GAO

June 1997

DEFENSE FINANCIAL MANAGEMENT

Immature Software Development Processes at Indianapolis Increase Risk
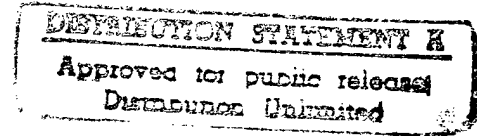
19970626 077

GAO/AIMD-97-41

GAO

United States
General Accounting Office
Washington, D.C. 20548

Accounting and Information
Management Division

B-276764

June 6, 1997

The Honorable John J. Hamre
The Under Secretary of Defense (Comptroller)

Dear Mr. Hamre:

In conjunction with our responsibilities to audit the U.S. government's financial statements, we are reviewing the Department of Defense (DOD) financial management systems. As you know, Defense's ability to produce accurate, auditable financial statements and other reliable management reports as required by the Chief Financial Officers (CFO) Act of 1990, as expanded upon by the Government Management Reform Act of 1994, has been hampered by inadequate financial systems. This report discusses the results of our evaluation of the Defense Finance and Accounting Service (DFAS) Financial Systems Activity (FSA)-Indianapolis' capability for developing and maintaining software for its information systems. Our objective was to evaluate software development processes used at FSA-Indianapolis. The four projects we reviewed were selected by the FSA director as those best representing its software development processes and practices.

Background

DFAS was created in 1991 from the financial centers of the military departments as the executive agent responsible for finance and accounting functions within DOD. Through consolidation, DFAS acquired the responsibility for more than 200 existing "legacy" finance and accounting systems, commonly referred to as automated information systems. Some are being eliminated and others are being further consolidated into a smaller number of "migratory" and "interim migratory" systems. In October 1993, the newly formed DFAS organization, called the Financial Systems Organization (FSO), was created to provide traditional central design activity services, as well as technical support, on a fee-for-service basis. FSO headquarters staff were in Indianapolis with additional personnel at six geographically dispersed FSAs having the primary mission to develop, modify, and maintain DFAS' automated information systems and secondarily to provide technical support in a number of systems-related areas.

Just prior to this reorganization, in June 1993, the Indianapolis center completed an assessment of the software engineering processes associated with its role as a central design activity. This internal software

process assessment concluded that the overall software engineering process practiced at Indianapolis was consistent with level 1 of the Software Engineering Institute's (SEI) capability maturity model.[1] SEI characterizes a Level 1 software process, which is the initial and most basic of the five levels, as ad hoc, and occasionally even chaotic, with few processes defined, and success depending on individual effort. Table 1 describes each level of this model.

Today, the FSAS are mainly concerned with maintaining and modifying the 109 existing automated systems, with software development, modification, and maintenance of DFAS' mission-oriented and support systems consuming more than a reported 80 percent of the FSO's fiscal year 1995 budget. Recognizing this role, in 1994, FSO initiated a major effort to improve its underlying software processes. In March of that year, the original FSO-developed software process improvement (SPI) strategic action plan was approved.

Since 1994, FSO has been implementing its long-term SPI plan to improve and standardize maintenance and modification processes. The plan includes the implementation of a system modification scenario and achievement of a level 2 software engineering capability according to the criteria of SEI's model. (See appendix II for a list of the 10 major objectives in the strategic plan, and the names and locations of other activities and systems under the SPI umbrella.)

Results in Brief

Although FSA-Indianapolis does not yet satisfy the criteria for a level 2 (i.e., repeatable) software development capability on any of the four projects we reviewed, the two projects under its SPI program showed strengths and improvement activities in many of the key process areas (KPAS). For example, projects under the SPI program generally kept software-related work products consistent with requirements. In contrast, projects not under the SPI program had few such identifiable strengths or improvement activities.

While the SPI program is making progress in ensuring that its projects implement defined and documented processes, many of its processes were

[1]The Software Engineering Institute (SEI) is a nationally recognized, federally funded research and development center established at Carnegie Mellon University in Pittsburgh, Pennsylvania, to address software development issues. In the late 1980s, with assistance from the Mitre Corporation, SEI developed a process maturity framework designed to assist organizations in improving their software processes. In general, software process maturity serves as an indicator of the likely range of cost, schedule, and quality of results that can be expected to be achieved by projects within a software organization.

not yet institutionalized.[2] For example, many policies were still in draft form or were in the planning phase, and therefore were not yet an ongoing way of doing business. In addition, software quality assurance activities, such as audits, were not used to ensure that defined software processes and standards were being followed. Such deficiencies pose unnecessary risks to the success of the software project until they are addressed.

By more rigorously implementing its project management processes among its SPI projects, FSA-Indianapolis could accelerate progress toward reaching the level 2 capability. This would enhance its ability to repeat individual project successes within similar application areas.

Scope and Methodology

To evaluate FSA-Indianapolis' software development capability, version 3.0 of the Software Engineering Institute's (SEI) software capability evaluation (SCE) method[3] was used by an SEI-trained team of GAO specialists, including an authorized lead evaluator trained in this evaluation technique by SEI. The evaluation is a method of assessing agencies' and contractors' software development processes against industry-accepted criteria in SEI's five-level software capability maturity model (CMM), as shown in table 1. These levels and the key process areas described within each level define an organization's ability to develop software, and can be used to improve its software development processes. The findings generated from an SCE identify (1) process strengths that mitigate risks, (2) process weaknesses that increase risks, and (3) improvement activities that indicate potential mitigation of risks.

---

[2]The SEI defines institutionalization as the building of an infrastructure and corporate culture that suggest methods, practices, and procedures so that they become the ongoing way of doing business, even after those who originally defined them are gone. Software Capability Evaluation, Version 3.0, Method Description (CMU/SEI-96-TR-002, April 1996).

[3]Version 3.0 of the SCE method is based on SEI's capability maturity model, version 1.1.

Table 1: CMM Levels and Descriptions

Level	Name	Description
5	Optimizing	Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.
4	Managed	Detailed measures of the software process and product quality are collected. Both the software process and products are quantitatively understood and controlled.
3	Defined	The software process for both management and engineering activities is documented, standardized, and integrated into a standard software process for the organization. All projects use an approved, tailored version of the organization's standard software process for developing and maintaining software.
2	Repeatable	Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.
1	Initial	The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort.

Note: According to an SEI study (Moving on Up: Data and Experience Doing CMM-Based Process Improvement, Technical Report CMU/SEI-95-TR-008, August 1995) of 48 organizations that implemented software process improvement programs, the time required to increase process maturity from level 1 to level 2 took an average of 30 months, with a range of 11 months to 58 months.

Source: Capability Maturity Model for Software, Version 1.1, (Technical Report CMU/SEI-93-TR-24, February 1993).

We requested that FSA-Indianapolis identify for our evaluation those projects that best represented their software development processes implemented at Indianapolis. The Director, FSA-Indianapolis identified two SPI projects and two non-SPI projects, as follows:

SPI Projects

- Defense Transportation Payment System (DTRS)
- Standard Army Financial Inventory Accounting and Reporting System Modernization (STARFIARS-MOD)

Table 2: CMM Level 2 "Repeatable" Key Process Area Descriptions

CMM Level 2 KPAs	Description
Requirements management	Defining, validating, and prioritizing requirements, such as functions, performance, and delivery dates.
Software project planning	Developing estimates for the work to be performed, establishing the necessary commitments, and defining the plan to perform the work.
Software project tracking and oversight	Tracking and reviewing software accomplishments and results against documented estimates, commitments, and plans and adjusting these based on the actual accomplishments and results.
Software subcontract management	Selecting qualified contractors and managing them effectively.
Software quality assurance	Reviewing and auditing the software products and activities to ensure that they comply with the applicable processes, standards, and procedures and providing the staff and managers with the results of their reviews and audits.
Software configuration management	Selecting project baseline items, such as specifications; systematically controlling these items and changes to them; and recording and reporting status and change activity for these items.

The Department of Defense provided written comments on a draft of this report. These comments are presented and evaluated at the end of this report and are reprinted in appendix I. We conducted our review from August 1996 through February 1997 in accordance with generally accepted government auditing standards.

FSA-Indianapolis Software Development Processes Are Immature

In order for FSA-Indianapolis to be rated at CMM level 2, all evaluated projects would have to pass every level 2 KPA. As shown in appendix III, this is not the case. No project passed every KPA, nor was there a single KPA that was passed by every project. Therefore, we conclude that FSA-Indianapolis as an organization remains a long way from achieving the repeatable level of maturity (level 2).

Organizations that have not developed the process discipline necessary to better manage and control their projects at the repeatable level incur greater risk of schedule delay, cost overruns, and poor quality software. To mitigate this, such organizations typically rely upon the variable

capabilities of individuals, rather than on institutionalized processes considered basic to software development.

Highlights of our evaluation of the four projects follow.

- Requirements Management. The purpose of requirements management is to establish a common understanding between the customer and the software project of the customer's requirements that will be addressed by the software project.

  DTRS was the only project that met all of the goals for requirements management. Specifically, for this project, functional and performance requirements and delivery dates were defined, validated, and prioritized.

  Other than DTRS, the projects' functional requirements were not adequately reviewed at the early stages of the software development life cycle. Specifically, the configuration control boards (CCBS) used in FSA-Indianapolis were responsible primarily for funding decisions but not the review and authorization of the establishment of and changes to software baselines. This situation can lead to wasted effort developing requirements which may be technically infeasible.

  In addition, the CEFMS project and its prime contractor in FSA-Indianapolis depended on a subcontractor to perform the requirements management function, but the subcontractor did not satisfy any of the goals within the requirements management KPA. Specifically, although the contractor reviewed software change requests before they were incorporated into the CEFMS project, a baseline of requirements was not established. Without a baseline, it is difficult to manage changes to the project and maintain the stability of the software produced from release to release.

Table 3: Results for the Requirements Management Key Process Area

Requirements management goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
System requirements allocated to software are controlled to establish a baseline for software engineering and management use.	Unsatisfied	Satisfied	Unsatisfied	Unsatisfied
Activity: —The software engineering group reviews the allocated requirements before they are incorporated into the software project.				
Software plans, products, and activities are kept consistent with the system requirements allocated to software.	Unsatisfied	Satisfied	Satisfied	Partially satisfied
Activities: —The software engineering group uses the allocated requirements as the basis for software plans, work products, and activities. —Changes to the allocated requirements are reviewed and incorporated into the software project.				

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

- <u>Software Project Planning</u>. The purpose of software project planning is to establish reasonable plans for performing the software engineering and for managing the software project.

  DTRS and STARFIARS-MOD had software development plans and documented software estimates (e.g., effort, cost, and schedule) for the project. Further, STARFIARS-MOD had recently implemented function point analysis[6] to estimate software size. On the other hand, software risks associated with the cost, resource, schedule, and technical aspects of the projects were not adequately identified, assessed, or documented for any of the four projects evaluated. Without risk assessment, the reliability of estimates is questionable, and the ability of a project to meet its schedule is reduced.

---

[6]Function points are derived using an empirical relationship based on countable measures (e.g., number of user inputs, number of user outputs, number of user inquiries, number of files, and number of external interfaces) of software's information domain and assessments of software complexity.

Table 4: Results for the Software Project Planning Key Process Area

Software project planning goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Software estimates are documented for use in planning and tracking the software project.	Unsatisfied	Partially satisfied	Partially satisfied	Unsatisfied

Activities:

—Estimates for the size of the software work products (or changes to the size of software work products) are derived according to a documented procedure.

—Estimates for the software project's effort and costs are derived according to a documented procedure.

—Estimates for project's critical computer resources are derived according to a documented procedure.

—The project's software schedule is derived according to a documented procedure.

—Software planning data are recorded.

(continued)

Software project planning goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Software project activities and commitments are planned and documented.	Unsatisfied	Partially satisfied	Partially satisfied	Unsatisfied

Activities:

—Software project planning is initiated in the early stages of, and in parallel with, the overall project planning.

—A software life cycle with predefined stages of manageable size is identified or defined.

—The project's software development plan is developed according to a documented procedure.

—The plan for the software project is documented.

—Software work products that are needed to establish and maintain control of the software project are identified.

—The software risks associated with the cost, resource, schedule, and technical aspects of the project are identified, assessed, and documented.

—Plans for the project's software engineering facilities and support tools are prepared.

(continued)

Software project planning goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Affected groups and individuals agree to their commitments related to the software project.	Unsatisfied	Satisfied	Partially satisfied	Unsatisfied

Activities:

—The software engineering group participates on the project proposal team.

—The software engineering group participates with other affected groups in the overall project planning throughout the project's life.

—Software project commitments made to individuals and groups external to the organization are reviewed with senior management according to a documented procedure.

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

- **Software Project Tracking and Oversight.** The purpose of software project tracking and oversight is to provide adequate visibility into actual progress so that management can take effective actions when the software project's performance deviates significantly from software plans.

  FSA-Indianapolis projects that were evaluated underwent periodic status reviews at meetings with key personnel present, and changes to commitments were generally agreed to by the affected groups and individuals. However, software risks associated with cost, resource, schedule, and technical aspects of the projects were not tracked. Moreover, although the SPI projects tracked performance and actual results, a mechanism for making corrections if and when projects failed to meet estimates was not in place. As a result of these weaknesses, the projects reviewed are more likely to be affected by unplanned events and are less likely to meet schedule and cost commitments.

Table 5: Results for the Software Project Tracking and Oversight Key Process Area

Software project tracking and oversight goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Actual results and performances are tracked against the software plans.	Unsatisfied	Partially satisfied	Partially satisfied	Unsatisfied

Activities:

—A documented software development plan is used for tracking the software activities and communicating status.

—The size of the software work products (or size of the changes to the software work products) is tracked, and corrective actions are taken as necessary.

—The project's software effort and costs are tracked, and corrective actions are taken as necessary.

—The project's critical computer resources are tracked, and corrective actions are taken as necessary.

—The project's software schedule is tracked, and corrective actions are taken as necessary.

—Software engineering technical activities are tracked, and corrective actions are taken as necessary.

—The software risks associated with cost, resource, schedule, and technical aspects of the project are tracked.

—Actual measurement data and replanning data for the software project are recorded.

—The software engineering group conducts periodic internal reviews to track technical progress, plan, performance, and issues against the software development plan.

—Formal reviews to address the accomplishments and results of the software project are conducted at selected project milestones according to a documented procedure.

(continued)

Software project tracking and oversight goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.	Unsatisfied	Partially satisfied	Partially satisfied	Unsatisfied

Activities:

—The project's software development plan is revised according to a documented procedure.

—The size of the software work products (or size of the changes to the software work products) is tracked, and corrective actions are taken as necessary.

—The project's software effort and costs are tracked, and corrective actions are taken as necessary.

—The project's critical computer resources are tracked, and corrective actions are taken as necessary.

—The project's software schedule is tracked, and corrective actions are taken as necessary.

—Software engineering technical activities are tracked, and corrective actions are taken as necessary.

—Actual measurement data and replanning data for the software project are recorded.

Software project tracking and oversight goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Changes to software commitments are agreed to by the affected groups and individuals.	Unsatisfied	Satisfied	Satisfied	Partially satisfied

Activities:

—Software project commitments and changes to commitments made to individuals and groups external to the organization are reviewed with senior management according to a documented procedure.

—Approved changes to commitments that affect the software project are communicated to the members of the software engineering group and other software-related groups.

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

• Software Subcontract Management. The purpose of software subcontract management is to select qualified software subcontractors and manage them effectively.

A contractual agreement between the government and the software contractors was used as a basis for managing the contracts and contractors. The projects had also designated a contracting officer representative to be responsible for establishing and managing software task orders. However, FSA-Indianapolis was unable to produce a written organizational policy describing the process for managing software contracts. Also, the contractor-developed software plans (e.g., software development plan, configuration management plan, and quality assurance plan), which the government would use to track contractors' progress, either (1) did not exist or (2) were not specific to the particular project under contract. The lack of an approved organizational policy removes an important source of guidance for project personnel; hence, there is a higher risk that individual projects will manage software contractors inconsistently with wide-ranging results. Accordingly, it would be prudent for FSA-Indianapolis to ensure that software contractors have a CMM rating of at least level 2. Neither the DTRS nor the STARFIARS-MOD project had contractor support, and therefore were not evaluated against this KPA.

Table 6: Results for the Software Subcontract Management Key Process Area

Software subcontract management goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
The organization selects qualified software subcontractors.	**Unsatisfied**	**Not evaluated**	**Not evaluated**	**Unsatisfied**
Activities: —The work to be subcontracted is defined and planned according to a documented procedure. —The software subcontractor is selected, based on an evaluation of the subcontract bidders' ability to perform the work, according to a documented procedure.				
The organization and the software subcontractor agree to their commitments to each other.	**Unsatisfied**	**Not evaluated**	**Not evaluated**	**Partially satisfied**
Activities: —The contractual agreement between the prime contractor and the software subcontractor is used as the basis for managing the subcontract. —A documented subcontractor's software development plan is reviewed and approved by the prime contractor. —Changes to the software subcontractor's statement of work, subcontract terms and conditions, and other commitments are resolved according to a documented procedure.				

(continued)

Software subcontract management goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
The organization and the software subcontractor maintain ongoing communications.	Unsatisfied	Not evaluated	Not evaluated	Partially satisfied

Activities:

—The prime contractor's management conducts periodic status/coordination reviews with the software subcontractor's management.

—Periodic technical reviews and interchanges are held with the software subcontractor.

—Formal reviews to address the subcontractor's software engineering accomplishments and results are conducted at selected milestones according to a documented procedure.

—The software subcontractor's performance is evaluated on a periodic basis, and the evaluation is reviewed with the subcontractor.

(continued)

	Project			
Software subcontract management goal	STANFINS	DTRS	STARFIARS-MOD	CEFMS
The organization tracks the software subcontractors' actual results and performance against its commitments.	Unsatisfied	Not evaluated	Not evaluated	Unsatisfied

Activities:

—The contractual agreement between the prime contractor and the software subcontractor is used as the basis for managing the subcontract.

—A documented and approved subcontractor's software development plan is used for tracking the software activities and communicating status.

—The prime contractor's management conducts periodic status/coordination reviews with the software subcontractor's management.

—Formal reviews to address the subcontractor's software engineering accomplishments and results are conducted at selected milestones according to a documented procedure.

—The prime contractor's software quality assurance group monitors the subcontractor's software quality assurance activities according to a documented procedure.

—The prime contractor's software configuration management group monitors the subcontractor's activities for software configuration management according to a documented procedure.

—The prime contractor conducts acceptance testing as part of the delivery of the subcontractor's software products according to a documented procedure.

—The software subcontractor's performance is evaluated on a periodic basis, and the evaluation is reviewed with the subcontractor.

(Table notes on next page)

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

Not evaluated - Goal did not apply in the organization's environment or insufficient evidence to rate the goal.

- **Software Quality Assurance.** The purpose of software quality assurance is to provide management with appropriate visibility into the process being used by the software project and of the products being built. SQA involves reviewing and auditing the software products and activities to verify that they comply with applicable procedures and standards.

  FSA-Indianapolis had a software quality assurance group, but the STARFIARS-MOD and CEFMS projects did not perform SQA activities during our evaluation, nor were their SQA personnel available for interviews. Therefore, these projects were not evaluated against this KPA. Moreover, although DTRS performed SQA activities, these were focused on the individual product, rather than on the overall process.[7] This is important because while focusing on the product can improve that particular item, focusing on the process ensures the consistent quality of all products. Further, there was no evidence of verification of processes and standards by the project SQA staff. Without process-focused SQA, FSA-Indianapolis cannot be certain that (1) its established software development processes are being followed as intended and (2) deviations from software standards and procedures are identified. Unless these basic issues are addressed, FSA-Indianapolis will have difficulty improving its processes.

---

[7]According to SEI, the process used for developing products should be defined, understood, measured, and progressively improved. As process quality increases, management also has greater insight, understanding, and control of risks. (See Software Capability Evaluation (SCE) Version 2.0, Implementation Guide, CMU/SEI-94-TR-5, February 1994).

Table 7: Results for the Software Quality Assurance Key Process Area

Software quality assurance goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Software quality assurance activities are planned.	Unsatisfied	Partially satisfied	Not evaluated	Not evaluated
Activities: —An SQA plan is prepared for the software project according to a documented procedure. —The SQA group's activities are performed in accordance with the SQA plan.				
Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.	Unsatisfied	Partially satisfied	Not evaluated	Not evaluated
Activities: —The SQA group's activities are performed in accordance with the SQA plan. —The SQA group participates in the preparation and review of the project's software development plan, standards, and procedures. —The SQA group reviews the software engineering activities to verify compliance. —The SQA group audits designated software work products to verify compliance.				
Affected groups and individuals are informed of software quality assurance activities and results.	Unsatisfied	Partially satisfied	Not evaluated	Not evaluated
Activities: —The SQA group periodically reports the results of its activities to the software engineering group. —Deviations identified in the software activities and software work products are documented and handled according to a documented procedure. —The SQA group conducts periodic reviews of its activities and findings with the customer's SQA personnel, as appropriate.				

(continued)

Software quality assurance goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Noncompliance issues that cannot be resolved within the software project are addressed by senior management.	Unsatisfied	Not evaluated	Not evaluated	Not evaluated
Activity:				
—Deviations identified in the software activities and software work products are documented and handled according to a documented procedure.				

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

Not evaluated - Goal did not apply in the organization's environment or insufficient evidence to rate the goal.

- Software Configuration Management - The purpose of software configuration management is to establish and maintain the integrity of products of the software project throughout the project's software lifecycle.

  STARFIARS-MOD had a configuration management plan and DTRS had a draft plan. In addition, both projects identified the software work products to be placed under configuration management. However, no software configuration control board (SCCB) existed for any of the projects to authorize the establishment of a software baseline and the identification of configuration items. Moreover, (1) the SCM procedures were inconsistent within projects and these projects contained multiple library systems, (2) some of the SCM systems were poorly documented, and (3) some SCM staff were inexperienced and had inadequate training. For example, SCM staff from the STANFINS and CEFMS projects had no formal training in configuration management. These weaknesses increase the risk of being unable to control software integrity uniformly within various projects, thus potentially increasing software development time and costs, or decreasing software product quality.

Table 8: Results for the Software Configuration Management Key Process Area

	Project			
Software configuration management goal	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Software configuration management activities are planned.	**Unsatisfied**	**Partially satisfied**	**Partially satisfied**	**Unsatisfied**
Activities: —An SCM plan is prepared for each software project according to a documented procedure. —A documented and approved SCM plan is used as the basis for performing the SCM activities.				
Selected software work products are identified, controlled, and available.	**Unsatisfied**	**Partially satisfied**	**Partially satisfied**	**Unsatisfied**
Activities: —A documented and approved SCM plan is used as the basis for performing the SCM activities. —A configuration management library system is established as a repository for the software baseline. —The software work products to be placed under configuration management are identified. —Products from the software baseline library are created and their release is controlled according to a documented procedure.				
Changes to identified software work products are controlled.	**Unsatisfied**	**Satisfied**	**Satisfied**	**Partially satisfied**
Activities: —Change requests and problem reports for all configuration items/units are initiated, recorded, reviewed, approved, and tracked according to a documented procedure. —Changes to baselines are controlled according to a documented procedure.				

(continued)

Software configuration management goal	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
Affected groups and individuals are informed of the status and content of software baselines.	Unsatisfied	Partially satisfied	Satisfied	Unsatisfied

Activities:

—The status of configuration items/units is recorded according to a documented procedure.
—Standard reports documenting the SCM activities and the contents of the software baseline are developed and made available to affected groups and individuals.
—Software baseline audits are conducted according to a documented procedure.

Satisfied - Practices that achieve the intent of the goal were implemented.

Unsatisfied - Weaknesses that significantly impact the goal exist.

Partially satisfied - Practices that achieve the intent of the goal were implemented but not institutionalized.

Conclusions

FSA-Indianapolis has begun, through its software process improvement program, to improve its software process for a limited number of projects. This is a significant positive step, but it has yet to satisfy all of the requirements for a level 2 capability. A significant amount of effort remains until the entire organization can demonstrate that it meets level 2 criteria. Until then, significant risks remain that investments made in new software development will not achieve their operational improvement objectives and that software will not be delivered consistent with cost and schedule estimates and needs.

Recommendations to the Under Secretary of Defense (Comptroller)

To better position FSA-Indianapolis to develop and maintain its software successfully and to protect its software investments, we recommend that you direct the Director of the Defense Finance and Accounting Service to:

• Ensure that software configuration control functions are performed for each development project. The configuration control board that carries out these functions should include software engineering specialists, and authorize the software baseline, configuration items, and other relevant software products.

- Ensure that any future contracts or contract modifications for software development include as part of the evaluation criteria that the contractor(s) (1) have an independently assessed software development capability of at least CMM level 2, (2) develop project specific software plans, and (3) perform software quality assurance and software configuration management activities.
- Require that projects develop, document, and periodically update a risk management plan that identifies and assesses risks to cost, schedule, and quality goals. The plan should also outline strategies for mitigating the risks, including mechanisms for corrective action when projects exceed established thresholds.
- Require that each development project perform both product- and process-focused software quality assurance activities throughout the system life cycle.
- Ensure that each development project (1) prepares a software configuration management plan that addresses all work products to be placed under configuration management and (2) follows a documented procedure.
- Expedite the promulgation of FSA-Indianapolis policies and procedures for software development.
- Delay any major investment in software development for projects at FSA-Indianapolis beyond that needed to sustain critical day-to-day operations until the repeatable level of process maturity (level 2) is attained and validated through an independent performance audit or, at a minimum, until the above recommendations are fully implemented.

Agency Comments and Our Evaluation

In commenting on a draft of this report, DOD officials generally agreed with the intent of our recommendations, with one exception. DOD disagreed with our recommendation to delay major investments in software development for projects at FSA-Indianapolis beyond that needed to sustain critical day-to-day operations until it satisfies a level 2 capability. Moreover, they expressed concern relative to either the scope of the recommendations or how the recommendations should be implemented.

DOD disagreed with our assessment that FSA-Indianapolis does not yet satisfy the criteria for a level 2 (i.e., repeatable) software development capability for any of the four projects we reviewed, stating that the Defense Transportation Payment System (DTRS) achieved a level 2 software development capability in November 1996. As discussed in this report, in September 1996 when we evaluated the four projects chosen for the SCE, none of the projects satisfied all six of the key process areas to

qualify as level 2. However, one project, DTRS, showed more progress than the others by satisfying the criteria in one key process area. Further, as noted in our report, DTRS was rated as either partially or fully satisfying most of the goals within the remaining five key process areas. Accordingly, given the time lapse between our evaluation and the issuance of this report, it is possible that the internal SCE performed by DFAS could have resulted in a level 2 rating for this one project. The three other systems (i.e., the Defense Business Management System, the Marine Corps Total Force System, and the Defense Civilian Pay System) that DOD asserted were level 2 systems were not developed by FSA-Indianapolis, and therefore are not relevant to the FSA-Indianapolis software development capability rating. However, in our view, the major issue is not whether FSA-Indianapolis has any supportable level 2 projects but instead when will FSA-Indianapolis become a level 2 organization.

DOD partially concurred with our recommendations that, for each project, DOD should establish a Software Configuration Control Board that collaborates with the functionally-oriented Configuration Control Board. DOD agreed that the functions specified for an SCCB should be performed but stated that it would rather place the functions within the existing CCB as opposed to establishing another board. Our major concern was to ensure that software configuration control functions are performed for each project, and that a configuration control board consisting of software engineering specialists controlled this activity. Accordingly, the DOD-suggested alignment would be appropriate if the existing CCB membership includes appropriate technical representation.

DOD partially concurred with our recommendation that future contract modifications for software development require the contractor to have an independently assessed software development capability of at least CMM level 2. While agreeing in principle that CMM level 2 is desirable, DOD stated that ". . .to use the CMM level 2 measurement as a sole discriminator in future contractor awards would preclude the use of a much needed quality assurance service available from other contractors. On future projects under the subcontract management Key Process Area, an evaluation criteria will be included in contracts for consideration on contractors having level 2 or higher capability." We did not mean to imply that the CMM level 2 criteria should be used as the sole determinant in evaluating software development contractors; however, it should be a significant criterion. Accordingly, FSA-Indianapolis should use the level 2 requirement as an important criterion in software development procurements to reduce

the risk of schedule slippage, cost overruns, and poor software quality. We modified our recommendation to clarify this point.

DOD partially concurred with our recommendation that projects develop, document, and periodically update a risk management plan that identifies and assesses risks to cost, schedule, and quality goals. DOD agreed with the function and actions included in our recommendation, but stated that these are not required because a plan to manage risk is not a requirement at CMM level 2. Within the Software Project Planning (SPP) key process area for CMM level 2, our SCE team evaluated FSA-Indianapolis against activity 13 which states, "The software risks associated with the cost, resource, schedule, and technical aspects of the project are identified, assessed, and documented" as well as the other activities required to satisfy this key process area. Thus, contrary to DOD's assertion that this activity is not required until level 3, the CMM cites this activity as one of the specific requirements for satisfying the level 2 Software Project Planning key process area. During the SCE, no project, including DTRS, was able to demonstrate that it was identifying, assessing, and documenting the software risks associated with cost, schedule, resource, and technical aspects of the project. Further, the level 3 requirement referred to by DOD builds upon the level 2 requirement cited above.

DOD partially concurred with our recommendation to require that each project perform both product- and process-focused software quality assurance activities throughout the system life cycle. However, the Department disagreed that these functions should be performed on each project managed by FSA-Indianapolis stating that "The Department believes that the appropriateness of these activities should be determined on a project-by-project basis that considers cost versus the limited life cycle of many of the legacy and interim migratory systems." Similarly, DOD partially concurred with our recommendation to require that each project (1) prepare a configuration management plan that addresses all work products to be placed under configuration management and (2) follow a documented procedure stating that implementation of this recommendation should also be made on a project-by-project basis. We agree that these activities should not apply to systems that (1) are currently operating as legacy systems, (2) have a short (i.e., 1- or 2-year) life cycle, and (3) are due for replacement within 1 or 2 years. However, because the CMM level 2 rating is dependent on satisfying the requirement for all projects, FSA-Indianapolis cannot reach level 2 as an organization until all the key process areas, including software quality assurance and

software configuration management, are satisfied. We modified these recommendations to clarify this provision.

Finally, DOD disagreed with our recommendation that FSA-Indianapolis delay any major investment in software development beyond that needed to sustain critical day-to-day operations until a repeatable level of maturity is reached. DOD indicated that delaying major development efforts until all projects achieve a level 2 would result in significant impacts on their system development initiatives and schedules. We disagree with Defense's contention. The recommendation does limit DOD's ability to take on the development of major automated information systems; however, it still permits FSA-Indianapolis to (1) maintain and modify existing operational systems and (2) develop prototypes and proof of concept systems. As a level 1 organization, FSA-Indianapolis will still be taking on the risk of schedule slippage, cost overruns, and poor quality software. DOD will have to be selective in its choice of development initiatives so as not to repeat the failed startups of the past. For example, in its attempt at the Military Pay Redesign, the U.S. Army Finance and Accounting Center (now known as FSA-Indianapolis) experienced schedule slippage from September 1984 to September 1993 and a cost overrun from $14.6 million to $82 million. In our report, Army Decision to Use Air Force Military Pay System Appears Advantageous (GAO/IMTEC-89-28, March 1989) we found that there was a lack of planning and documentation of risks. Furthermore, in its report on the Corps of Engineers Financial Management System (Report No. 97-051, December 18, 1996), the DOD's Office of the Inspector General stated that the DFAS Indianapolis Center "had not developed detailed plans for reducing the risks of achieving the expected performance of CEFMS." We believe that this was instrumental in the decision to classify CEFMS as a special interest program subject to the Major Automated Information Systems Review Council review.

We are sending copies of this report to the Chairmen and Ranking Minority Members of the House Committee on Government Reform and Oversight, Subcommittee on Government Management, Information and Technology; and the Senate Committee on Governmental Affairs. We are also sending copies to the Director of the Office of Management and Budget and the Secretary of Defense. Copies will also be made

available to others upon request. If you have any questions or wish to discuss the issues in this letter, please contact me at (202) 512-6234. Major contributors to this report are listed in appendix IV.

Sincerely yours,

William S. Franklin
Director, Information Systems Methodology & Support

Contents

Abbreviations

AIS	automated information system
ASQC	American Society for Quality Control
CCB	configuration control board
CDA	central design activity
CEFMS	Corps of Engineers Financial Management System
CFO	Chief Financial Officer
CMM	capability maturity model
DFAS	Defense Finance and Accounting Service
DOD	Department of Defense
DTRS	Defense Transportation Payment System
FSA	financial systems activity
FSO	financial systems organization
KPA	key process area
SCCB	software configuration control board
SCE	software capability evaluation
SCM	software configuration management
SDS	system development scenario
SEI	Software Engineering Institute
SMS	system modification scenario
SPI	software process improvement
SQA	software quality assurance
STANFINS	Standard Finance System
STARFIARS-MOD	Standard Army Financial Inventory Accounting Reporting System Modernization

UNDER SECRETARY OF DEFENSE
1100 DEFENSE PENTAGON
WASHINGTON, DC 20301-1100

COMPTROLLER

MAY 2 3 1997

Mr. Gene L. Dodaro
Assistant Comptroller General
Accounting and Information Management Division
U.S. General Accounting Office
Washington, DC 20548

Dear Mr. Dodaro:

This is the Department of Defense (DoD) response to the General Accounting Office (GAO) draft report, "DEFENSE FINANCIAL MANAGEMENT: Immature Software Development Processes at Indianapolis Increase Risk," dated April 30, 1997 (GAO Code 511502/OSD Case 1346).

The Department appreciates the endorsement and confirmation that the projects being managed by the Defense Finance and Accounting Service (DFAS) Financial Systems Activity (FSA)-Indianapolis under the Software Process Improvement (SPI) program showed strengths and improvement activities in many of the key process areas. However, the Department disagrees with the GAO assessment that the FSA-Indianapolis does not yet satisfy the criteria for a Level 2 (i.e., repeatable processes) software development capability on any of the four projects reviewed by the GAO. One of the projects reviewed by the GAO, and managed under the SPI program, is the Defense Transportation Payment System (DTRS). The DTRS achieved Level 2 software development capability in November 1996. This was accomplished under the leadership of a lead assessor certified by the Software Engineering Institute (SEI) who served as a full voting member of the team performing the software capability evaluation (SCE) assessment. The SCE assessments also have been completed on three other DFAS FSA systems that have achieved the Level 2 software development capability against the SEI capability maturity model (CMM) using the SCE Version 3.0 Method Description. The three systems are: (1) the Defense Business Management System, (2) the Marine Corps Total Force System and (3) the Defense Civilian Pay System.

In addition to the above, the Department believes that it would be appropriate to include comments in the report that address industry standings with respect to SCE assessments. The Department understands that the majority of software development organizations both in the private and public sector (approximately 70 percent as documented in a March 1996 SEI report titled "The Benefit of CMM Based Software Process Improvement") are Level 1 of the CMM. Including such a comparison will clarify the findings presented in this report with respect to the capabilities that have been achieved by the FSA-Indianapolis.

The Department's overall strategy for accounting and finance systems is to first manage only the migratory systems under the SPI program (nonmigratory, or legacy systems, are maintained at minimal levels to ensure systems viability until replaced by migratory systems). This will allow the Department to apply its resources effectively and efficiently to those systems that are expected to continue to be operational over the long term--rather than spending limited resources on systems that will be eliminated in the near term.

Thus far, a total of eight migratory accounting and finance systems have been placed under the SPI program. Additional systems will be added as resources allow. The Department will continue to manage the SPI program aggressively with the ultimate goal of increasing the maturity level for each migratory system.

Enclosed are the DoD comments on each recommendation set forth in the draft report.

Sincerely,

Alice C. Maroni
Principal Deputy Under
   Secretary of Defense (Comptroller)

Enclosure

DEPARTMENT OF DEFENSE COMMENTS ON THE
GAO DRAFT REPORT, DATED APRIL 30, 1997
(GAO CODE 511502) OSD CASE 1346

"DEFENSE FINANCIAL MANAGEMENT: Immature Software Development
Processes at Indianapolis Increase Risk"

RECOMMENDATIONS

RECOMMENDATION 1: The GAO recommended that the Secretary of Defense direct that
the Under Secretary of Defense (Comptroller) establish, for each project, a software configuration
control board (SCCB) composed of software engineering specialists. The GAO noted that such a
board should authorize the software baseline, configuration items, and other relevant software
products. (p. 26/GAO Draft Report)

DOD RESPONSE: Partially concur. The Department agrees that the functions inherent in an
SCCB should be established. Furthermore, the entity established should include software engi-
neering expertise. The Department does not agree that an SCCB needs to be established for each
discrete project. Establishing an SCCB for each project would require a significant amount of
additional resources to staff each board, thereby resulting in redundant and duplicative functions.
The Department believes that the functions of an SCCB best can be performed by one configura-
tion control board (CCB) that includes both functional and technical managers, with their respec-
tive expertise. A CCB could authorize the software baseline, configuration items, and other
relevant software products. This has been determined to be a more efficient and effective process.

RECOMMENDATION 2: The GAO recommended that the Secretary of Defense direct that
the Under Secretary of Defense (Comptroller) ensure collaboration between the SCCB and the
functionally-oriented configuration control board (CCB). (p. 27/GAO Draft Report)

DOD RESPONSE: Partially concur. The Department agrees that collaboration between the
functional and technical managers and their staffs is vital and needs to be accomplished. The
Department does not agree that collaboration needs to occur between two different control
boards. The Department believes that collaboration between the functional and technical staffs
can best be accomplished through the vehicle of a single configuration control board.

RECOMMENDATION 3: The GAO recommended that the Secretary of Defense direct that
the Under Secretary of Defense (Comptroller) ensure that any future contract modifications for
software development require the contractor(s) to (1) have an independently assessed software
development capability of at least CMM level 2; (2) develop project specific software plans; and
(3) perform software quality assurance and software configuration management activities.
(p. 27/GAO Draft Report)

Enclosure to Letter--GAO
Draft Report--OSD Case 1346
Page 1

DOD RESPONSE: Partially concur. The Department agrees that future contract modifications for software development should contain specific software plans and the contractors should perform software quality assurance and software configuration management activities. The Department does not agree with item one of the recommendation, i.e., to ensure an independently assessed software development capability of at least CMM level 2. While the Department concurs in principle, it does not concur with the recommendation as specifically written because most contractual support for software development is required to follow governmental regulatory and statutory processes and procedures. Additionally, to use the CMM level 2 measurement as a sole discriminator in future contract awards would preclude the use of a much needed quality assurance service available from other contractors. On future projects under the subcontract management Key Process Area, an evaluation criteria will be included in contracts for consideration on contractors having level 2 or higher capability. Recently, a new contract supporting software development was awarded based on offers with CMM capabilities.

RECOMMENDATION 4: The GAO recommended that the Secretary of Defense direct that the Under Secretary of Defense (Comptroller) require that projects develop, document and periodically update a risk management plan that identifies and assesses risks to cost, schedule and quality goals. The GAO noted that the plan should also outline strategies for mitigating risks, including mechanisms for corrective action when projects exceed established thresholds.
(p. 27/GAO Draft Report)

DOD RESPONSE: Partially concur. The Department agrees with the functions and actions stated in the recommendation. The Department does not agree that these functions and actions be required because a plan to manage risk is not a requirement at CMM level 2. A formalized risk management process--together with its management in place within the program management office--is not required until CMM level 3, Activity 10 "Integrated Software Management-Key Process Area." It is the Department's intention eventually to move the entire organization first to a level 2 maturity, then to a level 3 maturity. As the organization moves to a level 3 maturity, a formal risk management program will be implemented. In fact, a number of the required risk management activities already have been implemented under the FSA-Indianapolis SPI program such as: (1) identifying, assessing and documenting the software risks associated with the cost, resource, schedule, and technical aspects of the project, (2) analyzing and prioritizing the risks, and (3) ensuring that contingencies for the risks are identified. For example, in the Defense Transportation Payment System (DTRS), risk assessments are performed and documented on individual software change requests as well as each software release. The DTRS Software Development Plan includes a section on "Risks and Concerns Identification and Assessment." That section addresses various critical areas such as resource unavailability, changing or additional requirements/taskings outside the scope of the release, and unavailability of the mainframe computer.

Enclosure to Letter--GAO
Draft Report--OSD Case 1346
Page 2

RECOMMENDATION 5: The GAO recommended that the Secretary of Defense direct that the Under Secretary of Defense (Comptroller) require that each project perform both product- and process-focused software quality assurance activities throughout the system life cycle. (p. 27/GAO Draft Report)

DOD RESPONSE: Partially concur. The Department agrees that both product and process focused software quality assurance activities must be performed throughout the systems life cycle. The Department does not agree that these functions should be performed on each project managed by the FSA-Indianapolis. The Department believes that the appropriateness of these activities should be determined on a project-by-project basis that considers cost versus the limited life cycle of many of the legacy and interim migratory systems. For projects being managed under the SPI program that have reached the CMM level 2 rating, both product and process quality assurance are being accomplished.

RECOMMENDATION 6: The GAO recommended that the Secretary of Defense direct that the Under Secretary of Defense (Comptroller) ensure that each project (1) prepares a software configuration management plan that addresses all work products to be placed under configuration management and (2) follows a documented procedure. (p. 27/GAO Draft Report)

DOD RESPONSE: Partially concur. The Department agrees with both provisions of the recommendation. The Department does not agree that these functions should be performed on each project managed by the FSA-Indianapolis. The Department believes that the appropriateness of these activities should be determined on a project-by-project basis that considers cost versus the limited life cycle of many of the legacy and interim migratory systems. For projects being managed under the SPI program that have reached the CMM level 2 status, both of these functions are being accomplished. However, for other projects being managed under the SPI program, guidance is being issued consistent with software configuration management plans and documentation procedures.

RECOMMENDATION 7: The GAO recommended that the Secretary of Defense direct that the Under Secretary of Defense (Comptroller) expedite the promulgation of FSA-Indianapolis policies and procedures for software development. (p. 27/GAO Draft Report)

DOD RESPONSE: Concur. The Defense Finance and Accounting Service (DFAS) already is in process of promulgating policies and procedures to all its six Financial Systems Activities for software development that is consistent with CMM provisions and in consonance with available resources and mission priorities. The implementation of the CMM provisions, however, is a long term endeavor. For example, the process from CMM level 1 to CMM level 2 took between 18-30 months for each major software project. The objectives of the CMM currently are being achieved through the FSA-Indianapolis SPI program; that instrument is being used for moving the Department toward higher levels of software maturity.

Enclosure to Letter--GAO
Draft Report--OSD Case 1346
Page 3

RECOMMENDATION 8: The GAO recommended that the Secretary of Defense direct that the Under Secretary of Defense (Comptroller) delay any major investment in software development for projects at FSA-Indianapolis beyond that needed to sustain critical day-to-day operations until the repeatable level of process maturity (Level 2) is attained and validated through an independent performance audit or, at a minimum, until the above recommendations are fully implemented. (p. 28/GAO Draft Report)

DOD RESPONSE: Nonconcur. The Department does not agree to delaying investments in major development efforts until all projects achieve a CMM level 2 maturity level. The Department has undertaken a major initiative to reduce, improve and standardize accounting and finance systems so that they comply with statutory, regulatory and audit requirements. The mission needs of the Department preclude the further delay of any planned investment on future software development projects. As stated in response to other recommendations in this report, the Department believes that the appropriateness of achieving a CMM level 2 capability should be determined on a project-by-project basis that considers cost versus the limited life cycle of many of the legacy and interim migratory systems. The Department's goal is for FSA-Indianapolis, together with the other DFAS FSAs, to become Level 2 organizations. Any significant restriction placed on funding for development projects would work against the ability of the Department to reach higher levels of maturity, as institutionalization of new processes is necessary for assessment of its maturity level.

Software Process Improvement Program Financial Systems Activities and Automated Information Systems

In August 1995, the FSO developed a strategic plan identifying the movement to a standard process as a key element in improving the efficiency of the FSO and in achieving its productivity goals. The plan's 10 major objectives were to

- achieve CMM level 2 for an initial FSO system (FY 96),
- identify and implement productivity metrics for SMS (FY 96),
- define system development scenario (SDS) rapid prototyping (FY 96),
- identify and implement productivity metrics for SDS rapid prototyping (FY 97),
- define SDS (FY 97),
- identify and integrate standard function point analysis tools (FY 97),
- achieve CMM level 2 for all migratory systems (FY 97),
- identify and implement productivity metrics for SDS (FY 98),
- complete full Ada development and maintenance capability (FY 98), and
- achieve CMM level 3 for first migratory system (FY 99).

In an August 1996 briefing to the GAO team of SEI-trained specialists conducting the software capability evaluations at FSA-Indianapolis, the SPI program director stated that SPI officially began in November 1993, with its objectives being to (1) create a single, standard set of development processes, (2) improve and standardize development, modification, and reengineering processes, and (3) establish a basis for measuring performance. The activities and systems under SPI were:

FSA-Cleveland	• Defense Retiree and Annuitant Pay System.
FSA-Columbus	• Defense Business Management System.
FSA-Denver	• Defense Debt Management System. • Defense Joint Military Pay System. • Defense Retiree and Annuitant Pay System.
FSA-Indianapolis	• Defense Transportation Payment System. • Standard Army Financial Inventory Accounting and Reporting System-Modernization.
FSA-Kansas City	• Marine Corps Total Force System.

FSA-Pensacola

- Defense Civilian Pay System.
- Fund Administration and Standardized Document Automation System.

According to the FSO director, a software capability evaluation of the DTRS project, which was performed in December 1996, found that it satisfied all level 2 KPAs. He further asserted that the FSA-Kansas City-based Marine Corps Total Force System and the FSA-Pensacola-based Defense Civilian Pay System projects were also rated as level 2 projects. These software capability evaluations were performed by FSO staff.

Summary of FSA-Indianapolis SCE Results

KPA	Project			
	STANFINS	DTRS	STARFIARS-MOD	CEFMS
RM	Fail	Pass	Fail	Fail
SPP	Fail	Fail	Fail	Fail
SPTO	Fail	Fail	Fail	Fail
SSM	Fail	Not evaluated	Not evaluated	Fail
SQA	Fail	Fail	Not evaluated	Not evaluated
SCM	Fail	Fail	Fail	Fail

Pass - All goals corresponding to the KPA were satisfied.

Fail - One or more goals corresponding to the KPA was not satisfied.

Not evaluated - KPA did not apply in the organization's environment or insufficient evidence to rate the KPA.

Major Contributors to This Report

Accounting and Information Management Division, Washington, D.C.

David Chao, ASQC Certified Software Quality Engineer, SEI Authorized Lead Evaluator

Gary R. Austin, SCE Team Member

K. Alan Merrill, SCE Team Member

Prithviraj Mukherji, Ph.D., SCE Team Member

Frank W. Reilly, SCE Team Member

Paul Silverman, ASQC Certified Software Quality Engineer, SCE Team Member

Nancy M. Donnellan, SCE Team Apprentice