NASA Software Engineering Initiative Implementation Plan

Responsible Office: Code AE/Office of the Chief Engineer

original signed by *1/11/02*

_____ _____

W. Brian Keegan **Date**
Chief Engineer

NASA Software Engineering Initiative Implementation Plan Change History

Change	Date	Description

NASA Software Engineering Initiative Implementation Plan

Table of Contents

1. Introduction

Software engineering is a core capability and a key enabling technology necessary for the support of NASA's Enterprises. Ensuring the quality, safety, and reliability of NASA software is of paramount importance in achieving mission success. Through surveys[1] and assessments, many software challenges within the Agency have been identified and documented. Additionally, the exponential growth in the scope, complexity, and importance of software within NASA systems experienced over the years is expected to continue, challenging our ability to manage it effectively. As a result, the NASA Software Engineering Initiative[2] was formed. This Plan specifies how the NASA Software Engineering Initiative will be implemented. In coordination with Center Software Engineering Improvement Plans, this Plan defines a NASA-wide comprehensive approach for improving software engineering to a quantifiable maturity level commensurate with mission criticality in order to meet the software challenges of NASA.

2. Goal

The goal of this Plan is to advance software engineering[3] practices to effectively meet the scientific and technological objectives of NASA.

This initiative strives to achieve the following through advancing software engineering practices:

- Improved cost and schedule predictability. Accurate schedules and budgets will ensure that software engineers are provided with adequate resources and realistic schedules to develop and maintain NASA products.
- Improved software reliability.
- Improved software quality.
- Reduced software cost.

3. Guiding Principles

NASA will employ common frameworks for software process improvement.

- Standard methods: Initiating, Diagnosing, Establishing, Acting, and Learning (IDEAL) Model; Process Change Method (PCM); Software Acquisition-Capability Maturity Model (SA-CMM); Software-Capability Maturity Model (SW-CMM); CMM Integrated (CMMI); Institute of Electrical and Electronics Engineers (IEEE) 12207,Standards for Information Technology-Software Life Cycle Processes; and International Organization for Standardization (ISO) 9000. The IDEAL Model and

---

1 Reference "Improving the Current State of Software within NASA" developed by the NASA Software Working Group, presented by Pat Schuler to the NASA Chief Information Office, March 2000, URL: http://ip-strategies.jpl.nasa.gov/

2 Reference the "NASA Initiative for Software Safety and Quality" presented by Lee Holcomb to the NASA Senior Management Council, April 12, 2000, for the presentation that introduced this Initiative and for the specific rationale for the Initiative.

3 For the purposes of this Plan only, the term "Software Engineering" refers to software development, assurance (i.e., safety, reliability, and quality), and management.

PCM provide guidance and structure for Agency and Center-level Software Process Improvement plans.

NASA will employ standard methods of assessment.

- CMM–Based Appraisal for Internal Process Improvement (CBA-IPI)
- Software Capability Evaluation (SCE)—performed by external source.
- Standard CMMI Assessment Method for Process Improvement.

NASA recognizes that the Software Engineering Institute (SEI) Software Capability Maturity Model® (CMM®)[4] and CMM-Integrated (CMMI℠) provide a proven framework for advancing software engineering practices and are key elements in achieving this Plan's goal. The criteria in Appendix D (and in pending NPG 2820, NASA Software Guidelines and Requirements) specifies the use of CMM or CMMI as a standard measurement of the capability to produce quality software engineering products. Appendix D shows the Centers' and offerors' flexibility in implementing software process improvement. The CMM's will be used as a benchmark to objectively measure progress toward software improvement; however, simply achieving a CMM/CMMI Level rating is not the objective. Software products must effectively meet NASA's scientific and technological objectives. As part of this initiative, metrics will be gathered to quantitatively measure improvements in software processes and products.

For software process improvement to be effective, it must be continuous and should not be considered complete, upon the establishment of any particular assessment milestone. The Centers' plans discussed below should prioritize software engineering improvement activities according to organization and program needs, making appropriate use of both CMM- and non-CMM-based software engineering improvement activities. The Centers' plans will include activities to infuse software research results as well as new methods and tools to advance their software engineering practices.

4. Scope

The scope of this Plan covers software process improvement as well as items related to software research; software safety, reliability, and quality; attraction and retention of software engineers; and improving NASA's software engineering knowledge and skills. This Plan covers both mission-critical and nonmission-critical software. Each Center will define the extent to which this Plan will be applied to their own nonmission-critical software classes. Improvements related to systems engineering will be addressed by the NASA Systems Engineering Working Group (SEWG) in coordination with the NASA Software Working Group (SWG)[5].

5. Ownership of Plan

---

[4] *The Capability Maturity Model: Guidelines for Improving the Software Process*, Software Engineering Institute at Carnegie Mellon University, Addison-Wesley (1994), ISBN 0-201-54664-7. Capability Maturity Model and CMM are registered (®) in the U.S. Patent and Trademark Office. Reference, herein, to any specific commercial product, process, or service by trade name, trademark, or otherwise, does not constitute or imply its endorsement by the United States Government.

[5] The NASA SWG is chartered under the auspices of NPD 2820.1, NASA Software Policies.

NASA's Office of the Chief Engineer (OCE) is the sponsor of the NASA Software Engineering Initiative and this supporting implementation Plan. OCE is responsible for coordinating the creation, execution, and maintenance of this Plan and is accountable for its overall success. This Plan will be revised on an annual basis and as deemed necessary, based on lessons learned in implementing the strategies.

NASA's Chief Information Officer (CIO), the Office of Safety and Mission Assurance (OSMA), and Institutional Program Offices (IPO) jointly support the Software Engineering Initiative.

Centers shall coordinate with their IPO's on Center Software Engineering Improvement Plans.

Activities performed under this Plan will leverage off and closely coordinate with research and software activities funded by the CIO, OSMA and the Office of Aerospace Technology (OAT).

Each Center Director is accountable for the success of their Center's Software Engineering Improvement Plan.

The SWG has a significant role in the development, coordination, maintenance, and deployment of this Plan under the leadership the OCE.

The purpose of the SWG is to develop and oversee the formulation and implementation of an Agencywide plan to work toward continuous, sustained software engineering process and product improvements in NASA; and to ensure appropriate visibility of software issues within the Agency. NASA SWG Charter, URL http://swg.jpl.nasa.gov/charter/index.shtml.

6. Summary of Strategies, Objectives, and Approaches

This section defines the strategies, objectives, and approaches to achieve the goal of this Plan. The strategies state how the goal will be achieved. The objectives provide measurable, time-related statements of what is to be accomplished. The approaches provide a set of tasks used to achieve the objective. Appendix A: Schedule Summary provides a high-level overview of the Plan schedule and Appendix B: Funding Breakdown contains a detailed profile of the funding for this Plan. The numbering of strategies and objectives does not necessarily reflect their order of priority. The schedule for objectives, approaches, and tasks, as well as the associated responsible groups for each, are provided at URL: http://ip-strategies.jpl.nasa.gov/. The Approach Details, a file containing a summary of the major deliverables and the resources required to achieve the approaches in FY 2002, is also provided at the same URL.

Strategy 1. Implement a continuous software process and product improvement program across NASA and its contract community.

S1. Objective 1. Have an approved Center Plan for each Center for advancing software engineering practice by September 2001, and annually updated each August.

S1.O1. Approach 1. Develop and refine Center Plans, align with Enterprises, and obtain Center Director and OCE approval. Examine Information Technology Security status at the Center and determine if any activities related to implementing the following need to be incorporated into the Center Plan: IT Security policies, procedures, and guidance for ensuring the security of the Agency's IT resources (data, information, applications, and systems) as defined in NPG 2810.1, Security of Information Technology.

S1. Objective 2. Monitor and track implementation of NASA Software Engineering Initiative Improvement Plan and annually revise the Plan. The first revision will be by March 2003.

S1.O2. Approach 1. Revise OCE's Program Operating Plan (POP) proposals and review Center Plan progress with OCE and SWG. The SWG summarizes the data from Center progress reports for the Agency's Annual Status Report Presentation. Appendix C: Center Plan Required Content contains an outline with the required content to be included in each Center Plan submitted to OCE. Center Plans must adhere to the numbering scheme and titles provided in the outline. Additional sections may be added at the end of the Center Plans.

S1.O2. Approach 2. Review progress on NASA Software Engineering Initiative implementation against the Plan. Analyze status data on all Plan strategies, obtain feedback from Center Software Engineering Process Groups (SEPG)[5], review failure reports, NASA Software Metrics Report, Independent Verification and Validation Summary Report, and usage data. Deliver briefing to the Engineering Management Council and appropriate Headquarters elements on progress and recommendations on Plan changes. OCE approves Plan updates.

S1. Objective 3. Put initial infrastructure in place to support the NASA Software Engineering Initiative by January 2002. Maintain ongoing infrastructure activities according to the annually updated NASA Software Engineering Initiative Implementation Plan.

S1.O3. Approach 1. Put Agency infrastructure in place. This includes gaining advocacy from the Enterprises; securing a budget; working with the Training and Development Division to secure training, putting NPG 2820.x, NASA Software Guidelines; NPD 8730.4, NASA Software Independent Verification and Validation Policy; and NPG 8730.x, Software Independent Verification and Validation (IV&V) Management, in place; and coordinating activities between the SEWG and SWG.

S1.O3. Approach 2. Put the SWG infrastructure in place. This includes securing the full level of support from the Centers for the SWG, training SWG members to become subject matter experts in software process improvement and CMM, developing contract wording for Centers, providing information exchange opportunities between Center SEPG's, providing a process asset

[5] The SEPG is the focal point for software process improvement at the Centers. The number of groups, their makeup, and management reporting structure are at the discretion of the Center Director.

library for the Centers to use, putting Center infrastructure in place, drafting and revising NASA standards, policies, and guidelines, and supporting OCE in the management of this Plan.

S1. Objective 4. Develop and phase in a long-term metrics collection and analysis program and fully implement it by October 2003. Metrics analysis and identification of new participating projects are performed annually in January.

S1.O4. Approach 1. Establish project metrics collection capability. This includes designing an initial set of metrics and metrics reports, developing a metrics collection system, piloting the metrics and using the results to develop an operational Agencywide system.

S1.O4. Approach 2. Establish Agency-level metrics analysis capability. This includes developing advanced metrics reports, documenting analysis methods, and developing metrics analysis training.

S1.O4. Approach 3. Roll out the Agency Metrics Program. This includes briefing Senior Management, delivering metrics training, and phasing in project participation.

S1.O4. Approach 4. Initiate and maintain regular metrics collection and reporting of analysis results. This includes the continuing functions of entering the metrics, maintaining the helpdesk, performing the analysis, and developing the reports.

S1.O4. Approach 5. Refine metrics set and collection techniques, if necessary, based on trends and user feedback.

Strategy 2. Improve safety, reliability, and quality of software through the establishment and integration of sound software engineering principles and standards.[6]

S2. Objective 1. Update and extend Agency policy, standards, and guidebooks on safety, reliability, and quality of software by May 2003, with periodic updates.

S2.O1. Approach 1. Collaborate with OSMA on the update of Software Safety, Reliability, and Quality Assurance (SW SR&QA) policy based on requirements of NPD 2820.1, NASA Software Policies; NPG 2820.x; NPG 8730.x, IEEE 12207; and CMM/CMMI. This includes performing a gap analysis to determine differences between existing and required roles, processes, and tasks.

S2.O1. Approach 2. Identify missing/needed software engineering principles, tools, and techniques for improving safety, reliability, and quality of software and infuse them into standards, and guidebooks.

[6] OSMA is responsible for the maintenance of policy and guidelines regarding software safety, reliability, and quality. This strategy is accomplished through effective collaboration between OSMA and other NASA organizations.

S2. Objective 2. Put in place mechanisms to capture, document, and promote proven software engineering practices for improving safety, quality, and reliability of software by January 2003, with periodic updates.

S2.O2. Approach 1. Encourage submittals of software engineering best practices and lessons learned to the NASA Lessons Learned Information System.

S2.O2. Approach 2. Create/Utilize forums to have Software Safety, Reliability, and Quality Assurance (SW SR&QA) engineers and software engineers meet to present and share problems and successes. Strategy 4, Objective 2, Approach 1 will be utilized to perform this Approach.

S2. Objective 3. Collaborate with OSMA in the establishment of a continuous improvement program for the advancement of software safety, reliability, and quality research by April 2003, with annual reviews. This objective will be conducted through exercising Strategy 3, Objective 1, Approach 2.

S2. Objective 4. Collaborate with OSMA in the establishment of an infrastructure to improve software safety, reliability, and quality assurance across the Agency and Centers by July 2004, with annual updates.

S2.O4. Approach 1. Provide assistance to OSMA in the management, oversight, and control of software quality assurance, safety, and reliability policies across the Agency.

S2.O4. Approach 2. At each Center, incorporate the functions defined in the updated Agency SW SR&QA policy and initiate the functions. Each Center augments its Center Plan to address this.

S2.O4. Approach 3. Process Verification Audits are reviewed to determine Center compliance with Agency SW SR&QA policy.

Strategy 3. Improve NASA's software engineering practices through research.

S3. Objective 1. Implement and transfer mature software engineering research results and new technologies to operational use within NASA, starting January 2002, and annually updated each October.

S3.O1. Approach 1. Identify a limited number of currently funded mature Agency research products that demonstrate exceptional potential for adding value to software engineering within NASA.

S3.O1. Approach 2. Identify promising new tools and technologies for pilot infusion within NASA.

S3.O1. Approach 3. Match the identified research products and new technologies with the Centers' interests and needs. Support pilots and transfer the identified research products and new technologies into NASA projects. Obtain support for pilot studies and initial startup training. Support the migration of successful research product and new technology pilots across NASA Centers.

S3.O1. Approach 4. Coordinate with key NASA organizations to utilize Strategy 4, Objective 2 to aid in transition of mature research products and promising new tools and technologies.

S3.O1. Approach 5. Communicate lessons learned[7] about technology transfer of research results.

S3. Objective 2. Cooperate with OAT[8], OSMA, NASA Centers, and other Enterprise programs to identify current software research and identify needed research areas (new tools, techniques, processes, methods) based on documented software problem areas, by January 2003, and annually updated in January.

S3.O2. Approach 1. Identify and characterize[9] the current software research performed by NASA.

S3.O2. Approach 2. Document deficiencies, and problems identified by Provide Aerospace Products and Capabilities (PAPAC) projects, SEPG Surveys, and NASA Software Metrics to identify potential research needs in software engineering.

S3.O2. Approach 3. Identify emerging software technologies and mission requirements that have the potential to significantly impact software research needs.[10]

S3.O2. Approach 4. Prioritize research needs for addressing the identified deficiencies and problems in software engineering. To the extent possible, document which prioritized research needs are being covered by NASA, other Government agencies, industry, or academia and which ones are not being addressed.

S3.Objective 3. Solicit support to address the highest priority software engineering research needs which at the present time are insufficiently addressed, by March 2003, and annually updated in March.

---

[7] E.g. The role of a long-term champion and subject matter expert to aid in advertising the product, aid in the insertion and mentoring of new technologies.

[8] OAT is responsible for performing leading edge research and advanced computing systems and user environments, revolutionary software technologies, and pathfinder applications that enable the achievement of aerospace technology goals.

[9] Characteristics include the research rationale, planned research products, and the primary point of contacts.

[10] Schedule Note: S3.O2.A3 will begin activities in FY 2003.

S3.O3. Approach 1. Communicate and advocate highest priority software engineering research needs with key NASA organizations at the onset of their funding cycle. In addition, offer assistance in reviewing proposal submissions to provide an integrated viewpoint of the Agency's software engineering problems.

S3.O3. Approach 2. Offer expertise in reviews of research products to improve the maturation and technology transfer.

Strategy 4. Improve software engineers' knowledge and skills and attract and retain software engineers.

S4. Objective 1. Identify training courses required to increase the level of knowledge of software engineers and establish a schedule of course offerings by June 30, 2002.

S4.O1. Approach 1. Establish a process to annually prioritize training needs and coordinate with the Training and Development Division and Center training offices for funding. Inputs from various sources including survey results will be used to create the prioritized list.

S4.O1. Approach 2. Update and maintain the Course Matrix of “NASA Software Training Course Listing,” NASA-TM-209370. This TM, a product of the SWG, describes courses that are available from a core group of providers for software managers, software engineers, and software assurance engineers. The course listing is based upon the Key Process Areas of the CMM. After surveying NASA courses (developed at the Agency and Center levels) and public course offerings, identify subject areas where no courses currently exist and evaluate the need for developing a course.

S4.O1. Approach 3. Recommend updates to Academy of Program and Project Leadership (APPL) courses, if required. The SWG will work with Training and Development Division to identify APPL courses in which to update or add software content to reflect appropriate aspects of the Initiative.

S4. Objective 2. Provide more opportunities for information exchange with other software engineers and for keeping current with state of the art and state of the practice by September 30, 2002.

S4.O2. Approach 1. Advocate the use of SWG deliverables (e.g., Web sites) as opportunities for information exchange among software engineers. The SWG and support staff will sponsor and set up periodic information exchange sessions for NASA software engineers and provide related Web sites for followon user groups and references for other exchange and networking opportunities.

S4. Objective 3. Collaborate with the responsible Headquarters codes to advocate the establishment and use of incentives to attract and retain top quality software engineering professionals by December 1, 2002.

S4.O3. Approach 1. Establish communications with other groups existing in the Agency that are advocating better compensation for software engineers. The SWG identifies other Agency initiatives and organizations that are working towards a similar objective and collaborate where possible.

S4.O3. Approach 2. Develop Dual Career Ladder (DCL) promotion evaluation criteria for engineering and advocate its use at Centers that do not already have a DCL for engineering. Investigate the development and promotion of an Agencywide DCL.

S4.O3. Approach 3. Review Agency Awards program and develop recommendations to expand it with new awards for software engineering. The SWG will work with the Inventions and Contributions Board to review the Agency Awards Program for software-relevant awards and investigate the creation of new awards that recognize software process improvement successes.

7. Funding

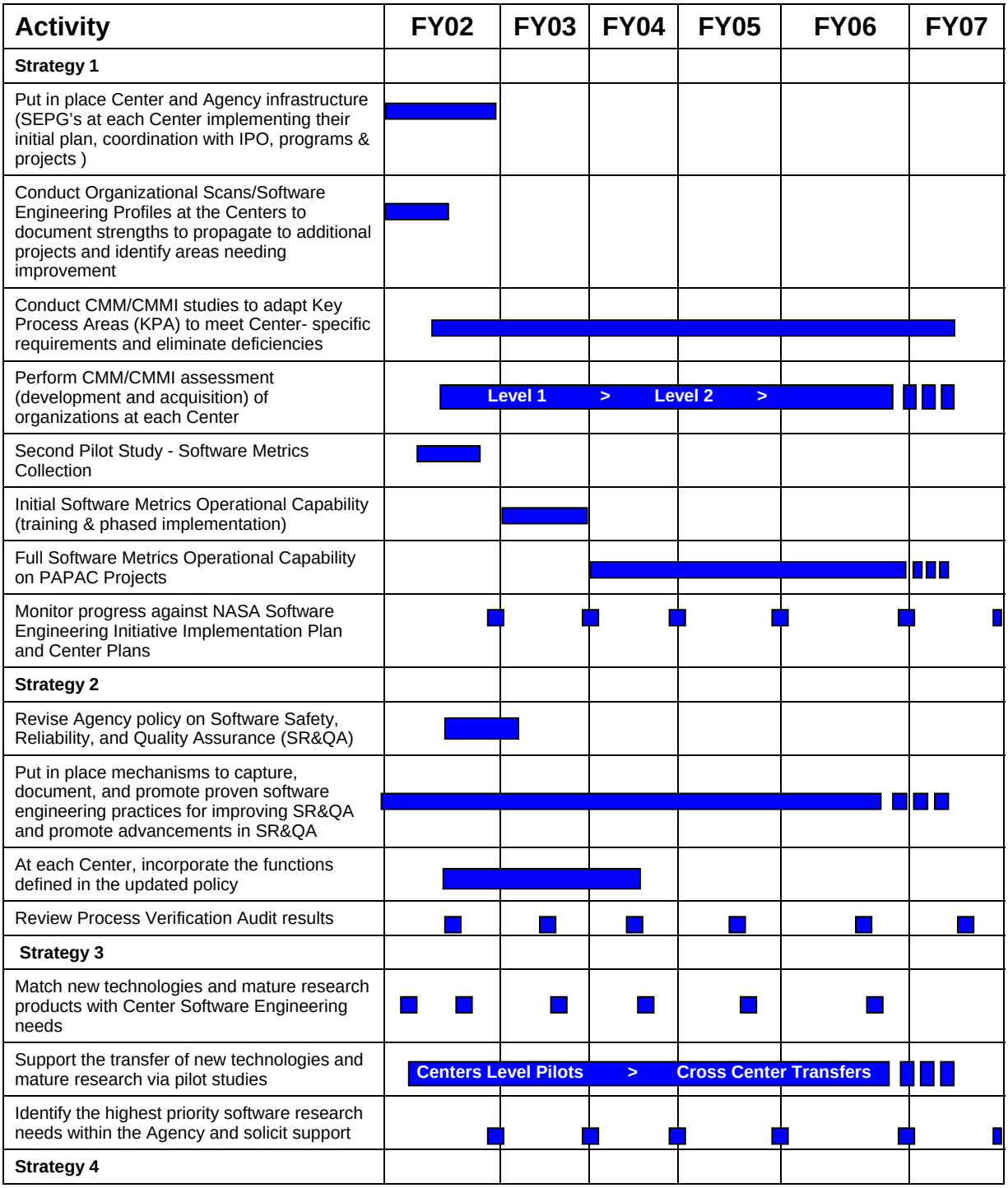
The following table shows OCE dollar values by fiscal year for funding the NASA Software Engineering Initiative Implementation Plan. Appendix A: Schedule Summary provides a high-level overview of the Plan schedule and Appendix B: Funding Breakdown contains a detailed profile of the funding for this Plan. On an as-needed basis, Centers will provide their own collateral funding over and above OCE funding to achieve their Center Software Engineering Improvement Plan goals. OCE-managed dollars provide the core funding for the initiative; the amount of additional collateral funds needed at each Center is determined on an individual Center-by-Center basis.

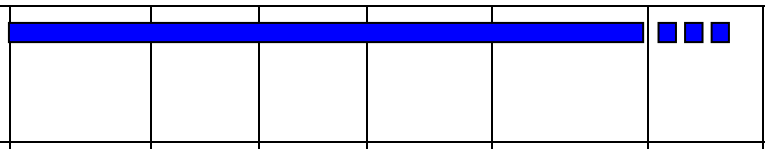
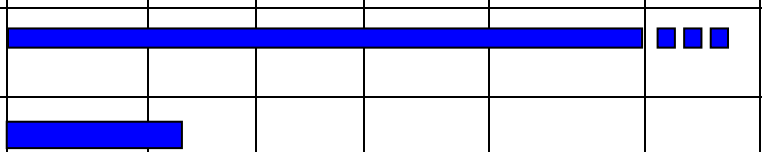
< Table deleted for general distribution>

Table 7-1. Funding by Fiscal Year ($ Millions)

Appendix A: Schedule Summary

The following table provides a high-level overview of the schedule for this initiative Plan.



Work closely with the Training and Development Division to supply the training necessary for successful implementation of the NASA Software Engineering Initiative Implementation Plan	
Provide opportunities for information exchange between software engineers on state of the art and state of the practice	
Advocate the establishment and use of incentives to attract and retain top quality software engineering professionals	

Appendix B: Funding Breakdown

As stated above, the goal of this Plan is to advance software engineering practices to effectively meet the scientific and technological objectives of NASA. To initiate and make progress toward that goal, substantial resources and effort are needed at both the Agency and Center levels over the next 6 years. Subsequent funding, after the completion of this Plan, to maintain software engineering practices will be relatively nominal, with only a small core of Agency funds set aside to maintain and coordinate the long-term elements of the Plan. These long-term elements include Agency-level policy and guideline updates, software metrics collection and analysis, assessments, and coordination and dissemination of improvement, research, and training products and activities among Centers. After the completion of this Plan, the Centers and IPO's will sustain local initiative elements to meet the needs of evolving challenges in software engineering.

The following table shows the breakdown of OCE projected core dollar values by fiscal year for funding the NASA Software Engineering Initiative Implementation Plan. Actual expenditures will be tracked and resource needs will be reassessed on an annual basis. In the initial years, new process improvement activities will be piloted at the Centers. Once experience is gained in implementing software engineering capability improvements, funding levels increase to perform improvements across all mission critical organizations. As the organizations complete their key software engineering capability improvements, funding levels decreases accordingly.[11]

<Budget table deleted for general distribution>

[11] On an as-needed basis, Centers will provide their own collateral funding over and above OCE funding to achieve their Center Software Engineering Improvement Plan goals. OCE-managed dollars provide the direct core funding for the initiative, the amount of additional collateral funds needed at each Center is determined on an individual Center-by-Center basis.

Appendix C: Center Plan Required Content

Center Software Engineering Improvement Plans (hereafter referred to as Center Plans) must include the following content:

i. Revision Record

1. Introduction

2. Goal

3. Scope

Includes the following:

- Specify the organizational elements implementing this Plan; at a minimum include all Center organizations responsible for the performance of mission-critical software development, management, and acquisition.
- Specify the Center's tactic for phasing in improvements (e.g., domain phasing, and organizational phasing).

4. Ownership of Center Plan

Includes the following:

Define how the Center Plan progress will be monitored and who is responsible. Include a process improvement organizational chart.

- Identify how oversight of Center Plan will be conducted (e.g., planned verses actual progress in Center Plan).
- Name of the Center software 'Champion' {the SWG primary representative or named delegate takes on this role}.
- List the organizations to be represented on the Software Engineering Process Group(s) (SEPG) and Management Steering Group (MSG). The number and organizational representation of SEPG(s) and the MSG(s) are determined by the individual Centers. However, the Center's Software Quality Assurance organization must be represented on the SEPG.

5. Strategies and objectives

Includes the following:

Identify how the Center Plan will support the implementation of all strategies of the NASA Software Engineering Initiative Implementation Plan.

6. Schedule

The schedule should include--

- Approaches and tasks for meeting the strategies and objectives for upcoming fiscal year. As Center Plans are updated annually, these will be added for subsequent years. Out years should be covered at a higher level.
- Tasks and milestones for meeting the criteria listed in the NASA Software Engineering Initiative Implementation Plan (See Appendix D).

- Date when Center SEPG(s) will be established to focus on initiative implementation.
- Date when Center MSG(s) will be established to focus on initiative implementation.
- Work Breakdown Structure (WBS) and associated Full-Time Equivalent (FTE) estimates for each major WBS element.
- Training activities.
- Assessments against the Software CMM or CMMI for all Center organizations responsible for the performance of mission-critical software development and management.
- Assessments against the Software Acquisition CMM for all Center organizations responsible for individuals performing mission-critical software acquisition and oversight.
- Milestones for documenting the current software engineering state/baseline/profile.
- Milestones for determining which software engineering elements from the baseline will be improved and when plans for improving those selected elements will be documented. The Center Plans should prioritize software engineering improvement activities according to organization and program needs, making appropriate use of both CMM- and non-CMM-based software engineering improvement activities (e.g., Requirements Management, and Risk Management).
- Technology transfers planned for infusion and implementation.
- Tasks that will flowdown the requirements from NPG 2820, NPD 8730, and NPG 8730 (when approved) to NASA and contractor developed software and related ISO procedures.
- Checkpoints for evaluating changes in mission critical software systems'
    a. cost estimation
    b. schedule estimation, and
    c. defect density.
- Date by which measures will be in place to ensure that the NASA Software Engineering Initiative Implementation Plan and the Center Plan are being followed. (e.g., Date which the Center software Champion will begin sitting on the Center GPMC's to assure program and project compliance with the NASA and Center Plans).
- Task for performing Annual Status Report on Center Plan to OCE. This includes the SEPG feedback of successes and concerns.
- Tasks for the identification of external partners (including international partners) that are supplying mission-critical software and whether or not the Center can negotiate with the partner to have mission-critical software be accomplished by Level 3 organizations. Define how the Center will attempt to have all plans and agreements with external partners conform to the criteria listed in the NASA Software Engineering Initiative Implementation Plan for mission critical software.)

Appendix D: Criteria for Using CMM[13] As A Benchmark In Support Of This Plan's Goal

The criteria in this Appendix (and in pending NPG 2820) specifies the use of CMM or CMMI as a standard measurement of the capability to produce quality software engineering products. It shows the Centers' and offerors' flexibility in implementing software process improvement.

	NASA	New Contractor	Current Contractor
Organizations responsible for individuals performing <u>mission critical software</u> management, development, or acquisition	Each Center shall develop and implement a plan to advance software engineering capability for their organizations. Each Center shall show ongoing progress toward software process and product improvement goals in the Center plan and against CMM[12] Level 3 showing increase in capability over time. Center organizations that develop systems containing software shall be periodically assessed against CMM Level 3 capability as a benchmark. Center organizations acquiring systems containing software will be periodically assessed against the Software Acquisition CMM.	NASA's preference is that contractors be measured at CMM Level 3. NASA will specify in Request For Proposals (RFP) that mission-critical software be developed by CMM Level 3 organizations (as measured by a CMM appraisal[13] by a Software Engineering Institute authorized lead appraiser from an external organization) **<u>or</u>** require offerors[14] to provide within their proposal a plan and schedule that will describe, in detail, actions that will be taken to show how the process areas in Level 2 and selected process areas in Level 3 critical to mission success will be performed. This offeror-supplied CMM information will be used as an evaluation factor in the final source selection. If the selected offeror is not at CMM Level 3, NASA will put in place appropriate contract control and monitoring mechanisms to track the selected contractor's plan. Note: The above criteria are flowed down to all subcontracts providing mission critical software.	A) Starting in FY 2002 Centers shall review all existing contracts that include mission critical software to document the supplier's status with respect to CMM Level 3 and identify software risk mitigation measures currently in place on these contracts. B) For contracts that are not currently at CMM Level 3 (as measured by a CMM appraisal by a Software Engineering Institute authorized lead appraiser from an external organization), the Center shall document and implement a risk mitigation plan by June 2002. Examples of risk mitigation measures include, but are not limited to, Independent Verification & Validation, peer review, end-to-end testing, requirements trace matrices, additional software quality assurance measures, and test witnessing. Centers will put in place appropriate control and monitoring mechanisms for tracking the progress against the documented

[12] CMM-Integrated (CMMI[SM]) is an acceptable substitute for "CMM" in this Table.
[13] A Software Capability Evaluation[SM] or a Standard CMMI Assessment Method for Process Improvement are the only acceptable appraisal methods.
[14] The term "offeror" in this table includes the specific implementing organization performing the work.

			risk mitigation plan.
Organizations responsible for individuals performing only <u>non-mission critical</u> software management, development, or acquisition	Each Center shall develop and implement a plan to advance software engineering capability for their organizations. Each Center shall show ongoing progress toward meeting software process and product improvement goals in the Center plan. Each Center shall define the CMM capability that will be achieved for each non-mission-critical software class[15] within their Center plan.	NASA will specify in Request For Proposals (RFP) that nonmission-critical software be developed by organizations meeting the Center's specified CMM Level (as measured by a CMM appraisal by a Software Engineering Institute authorized lead appraiser from an external organization) for each nonmission-critical software class applicable to the Statement of Work (SOW). If an offeror does not meet the Center-specified CMM Level, then that offeror is required to provide within their proposal a plan and schedule that will describe, in detail, actions that will be taken to show how the deficient or unassessed process areas will be performed. This offeror-supplied CMM information will be used as an evaluation factor in the final source selection. If the selected offeror does not meet the Center-specified CMM Level, NASA will put in place appropriate contract control and monitoring mechanisms to track the selected contractor's plan. Note: The above criteria are flowed down to all subcontracts providing software.	

[15] Software classifications are discussed in pending NPG 2820.

Appendix E: Risk Management Plan

A Risk Management Spreadsheet (enclosed) will be maintained on a continuous basis to identify, analyze, plan, track, control, and communicate risks associated with the NASA Software Engineering Initiative. Responsibility for maintaining the Risk Management Spreadsheet will be assigned to the SWG Risk Manager (a member of the SWG) and will be rotated over time. In addition to the Risk Spreadsheet maintained by the SWG Risk Manager, each strategy subgroup lead will be responsible for keeping a separate spreadsheet for its strategy current. Only those risks requiring SWG attention will be elevated from the strategy risk spreadsheet to the SWG risk spreadsheet. Risk spreadsheet review will be incorporated into the SWG meeting agendas at a minimum on a quarterly basis.

For each risk in the Risk Management Spreadsheet, the following will be recorded:

- Risk identification number and risk statement (in condition: consequence format).
- Priority, Probability, Impact, Timeframe.
- Mitigation strategy.
- Responsible person(s).
- Status, Date Opened, Date Updated, Date Closed.

The following definition of risk attributes will be used:

- High Probability—Likelihood of occurrence is greater than 70 percent.
- Medium Probability—Likelihood of occurrence is between 40 and 70 percent.
- Low Probability—Likelihood of occurrence is less than 40 percent.

- High Impact—Schedule delay of more than 3 months or cost overrun of greater than 15 percent.
- Medium Impact—Schedule delay of 2 to 3 months or cost overrun of 10 to 15 percent.
- Low Impact—Schedule delay of less than 2 months or cost overrun of less than 10 percent.

- Near-term Timeframe—1 month.
- Mid-term Timeframe—3 months.
- Far-term Timeframe—6 months or more.

The Timeframe specified should be when the problem is expected to occur if the risk is not handled.

Appendix F: Definitions

Computer Software Configuration Item (CSCI): An aggregation of software that is designated for configuration management and treated as a single entity in the configuration management process.

Mission-critical software:

(a) All CSCI's developed, reused, or acquired for inclusion in an NPG 7120.5A project, which fall under one of the two following items:

1. flight CSCI's, in which failure of the software could cause mission failure, harm to humans, damage to facilities or equipment, or risk to NASA's public reputation, or

2. ground CSCI's[16] designed for use in mission operations in which failure of the software could cause mission failure, harm to humans, damage to facilities or equipment, or risk to NASA's public reputation, or

(b) other software development items as designated by the GPMC, the NASA CIO, the NASA OCE, the NASA's Office of SMA or by the Center SMA office.

Nonmission-critical software: All software developed, reused, or acquired for or by NASA except for (a) mission-critical software as defined above, or (b) common desktop COTS software.

[16] Ground CSCI's are defined as software products associated with controlling flight hardware and software.

Appendix G: Acronyms

APPL	Academy of Program and Project Leadership
CBA-IPI	CMM-Based Appraisal for Internal Process Improvement
CIO	Chief Information Officer
CMM	Capability Maturity Model
CMMI	CMM Integrated
COTS	Commercial Off The Shelf
CSCI	Computer Software Configuration Item
DCL	Dual Career Ladder
FTE	Full-time equivalent
FY	Fiscal year
GPMC	Governing Program Management Council
HQ	Headquarters
IDEAL	Initiating, Diagnosing, Establishing, Acting, and Learning (of process improvement model
IEEE	Institute of Electrical and Electronics Engineers (Standards Association of Piscataway, New Jersey)
IPO	Institutional Program Office
ISO	International Organization for Standardization
IV&V	Independent Verification and Validation
KPA	Key Process Areas
MSG	Management Steering Group
NPD	NASA Policy Document
NPG	NASA Procedures and Guidelines
OAT	Office of Aerospace Technology
OCE	Office of the Chief Engineer
OSMA	NASA Office of Safety and Mission Assurance
PAPAC	Provide Aerospace Products and Capabilities
PCM	Process Change Method
POP	Program Operating Plan
RFP	Request For Proposals
SA-CMM	Software Acquisition-Capability Maturity Model
SCE	Software Capability Evaluation
SEI	Software Engineering Institute
SEWG	Systems Engineering Working Group
SEPG	Software Engineering Process Groups
SOW	Statement Of Work
STI	Scientific and Technical Information
SW-CMM	Software- Capability Maturity Model
SWG	Software Working Group
SW SR&QA	Software Safety, Reliability, and Quality Assurance
TM	Technical Manual
URL	Universal Resource Locator
WBS	Work Breakdown Structure

Risk Spreadsheet Template

ID #	Priority	Probability	Impact	Time Frame	Risk Statement (Condition; consequence format)	Assigned To	Mitigation Strategy	Status	Date Opened	Date Updated	Date Closed

Enclosure