define an inspection process. It does not describe most process details. Instead, it identifies the required elements that a project must define to have a conforming inspection process. For example, it does not stipulate specific entry criteria for an inspection, but rather states that specific entry and exit criteria must be present at each stage in an inspection process for it to conform to the standard.

The Software Formal Inspections Guidebook (NASA, August 1993) presents additional details regarding how a development team can implement an inspection process that follows the Standard. It is a companion document that "provides additional information on how to establish and implement the [inspection] process."

What? Inspections are suggested for several different phases of the overall development process. The typical products found in these phases are:

- System Requirements
- Software Requirements
- Architectural Design
- Detailed Design
- Source Code
- Test Plan
- Test Procedures

A given project is expected to tailor its inspection procedure to include whatever products are needed to meet the software assurance and safety needs of the project.

The Guidebook suggests inspections for products developed in all phases of the standard NASA waterfall model through the implementation phase. The Guidebook describes more specifically the products that are likely to be inspected during each phase of the standard development process.

Who? There must be a minimum of three inspectors in a conforming inspection procedure. The author of the documents under review and the inspection moderator must be different individuals. Additionally, the roles of reader and recorder at the meeting can be assigned to any inspector except for the author. The other inspectors are selected from developers working on the project in its current phase, working on interfacing components of the system, and who have worked or will work on the product in the previous and next phase of development.

The Guidebook helps enumerate additional sources of personnel, suggesting Systems Engineering, Testing, Software Assurance, and other interested stakeholders. Outside inspectors

should be brought in if the moderator feels the need for expertise not present among the project team.

When? An inspection is called when the team leader and the author agree that a product is ready for review. Entry criteria need to be defined and may address issues such as formatting, passing a first compilation, or passing some minimal number of unit tests, among other possibilities.

The Guidebook does not specify any process for generating entry criteria. It simply assumes that the criteria have been decided upon by or for the inspection team already.

How measured? The moderator is responsible for maintaining a record of process metrics that can be used to evaluate whether the process is working or not. These measures must include, at a minimum, defect counts and densities (per page/LOC) and effort expended in person-hours in each phase of the inspection process.

The Guidebook suggests that data should be collected and analyzed routinely and continuously to judge the effectiveness of the inspection process. Typical metrics include the amount of product inspected per meeting, inspector preparation and meeting time, defects found per inspection meeting and per module, types of defects found, and defects found per phase.

3.4 CSC SSDM Inspection Standard

The SSDM (CSC, April 1999) was developed by CSC as a revision and extension of their earlier Digital System Design Methodology (DSDM). SSDM was originally developed specifically to support work on the SEAS contract. It is a full life-cycle methodology that provides general guidelines for planning and managing the design, development, and installation and maintenance of software systems. It includes several standards and procedures related to software inspections.

In addition, the process provides some limited provisions for less formal inspections that gather similar data as the formal process but require only a single inspector besides the author. In such an informal review a list of defects and open issues is generated and sent to the Quality Assurance Office (QAO).

What? Almost any document is open to review. There are several standards describing the qualities needed in each type of product along with specific procedures tailoring the basic inspection standard to each document type. Standards and procedures are provided for:

- Data Flow Diagrams
- Data Dictionaries
- Functional Specifications
- Unit Designs
- Structure Charts
- Unit Code
- Unit Test Plans

SSDM also provides a mechanism for creating new or tailoring existing standards and procedures for a specific project or task.

Who? The author and the moderator are the only specified roles in the process. Reader and recorder (here, called the "scribe") are mentioned as meeting roles, but the description seems to assume that the moderator is responsible for both, although the option is available to assign them to another inspector. A QAO representative is always nominally on the attendance list.

When? An inspection occurs when the author of a product decides that his work meets all the criteria that have been defined as marking the product ready for the next phase of development. The moderator does a quick check to make sure there is nothing seriously wrong with the product and sets up the meeting.

How measured? Several data collection forms are described for maintaining data about the effectiveness of the inspection process. The time of preparation is aggregated across all inspectors, by product type. And the defect list recorded at the meeting provides counts of defects in each product.

3.5 SEL Recommended Approach

The SEL at GSFC developed the Recommended Approach (RA) as a guide for software development in the Flight Dynamics Division (NASA, June 1992). Over time it has been adopted as a non-mandatory, but de facto, standard for many GSFC projects. The RA is also a full life-cycle methodology that places a strong emphasis on inspections and less formal reviews as part of the process. It does not describe inspections in detail, but the process described follows a Fagan inspection process and is consistent with it.

Throughout, the RA considers walkthroughs, inspections, and reviews as processes that differ along a continuum of formality but are all part of the same quality assurance process. In part this is because this is a recommendation, rather than a specific standard to follow, but it also reflects the authors' intention that all products should be reviewed by at least one person other than the author. *Walkthroughs* give developers and designers a chance to present the products to others on the team who need to understand the product. *Inspections* are more formal events with an explicit defect reduction goal. *Reviews* are the most formal with primarily a management focus.

What? The RA makes it clear that every work product is open to at least a walkthrough style review. Walkthroughs are explicitly recommended in several phases:

- Requirements analysis
- Preliminary design
- Detailed design
- Implementation

While walkthroughs and inspections should occur during the above phases, reviews are recommended for use at the completion of each phase.

Who? Walkthroughs are generally small groups containing one to five people from interfacing systems or sub-systems, along with the author. In an inspection, the moderator runs a meeting of three or more members of the development team.

When? Walkthroughs are done on small parts of a design, code, or specification as the product is developed. Inspections are performed on complete or nearly complete products. In both cases, the head of the team or project decides when review activities are necessary.

How measured? There are no specific suggestions on metrics for measuring the success of the inspection process. However, the RA suggests metrics be collected during each lifecycle phase. It is implicit that some may be collected as a result of the review process (e.g., errors by category during the implementation phase).

3.6 JPL Inspection Process

The JPL inspection process described here (dating from October 2000) is based on training materials that have been used at JPL and elsewhere within NASA. This process is clearly related to the NASA Standard (which incorporates by reference several JPL checklists).

What? Similarly to the Guidebook, this process covers all types of inspectable products. Specifically mentioned are:

- System and subsystem requirements
- Subsystem functional description
- Architectural design
- Detailed design
- Source code
- Test plan
- Test procedures and functions
- Operator's manual

Who? An inspection requires a minimum of three people, following the same roles described in the NASA Formal Inspection Guidebook. Some prescriptions have been made for combining roles due to the smallness of the inspection team: the "author" and "reader" roles should never be played by the same person; the "moderator" and "author" cannot be the same; nor can the "moderator" and "reader." Several support roles are identified, including:

- Manager
- Chief moderator
- Educator
- Data manager
- Librarian

Only the chief moderator and librarian roles are also mentioned in the Guidebook.

When? The author determines when the product is ready for inspection.

How measured? The time spent in each phase of the inspection process is collected. Individual inspectors prepare inspection logs with defects listed and categorized, which are then used as a basis for the defect list generated at the inspection meeting.

3.7 Comparison of the Processes

In general, the processes reviewed here are more similar than they are different. All are based on the Fagan inspection process. Differences mainly center around terminology, the specific activities surrounding the meeting, the specification of roles for the inspectors, and the collection of metrics.

Some of these differences are due to the different purposes of the process descriptions. The NASA Standard and Guidebook and the JPL materials concentrate on inspections alone. SSDM and SEL-RA are concerned with the entire development lifecycle with inspections as a part of that lifecycle. Created for use by a contractor, rather than by civil servants, the SSDM describes a process of certification where the author, moderator and a QA representative sign off on the quality of the work product. This certification step is not present in the other processes.

The processes are compared here within the framework of the questions used to organize the process descriptions in previous sections.

What? There were some differences in the specific artifacts mentioned, but all processes recommend that inspections be used for software specifications, design, and code for all “large” projects. (The definition of a large project is left to the members and managers of projects to decide.)

SSDM specifies the Structured Design methodology. It specifies a particular set of documents (Data Flow Diagrams, Structure Charts, and Functional Specifications) that are to be generated and reviewed during the requirements and design phases that are not specifically mentioned in the other processes. The other processes simply prescribe inspection of the appropriate artifacts, documents, or work products within those phases. Another terminology difference is that all of the inspection guidelines use slightly variant names for the different development phases and stages or iterations within those stages, based on the underlying terminology at their points of origin.

Who? For all the processes, guidelines for the selection of inspectors were basically the same. However, the tasks assigned to the different roles did vary. The most significant differences across the processes involved the allocation of tasks to the author and moderator during the planning and execution of an inspection. There were also differences in the way meetings were to be run; some of the processes allow the reader role to be taken on by the moderator or the author, while others specifically disallow such role mergers.

Another difference concerns the overview meeting. The moderator is responsible for running the overview meeting, before individual preparation by the inspectors commences, but the overview meeting is optional in both the NASA Standard and the JPL process, while the CSC SSDM and SEL RA do not even mention it. It seems, though, that the author is expected to conduct a meeting similar to the overview in the SEL RA.

The SEL RA and CSC SSDM both mention the use of a less formal review process using one or two individuals beyond the author. The SEL RA recommends that these small reviews be used in preparation for larger, formal inspections.

When? Deciding when a document is ready for a review is a task variously assigned to the document's author, the inspection moderator, or the project team lead or manager. Once again, the differences seem to derive more from differences in the background methodology rather than differences in the inspection processes themselves.

How measured? All processes recommend that defects are tracked at inspection meetings, but not all make a point of advocating that open issues should be recorded. If open issues are not tracked, the "Third Hour," a post-inspection meeting to address open issues or solutions to defects found during the inspection, loses some of its effect. The RA doesn't mention this activity, although the other process descriptions do.

The SSDM and the JPL process both describe in some detail how to use the collected metrics for project and process improvement, while the SEL RA and NASA Standard simply suggest that they be used for such purposes. The SSDM suggests that the results of the small reviews should be collected in much the same manner as the formal meetings for project and process management purposes.

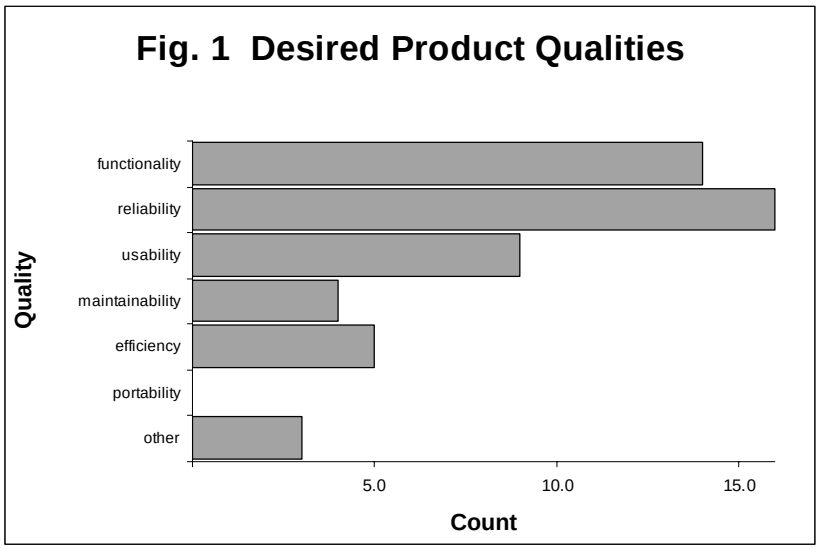
Section 4 STUDY/ANALYSIS RESULTS

4.1 State of the Practice - Quantitative

The description in this section is derived primarily from the questionnaires each respondent completed before the interview.

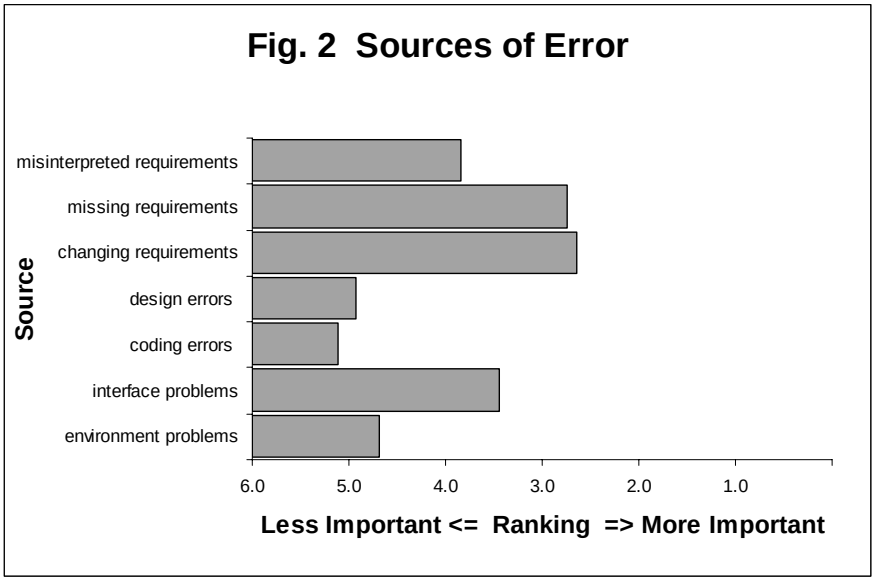
4.1.1 Context description

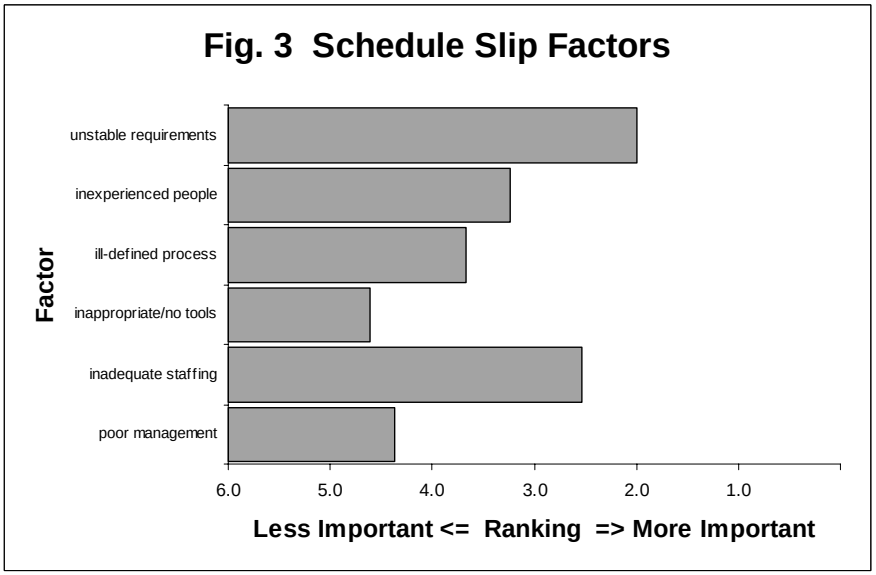
The questionnaire contained several questions regarding the context of the respondent's projects. Figure 1 summarizes the responses concerning the most important product qualities for the systems on which respondents worked. Sixteen of the seventeen respondents listed reliability as one of the three most important product qualities for their systems, and fourteen included functionality as one of the most important qualities for software products.



Figures 2 and 3 illustrate the relative *average importance* assigned to various factors by respondents. These figures show that issues pertaining to requirements were clearly ranked above other issues relating to the quality of software products and process. For example, although all respondents listed “changing requirements” and “coding errors” as major sources of error in projects they had worked on, they did not feel that these problems contributed equally to development difficulties. In the scaled rankings, with 1 being most important, “changing requirements” was ranked highest, with an average ranking of 2.6. “Coding errors” had the lowest ranking with an average of 5.1 out of seven provided choices.

“Unstable requirements” was reported as the leading factor affecting slippages against plan with an average ranking of 2.0, with 1.0 being most important. The second and third most important factors listed were inadequate staffing and inexperienced people.

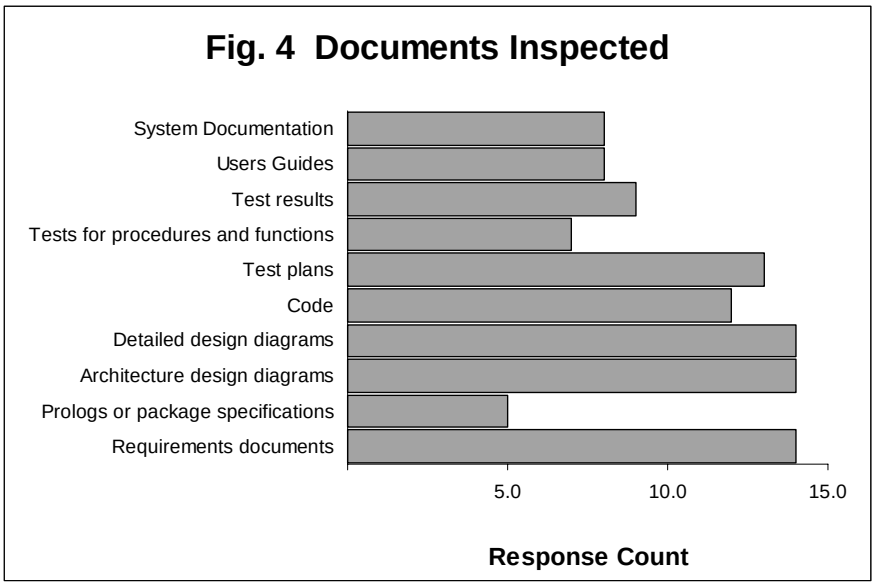




4.1.2 State-of-the-practice regarding inspections

All of the respondents had participated directly in inspections at some point in time. Roughly three-quarters of respondents are still doing them. Of the remaining one-quarter of respondents, the majority had been promoted to a position where they no longer were able to directly participate. However, two respondents were in situations where the size of the teams and the size of the projects made the costs of formal inspections appear to be much larger than their potential benefit.

Figure 4 identifies the total number of respondents who reported inspecting various software development artifacts. Requirements and design documents (both high-level architecture and detailed designs) are almost always inspected. Test plans and code are also inspected on most projects. In general, the first documents generated by a team, whether requirements, designs, or code, are inspected against whatever document is received by a team from their customer. Five of the respondents had little or no input into the requirements development process. All five felt that code inspections were an important part of their quality assurance program, while half of the other respondents felt that code inspections were not significant in terms of detecting defects.



The highest-priority item checked for during inspections was the consistency of the current documents with those from the previous development phase. Respondents look for inconsistencies with understood customer needs during requirements inspections, for mismatches with the requirements specification during design inspection, and for comparisons to the design during code inspections. Checking for omitted functionality during requirements and design inspections was also a high priority, indicated both by the number of respondents who mentioned this and the importance they assigned it. There was clear consensus that the lowest-priority item to check was adherence to coding standards[1].

[1] Some respondents later suggested reasons why standards were rated so low: Most developers are professional enough to follow the standards adequately, so checking that level of detail during inspections is not necessary. Also, other factors are just "more" important in that they have more impact on the quality of the final system. The one

There was no clear correlation between project type and the use of inspections: Some large or mission-critical projects did not do inspections, while some small or low criticality projects did use them. However, a larger percentage of the larger and more critical projects among the respondents did use inspections fairly regularly.

The projects described by respondents varied from very short, small, non-mission-critical projects to long, large, mission-critical projects. Of the seven interviewees whose projects took 24 months or longer, five made a concerted effort to conform to an inspection process description. Of the five whose projects took 12 months or less, two followed some inspection process on inspections.

The project with the largest team size (>20) was not mission critical and did not use inspections as much as the interviewee would have liked. But eight out of the ten mid-sized (5-20) projects used formal inspections, while four of the five smallest projects did not emphasize inspections. (These numbers do not add to 17 due to missing responses.)

4.2 State of the Practice - Qualitative

Whether extensively tailored or not, all of the respondents who described their inspection process described processes that were clearly related to the NASA Standard, but each of the four questions has a range of answers in the field.

What? All of the respondents review the first document they receive from a customer. For the smallest projects (web page development, for example) the review may be just a few questions asked over the phone or through email. For several respondents (5), the first artifacts they can effectively change are the design documents because the requirements have been fixed. As a result, those respondents concentrated on design and code inspections.

Who? Almost all interviewees discussed the importance of getting people with the right perspectives involved in an inspection; a subset (3) explicitly mentioned basing their decisions on the recommendations from the JPL formal inspection process. In general, the inspectors were expected to bring their own expertise and perspective to the inspection rather than following checklists or any other specific inspection technique.

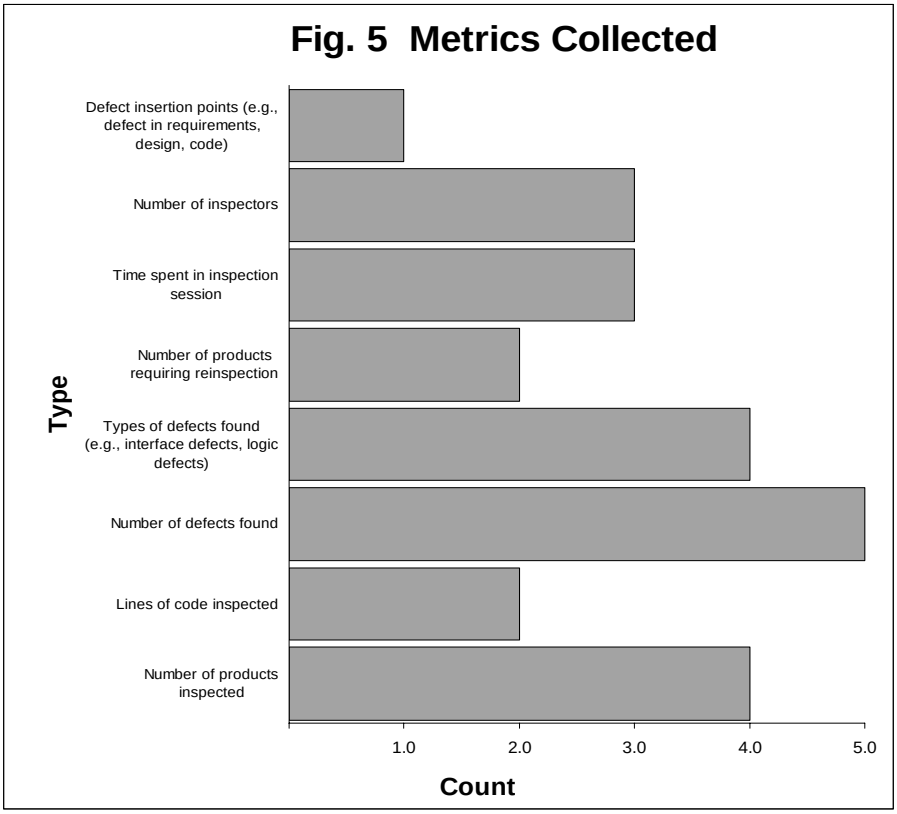
When? "Early and often" was the most frequently heard answer; this was true for 11 of the respondents. It was frequently not clear who decided when to inspect an artifact because on many projects the team lead, project manager and developer worked closely together on making the decisions.

How measured? Almost no one collects or uses data in the inspection process itself. The one exception is when the data can be used with customers, e.g. as proof of team's development quality. Very few (only 3 respondents) collected data for internal use. There is some indication that people who have collected internal data in the past find it useful and would *like* to continue

---

respondent who did rank standards as highly important cited the contribution of standards to maintainability, in that seeing the effect of standards use on software artifacts was a learning experience for new developers.

doing so. However, there is overhead involved and without support from outside the project, process measurement effort is sacrificed for more direct effort on the project itself. As shown by Figure 5, the type of information collected tends to be relatively simple measures, such as the number of defects.



While many (8) respondents reported early resistance to inspections, most of them reported that participants in inspections often were very supportive of further inspections. One respondent reported that participants from large projects at that site often asked for inspection support for small projects because of the benefit they had seen from their participation in inspections on their earlier projects.

4.3 Lessons Learned

4.3.1 Lessons Learned - Tools and Techniques

The interview began with questions to augment and clarify the information gathered from the survey. These questions were intended to identify specific tools, techniques and tailoring approaches that have proved helpful.

- Are there process standards or guidelines that cover how inspections should be performed in your organization?
- Who participates in inspections in your project or organization? How are participants selected?

- Are there distinct roles that people play in an inspection? Are all reviewers looking for the same things in the work product?
- Are there other people or resources that you consult while performing an inspection?
- Are there job aids that you use during inspections (e.g. checklists, automated data collection)? Are they helpful?

Eleven people reported that they used a standard. Of these,

- Seven described their standard as having been based on or tailored from one of the four standard processes discussed in Section 3.
- The remaining four have developed local standards.
- Two described these local standards as very informal. Roles are combined and data collection is reduced or eliminated to reduce the process overhead and to deal with very small teams.
- Two more describe approaches that while not documented as standards seem to be local *de facto* standards. One said that his group "made it up as we went along" and "evolved the process through cultural consensus."

Of those who perform inspections but do not follow a standard, one attributed this choice to a lack of time to follow a standard, and another said that he often works alone so review of his products is ad hoc.

In most cases the project lead identifies participants. The participants are chosen with considerable thought to the particular expertise or perspective that they can provide. In some cases the developers recommend participants. In other cases next level managers and quality assurance personnel contribute to selecting participants. With few exceptions respondents identified the value of independence and breadth of perspective in the participants. Two groups with distributed teams did their inspections via teleconference to get adequate participation. Eleven people mentioned testers, quality assurance personnel, and software system engineers, along with developers from the project and interfacing projects. Nearly as many said that they looked for expertise outside the project: operations personnel, spacecraft and instrument hardware engineers, and users were identified. Two respondents said that they call on experts external to their organization and have on occasion requested expertise from other centers.

When discussing tools or other job aids, three people identified automated tools to check consistency or adherence to standards. These were viewed as valuable precursors to the inspection process. Six respondents used checklists. Some of the checklists were tailored from the standards and others were developed specifically for the project environment. One person cautioned that checklists may be a starting point, but should not be used "as a replacement for thinking."

Lessons Learned. Most people use a standard, written or *de facto*, to guide their inspections. Where there is a written standard it is often based on one of the processes discussed in Section 3, but may be heavily tailored to meet the project needs and constraints. In addition almost all participants agreed that it is very important to have a variety of people participate in an inspection. Great care is taken to ensure that as many different perspectives are represented as

possible. Participants are selected for their knowledge in different development disciplines as well as their knowledge of different technical aspects of the project.

4.3.2 General Lessons Learned

Several questions asked during the interviews were intended to elicit lessons learned and overall "truths" from the interviewees. The questions were asked near the end of the session to allow time for the interviewee to have recalled various situations associated with reading software products and various types of inspections. The questions were open ended to encourage general conversation about experiences, good and bad, with inspections.

- What would you say are the main benefits to your organization from doing inspections? Are there conditions under which you aren't likely to see those benefits?
- Do you think the time spent on inspections for your projects is generally well spent? Do you ever have inspections where you feel the time spent isn't worth the payoff? Do you have any ideas what causes those situations?
- Are there any problems or frustrations that you have with inspections?
- Any other comments about inspections or other relevant issues?

Eleven of fourteen interviewees with whom organizational benefits were discussed identified communications related benefits. Five identified improving communications among team members and with members of other teams on the same project. In these cases the respondents noted improved understanding of the product under development. One person said that people "learn things that they didn't know they needed to know, especially in requirements and preliminary design" through inspection meetings. In one case inspection meetings are held for nearly all artifacts and are the primary vehicle for training new team members. Four said that inspections are essential - "I don't know how we would meet deliverable deadlines without them" and "there would just be too much risk [without inspections]."

Four people identified team building, especially in the early phases of a project, as a benefit. Training and cross training was identified by four other participants. In these cases respondents discussed improving the skills of individuals and improving the effectiveness of the team as benefits beyond those that accrue to the product under development.

Thirteen of seventeen agreed that time dedicated to inspections is well spent. Five respondents identified cases where the time had not, in their opinion, been well spent. Scheduling the inspection before the product was ready, or trying to review too much at one time were mentioned as causes of inspections that were not worth the time expended. Other causes of ineffective inspections noted were people who did not take the process seriously and the learning curve in introducing inspections where they have not previously been performed. Not having the right people or expertise available was also mentioned. One person reported having rescheduled meetings rather than hold them when a key person was not available, and mentioned that the Mars Climate Orbiter failure validated his insistence on having the right people available, since the defect that led to failure of the mission could have been found with the involvement of the right person in the inspections.

Six people identified either defect detection or improvement of design or code as benefits and one said that defect detection was a secondary benefit. But other data did put further emphasis on this benefit, since sixteen of seventeen surveys indicated that defects were always or usually discovered during inspections.

Nine people responded to the questions about problems or frustrations with issues related to inadequate resources. One described an earlier environment where that support had been available and had been valuable. More recently, with increased pressure to deliver systems faster and less expensively, that support is no longer available. Another said that lack of meeting support has sometimes resulted in action items not being followed up adequately. Limited resources have contributed to "tailoring out" formality and process steps. Some viewed the tailoring as appropriate for the project and constraints; others felt that while a somewhat less formal process was appropriate perhaps too much had been cut. However, one respondent said that even though there were inadequate resources and little management support for inspections the culture of the team (as a result of a previous project) included inspections so they found a way to hold inspections.

Five people identified the learning curve associated with introducing inspections to a new team as an issue. But all of them felt that the inefficiencies associated with the learning curve were acceptable and that once people got used to the process, as locally tailored, it was worthwhile. Two indicated that while people often don't see the benefit at first they quickly recognize the benefits.

A third area where people experience frustrations is in moderating the meetings. Five said that good moderators are those who can handle difficult people: participants who are not constructive in their comments or who are unwilling to accept constructive criticism can affect a meeting's effectiveness adversely.

Lessons Learned. Practitioners find benefits in inspections. Inspections serve their intended purpose of identifying defects in the products under review. But the team gains benefits beyond defect detection by improving communications both among the members of the team and with others, external to the team, but important to the outcome of the project. Inspections also serve important team building and training functions. This is not to say that people are completely satisfied with the process. In the years since the NASA and JPL formal processes were introduced, resource limitations have led to reduced support for metrics collection and analysis, training, and other support infrastructure. Practitioners who have had that support and the discipline it encourages miss it and report that the process, while still viewed as "essential" is less effective. Two felt strongly that the value added was well worth the cost. Those who have not had that support generally do not recognize the value that it could add.

Section 5 VALIDITY CONCERNS

The most important concern regarding the validity of this study's conclusions must be the small size of the sample population (17 interviewees). As stated in Section 1.3, the aim of this work is not to give a comprehensive description of inspections at NASA, but to describe lessons learned in representative environments. To achieve this goal, participants were solicited representing

multiple system domains within NASA (see Section 2.1.1). A range of team sizes and project durations were covered that seem to represent the spectrum of NASA projects.

A more significant concern is that participants did not represent a real cross-section of attitudes that NASA personnel have toward inspections. There was a chance of selection bias at both the referral and acceptance steps in arranging the interviews; that is, people who feel they have had good experiences using inspections are more likely to both participate as interviewees and suggest other people to participate. A consequence of this is a lack of lessons learned on the topic of situations for which inspections are not suitable or cannot be used effectively – although some respondents did have experiences on this topic to share. However, the fact that many of the participants were highly experienced in using inspections is very positive and crucial to effectively elicit lessons learned.

Throughout this study, every attempt has been made to avoid bias concerning results, not only in analysis but also in the information solicited during interviews. Although the researchers set the initial content of the questionnaires and interview questions, the semi-directed nature of the interviews meant that interviewees had the chance to raise important issues other than those originally planned for. This was important to ensure that the interviews did not merely confirm or deny the original ideas of the researchers, but could raise additional issues as necessary. Also, misinterpretations of the interview data were avoided by having multiple researchers conduct every interview and categorize the responses.

Section 6 RECOMMENDATIONS

Recommendations for this Initiative. In the later phases of this Initiative, state-of-the-art inspection approaches shall be introduced to NASA projects through a series of controlled studies during training sessions and pilot studies. This lessons learned report provides a basis for selecting specific inspection approaches for use in these later phases.

Based on the information elicited during the interviews and analysis, the following candidate inspection techniques have been selected for follow-up work:

- **Reading Techniques.** Reading techniques provide step-by-step procedures to guide individual inspectors when they review a software artifact. Unlike many other inspection approaches, which refine the meeting roles and responsibilities, reading techniques focus on providing guidance for the individual "preparation" phase of the inspection – where inspectors need to effectively review the given artifact and recognize defects. Reading techniques provide a systematic and well-defined way of inspecting a document, allowing feedback and improvement for both the inspector and the technique. Each reviewer takes on the perspective of a particular stakeholder of the document, using a procedural technique to focus his or her attention on only the relevant subset of defects.

    - Several items from the lessons learned analysis indicate that reading techniques are a promising choice for further work on this Initiative. Respondents felt that having the right perspectives represented on an inspection team was crucial (Section 4.3.1); not having the right people or perspectives available was cited as one of the leading causes of ineffective inspections (Section 4.3.2). Reading techniques address both of these

concerns by making explicit the necessary responsibilities and expected expertise. Perhaps more important is the assistance which the procedural approach provides to novice reviewers, to better represent these perspectives when more experienced personnel are not available. By training novices in inspection techniques at a step-by-step level, reading techniques help them get up to speed in inspections more quickly. These benefits can contribute to effective team building and cross-training, already listed as important benefits expected from inspections in Section 4.3.2. Existing reading techniques have focused on requirements and design inspections (which were shown to be of highest importance to the respondents in Section 4.1.2). Previous studies, some using NASA developers, have shown improvement in effectiveness at both the individual and team level due to using reading techniques (Basili, 1996).

- Reading techniques also provide benefits for more experienced developers. Exposure to other perspectives was cited by interviewees as one of the intangible benefits of inspection meetings. Procedural techniques that encapsulate some of the experience of another perspective provide quick access to that perspective to others. Additionally, the format of a reading technique provides a template (similar to a perspective-based checklist) for recording the outlines of specific experienced developers' perspectives.

- Current follow-up plans include making training in reading techniques available to NASA personnel and using the opportunity to collect data to verify or deny the effectiveness of the techniques in a NASA context. Information gathered from these training sessions will also be used to further tailor the techniques. (A first pilot study of the training and data-collection was run September 4, 2001, at the NASA IV&V Facility in West Virginia.)

- **Formal inspections process.** The JPL inspection process is based on training materials that have been used at JPL and elsewhere within NASA. These materials were cited by a majority of respondents, from several centers, as an important part of their inspection training.

  - Section 2.1.3 showed that a large majority of respondents had trained on the JPL formal inspection process. The impact of this training was quite far-reaching; it was consistently mentioned as a positive influence on inspection practices by the respondents. It has formed a major basis for inspection processes even when support is not available for the full-blown process. This training also includes helpful guidelines for recommending which personnel to consider involving in inspections for the valued perspectives they can provide. (Involving the right perspectives was described as a key component of effective inspection practices in Section 4.3.1.) Additionally, some respondents specifically reported that the full formality of the process was useful for ensuring that follow-up of reported defects was completed. Training in the JPL process has an excellent potential to complement the use of other standards, such as the SEL RA, which provides many suggestions about when more or less formality of the inspection process is required, but does not describe in detail what those processes should entail.

  - Current follow-up plans include offering this training to more NASA personnel. Since there is already extensive experience with this training throughout NASA, it is being considered whether a pilot study of the process is more appropriate (where we provide

long-term support and observation of the inspection approach on a real project) than trying to demonstrate its effectiveness quantitatively in a small controlled experiment outside of a real development environment.

- **Remaining Defect Estimation Techniques (RDET).** One thrust of current research on inspection is the investigation of techniques that use the results of inspections to extrapolate the remaining defect density. The goal of such a technique is to assess, based on effort already spent and defects already found, whether further inspections or re-inspections are necessary and cost-effective.

  - Since inadequate resources (a lack of management support and time/schedule resources) were cited in Section 4.3.2 as a key frustration in performing inspections, RDET could provide support for projects by minimizing the amount of effort spent on inspections, and providing some indication of the cost-effectiveness of further expenditures. Some reports in the literature have called RDET "easy-to-calculate" and "a simple but effective mechanism for obtaining a rough idea about the magnitude of potentially undetected defects" (Houdek, 2001), indicating that this could be a useful technique for development teams that would not require extensive new expertise. Such techniques also have the potential to be useful in situations involving the review of extensive amounts of work products received from another party, for example IV&V or subcontracting environments, in which an initial assessment of product quality is desired to assess the degree of review efforts necessary.

  - Current follow-up plans involve further investigation of published RDET. Initial feasibility assessments to gauge the accuracy of existing models will be taken off-line. Based on those results, a decision on whether or not to go ahead with offering training to NASA personnel and controlled experiments will be made.

Recommendations for inspection application at NASA

The report concludes with some observations about inspection use at NASA, based on a collection of common lessons learned from the participants.

The most important observation that can be made is that the majority of the participants found the use of some kind of inspection activities on their projects to be cost-effective. Somewhat surprisingly, respondents focused less on the defect detection benefits of inspections and more on the communication benefits. The most often-emphasized benefits included: getting the right people to communicate the right information; training new members and cross-training experienced developers; building a cohesive team culture. This may be because many respondents felt that there are other defect-detection activities available, such as testing; while effective communication was something that all teams felt was necessary, yet had few effective strategies for. This is not to say that inspection effort does not lead to significant defect detection, but that defect detection seems expected as a matter of course, both directly at the time of the inspection and over time as a result of a better-trained, more knowledgeable development team.

The only teams interviewed who were able to do without inspections and yet achieved the same benefits were small teams whose nature facilitated communication: small size, co-location;

stability of the group over time; and an existing knowledge of each other's areas of expertise. These teams already had all the benefits of inspections listed above, without spending time on formal inspection activities. However, it is not clear that the benefits would continue if the characteristics of the team changed. If, for example, there was high turnover of personnel or many new developers who had to be integrated into the existing team culture, communication breakdowns could occur. Also, it is not clear that those team characteristics hold for many development environments at NASA.

For these reasons, efforts should be made to give all NASA developers training in inspection techniques and encourage all development teams to make use of such techniques on their projects. Feedback from the respondents has indicated that developers and managers are convinced of the cost-effectiveness and value of inspections after participation in their first effective inspection meeting. (An additional benefit, not mentioned in this study, is that such training satisfies a Level 3 key process area of the CMMI software process improvement model.)

An important point (yet perhaps the most overlooked) is that the tailoring of inspection processes for various environments is necessary. The majority of the participants found inspection training useful though few followed the full process with much formality. Instead, they used a tailored version of the process to achieve the benefits listed above at a level of formality that seemed cost-effective to them. One of the keys to developer acceptance of formal processes was the inclusion of the developers in the tailoring process. This is especially true of projects starting up. When projects have been running for a while, new staff expect to fit into an existing process and the old staff can help the new fit in. On new projects though there is no "customary" process and buy-in from the staff is needed.

However, some of the participants felt there was benefit to be gained by doing inspections at a higher level of formality than was currently possible for the team. These respondents had previous experience performing inspections in a situation where their Center provided additional support for inspection activities. Metrics collection was often cited as an example of an activity that development teams did not have the resources to perform on their own, but which could be performed when Center-level support was provided. These participants felt there was a cost-effective benefit gained from metrics collection but that teams were unable to provide the effort themselves because 1) the benefits were not immediately available to the team, so it was hard to connect the benefits to the effort spent, and 2) the effort was required to be spent over an extended period of time, and schedule and time pressures for project deliverables made it difficult for development teams to make the sustained commitment. A key issue raised in several interviews was that many of the more detailed metrics are for organizational improvement, not project improvement. Therefore when budget and time pressures come to bear on a project, the metrics that are of no use to the project itself are jettisoned to concentrate on those efforts that do help the project. ***NASA centers should provide support to teams for metrics collection, including common forms and analysis of the data gathered.***

Finally, one technique that was consistently mentioned for performing effective inspections was that of getting the right perspectives represented by inspection participants. This was something that all teams performing inspections, regardless of the level of formality, were making some attempt to do. Of all activities associated with inspections, making an effort to get the right people to review the product was most clearly connected to the greatest list of benefits, including

communication (effort is used to get more people, possibly from outside the team, familiar with the system being built) and defect reduction (the more unique perspectives that are brought to bear on a document, the more likely it is to discover any problems that may exist). ***The guidelines for selecting perspectives from the JPL formal inspection training should be advocated for use by team managers and QA, as a starting point to get personnel thinking about what perspectives might be useful for a product review.***

Those guidelines include selecting inspectors from the developers' peers on the project; from developers in both earlier and later life cycle phases of the project; from the testing and quality assurance teams; from the user organization; from areas with which the work product has an interface; and finally as a last but important catch-all areas where communications are likely to break down.

Section 7 REFERENCES

V. Basili, S. Green, O. Laitenberger, F. Lanubile, F. Shull, S. Soerumgaard, and M. Zelkowitz. "The Empirical Investigation of Perspective-Based Reading." *Empirical Software Engineering: An International Journal*, 1(2): 133-164, 1996.

V. Basili, F. Lanubile, and F. Shull. "Investigating Maintenance Processes in a Framework-Based Environment." In *Proc. of the International Conference on Software Maintenance*, Bethesda, MD, November 1998.

CSC. *SSDM Guidebook*. April 1999.

K. Eisenhardt, "Building Theories from Case Study Research", *Academy of Management Review*, vol. 14, no. 4, pp. 532-550, 1989.

M. E. Fagan. "Design and Code Inspections to Reduce Errors in Programming." *IBM Systems Journal*, Vol. 15, No. 3, pp. 182-211. 1976.

M. E. Fagan. "Advances in Software Inspections." *IEEE Transactions on Software Engineering*. Vol. 12, No. 7, pp. 744-751. July 1986.

T. Gilb and D. Graham. *Software Inspection.* Addison-Wesley, Reading, Mass, 1993.

F. Houdek, T. Schwinn, and D. Ernst. "Defect Detection for Executable Specifications – An Experiment." In *Proc. of the 13th International Conference on Software Engineering and Knowledge Engineering*, Buenos Aires, Argentina, June 2001.

JPL. *JPL Software Development Process Description*. JPL D-15738, Revision D, November 1999.

M. Miles, "Qualitative Data as an Attractive Nuisance: The Problem of Analysis", *Administrative Science Quarterly*, vol. 24, no. 4, pp. 590-601, 1979.

NASA, Software Engineering Laboratory. *Recommended Approach to Software Development, Revision 3*. SEL-81-305. June 1992.

NASA, Office of Safety and Mission Assurance. *Software Formal Inspections Standard*. NASA-STD-2202-93. April 1993.

NASA, Office of Safety and Mission Assurance. *Software Formal Inspections Guidebook.* NASA-GB-A302, August 1993.

C. B. Seaman, V. R. Basili. "Communication and Organization: An Empirical Study of Discussion in Inspection Meetings". *IEEE Transactions on Software Engineering,* 24(7): 559-572, July 1998.

F. Shull, F. Lanubile, and V. Basili. "Investigating Reading Techniques for Object-Oriented Framework Learning." *IEEE Trans. on Software Engineering*, Vol. 26, No. 11, November 2000.

Appendix A – Fagan Inspection Process

Fagan defines five roles for participants in the process:

1. **Author**. The author of the product to be inspected by the group.
2. **Moderator**. The coordinator of the inspection process as a whole.
3. **Inspector**. A knowledgeable person who will review the product and attend the inspection meeting.
4. **Reader**. The leader of the inspection meeting.
5. **Recorder**. The recorder of the defects and open issues found at the meeting.

The process

The Fagan process comprises six operations: planning, overview, preparation, inspection, rework, and follow-up.

Planning
The moderator checks that all of the entry criteria have been met. The moderator selects the inspectors, and schedules the inspection meeting. The author or moderator ensures that the inspection package is assembled and distributed to all of the inspectors.

Overview
The author runs a one to two hour meeting to familiarize the inspectors with the appropriate aspects of the system. If every member of the inspection team is part of the project team, this step is not usually necessary.

Preparation
Each inspector works through the inspection materials individually to prepare for the meeting.

Inspection Meeting
The moderator may schedule multiple meetings to provide enough time to review all the materials. No single meeting should take more than two hours. The moderator or recorder makes a list of defects categorized into some classification scheme, consisting of at least major and minor errors. The discussion should concentrate on finding problems, *not* on their solution.

Rework
The author repairs the defects found at the meeting, and addresses any other open issues.

Follow-up
The moderator checks the revised product against the exit criteria.

Appendix B – Inspections Questionnaire

Name:
Position:
Organization:
Phone/email address:

Background on Your Organization

Please answer the following questions to the best of your knowledge. For the purposes of this survey, an "inspection" is defined as any "technical examination process during which a product is examined with the purpose of finding and removing defects as early as possible in the software life cycle." For this survey, inspections can range from someone reviewing a software document to find defects with little or no formal guidance, to highly formal inspection meeting processes.

1. What does your organization do?

2. What is your current role in the organization?

3. How long have you been involved with this organization?

4. In what domain would you classify your projects?
[ ] flight software [ ] control centers [ ] mission planning & scheduling
[ ] flight dynamics [ ] data capture & processing [ ] other:
_____

5. What are the 3 most important product qualities for the systems you work with?
[ ] functionality [ ] usability [ ] efficiency
[ ] reliability [ ] maintainability [ ] portability
[ ] other (Please explain:_____)

6a. What have you observed to be the major sources of errors in your products under development? (Please rank most to least significant, where 1=most significant) ____ misinterpreted requirements ____ missing requirements ____ changing requirements ____ design errors ____ coding errors ____ interface problems ____ environment problems ____ other _____	**6b**. In your project(s), why are estimates of schedule, cost, or system size sometimes exceeded? (Please rank these causes from most to least significant, where 1=most significant) ____ unstable requirements ____ inexperienced people ____ ill-defined process ____ inappropriate/no tools ____ inadequate staffing ____ poor management ____ other _____

Inspections at Your Organization

7. In your current project(s), do you participate directly in inspections? Circle one: Yes; No.

8. How many inspections have you participated in at this organization?
Circle one: 0-2; 3-5; 6-10; >10

9. Over what time period did you participate in these inspections? (e.g. "1995 to present")

10. What, if any, kinds of training in inspection techniques or procedures have you received?
Check all that apply: □ Formal class or course; □ Computer-based training; □ On-the-job training;
□ Other:_____

11. What kinds of work products get inspected within your projects? Check all that apply:

- □ Requirements documents
- □ Prologs or package specifications
- □ Architecture design diagrams
- □ Detailed design diagrams
- □ Code
- □ Other _____
- □ Test plans
- □ Tests for procedures and functions
- □ Test results
- □ Users Guides
- □ System Documentation

12. If applicable, what do you look for when you participate in a *requirements* inspection? Rank by priority, where 1 = most important:

- __ Inconsistencies with high-level requirements documents or statements of customer needs
- __ Omitted requirements
- __ Omitted functionality
- __ Other_____
- __ Unclear or contradictory requirements
- __ Deviations from requirements standards
- __ Non-testable requirements

13. If applicable, what do you look for when you participate in a *design* inspection? Rank by priority, where 1 = most important:

- __ Inconsistencies with requirements specifications
- __ Omitted functionality
- __ Internal inconsistencies
- __ Other _____
- __ Interface errors
- __ Deviations from design standards
- __ Data definition/handling errors

14. If applicable, what do you look for when you participate in a *code* inspection? Rank by priority, where 1 = most important:

- __ Inconsistencies with design
- __ Logic errors
- __ Interface errors
- __ Syntax errors
- __ Data/variable definition errors
- __ Other _____
- __ Unclear code
- __ Non-reusable code
- __ Non-testable code
- __ Deviations from coding standards
- __ Data passing (e.g., argument) errors

15. How long do you typically take preparing for an inspection (i.e. understanding the work products being inspected and finding defects on your own)?
Circle one: <15 minutes; 15-30 minutes; 30 minutes – 1 hr.; 1 – 3 hours; >3 hours

16. How long does a typical inspection meeting last?
Circle one: 15-30 minutes; 30 minutes – 1 hr.; 1 – 2 hours; 2 – 4 hours; >4 hours

17. Are work products ever reinspected? Circle one: Usually, Occasionally, Seldom, Never

18. Are the defects discovered during inspections corrected?
Circle one: Usually, Occasionally, Seldom, Never

19. Does your project or organization collect any measures of the inspection process or inspection results? Circle one: Yes, No, Don't know If "Yes", what measures are collected? Check all that apply:

- □ Number of products inspected
- □ Lines of code inspected
- □ Number of defects found
- □ Types of defects found (e.g., interface defects, logic defects)
- □ Other _____
- □ Number of products requiring reinspection
- □ Time spent in inspection session
- □ Number of inspectors
- □ Defect insertion points (e.g., defect in requirements, design, code)

Projects using Inspections

20. For the projects on which you have been involved in inspections:

a. How many people were on a *typical* development team? Circle one: <5; 5-10; 10-20; >20

b. What was the *typical* length of a project, in months? Circle one: <6; 6-12; 12-24; >24

c. Is work typically distributed across multiple sites? Circle one: Yes; No

21. For these projects:

a. What programming languages were used? Circle as many as relevant:

C; C++; Java; Ada; Fortran; VB; Other:_____

b. What percent of the projects were mission critical? _____

(Mission critical is defined as: "Software where failure could cause mission failure (includes significant unrecoverable data loss), harm to humans, damage to facilities or equipment, or cause risk to the organization's public reputation, or software otherwise designated as mission critical.")

c. What percent of projects are:

_____% new development.	How many deliveries are typically planned? _____ What customer reviews are typically part of the process?
_____% enhancement/ maintenance of older systems.	Can you estimate the typical number of releases per year?
_____% other:	Describe:

When you are done with this survey, please return it to:

Forrest Shull Fraunhofer Center -- Maryland University of Maryland 4321 Hartwick Road Suite 500 College Park MD 20742-3290 Fax: (301) 403-8976 Email: fshull@fc-md.umd.edu	**- or -**	Judith Bachman SEL/CSC Goddard Space Flight Center Mail Code 581 Building 23/N218 EE Greenbelt Maryland 20771 Email: jbachman@csc.com

Appendix C – Predefined Interview Questions

INTERVIEW NOTES

Instructions to the interviewers are enclosed in [square brackets].

[Begin by asking the interviewee to answer to the best of his/her knowledge. "Don't know" is a valid answer! Remind the interviewee of the definition of an "inspection": any "technical examination process during which a product is examined with the purpose of finding and removing defects as early as possible in the software life cycle."]

General

1. Are there process standards or guidelines that cover how inspections should be performed in your organization?

    a) *If so*, have your projects followed this standard or guideline? Was it useful for getting started and/or as an ongoing reference?

    b) *If not*, why not?

2. On the questionnaire, you described the work products that get inspected at your organization. Please discuss a little bit about which of these have the highest payoff on inspections. Which of these receive the most effort or formality in the inspections?

3. What kinds of activities are performed during a typical inspection? Please estimate the amount of effort you use for each activity. How easy it is to perform each of these activities effectively? [Use the categories below to organize responses. If the interviewee doesn't bring up an activity on his/her own, ask them: "Do you spend any time on…"]

    a) Distributing the work products to be inspected

    b) Inspectors individually reading or inspecting the work products for defects

    c) Inspectors meeting to discuss their individual findings

    d) Inspection leader conducting a walkthrough of the work products

    e) Documenting inspection results

    f) Noting and assigning action items resulting from the inspection

    g) Tracking action items resulting from the inspection. *Are action items tracked, or meeting notes reviewed to capture lessons learned?*

    h) Other

Planning

4. Are inspections used on all projects you've been involved in, in this organization? If not, is there a particular difference between projects for which inspections are used (that you described on your questionnaire) and projects where they're not?

5. Who participates in inspections in your project or organization? [If the interviewee doesn't bring up all of the following responses on his/her own, ask them: "Do any of the following ever get involved in inspections…"]

   a) Authors of work product being reviewed

   b) Peers of the authors

   c) Members of the spacecraft project team

   d) Members of the development team

   e) Team lead

   f) Project manager

   g) Quality assurance representative(s)

   h) End users or their representatives

   i) Other

6. Who usually decides on what exactly should be inspected and selects the inspectors? [If the interviewee doesn't bring up all of the following responses on his/her own, ask them: "Do any of the following have any say in planning inspections…"]

   a) Project manager

   b) Team lead

   c) Technical expert on work products to be inspected

   d) Author of work products to be inspected

   e) Quality assurance representative

   f) Other

7. Do you ever get a work product that you would refuse to review in its current form? What might make you decide this?

Preparation

8. On the questionnaire, you estimated the time required to prepare for an inspection. How do you typically spend this time? What factors can increase or decrease the preparation time?

Inspecting the work product

9. Are there distinct roles that people play in an inspection? Are all reviewers looking for the same things in the work product?

10. Is there a lot of difference in the work products you see from one inspection to the next?

    a) *If so*, are certain types of work products easier to inspect, or have more productive inspections?

11. Are there other people or resources that you consult while performing an inspection?

12. Are there job aids that you use during inspections (e.g. checklists, automated data collection)? Are they helpful?

    a) *If checklists,* were they tailored by your team? If so, could we get a copy?

Results

13. What would you say are the main benefits to your organization from doing inspections? Are there conditions under which you aren't likely to see those benefits? [If the interviewee doesn't bring up all of the following responses on his/her own, ask them: "Do any of the following result from inspections…"]

    a) Improve requirements

    b) Reduce numbers of defects

    c) Improve design

    d) Reduce rework necessary

    e) Reduce time required for testing

    f) Improve code

    g) Improve documentation

    h) Improve testing

    i) Other

14. Have you ever had a situation where inspections were skipped, and a problem was allowed to slip to a later phase as a result?

15. Do you think the time spent on inspections for your projects is generally well-spent? Do you ever have inspections where you feel the time spent isn't worth the payoff? Do you have any ideas what causes those situations?

16. Are there any problems or frustrations that you have with inspections?

Appendix D – Abbreviations and Acronyms Used in This Report

CSC	Computer Sciences Corporation (contractor to GSFC)
GRC	Glenn Research Center
GSFC	Goddard Space Flight Center
IV&V	Independent Verification and Validation
JPL	Jet Propulsion Laboratory
JSC	Johnson Space Center
LRC	Langley Research Center
NASA	National Aeronautics and Space Administration
QAO	Quality Assurance Office
RA	Recommended Approach
SEAS	System Engineering Analysis Support (the CSC contract under which the SSDM methodology was developed)
SEL	Software Engineering Laboratory
SSDM	SEAS System Development Methodology (originally developed for the SEAS contract at CSC)