U.S. Small Business Administration

Business Owner:	SBA_ITPMO
Creation Date:	24-Jun-09
Last Updated Version:	24-Jun-09
Audience:	Internal
Document Control ID:	SBAITPMO_XX_000X_2.1-D

SBA System Development Method 2.1

Document History

Revision History

Revision Number	Revision Date	Summary of Changes	Author
01			
02			

Approvals

This document requires following approvals:

Name	Title

Distribution

This document has been distributed to:

Name	Title

TABLE OF CONTENTS

Table of Figures

Introduction

SBA SDM presents a System Development Method applicable to the acquisition, development and maintenance of all SBA information technology (IT) solutions. Since solutions vary greatly in size, scope of application, complexity of processing, and technologies used, the SBA SDM is intended to be easily tailored to be used for all SBA IT projects. The SBA IT Project Manager or Project Leader is responsible for tailoring the SBA SDM to meet their project objectives.

An important concept of this SDM is that it can be tailored to satisfy the requirements of any SBA system development project, providing every project a documented system development approach.

Purpose (and objectives of guide)

The Small Business Administration (SBA) System Development Method (SDM) provides a structured and integrated approach to acquiring IT solution including planning, developing and operating those solutions. This approach is to be used on all IT projects related to SBA programs and is intended to be flexible so that it can be tailored to meet the needs of each project.

The objective of every SBA IT project is to provide IT solutions that are engineered to satisfy the user's requirements, within determined cost, schedule, and quality guidelines. These IT projects encompass new development, maintenance, reuse, reengineering, and all other activities resulting in IT solutions that have a wide spectrum of characteristics:

- Target operating environments including but not limited to personal computers (PCs), mainframes, and LAN-based and client/server-based systems.

- Solution sizes include Major and Minor Investments

- Cost and cycle-time requirements vary.

- Desired end-product qualities, such as reusability and reliability (consequences of system failure ranging from minor inconvenience to loss of several million dollars), also vary.

One significant lesson learned from many years of IT system acquisition and development at SBA is that no single solution can solve every problem. No one lifecycle model; analysis and design, testing, or product evaluation method; or degree of formality for documents and reviews is appropriate for all SBA IT projects. To accommodate these variations, the project manager of each project must shape its system process to acknowledge customer requirements and constraints; goals and objectives for cost, cycle time, and product qualities; and management's tolerance for risk.

This SDM has the following objectives:

- Clarify the importance, objectives, and benefits of system lifecycle management to all potential participants in the system lifecycle.

- Describe the progression of the lifecycle through individual activities and processes, in terms of their respective objectives and products, and describe the relationships among the activities.

- Guide organizations to comply with all relevant Federal and Agency legislation and requirements.

This SDM presents a method applicable to the development and maintenance of all SBA IT projects. Systems that support SBA programs vary greatly in size, scope of application, complexity of processing, technologies used, and the methods and tools employed to support the evolution of the system from initial need statement through operation and eventual system termination. To accommodate the diversity of system development needs in SBA programs, this approach offers a structured, disciplined approach.

A majority of systems being fielded across the Federal government are acquired as more or less complete software packages with attendant hardware requirements. This does not preclude ensuring that the proposed acquisition fulfills SBA business, process, technical and data requirements. It is critical then that the steps outlined in this document be carried out as prescribed and that any proposed exception to SDM guidance be thoroughly documented, proposed and accepted before moving forward. Additional information for exceptions can be found in.

Scope of SDM

The SBA SDM covers the acquisition and development of both IT systems originating at the Program Offices as well as Agency-wide/cross-cutting and multi-agency systems originating in SBA executive-level offices and OMB. All SBA IT investments are identified in the SBA IT Portfolio and all IT projects must employ either this SDM or an SDM in compliance with the requirements of this SDM. The requirements will scale according to the scope of the system project.

Key SBA SDM Principles

The following principles serve as the foundation for managing the system development approach within SBA:

Ensure a structured approach to solving information management needs. SBA's SDM describes an overall approach for achieving solutions to identified information management needs. Primary emphasis is placed on the decisions that need to be made and the proper timing of these decisions. This SDM provides the flexibility for tailoring the approach, enabling information system projects to combine activities, processes, and products as appropriate and to select the tools and methodologies best suited to solving the need.

Designate an accountable sponsoring organization for each solution project. The project sponsor helps to ensure effective planning, management, and commitment to information system projects. The sponsor serves in a leadership role, providing guidance to the project team and securing from senior management the required reviews and approvals at specific points in the lifecycle.

Appoint a single project manager for each system project. The project manager has lead responsibility for the project's success and works through a project team and other supporting organization structures (e.g., working groups and user groups) to accomplish the objectives of the project. Regardless of organizational affiliation, the project manager is responsible for ensuring that project activities and decisions consider the perspectives of all affected organizations.

Introduction

Require a comprehensive project plan for each system project. The project plan is a pivotal element in successfully solving an information management need. The document describes how the system development approach will suit the project's specific characteristics. It also provides direction to the many activities of the lifecycle. It is developed in its initial form during the Initiate Project stage and is refined and expanded throughout the system lifecycle.

Emphasize data management throughout system development. Data processed by a system is a valuable agency resource; therefore, this approach stresses effective data management, conducted in accordance with Federal guidelines. Because of the large volume of data handled by SBA systems, the increasing trend toward sharing data across systems and programs, and the importance of data quality, it is essential that the selected system lifecycle activities stress the clear definition of data and the design and implementation of automated and manual processes to ensure proper safeguarding of the system's data.

Review and formally approve each system project. To help ensure that systems effectively address the targeted information management need, each project is subject to management review and approval. Skilled professionals should perform the reviews by examining tangible products from a programmatic, technical, and project management perspective. Reviews aid the project team and those who provide the required project approvals. Approvals are provided by the suitable level of management and confirm the continued commitment to the project scope, direction, and resource requirements in view of known risks and uncertainties.

Assign specific individuals to perform key roles throughout system development. Certain roles are considered vital to a successful system project, and at least one individual must be assigned to fulfill each role on a full- or part-time basis, as appropriate. These roles include program management, program staffing, QA, and CM. For most projects, more than one individual should represent the actual or potential users of the system (e.g., program staff) and should be designated by the program managers of the affected programs and organizations.

Obtain the participation of skilled individuals. Individuals responsible for conducting system lifecycle management projects are encouraged to obtain assistance from experienced information management professionals. The skill of the individuals participating in a system project is the single most significant factor for a successful project. This SDM is not intended as a substitute for information management skills or experience. Rather, it was developed to provide an environment in which skilled personnel could perform to the best of their abilities.

Document completely and accurately activity results and decisions. Effective communication and coordination of activities throughout system development depends on complete and accurate documentation of decisions and activities leading up to decisions. Activities and decisions should not be considered complete until there is tangible documentation of the activity or decision. Documentation for large, complex and/or mission critical projects will provide a greater level of detail than smaller, less complex projects.

Clarify resource availability before a system project proceeds. Beginning with the approval of a project, the continuation of a system project is contingent on a clear commitment from the sponsoring management. This commitment is embodied in the

assurance that the necessary resources will be available, not for the next activity only, but for the remainder of the lifecycle.

Assure, within reason, that information processed by the system is reliable and properly safeguarded. Proper precautions should be followed throughout development, testing, and operation of the system to ensure that data processed by the system is accurate and that requirements of the Federal governing directives are being met.

SBA SDM Lifecycle Models

The SDM acknowledges that a single lifecycle model is not appropriate for all SBA system development projects and, if necessary, a project may use multiple lifecycle models to accomplish the project's objective. (For example, a waterfall development lifecycle model may be the best overall model to use on a specific project, but an evolutionary lifecycle model may be inserted in the overall lifecycle to prototype a high-risk area.) The SDM also acknowledges that the appropriate method for accomplishing a required activity depends on characteristics of the system project.

After the project manager defines a reasonable scope for the work, the next step is to define a technical approach that best accomplishes the work. The project manager defines the project's technical approach by selecting an appropriate lifecycle model for performing required activities and appropriate packaging techniques for required products. To facilitate developing each project's technical approach and to avoid "reinventing the wheel," this section contains a summary of effective lifecycle models from among which the project manager can choose to satisfy the project goals.

After selecting an appropriate lifecycle model, a project manager must determine the activities, methods, techniques, and products that will help achieve the goals and objectives established for the project. The project manager should incorporate into the Project Plan the appropriate project activities and SBA control and reporting mechanisms provided in the SDM.

A lifecycle model is composed of one or more phases (for example, a define phase, a design phase, a build phase). Within each phase, one or more activities are executed. For example, during the waterfall model's design phase, the design activity is performed, as is the test planning activity for acceptance testing. In most cases, activities neither begin nor end precisely at the phase boundaries; rather, they overlap adjacent phases. Various methods may be used in performing an activity. For example, object-oriented design is one proven design method; structured design is another.

This document does not mandate any particular lifecycle model, and the order of activities described here is not intended to conform to any particular model. Few specific methods are mandated for required activities. These decisions are left to the project manager, who selects an appropriate lifecycle model and activity-related methods and defines them in the Project Plan. This section gives guidance on selecting an appropriate, recommended lifecycle model. For convenience, Table A-3 provides the definitions of several important terms used extensively in this section.

Table 0-1 Defining a Lifecycle

Term	Definition
Software lifecycle	The period of time that begins when a software product is conceived and ends when the software is no longer available for use. A lifecycle is typically divided into lifecycle phases.
Lifecycle model	A framework on which to map activities, methods, standards, procedures, tools, products, and services (e.g., waterfall, evolutionary).
Lifecycle phase	A division of the software effort into non-overlapping time periods. Lifecycle phases are important reference points for the project manager. Multiple activities may be performed in a lifecycle phase; an activity may span multiple phases.
Activity	A unit of work that has well-defined entry and exit criteria. Activities usually can be broken into discrete steps.
Method	A technique or approach, supported by procedures and standards that establishes a way of performing activities and arriving at a desired result.

The lifecycle models listed below are summarized in the following subsections.

Waterfall Development Lifecycle

The waterfall (single release) lifecycle is essentially a once-only, sequential-step approach. In simple terms, determine user needs; define requirements; design the system; implement the system; and test, fix, and deliver the system. Table A-4 outlines the advantages, disadvantages, and best circumstances under which to use the waterfall development lifecycle model. This lifecycle is illustrated in Figure A-1. Products and milestones for this lifecycle are summarized in Table A-5.

Table 0-2. Summary of Waterfall Development Lifecycle

Advantages	Model is well studied, well understood, and well defined. Easy to model and understand. Easy to plan and monitor. Many management tools exist to support this lifecycle model.
Disadvantages	Most, if not all, requirements must be known at the start. Model does not readily accommodate requirement changes. Product is not available for initial use until the project is nearly done.
Most appropriate when	Project is similar to one done successfully before. Requirements are stable and well understood. Design and technology are proven and mature. Total project duration is relatively short (less than one year). Customer does not need any interim releases.

Introduction
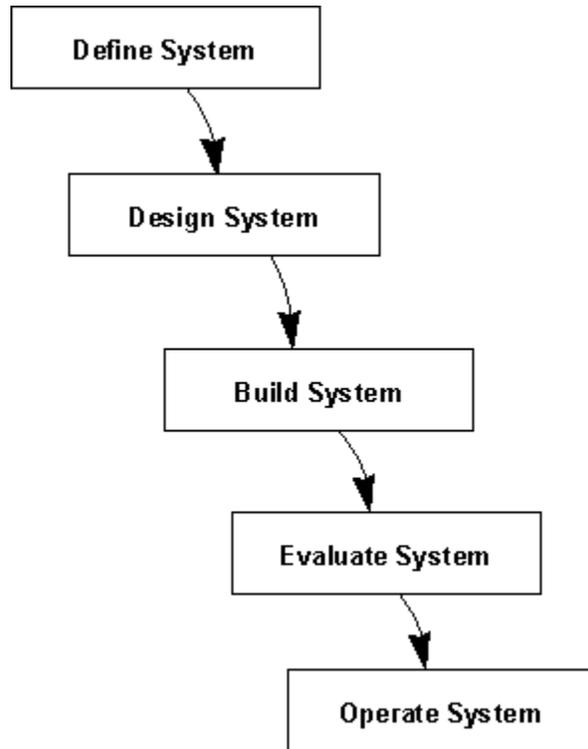
Figure 0-1 Waterfall Development Lifecycle

Table 0-3 Products and Milestones for the Waterfall Development Lifecycle

Lifecycle Phase	Major Products	Milestones
Initiate Project	Needs Statement Feasibility Study Benefit/Cost Analysis Risk Analysis System Decision Paper Project Plan	Initiate Project Phase Review
Define System	Functional Requirements Document Data Requirements Document Requirements traceability matrix	Define System Phase Review

Lifecycle Phase	Major Products	Milestones
Design System	System/Subsystem Specifications Database Specifications Program Specifications Updated requirements traceability matrix Acceptance Test Plan Preliminary User's Manual Prior phase products updated	Design System Phase Review
Build System	Unit-level design Implemented, tested software Database Validation, Verification, and Test Plan and procedures Updated requirements traceability matrix Operations Manual Maintenance Manual Prior phase products updated	Build System Phase Review
Evaluate System	Test Results and Evaluation Report Installation and Conversion Plan Final User's Manual Updated requirements traceability matrix Prior phase products updated	Evaluate System Phase Review
Operate System	System problem reports	Project Closeout

Incremental Development Lifecycle

For the incremental, (multiple small releases) lifecycle, determine user's needs, define system requirements, and perform the rest of the development in a sequence of small releases. The first small release incorporates part of the planned capabilities; the next small release adds more capabilities, and so on, until the system is complete. Table A-6 lists the advantages, disadvantages, and the best circumstances under which to use the incremental development lifecycle model. This lifecycle is illustrated in Figure 0-2. Major products and milestones for this lifecycle are summarized in Table A-7.

Table 0-4. Summary of Incremental Development Lifecycle

Advantages	Reduces risks of schedule slips, requirements changes, and acceptance problems.
	Increases manageability.
	Facilitates feeding back changes in subsequent small releases.
	May be delivered before the final version is done, allowing end users to identify needed changes.
	Breaks up development for long lead-time projects.
	Allows users to validate the product as it is developed.
	Allows software team to defer development of less understood requirements to later releases after issues have been resolved.
	Allows for early operational training on interim versions of the product.
	Allows for validation of operational procedures early.
	Includes well-defined checkpoints with users via reviews.
Disadvantages	Like the waterfall lifecycle, most, if not all, requirements must be known at the start.
	Sensitive to how specific releases are selected.
	Places products (particularly requirements) under configuration control early in lifecycle, thereby requiring formal change control procedures that may increase overhead, particularly if requirements are unstable.
Most appropriate when…	Project is similar to one done successfully before.
	Most of the requirements are stable and well understood, but some may not yet be identified.
	The design and technology are proven and mature.
	Total project duration is greater than one year or customer needs interim release(s).
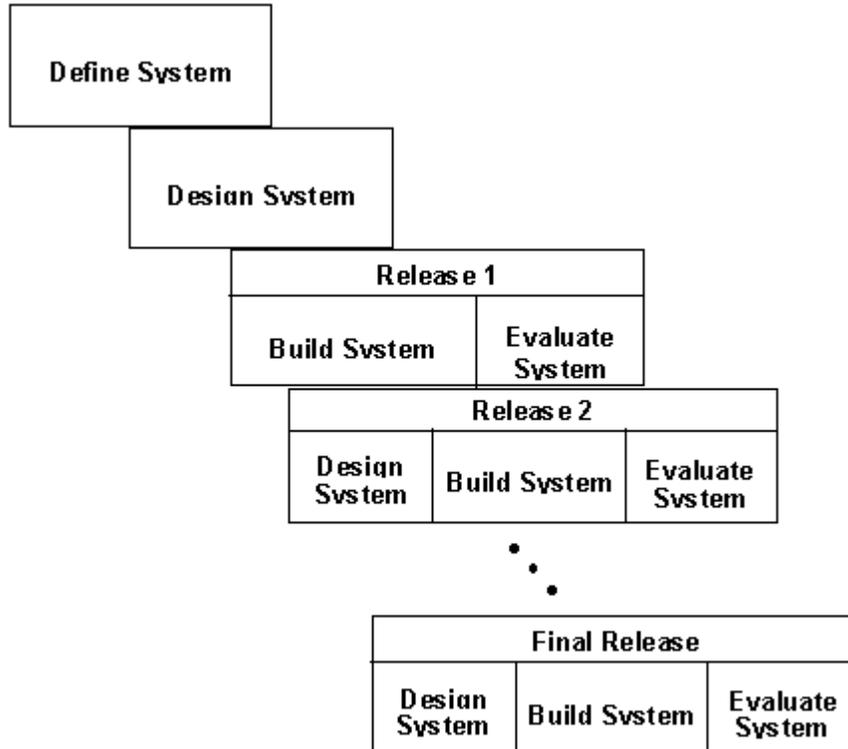
Figure 0-2 Incremental Development Lifecycle

Table 0-5 Products and Milestones for the Incremental Development Lifecycle

Lifecycle Phase	Major Products	Milestones
Initiate Project	Needs Statement Feasibility Study Benefit/Cost Analysis Risk Analysis System Decision Paper Project Plan	Initiate Project Phase Review
Define System	Functional and data requirements documents Requirements traceability matrix	Define System Phase Review

Lifecycle Phase	Major Products	Milestones
Design System	System/subsystem specifications through next small release Database specifications through next small release Program specifications Updated requirements traceability matrix Acceptance Test Plan Preliminary User's Manual Prior phase products updated	Design System Phase Review
Build System	Unit-level design Implemented, integrated, and tested software Database Validation, Verification, and Test Plan and procedures Updated requirements traceability matrix Operations Manual Maintenance Manual Prior phase products updated	Build System Phase Review
Evaluate System	Test Results and Evaluation Report Installation and Conversion Plan Final User's Manual Updated requirements traceability matrix Prior phase products updated	Evaluate System Phase Review
Operate System	System problem reports	Project Closeout

Iterative Development Lifecycle

Like the incremental development lifecycle, the evolutionary lifecycle also develops a system in small releases; but it differs from the incremental lifecycle in acknowledging that user needs are not fully understood and not all requirements can be defined at the start. In the evolutionary approach, user needs and system requirements are partially defined at the start, and then are refined in each succeeding small release. The system evolves as the understanding of user needs and the resolution of issues occurs. Prototyping is especially useful in this lifecycle model. Table A-8 points out the advantages, disadvantages, and the best circumstances in which to use the evolutionary lifecycle development model. This lifecycle is illustrated in Figure A-3. Major products and milestones for this lifecycle are summarized in Table A-9.

Table 0-6 Summary of Iterative Development Lifecycle

Advantages	Not all requirements need be known at the start.
	Like the incremental lifecycle, interim small releases of the product facilitate feeding back changes for subsequent small releases.
	Users are actively involved in definition and evaluation of the system.
	Prototyping techniques enable developers to demonstrate functionality to users with minimal effort.
	Even if time or money runs out, some amount of operational capability is available.
Disadvantages	Because not all requirements are well understood at the start, the total effort involved in the project is difficult to estimate early; therefore, expect accurate estimates only for the next cycle, not for the entire development effort.
	Less information is available for managing (progress is difficult to measure).
	There is risk of never-ending evolution (e.g., continual "gold plating").
	May be difficult to manage when cost ceilings or fixed delivery dates are specified.
	Will not be successful without user involvement.
Most appropriate when…	Requirements or design are not well defined, not well understood, or likely to undergo significant changes.
	New or unproven technologies are being introduced.
	System capabilities can be demonstrated for evaluation by users.
	Diverse user groups exist with potentially conflicting needs.

Diagram labels (inner to outer):

Sys. Concept Def.

Define System (repeated across rings)

Build System (repeated across rings)

Evaluate System (repeated across rings)

Operations and Maintenance (repeated across rings)

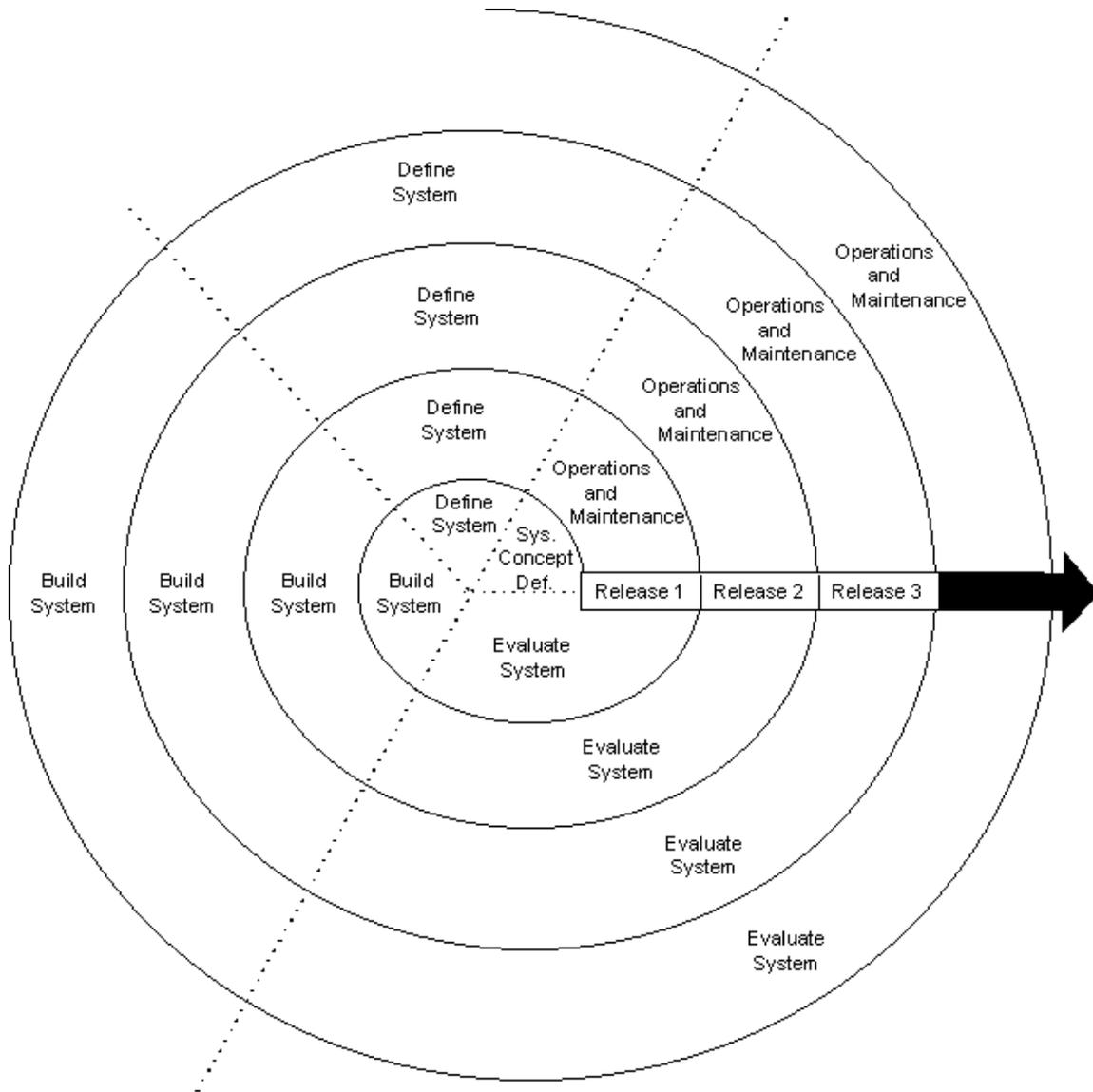Release 1 | Release 2 | Release 3

Figure 0-3 Iterative Development Lifecycle

Table 0-7 Products and Milestones for the Iterative Development Lifecycle

Lifecycle Phase	Major Products	Milestones
Initiate Project	Initial Project Plan, to be updated in later phases	Initiate Project Phase Review

Lifecycle Phase	Major Products	Milestones
Define System and Design System	Preliminary requirements document, including functional and database requirements Architectural design document containing the infrastructure plus the architecture of each release as it evolves Requirements traceability matrix	Combined Define System and Design System Phase Review
Build System	Prior phase products updated Software product baseline combining new, reused, and off-the-shelf products Updated requirements traceability matrix Draft User's Manual Integrated, tested software Acceptance test plan Prior phase products updated	Integration testing after all small releases have been completed
Evaluate System	Test Results and Evaluation Report Installation and Conversion Plan Final User's Manual Updated requirements traceability matrix Prior phase products updated	Evaluate System Phase Review
Operate System	System problem reports	Project Closeout

Package-Based Development Lifecycle

The package-based development lifecycle for system development is based on the use of commercial off-the-shelf (COTS) and Government off-the-shelf (GOTS) products and reusable packages. Table A-10 summarizes the advantages and disadvantages and indicates when the package-based development lifecycle model is most appropriately used. This lifecycle is illustrated in Figure A-4. Major products and milestones for this lifecycle are summarized in Table A-11.

Table 0-8 Summary of Package-Based Development Lifecycle

Advantages	Lower cost than developing equivalent functionality from scratch
	Cycle time often lower than developing equivalent functionality from scratch
	More confidence in quality of the end product (because quality of reusable products is already known)
Disadvantages	May result in compromises between desired functionality and functionality provided by reusable products
	Maintainability may be more of a challenge because source of reusable products may not be the same organization (for example, requires third party to make changes, raises CM issues when vendor releases updated versions)
Most appropriate when…	A significant portion of the functionality of a system can be provided by reusable products
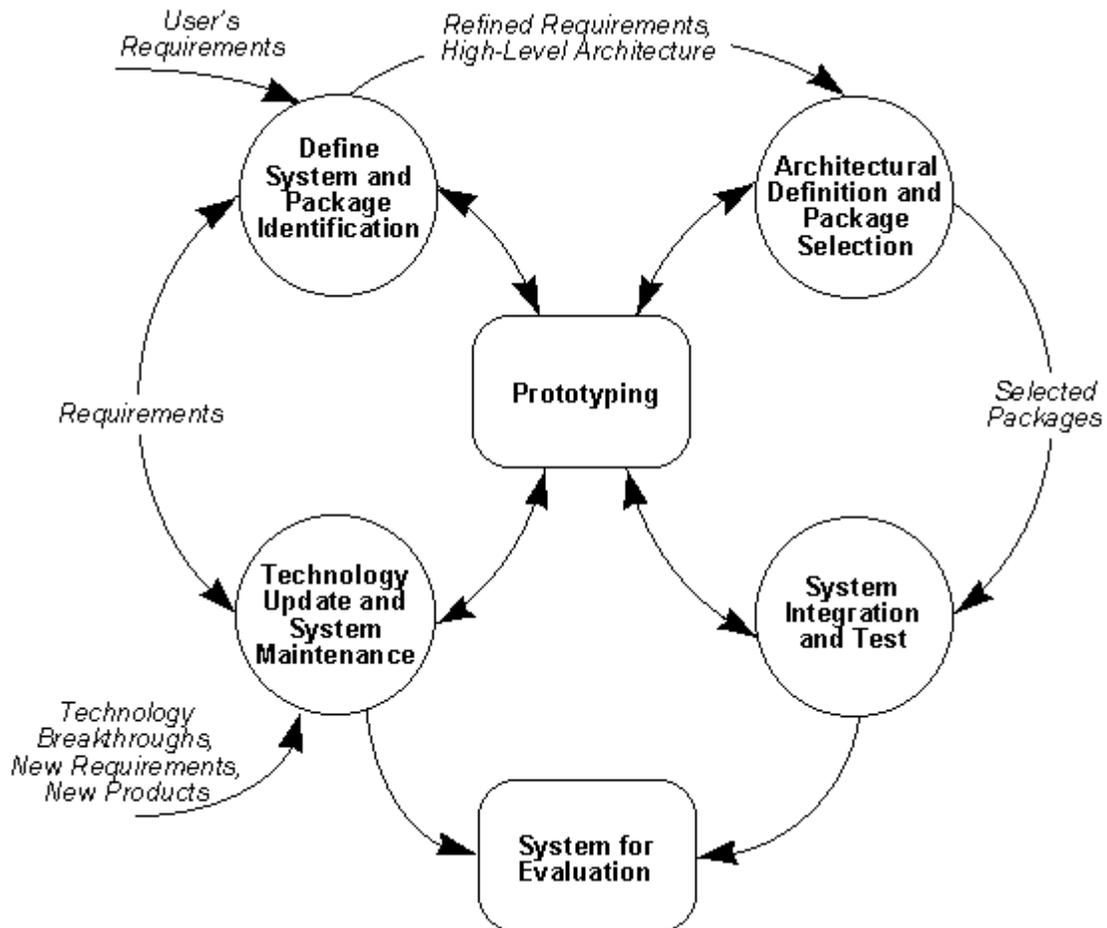
Figure 0-4 Package-Based Development Lifecycle

Table 0-9 Major Products and Milestones for the Package-Based Development Lifecycle

Lifecycle Phase	Major Products	Milestones
Define System and Package Identification	Requirements High-level architecture Candidate COTS packages	Define System Phase Review
Architectural Definition and Package Selection	Modified requirements System architecture Final packages	Design System Phase Review
System Integration and Test	System for evaluation	User demonstrations Build System Phase Review
Technology Update and System Maintenance	Enhanced system for evaluation	User demonstrations Build System Phase Review

Legacy System Maintenance Lifecycle

The legacy system maintenance release lifecycle is used to apply fixes or minor enhancements to an operational system. Use a waterfall or incremental lifecycle for major enhancements. Selected and sometimes abbreviated activities performed in the software development lifecycles are also performed during maintenance. The legacy system maintenance lifecycle is similar to the waterfall lifecycle; the primary difference is that in the legacy model the architectural design has already been established. This lifecycle is illustrated in Figure A-5. Major products and milestones for this lifecycle are summarized in Table A-12.
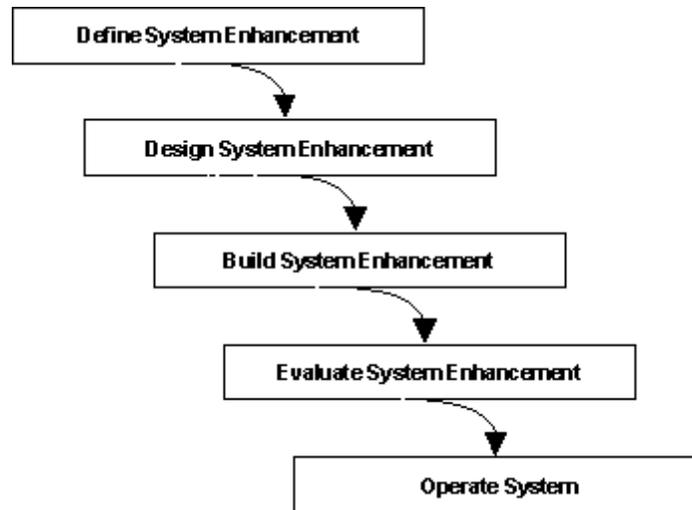
Figure 0-5 Legacy System Maintenance Lifecycle

Table 0-10 Products and Milestones for the Legacy System Maintenance Lifecycle

Lifecycle Phase	Major Products	Milestones
Initiate Project	Release Contents Agreement	Release Contents Review
Define System	Functional Requirements Document (updated)	Define System Phase Review
	Data Requirements Document (updated)	
	Updated requirements traceability matrix	
Design System	Design specifications (system/subsystem, database, program) (updated)	Design System Phase Review
	Updated requirements traceability matrix	
Build System	Unit-level design	Build System Phase Review
	Implemented, tested software	
	Updated requirements traceability matrix	
	Acceptance Test Plan	
	Draft User's Manual updates	
Evaluate System	Acceptance tested software	Evaluate System Phase Review
	Test Results and Evaluation Report	
	Final User's Manual updates	
	Updated requirements traceability matrix	
Operate System	Post-implementation review report	Release Sign-off

Solution Approach

SBA system development addresses a broad range of activities, from identifying a need for building or acquiring a system to the final disposition of the system.

The SBA SDM is intended to provide the flexibility to select the activities, methods, tools, and technologies that are appropriate to solving each information management need and to permit selection and refinement of the basic lifecycle as appropriate for a given need. However, this flexibility exists within a framework that requires the active participation of SBA program managers and information management professionals throughout the lifecycle.

The SDM is a structured process in which SBA systems are conceived, defined, developed, tested and continually monitored throughout their lifecycle. Both successful developments and those that are terminated or delayed are evaluated to assess the impact on future development and to benefit from any lessons learned. The SDM contains eight stages: Initiate, Plan, Define, Design, Build and Test, Implementation, Operate and Maintain and Retire. In the following paragraphs, each SDM stage is described in the following format:

Purpose: Describes the objective of the stage

Entry Criteria: Describes the stage requirements, and thresholds for entering the stage

Process: Describes the type of justification, and review that will occur in the stage

Performance & Governance Considerations: describes the how the IPT should use the EA in the stage and the governance activities that will take place during the stage; and

Exit Criteria: Describes the action necessary for proceeding to the next stage.

Completing one stage is necessary before beginning a subsequent stage. Each stage is overseen by the BTIC, which ultimately approves or rejects a project's advancement to the next stage. This ensures that each investment and its projects receive the appropriate level of managerial review and that coordination and accountability exist.

Some stages may be reiterated depending on lifecycle development method adopted. For example, the Acme Small Business Information System using an iterative development method will cycle through Design, Build and Test, Implement and Operate & Maintain stages as each system component or release is developed and goes live. It is also possible that failure to satisfy the requirements of one or all components of the SDM Process Stage would result in one or more cycles through that same Process Stage until those requirements are fulfilled.

SDM Stages

The eight stages that make up the SBA SDM are Initiate, Define, Design, Plan, Build and Test, Implement, Operate and Maintain, and Retire and are sequenced as shown in Figure 0-6 System Development Method Stages.

System Development Method (SDM)

Initiate → Plan → Define Req'ts. → Design → Build & Test → Implement → Operate & Maintain → Retire
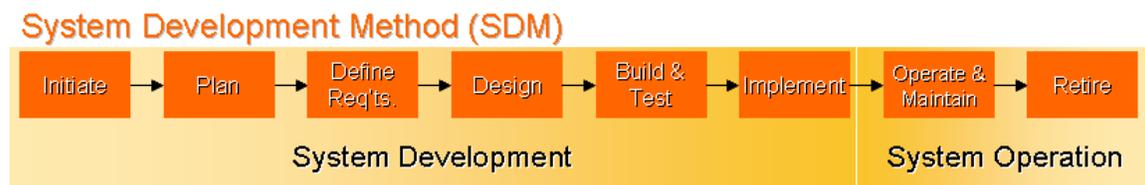
System Development | System Operation

Figure 0-6 System Development Method Stages

Initiate Stage

The Initiate Stage builds on the system concept developed as part of a CPIC Pre-Select stage. It identifies through research, analyses and system and modeling, the best approach to a given IT project based on maximizing benefit while minimizing risk.

Plan Stage

The budget, support resources, activities, schedules, security requirements, tools, and required reviews for developing a system are determined in the Plan Stage. A document management strategy is introduced to ensure that all documents/plans/artifacts are maintained and controlled in an orderly, consistent manner.

Define Stage

During the Define Stage, the exact requirements necessary to design and implement the system, as well as the security requirements and tests necessary to establish compliance are specified.

In this stage, complete sets of software and hardware designs and test plans are written based on the detailed specifications and security requirements created in the analysis stage.

Design Stage

In the Design Stage, a complete set of software, hardware and overall system designs and test plans are written based on the detailed specifications and security requirements created in the Define Stage.

Build and Test Stage

The detailed specifications produced during the Design Stage are translated into hardware, communications, and an executable system. Software is unit tested, integrated, and retested in a systematic manner. Hardware is assembled and tested.

The system specified is assembled from the parts developed during the previous stages and methodically tested. Users test the system to ensure that the documented requirements, as well as all privacy, security and other mandated requirements are satisfied by the developed or modified system.

Implement Stage

During the Implement Stage, the system is installed, and made operational in the production environment. This stage continues until the production system is operating in accordance with the defined user requirements.

Operate and Maintain Stage

System operation is ongoing during the Operate and Maintain Stage or the CPIC Steady State Stage. All documentation is finalized and archived for future reference. The system is monitored for continued performance in accordance with user requirements, and any needed modifications are incorporated.

Retire Stage

When the system has outlived its useful life and no longer supports the SBAs Mission and Goals. IN this final stage, all system disposition decisions are documented and archived and all capital material and equipment is disposed of in accordance with Federal and Agency requirements.

SDM Stage Components

Components of each process step consist of Purpose, Entry Criteria, Process, Performance Considerations and Exit Criteria.
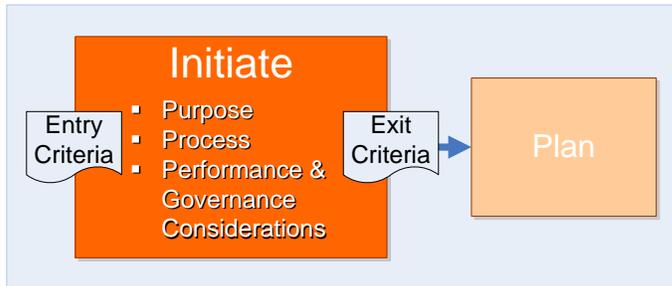

Figure 0-7 System Development Process Stage Components

Process Stage Purpose: Describes the purpose and objective of the stage

Process Stage Entry Criteria: Describes the stage requirements, artifacts and thresholds for entering the stage

Process Stage Process: Describes the type of justification, planning, and review that will occur in the stage

Process Stage Performance Considerations: describes the how the IPT should use the EA in the stage and the governance activities that will take place during the stage; and

Process Stage Exit Criteria: Describes the action and artifacts necessary for proceeding to the next stage.

Completing one stage is necessary before beginning a subsequent stage. Each stage is overseen by the E- Board, which ultimately approves or rejects a project's advancement to the next stage. This ensures that each project receives the appropriate level of managerial review and that coordination and accountability exist.

Crosscutting Considerations for All SDM Stages

A number of considerations exist throughout system development for every system and are addressed in multiple activities. These considerations are summarized as follows:

Project Plan – The Project Plan is produced during the Initiate Project stage and is updated, expanded, and refined continually throughout system development. It covers project scheduling, work breakdown structure (WBS), staffing, resources, adjustments to the system development cycle structure, selection of tools and techniques, identification of applicable reviews and approvals, configuration management (CM) methods, and other related topics. The plan is subject to review and approval by SBA management.

Project Reviews – Project reviews help to ensure that key issues are identified and addressed as early as possible, which will help to avoid major and expensive modifications to the original project. They provide feedback to the project team and also

assist management in supporting required system approvals. Specific organizations and individuals who will participate in project reviews and related activities are designated early in the lifecycle and are selected by of the nature of the information management need and the recommended solution.

Phase Reviews. Phase reviews are conducted at each phase of system development by the TPOC and where applicable, approved by the Program office before proceeding to the next phase. These reviews help ensure that the system established is programmatically and technically sound.

Quality Assurance (QA) – During the early stages of a project, the QA group works with the project development team to establish plans, standards, and procedures that will add value to the System project and satisfy the constraints of the project and the SBA CIO's policies. By participating in the early stages in establishing these guidelines, the QA group helps ensure that the plans, standards, and procedures fit the project's needs and verifies that they will be useful for performing reviews and audits throughout the system lifecycle. The QA group continuously reviews project activities, audits system work products throughout the lifecycle, and provides management information with which to judge whether the system project is adhering to its established guidelines. The QA group reviews and audits system products (system and documentation) and process activities to verify that these products comply with the SDM and the applicable procedures and standards. The QA group provides the system project and other appropriate managers with the results of these reviews and audits. A sound QA program ensures that the project is following approved SBA standards and procedures and that quality is checked throughout the product lifecycle.

After approval, all documentation is placed under the configuration management change control process. This change control process is observed whenever any products of the project's baseline are revised. Change control activities include verifying changes made to the products, assigning a new version number to the revision, updating a logbook with the change information, updating the central library with the new version, distributing copies of the new version, and archiving the old version. This process of change control over the lifecycle products continues throughout each project stage.

Quality Assurance (QA) Applications Management Review – The QA Manager continuously reviews project activities, audits software, works products throughout the lifecycle, and provides management information with which to determine whether the project is adhering to established guidelines. A sound QA program ensures that the project follows approved SBA standards and procedures and that quality is checked throughout the product lifecycle.

Before a project is implemented in production, the QA Manager will verify the documentation in the software release package. This documentation will at a minimum consist of:

   a. Product Description

   b. Product Version Number

   c. Change Description (Release Notes describing the purpose and features of the release)

   d. A Test Plan

   e. Test Procedures

      f.   Evidence of successful test completion

      g.   Testers Name

      h.   Customer Proponents Sign off that the product is acceptable

Project Approvals – Formal approvals also occur at each stage in the lifecycle to confirm management support of the project and the resulting system. Conducting reviews and obtaining approvals is not the goal of the lifecycle process, but they are key means to the desired end: successfully solving an information management need. Program management approvals are obtained during all activities from project approval through operation. In selecting organizations and individuals to provide approvals for a particular project, consideration is given to the specific characteristics of the information management need and the proposed solution.

Configuration Management (CM) – Continual, consistent documentation of the development and evolution of the system is essential to ensure that all milestones are met, key decisions are recorded, the system can be accurately described, and a consensus is reached on what products are required for delivery. CM must maintain a controlled library of project products, software, hardware, and documentation and provide a process for the consideration and disposition of requested modifications to the system.

All documentation is placed under the configuration management change control process. This change control process is observed whenever any products of the project's baseline are revised. Change control activities include verifying changes made to the products, assigning a new version number to the revision, updating a logbook with the change information, updating the central library with the new version, distributing copies of the new version, and archiving the old version. This process of change control over the lifecycle products continues throughout each project stage.

Security Management – To ensure effective management of SBA's data resources, all systems are created and maintained in accordance with SBA's IT security policy and practices. Project activities are carried out consistent with the data management environment, and data administration concerns are addressed during all activities of the lifecycle.

Methods, Techniques, and Tools – All systems created and maintained by SBA use clearly articulated methods, techniques, and modern system development and maintenance tools to the greatest extent practical. Specific selections are confirmed in subsequent activities as appropriate. However, no methods or tools can replace the application of system development management. For example, prototyping may be used as a method of performing design or development tasks, but the final products always include the documentation and reviews prescribed by the development approach. Use of expert system development tools may require that some activities be performed in a different sequence or iteratively; but, again, they are documented and reviewed as prescribed by the lifecycle management model.

SBA Internal Service Requests Management Procedures

This procedure documents processes used by the OISS to manage user requests concerning new systems development and maintenance of existing systems and applications. Further, this process records the approval or denial of software releases by OISS and application business owners.

Introduction

Change Requests (CRs) are the means by which OISS tracks system changes requested by the system owners/users. OISS defines, estimates, assigns and tracks work to be performed and CRs become Service Requests (SRs) once they are approved and resources are allocated. A CR can be initiated by one of several sources. These include the following:

    a. Customers. Individuals outside of OISS that use systems supported by OISS;

    b. OCIO/OISS Management;

    c. SBA Others. This includes other SBA personnel within OCIO/OISS that may request services.

CRs are categorized as follows:

    a. *Anticipated:* A change request that is not yet approved or clearly defined.

    b. *Approved:* A change request that is submitted with prior approval from associated change control boards or SBA Management.

    c. *Assigned:* A change that is approved and ready for implementation. It is assigned to a project and associated with a Service Request (SR) number.

    d. *Completed:* A change request that is implemented and closed.

    e. *Withdrawn:* A change request that is not approved or no longer requested.

All assigned and completed CRs are associated with an SR and SRs are categorized based on estimated size, scope of application, complexity of processing, and technologies used. SR categories are as follows:

    a. Small, medium or large (we need to define what small, medium or large projects mean);

    b. Maintenance/repair;

    c. IT strategic major enhancement; and

    d. New initiatives.

SBA Internal Service Request Tracking Methodology – The Service Request Tracking And Reporting System (SRTARS) provides the capability for managers and developers to track CRs and SRs from initiation through completion the efforts expended in resolving requests for services. All work performed by contractors must be entered into SRTARS.

SRTARS is a project management tool that consists of four (4) modules:

    a. **Change Request - Initiation Stage**: Supports the recording and tracking of change requests received from system owners/users. All CRs recorded and initiated via Change Request module of SRTARS.

    b. **SrTars Projects - Development Stage**: Supports the recording and implementation of the projects once they are approved. Includes all active projects and implementation activities.

    c. **Move Request – Migration Stage:** Supports the migration activities from development-to-test and test-to-production phases of the project cycle. Once implementation completed, codes migrated via Move Request module of SRTARS.

    d. **Management Report – Reporting:** Provides project management reports in summary for each cycle of the projects from initiation to completion.

For further instructions on the use of SRTARS, reference the SRTARS user's manual on the help section via http://yesapp.sba.gov/apps/srtars/index.cfm

Initiating Internal Service Request – Users/Individuals of OISS supported systems and applications (those outside of OISS) must initiate requested system and database changes using a SBA User Request (UR) Form 863. The completed UR form must be submitted to the System Owner/Program for review and approval. The applicable Program Offices reviews the UR; performs impact analysis and makes a determination on the UR, approve, defer, or deny. If approved,   it is submitted to OISS Director if is it a new application development request, otherwise it is submitted to the OISS Technical Point of Contact (TPOC) for review and approval.

Note: The OISS Technical Point of Contact (TPOC) for the applicable system will not accept system or database change requests that have not been submitted through the proper channels (this does not include bug/error reports).

Internal Applications Development and Maintenance Procedures – When requests for various system changes are received, they are initially recorded in the Change Request module within SRTARS as anticipated changes. If they are approved, they are marked as "approved changes" and ready for OCIO Management review and approval for implementation unless they are bug fixes. OCIO Management reviews and prioritizes requests based on resource availability as well as strategic direction of the Agency. The Management also selects change request to be reviewed by the Enterprise Architecture Change Control Board (EACCB) based on the scope and technical requirements.

Once the change request approved by the OCIO Management and resources allocated, OISS Team assigns the change request to a project for implementation. At this point, the change request becomes service request and is entered into SrTars project module of SRTARS.

Project Initiation: OISS IT projects are initiated once approved by the OCIO management for implementation. All IT projects are planned, managed and executed from a Software Development Methodology (SDM) approach. Once IT projects are identified, OISS staff follows the steps defined below:

Upon receipt, the TPOC enters the UR information into the SRTARS, which generates a formal SR with a tracking number. The TPOC then forwards the SR to the responsible OISS (contractor) Task Assignment and Control (TAC) Manager.

Definition and Design: The Definition and Design phase involves developing a detailed statement of the user's need and a project plan covering the estimated cost, schedule and technical parameters of the project; review and approval depend on detailed specifications to fulfill the stated functionality requirements.

The Maintenance Requirements Document (MRD) and project plan (PMP, if applicable) must be approved by the TPOC and Customer before proceeding to the next phase. Upon receiving MRD and/or PMP approval from the Program Office, the TAC Manager assigns development staff to work on the SR. The TPOC will enter budget information and activate the SR in SRTARS, which will allow the TAC to enter other applicable information (i.e.: hours, starting and completion dates) in SRTARS.

Building/Development: The Building/Development phase involves developing and testing computer programs in response to approved specifications.

A Verification Validation and Testing (VV&T) plan is used to prepare a User Acceptance test plan.

When work is completed, the developer moves the system to the staging area and notifies the TAC Manager who reviews the code and optionally conducts further tests.

The TAC Manager prepares the release of documentation, which is sent to the TPOC to indicate that the system is ready for user acceptance testing. The release documentation will also identify all updated System documentation included with the release. For Web based applications and changes involving Database releases, the request is submitted by the TAC Manager/developer via the automated *Move Request System.*

Testing and Evaluation: The Testing and Evaluation phase involves testing of the system to ensure that it functions as desired and certifying that it has completed development.

Upon receipt of release documentation and/or request via the automated Move Request System, the TPOC reviews the new/updated system documentation for accuracy, completeness and coordinates releasing the files from staging to the test area. The TPOC also notifies the user that the system is ready for testing and delivers a copy of the test plan.

The customer/user completes the tests listed in the test plan and any others that are appropriate. If there are any problems, the customer notifies the TPOC for corrective action. If there are no problems, the customer returns the test plan/results to the TPOC who in turn submits the release to production.

Production Release and Maintenance: The Production Release and Maintenance phase involves installing the system at all sites at the onset of operational activities. This includes all controls on proposed computer hardware and software changes.

The TPOC coordinates releasing the files from test to the production area via the Move Request System (MRS). The MRS will forward the production release request (in this order): to the Branch Chief, Customer (Program Office), OCIO QA, and the OISS Director for electronic approval before it is forwarded to the System and Database Administrators for implementation. The release must be approved at each point prior to being moved to production.

Once a release has been implemented, the TPOC will copy all the updated System documents to the root system/project directory (current system documentation) and move the project work-folder containing working files to the "History" folder on the OISS Portal.

When a software release results in a production work stoppage, an emergency software release will be performed by the designated contractor staff to restore the production environment. Prior government approval for the emergency release will not be required at the time of the release. Emergency releases will be entered into the Move Request system annotating the release was required to fix a work stoppage on the production system.

Roles and Responsibilities

Personnel and decision-making boards, e.g., System Sponsor, PCCB, QA, CIO, PM, BTIC, BTI-AC.

Integrated Project Team (IPT) Integrated Project Teams are tasked with three key functions for projects:

1. Day-to-day oversight of solutions through their development and operational stages
2. Change control of those solutions
3. Quality management of those solutions

Members of the IPT must include at a minimum: the project sponsor, project manager, key project team leads, and acquisition and OCIO IT representatives including security, privacy, communication technology support, and information systems support.

Day-to-Day Oversight These teams are required as part of the Fiscal Year President's Budget Pre-Select and Select Phases. The primary members of a project review board are: the Project Sponsor, who has been selected during the CPIC Pre-Select Phase and is responsible for articulating this need within the organization; and the Project Manager, who is responsible for the actual development of the system. The Project Sponsor and Project Manager select the remainder of the IPT. IPTs are structured to oversee the development activities of systems across the entire life cycle. Specifically, IPTs are structured to the following activities:

- Review on-going system development activities to ensure that their status, progress, and outlook are satisfactory and consistent with project plans.

- Identify deficiencies in project development, develop corrective actions and monitor their execution.

- Provide recommendations to support their decision to continue reduce or terminate system development activities.

- Conduct periodic reviews of project status, control, performance, risk and outlook.

- Establish and execute the necessary project controls to manage requirements; risk; cost, schedule, and technical baselines; and performance outcomes.

Change Control The IPT also provides for a change control function for small or non-critical projects. This project level function is the control mechanism for the program office that requested the need for which the project has been initiated and for any changes required during the lifecycle of the particular project. For each project, the project sponsor establishes and heads the function.

In the case of larger projects this function is formalized under a Project Change Control Board (PCCB) established and headed by the project sponsor and comprised of program officials, project manager, project technical leads, system users, external stakeholders, and IT/OCIO representatives

Project level change control evaluates the scope, applicability, and effect of each requested requirement change (RC). As part of this function, the IPT focuses on items that could affect cost, schedules, or compliance with technical requirements. It acts on any request for changes to the system and provides change approval or disapproval based on defined strategic initiatives, program business objectives, and budgetary parameters. In addition, the project CCB meets regarding impacts of changes proposed by other program areas or organizations, especially schedule and cost impacts. The P CCB has the authority to establish project baselines, initiate or change system, accept testing results, and approve the release of systems into production.

Quality Assurance (QA) For small projects, the IPT can be responsible for coordinating and implementing quality assurance for IT projects. QA verifies that plans, standards, and procedures are in place and can be used to review and audit the project. In addition, QA reviews system development and maintenance activities to verify compliance. Results of QA activities are reviewed with IT management and project personnel.

In larger, sensitive or otherwise critical projects, QA will be staffed and undertaken separately from the IPT to include personnel separate from the project to ensure objective and impartial QA assessments.

Business Technology Investment Council (BTIC) Responsible for ensuring that Agency resources are directed to the highest information technology priorities of the Agency and that these limited resources are efficiently utilized. The BTIC tracks and reviews progress toward achieving key events and target dates for critical project activities.

Process Coordination

Approved investments and their component projects must move through the CPIC and budget processes to obtain investment funding. They must conform to any guidance issued by the BTIC. The sponsoring program office is responsible for preparation and submission to the Agency of all budget and/or Working Capital Fund requests for its investments.

Document Structure

The SBA SDM document consists of this introduction, six major sections that describe the major stages, and additional supplemental appendices, acronyms and references lists.

The Introduction presents the purpose and benefits of the SDM. An important concept of this SDM is that it can be tailored to satisfy the requirements of any SBA system development project, providing every project a documented System Development Methodology.

Section 1, Initiate, describes the activities required to identify information management needs and the activities to decide whether to commit resources to satisfy or solve a need.

Section 2, Plan, describes the process and activities to effectively plan and document the project's tasks, activities and artifacts

Section 3, Define, describes processes used to establish and document the system's functional and data requirements.

Section 4, Design, discusses developing a system design from the system functional and data requirements.

Section 5, Build and Test, presents the process used by the development team to build and test the system before releasing it to the test team.

Section 6, Implement, discusses the process used by the test team to test the system to ensure the user community that the system functions as expected.

Section 7, Operate and Maintain, gives operational guidelines for system user activities, including system installation and operation and reporting of system deficiencies and enhancements.

Introduction

Section 8, Retire, discusses the processes, activities and artifacts for the closing and final disposition of the system

1   Initiate

Describes the activities required to identify information management needs and the activities to decide whether to commit resources to satisfy or solve a need. It includes the establishment of the project governance structure and processes to ensure the effective delivery of a specific solution.

1.1   Purpose

The Initiate Project stage of system development is the period in which an information management need is identified and the decision is made whether to commit the necessary resources to solve the deficiency. Figure 1-1 highlights the process flow for the Initiate Project stage. The Initiation Stage also supports the CPIC Process Pre-Select Stage by:

- Defining and validating a solution concept for improving business accomplishments or removing a deficiency related to a business need

- Identifying significant assumptions and constraints on systems to that need

- Exploring alternative methods and technologies that implement the solution concept

1.2   Entry Criteria

The Project Sponsor has briefly documented the business need as well as a notional description of the proposed solution including related high-level business processes and IT system(s).

1.3   Process

The Initiate Stage is the first step in the process of answering a business need and transforming it into an actual solution. The need is documented in the form of a Needs Statement or an Advanced Requirements Notice (ARN). Large-scale development and major system enhancements are described in a Needs Statement by the project sponsoring organization and approved by the Business Technology Investment Council (BTIC). Smaller development efforts, minor enhancements, or maintenance efforts are described in an ARN and approved at the Program Office / ADA level. An Evaluation of Request (EOR) is used to document the evaluation of the ARN.

> **SDM Initiate Stage and CPIC Pre-Select Touch Points**
>
> *Pre-Select Phase*
>
> The Needs Statement and preliminary documents developed during the Initiate Stage constitute requirements for the Initiate Sub-Phase of SBA's CPIC PreSelect Phase and are reviewed by the BT-IAC/BTIC before recommending the project to proceed.

Figure 1-1, SDM Initiate Stage/CPIC Pre-Select Phase Touch Points

The process begins with identification of an information management need. A user or system sponsor may identify the need or it may be the result of the Mission or Business Goal need described in the SBA Enterprise Architecture (EA). In either case, it is critical that the proposed solution is addressed by the EA. The Project Sponsor articulates this

need within the organization and appoints a qualified project manager to initiate the project life cycle. Next, an IPT is identified and the Project Charter is expanded to identify each member, their anticipated length of time on the project and their individual roles and responsibilities. After an internal review, the Needs Statement is reviewed by EA and CPIC before being presented to the BT-IAC as part of the CPIC process.

Figure 1-2 Initiate Stage Process Steps

During this stage, the project manager prepares a draft Project Charter and a Needs Statement. The PM also identifies the team that will be responsible for the actual realization of the system. The team helps begin the process of defining business requirements and associated system performance metrics, business performance measures, benefits and costs, as well as initial project planning efforts in preparation for inclusion in an investment and the Agency's IT portfolio.

The team uses the information provided in the draft Project Charter and the draft Needs Statement to develop a high-level scope statement, a description of the project and an initial system concept and associated project documentation described below.

Based on the Needs the team performs an Alternatives Analysis with a Cost/Benefit Analysis from which a Risk Management Strategy and a High Level Project Plan are developed for the preferred alternative.

Decisions throughout the Initiate Stage are recorded in a System Decision Paper (SDP), as is the schedule of activities already accomplished as well as the projected schedule of project activities.

Once accepted and approved, the team completes the rest of the required documentation for the Initiate Stage is reviewed and presented to the BT-IAC/BTIC for recommendation the solution be included in the SBA IT Portfolio as well as for possible funding.

At any of the steps in the Initiate Stage process, it may be required to go back to the next previous step to rework documentation to accommodate requests to amend or provide additional information before proceeding to the next process step.

Having completed these steps, making any requested changes and documenting solution decisions in the System Decision Paper, the project should be ready to proceed to the Plan Stage.

1.3.1   Establish Project Organization

This activity involves the appointment of a project manager who carries both the responsibility and accountability for project execution. For small efforts, this may only involve assigning a project to a manager within an existing organization that already has a project support structure. For new, major projects, a completely new organizational element may be formed requiring the hiring and reassignment of many technical and business specialists.

Each project will have an individual designated to lead the effort. The individual selected will have appropriate skills, experience, credibility, and availability to lead as project manager. Clearly defined authority and responsibility must be provided to the project manager.

The project manager will work with Stakeholders to identify the scope of the proposed system project, the participation of the key organizations, and individuals who participate in the formal reviews of the project. This decision addresses both programmatic and information management oriented participation as well as those technically interested in the project.

The output from this step is an organization chart and list of project participants and stakeholders with their name, title, organization, project role and contact information.

1.3.2   Prepare Initiate Stage Documentation

The documentation identifies the business needs, capabilities, scope of a system. It includes high-level requirements, benefits, business assumptions, and estimated system project costs and schedules. It records management decisions on the envisioned system early in its development and provides guidance on its achievement.

1.3.3   Develop Project Charter

The Project Charter serves as a high-level overview, identifying a project's purpose, objectives, scope, and critical success factors. It also establishes project authority and accountability by designating a project (executive) sponsor, a project initiator, and a project manager who together move a project through the vetting process into the Plan Stage. Include a participant and stakeholder list – this will be the basis for tracking requirements for system acquisition and decision making as well as providing the audience for project communications. This document often serves as an executive overview of the project: a single source for a high-level understanding of the business factors driving the development effort, the products of that effort, and those departments and personnel with a stake in its successful outcome. All of this information will be offered in greater detail in other project documents.

1.3.4   Develop Needs Statement

The Needs Statement describes the need and the benefits to be expected and identifies the estimated costs and resources to be expended to fully develop and implement the project.

Based on the information technology management need(s) the Needs Statement includes:

- Architecture Blueprint or Enterprise Architecture context for project

- Solution concept and performance gaps & goals[1], a brief exploration of alternative solutions from the Alternatives Analysis document

- Initial acquisition approach from the Acquisition Strategy document

- Initial Project Plan and Schedule from the Alternatives Analysis document

- Initial Spending Summary from the Alternatives Analysis document

- Critical Investment-level risks from the Risk Management Strategy

---

[1] This should include Mission Goals from the solution specific Performance Reference Model developed in consultation with the Enterprise Architect.

The approval of the Needs Statement must ensure that the document is correct and complete, as is, without further changes. Any changes or updates date that need to be made should be minor and should be made with change pages distributed by an agreed-upon date without requiring further review.

Any further required changes will have a major impact on the document. A second review must be scheduled. The changes must be incorporated and the resulting change pages distributed at least 10 working days before the second review.

1.3.5   Develop Alternatives Analysis

An alternatives analysis is compiled and submitted for senior management or board study. For each alternative considered, an Alternatives Analysis with a Cost/Benefit Analysis (CBA) is performed. All economic benefits and costs, including development and operation, one-time, and recurring costs, are calculated in current dollar values. Benefits that cannot be quantified in dollar amounts are identified. Each alternative is then evaluated based on the net economic benefits of developing and operating the system, as calculated during the analysis. This evaluation is documented in the benefit/cost analysis, including the analysis methodology, the alternatives with their associated benefits and costs, and a recommendation as to the best development alternative.

1.3.6   Develop Acquisition Strategy

The Acquisition Strategy (AS) is a formal planning document that is based on and refines the acquisition strategy objectives of the approved Mission Need Statement (MNS). It must address all technical, business, management, and other significant considerations that will control the acquisition and describes the program's plans for primary and support procurements, inter-agency agreements, competition, and source selection.

1.3.7   Develop Risk Management Strategy

A Risk Strategy is developed for the proposed preferred solution is then performed, identifying the strengths and weaknesses of the proposed project as well as alternatives for improvement. The purpose of the risk strategy is to select cost-effective safeguards that will reduce project-related risks to an acceptable level. The following determinations are included:

- Security requirements, based on the sensitivity and criticality of the proposed project to the project sponsor and user organizations

- Baseline security requirements that must be satisfied to ensure that applicable SBA guidelines are met

- Threats to the proposed project and the vulnerabilities that could be exploited

- Countermeasures to the identified threats and vulnerabilities

1.3.8   Develop High Level Project Plan

After the Needs Statement is defined, the project manager establishes a high-level project plan outline for all development stages so that adequate resources are defined. Because project planning is an iterative activity, detailed planning of the subsequent stages can occur after the Initiate Project stage is completed and the project is approved

for continuation. Included in the high-level plan is the preliminary strategy for handling programmatic risks that would have a direct impact on schedule, cost, and technical performance. It includes:

- Executive overview of the project

- Schedule and Levels 1 and 2 Work Breakdown Structure (WBS)

- Preliminary strategy for handling programmatic risks that would have a direct impact on schedule, cost, and technical performance

1.3.9   Develop System Decision Paper (SDP)

All decisions made during the Initiate Project stage are recorded in a System Decision Paper (SDP), as is the schedule of activities already accomplished. In addition, a projected schedule of activities for the full lifecycle of the system, by stage, is provided in the SDP. Included in this schedule are the actual or projected beginning and ending dates for each stage, as well as a running total of the actual or projected development and operation costs, broken out by stage and totaled by fiscal year. Major, complex development efforts, generally those addressing departmental priority needs, should follow the activities and tasks as outlined in the System Development Methodology (SDM). Minor efforts may need to tailor the SDM to fit the project's specific needs.

1.3.10   Develop Review Presentations & Material

Compile the Mission Needs Statement with the Alternatives Analysis, Acquisition Strategy, Risk Management Strategy and the high level Project Plan appended. Prepare briefings based on the documentation developed with a BTI-AC/BTIC presentation template.

1.3.11   Review Proposed Solution & Recommend Appropriate Action

Reviews of all project documents are performed by:

- Project Team

- Project Change Control Board (large, complex and/or mission critical projects only)

As described in Table 1.1 Documentation Review Responsibilities, particular documents are reviewed by:

Table 1-1 Documentation Review Responsibilities

Documentation Review Responsibilities		
• Project Sponsor	• CIO	• EA
• CPIC	• Security (OCTS)	• OISS

Based on its review of the Needs Statement and associated documents, the BT-IAC and then the BTIC decide whether to continue with the proposed development effort. If the review board decides to continue, the project manager documents the decision and which development alternative is selected in the SDP. After approval, all documentation

is placed under the configuration management change control process. The project manager then updates the Project Plan and all appropriate documentation accordingly.

1.3.12 Document Project (System) Decisions

All decisions made during the Initiate Project stage are recorded in a System Decision Paper (SDP.) as is the schedule of activities already accomplished. In addition, a projected schedule of activities for the full lifecycle of the system, by stage, is provided in the SDP. Included in this schedule are the actual or projected beginning and ending dates for each stage, as well as a running total of the actual or projected development and operation costs, broken out by stage and totaled by fiscal year.

The SDP is maintained by the project IPT for small projects and by the PCCB for large, complex and/or mission critical projects,

1.4   Performance & Governance Considerations

Key considerations for solution acquisition and integration are avoiding duplication with existing systems and making sure that the system concept being developed satisfies the business requirements. In order to minimize the risk of duplication, the SBA EA Repository (EAR), the Federal Transition Framework (FTF), and agency EA tools will be carefully researched to ascertain what currently exists. In order to ensure that the business focus is being properly maintained, the solution will establish those requirements, metrics and guidelines in the SBA Initiate Stage through development of the system's Performance Reference Model (PRM). It ensures that the "line of sight" from business needs to the conceptual technical system is established and remains unbroken.

From the EA governance perspective the proposed solution must be carefully investigated to ensure that it is contributing, to the maximum extent possible, to the attainment of SBA strategic goals and that it will adhere to all major policies, mandates, and procedures.

1.5   Exit Criteria

Table 1-2 provides a summary of the documents and models generated during the Initiate stage process, as well as the whether the document requires approval or whether the document is required only for the file for recordkeeping purposes.

Table 1-2 Initiate Stage Document Deliverables

SDM Initiate Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Organization Chart and Participant/Stakeholder List.	X	X			
Project Charter	X				
Architecture Blueprint	X		X		X
Needs Statement	X		X	X	

Initiate

SDM Initiate Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Preliminary Project Plan & Schedule	X	X		X	
Risk Assessment Strategy	X	X		X	X
Alternatives Analysis	X	X		X	X
Acquisition Plan	X	X	X	X	X
Review Presentations & Material	X				
System Decision Paper	X	X			
Exhibit 53 (Partial)	X	X	X	X	X
Draft Business Case (Partial – Exhibit 300)	X	X	X	X	X
Approved Initiate Stage SDM Decision Documentation	X	X			
Approved CPIC Planning Milestone Decision Documentation	X	X			

2 Plan

Describes the activities and documents required to define and effectively manage the development, operation and disposition of a system.

2.1 Purpose

The Plan Stage further extends the system concept, describing how the business will operate once the approved system is implemented and assessing the system's impact on business processes, employee and customers. To ensure that the system can deliver the products and/or services required on time and within budget, the project resources, activities, schedules, tools, and reviews are defined and detailed and documented in plans. Additionally, security certification and accreditation as well as record management activities begin with identification of system security requirements and the completion of a high-level vulnerability assessment. These plans and certifications are essential to the success of the entire project; they are carefully reviewed and updated throughout the remaining SDM stages.

The SDM Plan Stage is very closely associated with the CPIC Select Phase through the plans, many of which are used in the CPIC Select Phase to construct the System Boundary Document. Additionally, both the CPIC and SDM processes work closely together during this stage in the areas of risk management, lifecycle costing and overall project planning.

2.2 Entry Criteria

Prior to entering the Plan Stage, the Initiate Stage and documentation must be complete and the proposed project must have obtained BTIC approval.

2.3 Process

The Plan Stage is the most critical step in completing development, acquisition, and maintenance projects. Careful planning, particularly in the early stages of a project, is necessary to coordinate activities and manage project risks effectively. The depth and formality of project plans should be commensurate with the characteristics and risks of a given project. The SDM requirements will scale according to the scope of the system project. The project plans refine the information gathered during the Initiation Stage by further identifying the specific activities and resources required to complete a project. A critical part of a project manager's job is to coordinate discussions between user, audit, security, design, development, and network personnel to identify and document as many functional, security, and network requirements as possible.

Figure 2-1 Plan Stage Process Steps

2.3.1 Define and Document Internal Processes

Project management entails the internal management, engineering, business management, and contract management internal processes used by a project office for all subsequent life-cycle phases. This involves the appointment of individuals in small

Plan

projects or the establishment of teams or working groups in complex projects for specific tasks.

At a minimum, projects should define:

Project Scheduling – Following the positive BTIC ruling, the initial project schedule is detailed and expanded. Major phases and tasks are defined and given dates for completion. This schedule should be flexible to allow for uncertainties in planning.

Deliverable: Updated Project Plan – Project Schedule

Project Budgeting – Initial budget estimations and spending plans of the overall project costs determine if projects are feasible. With the added definition of project requirements and activities in the Plan Stage, Project Managers refine those estimates. These revisions form a baseline budget against which PMs monitor throughout a project, and adjust them as needed. In addition to hardware, software, contractor, personnel expenses and outsourced activities, the costs associated with project overhead such as office space, hardware, and software used during the project must be included.

Deliverable: Updated Project Plan – Lifecycle Budget and Cost Projections

Performance Management – In addition to managing earned value throughout acquisition it is imperative that the project operational performance is assessed against related investment performance measures as outlined in the latter's Performance Reference Model. These elements need to be included in the Project Plan.

Deliverable: Updated Project Plan – Earned Value Management and Performance Goals & Measures

Acquisition Management – Based on the Acquisition Strategy from the Initiate Stage, the team details and documents in a System Support and Acquisition Plan all technical, business, management, and other significant considerations that will control the acquisition and describes the program's plans for primary and support procurements, inter-agency agreements, competition, and source selection. Those significant considerations include, but are not limited to:

- Acquisition activities sequencing
- Processes to monitor and report acquisition activities, status, risks and metrics
- Contractor contract management processes and reporting

Deliverable: – Acquisition Management Plan

Risk Management – The ultimate purpose of risk management is to define and mitigate risks to system acquisition and development. To accomplish this, ongoing analysis is performed to estimate the potential losses that may result from system vulnerabilities and the damage from the occurrence of threats. Critical assets that must be protected are identified with consideration given to the environment in which these assets are stored and processed. One or more countermeasure(s) should be developed for each potential threat and vulnerability.

Based on the Project Risk Strategy from the Initiate Stage, projects should establish and document in a Risk Management Plan, procedures to appropriately assess, monitor, and manage internal and external risks throughout a project's life cycle. The procedures should include risk acceptance, mitigation, and/or transfer strategies.

Deliverable: Project Risk Management Plan

Plan

Configuration Change Management – Configuration management (CM) is the ongoing process of identifying and managing changes to deliverables and other work products as they evolve through system development and maintenance. The objectives of CM are to ensure that baselines are defined, the changes proposed are necessary and appropriate, the integrity of the system is maintained, and a record of changes to the system is kept to provide traceability throughout the lifecycle.

Although the basic elements of CM are the same, the implementation of CM activities varies depending on the type of system and environment requiring control. As such, the CM requirements imposed on a project and/or system must be selected to provide the optimal approach for fulfilling the goals of CM.

The elements for implementing CM are:

- Configuration management program
- Configuration management standards and procedures
- Configuration management measurements
- Training
- Implementation strategy

Representatives from the same departments involved in establishing requirements should be involved in evaluating and approving proposed changes. Large, complex, or mission-critical projects should include formal change management procedures.

Deliverable: Configuration Management Plan

Standards and Quality Assurance – Project plans should reference applicable standards relating to project oversight activities, system controls, and quality assurance. Oversight standards should address project methodology selections, approval authorities, and risk management procedures. System controls standards should address functional, security, and automated-control requirements. Quality assurance standards should address the validity of project assumptions, adherence to project standards, and testing of a product's overall performance. SBA Quality Assurance Board should review, approve, and document any deviations from established standards.

Deliverable: Quality Assurance Plan

Security & Privacy Management – An objective of the project is to ensure protection of information and information processing resources for SBA's sensitive application systems. A System Security & Privacy Plan's purpose is to provide a basic overview of the security and privacy requirements of the project and its plan for meeting those requirements. The System Security Plan also may be viewed as documentation of the structured process of planning adequate, cost-effective security protection for a system. Therefore, it should reflect input from all organizational areas with responsibilities concerning the system.

The project team will need to work with the OCIO Security staff to develop a System Security Plan that describes the controls needed to ensure compliance with Federal and Agency security requirements and to manage the security risks determined in the Initiate Phase and captured in the Risk Management Strategy.

Deliverable: System Security Plan

2.4 Performance & Governance Considerations

The restrictions placed on a system by the Business, Performance and Technical Reference Models should be carefully considered during the system planning, since these restrictions could have a significant effect on project cost and schedule. Existing repositories should also be re-visited at this stage to identify potentially reusable elements from other projects.

2.5 Exit Criteria

Prior to exiting the Plan Stage, the following areas must have been planned and documented in detail:

- Established performance goals and quantifiable performance measures.

- Developed a project plan, which details quantifiable objectives including an acquisition schedule, project deliverables, and projected and actual costs.

- Identified costs, schedule, benefits, and risks.

- Established security, Section 508 (IT accessibility), Privacy Act assessment, data, and architecture goals and measures.

- Established an EA governance review schedule.

- Obtained management approval to enter the Analysis Stage.

Table 2-1 Plan Stage Document Deliverables

SDM Plan Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Detailed Project Plan & Project Schedule	X	X	X	X	X
Acquisition Plan	X	X	X	X	X
Risk Management Plan	X	X	X	X	X
Configuration Management Plan	X		X		X
Quality Assurance Plan	X	X	X	X	X
System Security Plan	X	X	X		X
Approved CPIC Milestone Document	X	X			
Approved Plan Stage SDM Decision Document	X	X			

3 Define

The Define Stage expands system objectives into specific, detailed functional and data requirements, which then form the basis for the detailed design of the system during the Design System phase. These in turn become metrics by which the success and effectiveness of the development and operation of a system is measured. Outputs from these activities could necessitate changes to outputs from previous stages. At the end of the Define Stage, a complete detailed description of the system functions is available to guide the design and subsequent activities of system development. Figure 3-1 highlights the process flow for the Define Stage.

3.1 Purpose

The purpose of the Define Stage is to transform the needs specified in the previous stages into explicit statements of requirements in terms of system inputs, processes, outputs, and interfaces. This process takes place at the functional level, that is, the system is described in terms of the functions to be performed (what it will do), not in terms of software and hardware operations (how it will do it).

The Define Stage begins when the project is cleared to proceed from the Plan Stage. Documentation related to user requirements from the Plan Stage forms the basis for further user needs analysis and the development of detailed user requirements. Because this analysis may reveal new insights into the overall information systems requirements, the team should be prepared to revise deliverables from the Plan Stage, if necessary.

3.2 Entry Criteria

Prior to entering the Define Stage, the following areas must have been planned and documented in detail:

- Established performance goals and quantifiable performance measures

- A project plan which details quantifiable objectives, including an acquisition schedule, project deliverables, and projected and actual costs

- Identified costs, schedule, benefits, and risks

- Established security, Section 508 (IT accessibility), Privacy Act assessment, data, and architecture goals and measures

- An established EA governance review schedule

- Obtained management approval to enter the Define Stage

3.3 Process

During the Define Stage, the user's business need is translated into an IT solution. The users and developers work jointly to define the system's functional and data requirements, as follows:

- Emphasize what needs to be designed, not how to design it.

- Define and allocate requirements from the top down into subcomponents that are as independent as possible so that the system is easy to maintain, understand, and review in segments.

44

- Provide a conceptual or logical data model of the proposed system, independent of any data store, access method, or hardware.

- Document the requirements in a matrix that traces system objectives to functional and data requirements and to subsequent stage products that satisfy and validate the requirements.

- Submit requirements to walkthroughs (peer reviews) to discover potential product flaws and provide critical feedback.

Figure 3-1 Define Stage Process Steps

The following activities are conducted during the Define Stage:

- Determine the scope of the system objectives and major functions.

- Define the boundary of the existing system along with its inputs and outputs. Break down system objectives and major functions to processes, and processes to the single-function level. Write function definitions to describe the actions or activities taking place within the processes. Document definitions in the Business Process Definition and Requirements and Functional Requirements documents.

- Develop a logical data model showing how the data within the system are related in order to minimize duplication of data, provide flexibility, and allow the data to be mapped to a wider variety of database designs. Define data attributes in the model. Document definitions in the Data Requirements Document.

- Crosscheck functional and data requirements to ensure consistency among requirements.

- Use the requirements matrix to trace functional and data requirements to system objectives and needed functions approved by the Business Technology Investment Council (BTIC).

- Determine data conversion and report requirements.

- Maintain the Project Plan. Document project expenses for this stage and any new or revised estimates for subsequent stages to reflect cost and schedule for completing the project.

- Present the System Decision Paper and all required supporting documentation to the appropriate review board at the end of this stage for approval. Projects that receive review board approval will continue to the next stage of system development.

During this stage, the system's functional architecture is defined in terms of the operations and events that must be performed in order to meet the mission goals of the system (business requirements.) This proceeds in a top-down fashion and is independent of allocation to hardware, software, or personnel. The goal of defining the functional requirements is to assist in defining and allocating functions to the system and its users that are best suited to their capabilities and limitations. Requirements are divided into functional and nonfunctional types: functional requirements describe how the system will respond to external stimuli such as a user or external system input, a clock tick or other type of signal; non-functional requirements include performance

requirements, physical and environmental requirements, legal mandates, e.g., Section 508, and service level agreements.

3.3.1   Initiate Acquisition and Support Activities

Begin activities related to acquisition of hardware, software, communications support, and any professional services required to support the project. Perform these activities in coordination with the effort for determining functional requirements. Depending on the scope of the project, such activities may include acquisition of operating system hardware; supporting software packages; services of the vendor(s) who will provide and maintain these tools; and contractor support for the development, testing, and maintenance of the system. Include the following activities:

Initiate System Support Activities – Ensure that supporting organizations, such as Sybase DBgroup Network Integration Branch (NIB) and the ACSPM staff are aware of the project's requirements and currently have the necessary resources to meet those requirements (or are taking appropriate steps to acquire them). Coordinate with other projects being developed to determine if they have similar requirements for hardware or software; if so, steps should be taken to combine the resources of those projects to acquire the necessary hardware or software.

Ensure that NIB and the Sybase Operations Administrator, as supporting organizations, are working concurrently with the project to meet its requirements. NIB must be kept informed regarding the hardware and software requirements of the project, as well as changes to the implementation and processing schedule. Include the following activities:

Coordinate with NIB to ensure that they have ample time to prepare for production involved in the project. Inform NIB of such elements as performance requirements, amount and type of output expected, distribution and archiving of output, and number of expected users. Update NIB on any changes made to production requirements so that they will be able to support the system when it is implemented.

Provide NIB with an approved copy of the project's schedule and ensure that they are made aware of any changes to it.

Initiate Acquisition Activities – Based on the System Support and Acquisition Plan, Project Plan and decisions made for support requirements, estimate hardware, software, communications support, and any professional services requirements for development and operation of the system, and begin acquisition activities for each.

3.3.2   Refine Business Requirements

Some development or acquisition efforts involve creating a new or modified business process. To accurately estimate the level of effort required to achieve desired business goals, an analysis must first be made of the existing business process in the Business Process Definition and Requirements document. In it describe a "to be" business process that meets the business objectives of the proposed system. The document details new process requirements in business terms. Detailing the proposed system's business impact, technology impact, and related risks, business and technical managers may assess, at a high-level, the system's potential return on investment. Additionally, this document establishes the metrics and means of measure to assess whether the

system's presumed benefits are achieved. Please note that this is a non-technical document intended for distribution to a broad, business/operations-focused audience.

3.3.3   Develop and Document Functional Requirements

Functional requirements documentation builds upon the high-level description of the Business Requirements document to detail all operational aspects of the proposed system in the Functional Requirements document. This may include, but need not be limited to required operational support, operations of currently employed system(s), integration and dependencies of the proposed system with other existing or proposed systems, a detailed description of proposed system operations and environment(s), system design considerations, and system security. While this document establishes a technical approach, its description should remain as non-technical as possible, focusing its descriptions on functional/operational impact. This document should be addressed to operational managers employing applications to achieve tactical business objectives in support of high-level strategic goals. Other information determined during the identification of functional requirements includes control objectives, cost constraints, and performance requirements.

All functions defined are allocated to the available resources (humans, hardware, software or combinations). The allocation of some functions will be mandatory and predetermined by constraints established in the Plan Stage or from the Business Requirements analysis. Allocation is determined by comparison of performance between humans, hardware, and software; cost factors; and support for the operators. Allocation decisions are made so as to maximize total system performance and effectiveness. Allocation of a function may require its redefinition into one or more components. Function allocation is also be guided by what pieces of information and decisions are required to initiate, sustain, and otherwise support the functions. The designer must determine how decisions affect or alter the system performance of the system itself. Allocation may be done in static terms or it may be dynamic, with functions changing their allocation at different stages of the system. The system must be defined in terms of component functions, tasks, and subtasks, including the flow of information and the allocation of the functions, tasks, and subtasks to individual system components.

3.3.4   Define System Performance Requirements

Identify all performance requirements the system is expected to meet. Include performance-related characteristics, such as online response time, batch turnaround time, capacity limits, accuracy and validity requirements, input and output effects on performance, and failure contingencies. These must be traceable to the investment's Performance Reference Model.

3.3.5   Develop and Document Data Requirements

Data requirements provide a detailed description of the data model that the system must use to fulfill its functional requirements. Users and developers work jointly to identify requirements and with SBA Data Administration for defining the domain data model. In situations where users and Data Administration determine the model independently of developers, hold walkthroughs during the identification so that users can describe the requirements and the model to developers and receive feedback about the clarity and completeness of requirements. Separate the data description into two categories: static and dynamic data. Arrange data elements in each category in logical groupings, such as functions, subjects, or other groupings most relevant to their use. Describe the type of

information required to document the characteristics of each data element. Specify information, including that related to sensitivity and privacy issues, to be collected by the user and developer.

3.3.6   Identify Hardware and Software Support Requirements

Identify all hardware and software support requirements. Determine these requirements based on the following three factors:

- Needs of the specific application, taking into account available equipment resources, available funding, and SBA's overall systems plans.

- Current technical capabilities available to SBA, including the availability of technically trained personnel and special training necessary to accommodate new hardware or software.

- Assessment of the stability of the technology in place at SBA, primarily the potential impacts of adding a system to current hardware and staff responsibilities.

After these factors are investigated, the logical conclusion may be to procure additional hardware and software to absorb all or part of the requirements of the system/subsystem under development. If so, update the Project Support Plan to reflect the hardware and software needed for the project.

Define & Document Hardware Requirements – Based on the detailed design specifications, Functional Requirements Document, and the Needs Statement, identify the hardware needed to fulfill all or any part of the requirements of the system/subsystem under development, such as workstations, operational system components, and communication hardware. For each item of hardware to be procured, identify the performance and functional requirements it must meet and the preferred vendor(s). Also, include any development and testing and all related security requirements in this determination as well as expected maintenance requirements if the additional equipment will cause a significant change in the current maintenance schedule.

Determine Operational System Hardware Requirements – Estimate the impact of the proposed system's requirements on existing operational hardware as well as any new requirements that result from the detailed system design. Include any additional hardware requirements that may result due to development, testing, maintenance, and related security requirements.

Define Communication Hardware Requirements -- Analyze the specifications outlined in the detailed technical design documentation. Perform an evaluation of the estimated traffic flow and network capability requirements. Based on this analysis, determine the communication hardware requirements of the proposed system. Include additional requirements that may occur due to the impact of the proposed system on existing communication networks and any additional requirements that may result due to additional impacts on security.

Define & Document Software Requirements – Identify software requirements, including system support software and communication software. For each item of software to be acquired, identify the functional and performance requirements to be met, preferred vendor(s), estimated additional online and offline storage required, and estimated system resources necessary to process the application.

Determine System Support Software Requirements -- Based on the system/subsystem design, ascertain the functional and performance requirements for system support software. Estimate the additional online and offline storage requirements and any additional system resources that may be required. Operations should be consulted when making this determination.

Determine Communication Software Requirements -- Based on the detailed system/subsystem design, ascertain the functional and performance requirements for system communications software. Estimate the potential impact these requirements will have on existing SBA communication systems. SBA Operations (OISS) should be consulted when making this determination.

3.3.7   Develop Requirements Traceability Matrix

Requirements traceability ensures that the requirements defined in the functional specification are carried through to the design, build, and evaluation stages. Creating a traceability matrix requires the following steps:

- Decomposition of requirements into discrete statements

- Creation of a requirements table, indicating source(s) for each discrete requirement

- Identification of requirements though each stage of development, with emphasis on design, build, and evaluation

- Development of tests to ensure that requirements are fully met

Requirements traceability is facilitated by placing them in a tabular format. Because requirements are often initially captured in business or operational terms, they must be thoroughly reviewed and decomposed into discrete, consistent elements that can be tested/demonstrated in the operational system. By identifying each requirement with an individual statement, those needs may be more effectively traced throughout the development process.

The first traceability may occur as early as WBS completion. Each requirement is reviewed to ensure that there is a task defined for fulfilling that requirement. Such allocation of requirements to the WBS helps define WBS elements in indicates the scope of work covered by each item. In turn, a more careful estimate of schedule, budget, and resources is possible.

The SBA SDM recommends delaying the traceability matrix no later than the Design Stage as it is essential reference for the technical specifications and testing strategy development.

3.3.8   Update Project Plan

The Project Manager controls the project by monitoring stage activities; taking corrective action where necessary; refining the project plan to account for changes due to actions taken in the current stage and new information for upcoming stages; and reviewing the planned quality process developed by users, developers, testers, and QA for the next stage.

System requirements are complete when every mission need can be traced to a set of requirements and no requirement exists that is not traceable to a business need. Preparations for testing at this point involve the examination of each requirement to

ensure that it is testable and to develop a test plan outlining the requirements testing process.

3.4 Performance & Governance Considerations

During this stage, several factors should be considered:

Examine the Business Reference Model to ensure the functions being developed are similar or identical to existing BRM functionalities. If so, consider the possibility of reusing existing functionality. In the case where the business functionality being developed is new, consider whether this functionality might be usable by other business elements. If so, consider adding the functionality under development to the Agency's BRM and making it available to outside agencies.

Examine the current system development in the context of the Technical Reference Model. Does any element of the new development conflict with the standards, specifications, and technologies described therein? If so, what considerations justify overriding these requirements?

Ensure that the vocabulary being used in system development is consistent with the vocabulary specified by the Service Component Reference Model. If this is not the case, either modify the system vocabulary or develop a cross-reference. Next, try to group functions together into components based on the notion of coherence; that is, select functions that can be grouped around a common theme, such as "calculating performance measures" or "adding a new employee". Record these functional groupings for use in later stages.

Review the Performance Reference Model. Consider whether the system under consideration truly improves strategic and daily decision-making and contributes to the needs of the business.

3.5 Exit Criteria

Prior to exiting the Define Stage, projects must execute the following activities:

- Complete the description of all functional and non-functional system requirements.
- Map all system functional requirements to system functions.
- Allocate each system function to hardware, software or personnel.
- Validate all requirements and prepare for testing.
- Conduct a successful Preliminary Design Review.
- Demonstrate to the SBA BTIC conformance with any applicable guidance issued pursuant to an IT Modernization Blueprint.
- Demonstrate EA alignment.
- Obtain SBA approval to enter the Design Stage.

Table 3-1 provides a summary of the documents generated during the Define Stage process, as well as the whether the document requires approval or whether the document is required only for the file for recordkeeping purposes.

Define

Table 3-1 Define Stage Document Deliverables

SDM Define Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Business Process Definition and Requirements	X				X
Functional Requirements Specifications	X				X
Requirements Traceability Matrix	X				
System Performance Requirements	X				
Hardware & Software Requirements	X				
System Security & Privacy Plan (Update)	X				X
System Test Plan	X		X		X
Quality Assurance Plan & Reports (Update)	X				X
System Requirements Review Materials	X				
Project Plan & Project Schedule (Update)	X	X	X	X	X
Project Status Reports	X				
Quarterly or Monthly Reports	X	X	X	X	X
Approved Define Stage SDM Decision Document	X	X			

4 Design

After the functional and data requirements are documented in the Define Stage, they are further refined to low-level specifications (system/subsystem, program, and database specifications). The specifications are then organized in a way suitable for implementation within the constraints of a physical environment, e.g., servers, database, facilities). The Functional Requirements document and the Data Requirements Document are used to develop the System/Subsystem Specifications, Program Specifications, and Database Specifications documents.

4.1 Purpose

The purpose of the Design Stage is to transform the detailed, defined requirements developed in the Define Stage into complete, detailed specifications for the system that will guide the Build and Test Stage. The decisions made in this stage address, in detail, how the system will meet the defined functional, physical, interface, and data requirements. Design Stage activities may be conducted in an iterative fashion, producing first a general design that emphasizes the functional aspects of the system, then a more detailed design that expands the general design by adding all the necessary technical detail.

Throughout this stage, both the design process and the designs themselves are continuously monitored as part of the CPIC Control Stage. The CPIC Control Stage extends to include the Build & Test and Implementation Stages, continuously evaluating the products and process of the project development.

4.2 Entry Criteria

Prior to entering the Design Stage, the following areas must have been developed and documented in detail:

- Functional and non- functional requirements
- System functions
- Initial Test Plans and Procedures
- A CPIC review schedule
- Management approval to enter the Design Stage

4.3 Process

During this stage, the functional architecture is translated into the information required to create the system and to verify that it meets the requirements specified for it. Attention is focused on decision analysis and task design; information flow (human to computer, human to human); human workload (cognitive, physical, and temporal); human-machine interface; anthropometrics; etc. Trade-offs are considered to ensure that the system architecture optimizes all resources and yields maximum performance. Throughout this stage, the system is continually monitored for performance, maintenance activities, costs, resource allocation, defects, problems, and requirements changes as part of the CPIC Control process.

Lessons learned for both the project and the CPIC process are collected and fed back to prior CPIC stages.

Figure 4-1 Design Stage Process Steps

4.3.1 Continue Acquisition and Support Planning Activities

Those acquisition and support planning activities that were initiated during Define System are continued. These activities are related to the acquisition of hardware, software, and system support items required to continue with the project. At this point in the lifecycle, such tasks typically include continuing the acquisition process for the operating system hardware and environment; supporting software packages; vendor(s) services to provide and maintain these tools; and contractor support for the development, testing, and maintenance of the system.

Continue System Support Activities – Continue the system support activities initiated during the Define stage to ensure that supporting organizations, such as Operations, are aware of the project's requirements and any revisions to those requirements and are currently (or are in the process of becoming) capable of meeting those requirements. Continue coordination to determine if the resources of other projects under development may be combined to acquire the necessary hardware or software.

Coordinate with Operations – Based on the platform and the requirements of the system, contact the appropriate Operations organizations to coordinate support activities to include the following areas:

- Hardware and software installation and checkout for each site

- Supply requirements

- Online and offline storage requirements

- Communications hardware and software

- Report routing

Ensure that all planned system support activities are consistent with the operations guidelines. Before beginning these activities, Operations must be given sufficient advanced notification to prepare the system environment and determine the impact on current operations.

Continue Acquisition Activities – Continue the tasks begun during the Define stage that are necessary to acquire the hardware and software for the development and operation of the system. If necessary, document in the System Support Plan any changes in the strategy.

Continue the software acquisition activities initiated during the Define Stage to ensure the ongoing review and evaluation of potential supporting software packages (such as database management system, data entry system, communication software, security software). Ensure that the necessary paperwork for acquisition of the required software is being prepared and submitted on schedule.

Continue hardware acquisition activities initiated during the Define Stage, including the review, evaluation, and acquisition of operating system hardware, communications hardware, online terminals, printers, security devices, or additional disk storage. Ensure

that the necessary paperwork for acquisition of the required hardware is being prepared and submitted on schedule.

4.3.2 Develop & Document Technical Specifications

The Technical Specifications document is the primary means by which the expectations of business stakeholders are communicated to designers and builders. Functional/Operational requirements are addressed in a specific technical architecture, comprising the system technical solution. There is significant overlap between the Technical Specifications document and the System/Subsystem Specification document. The latter has detail that is more technical and may be considered an expanded subset of the Technical Specifications document.

4.3.3 Develop & Document System/Subsystem Specifications

The primary focus of this activity is to develop and describe the detailed design of the system/subsystems. The development team analyzes the functions, described during the Define Stage, and determines which functions logically may be processed together to form a subsystem. The development team documents the system/subsystem specifications that are used to partition the system/subsystem and hierarchically decompose it to the lowest-level functions. The system/subsystem specifications comprise the system detailed design.

Analyze Functional Requirements and Data Requirements – The project development team performs a detailed study of the functional and data requirements developed during the Define Stage to determine which requirements and data logically may be processed together. The logical groupings of requirements and data are subsystems.

Describe System/Subsystem Specifications – Develop a description of the system/subsystem specifications. If a prototype was developed earlier in the lifecycle, include those specifications defined with the prototype. For each system/subsystem, include such information as timing requirements, equipment needs, communication environment, support software, security, and input and output records. During the process of describing the system/subsystem specifications, the project development team performs the following actions:

- Partition the functional and data requirements into logical subsystems.

- Document the traceability between the functional and data requirements and the defined subsystems.

- Characterize the features incorporated in the design of the system/subsystem that meet the accuracy and validity requirements imposed on the system/subsystem during the Define Stage.

- Determine the timing requirements imposed on the system/subsystem under varying conditions.

- Determine the system/subsystem's capabilities for adapting to changing requirements, and document how these capabilities are incorporated into the system/subsystem's design. Identify those procedures and components designed to support failure contingencies.

- Identify the equipment needed for the operation of the system/subsystem. Include a description of the equipment that is currently available as well as the

characteristics of any new equipment requirements that may be necessary for the system/subsystem to operate correctly.

- Identify the system/subsystem's data communications environment. Develop detailed schematics of the portions of the communications environment that directly relate to the system/subsystem.

- Identify the software with which the system/subsystem must interact. Include both support software and test software, if needed. Provide the correct name, description, and documentation references of each such software system, subsystem, and program. Include a reference to the language (compiler, assembler, program, and query) and to the operating system to be used. If operation of the computer programs to be developed is dependent on forthcoming changes to other software, identify and discuss the nature, status, and anticipated availability date of such changes.

- Outline the potential threats to the system/subsystem and the security measures that have been incorporated to reduce or eliminate those threats. An example would be the threat of unauthorized access being reduced by incorporating into the design a password requirement at all system/subsystem access points. List and describe all controls that have been incorporated in the system/subsystem design; for example, batch controls on the entry of financial data.

- Identify the input and output data and reports. For inputs, provide information about the characteristics of each input to the system/subsystem, such as file name, format and type of data, validation criteria, volume, frequency, means of entry, source documents and their disposition, and security and privacy considerations. For output, provide file names, report formats, selection criteria for displays or outputs, types of output media, and disposition of products.

Perform Walkthroughs – Arrange one or more review walkthroughs to educate the attendees on the system/subsystem design and specifications. Include representatives from 1) other development and maintenance teams whose programs interface the system/subsystems being designed, 2) the system acceptance test team staff, 3) and user and sponsor organizations. Choose a walkthrough moderator to record action items and track them to completion. Incorporate feedback from the attendees into the design and specifications.

Document System/Subsystem Specifications – Prepare the System/Subsystem Specifications document in accordance with SBA SDM documentation standards.

Update the requirements traceability matrix to reflect the allocation of system objectives to system/subsystem design components.

The final element in this process is the Critical Design Review. This review is attended by the technical management team, project stakeholders and upper management. At this review, all hardware designs, software designs, and tasks assigned to systems operators and/or other personnel are examined in detail. Traceability between all functional system requirements and the designs is demonstrated. Testing strategies are presented and evaluated for complete testing of all functional and non-functional requirements. The results of this review form the basis for a management's decision as to whether the project should be allowed to proceed forward.

4.3.4 *Develop Database Specifications*

Develop and document the detailed design for the system's databases. Describe the database management system and identify the database administrator. Include the following activities:

Identify the Unique Database Names – Specify the code name, tag, or label by which each database table or file may be uniquely identified. Additional descriptive information also should be given, whether or not it is implied in the identification code, such as the system using the database, the database status, and a physical description of the database table or file.

Identify Any Special Instructions for Database Usage – Provide any special instructions to personnel who will contribute to the generation of the database or who may use it for testing or operational purposes, including criteria, procedures, and format for submitting data for entry into the database. Identify a data control organization.

Reference All Support Software – Describe all support software directly related to the database. Descriptions should include name, function, major operating characteristics, and machine run instructions for using the support software. Cite the support software documentation by title, number, and appropriate sections. Examples of support software are database management systems, query language, report writers, storage allocation software, database loading software programs, and file processing programs.

Identify Database Security Considerations, Sensitivities, and Critical Issues – Describe the security-related requirements of the database. Include such items as whether the database will contain sensitive data or data critical to the operation of the system. If sensitive or critical data will be part of the database, identify those measures put in place to safeguard the database, such as methods to guard against unauthorized entry into the database and to determine if unauthorized entry has been attempted, contingency plans if the database cannot be accessed, and steps to be taken to recover lost or damaged data.

Identify the System Administration and Control Personnel -- Identify the organizational areas and individuals who will serve in the following capacities:

Database administrator: The person responsible for the planning, definition, organization, protection, and efficiency of data and databases within SBA and specific to the project.

System administrator: The person in the system owner organization within SBA who has responsibility for the administration of the system. This individual gives the final approval to a user's request for access to the system.

System control officer: The person within each SBA organization who reviews and approves access request by users within their organization. This person determines the type of access the user needs to perform his/her tasks.

Security administrator: The person who controls the necessary activities that grant a user access to application systems based on access requests that have been approved by the system administrator.

Identify the Database Management System Configuration – Identify and describe the database management system (DBMS) configuration. Include such information as the operating system and storage device on which the DBMS is to be located, identification of the application software, all related computer programs that run the database and the amount of online and offline storage required. If special hardware or software is required to maintain backup copies or archived files, identify these also.

Describe the Database Schema – Describe the database schema and related sub schemas, including the names of all database-related application programs and data files, as well as the file sizes of each and data record layouts and descriptions.

Prepare the Database Specifications Document – The development team prepares the Database Specifications document in accordance with SBA SDM documentation standards.

4.3.5   Develop Program Specifications

For each lowest-level function identified in the System/Subsystem Specifications, prepare a detailed program specification. Develop in detail and document the functions, timing requirements, interfaces, input and output reports, and the accuracy and validity requirements.

Describe the Functions of the Software Units – Describe the functions that each program is expected to perform, encompassing a level of detail that will allow the program to be coded in software units. The functions should be taken directly from the corresponding sections of both the Functional Requirements Document and the System/Subsystem Specifications.

Describe Accuracy and Validity Characteristics Imposed on Software Units – Describe the data accuracy requirements and data validity requirements imposed on each software unit. Data accuracy may include identifying any mathematical calculations to be performed by a program. Data validation identifies the edit and error routines to be run on the data, such as ensuring that an all-numeric field contains only numbers and no alphabetic or special characters.

Describe Timing Requirements Placed on Software Units – Describe the timing requirements imposed on each software unit under varying conditions such as throughput time, response time to queries and updates of data files, response time of major program functions, sequential relationships of program functions and data flows, priorities imposed by types of input and change in modes of operation, timing requirements for the range of traffic load under varying operational conditions, and input/output transfer time required for disk and tape.

Describe Support Software – Provide a description of the support software with which each software unit must interact. Include support software, test software, and security software, if needed. Provide the correct name, description, and documentation references of each such software system, subsystem, and program. Included must be a reference to the language (compiler, assembler, program, and query) and to the operating system to be used. If the operation of the software units to be developed is dependent on forthcoming changes to other software, identify and discuss the nature, status, and anticipated availability date of such changes.

Describe Interfaces With Other Application Software – Describe the interfaces with other application software, including those from other operational organizations. For each interface, include the following information: type of interface, such as operator

control of a terminal or program interfaces with other programs; description of operational implications of data transfer, including security considerations; data transfer requirements to and from the subject program (including data content, sequence, timing, format, volume, and processing); current formats of interchanged data; interface procedures, including telecommunications considerations; interface equipment; and data conversion requirements.

Describe Storage Requirements – Describe the data storage requirements, including internal storage requirements, use of internal storage and auxiliary storage such as tape and disk, and the estimated quantity of storage required for each. Each program must consider the following types of information for the various storage media:

Internal: Describe and illustrate the use of internal storage areas, including indexing and working areas. Briefly state the equipment constraints and design considerations that affect the use of internal storage.

Device: List by device type all peripheral storage required and any constraints imposed on storage requirements by each storage device. State requirements for permanent and temporary storage, including overlays.

Offline: Describe the form, media, and storage requirements of all offline storage.

Describe the Degree of Security Required – Determine the security classifications of the data used by the programs and the degree of security required for the algorithms. Determine if the data is always in a sensitive state. If the data becomes sensitive upon the occurrence of specific events, describe those events. Specify the operational environment that must exist when sensitive data of the system/subsystem are being processed.

Describe the Input and Output Data and Reports – Describe the input and output data and reports. For inputs, provide information about the characteristics of each input to the program such as file name, format and type of data; validation criteria; volume, frequency, and means of entry; source document and its disposition; and security and privacy considerations. For output, provide file name, report format, selection criteria for display or output, output media, and disposition of products.

Define the Data Retention Period – Determine the period of time the system/subsystem is required to retain its data and include this retention period in the program specifications for each of the data files used and/or created by the system.

Describe the Logic of Each Software Unit – Describe the logic of each software unit in the Program Specifications. Present the logical flow in graphical format showing the hierarchical arrangement of the software units and their interfaces. Supplement the graphic with pseudo-code and English language narrative explanations. Explain in detail the methods for identifying error conditions and the resulting actions of the program.

Prepare the Program Specifications Document – The development team prepares the Program Specifications document in accordance with SBA SDM documentation standards.

Update the Requirements Traceability Matrix to reflect the allocation of requirements to software units.

4.3.6   Develop System Testing Strategy

Document the system testing strategy in the Verification, Validation, and Test (VV&T) Plan and provide for acceptance testing of all components of the system, including detailed requirements for all tests, testing methods and tools, and test evaluation criteria.

Identify Test Evaluation Criteria – Determine the specific criteria that each segment of the system/subsystem must meet. Such criteria are described by the user of the system/subsystem and typically are a mix of functional and performance requirements, such as processing data within a certain time frame, producing a report, or responding to an online query within a certain amount of time. The majority of the tests to be included in the VV&T Plan should be designed to prove the system's ability to meet the user's acceptance criteria.

Determine User System Acceptance Criteria – Determine the minimum function and performance criteria that must be met for the system to be accepted as "fit for use" by the user or sponsoring organization.

Obtain Concurrence(s) on Acceptance Criteria – Obtain the approval of the user or sponsoring organization on the minimum criteria the system must meet for it to be accepted as "fit for use" by their organization.

Identify Test Resources – Identify those items that will be needed during the execution of each system test. Prepare a testing schedule to reflect unit, integration, and system acceptance testing activities. Provide the following information:

- Software requirements

- Deliverable materials

- Security considerations

- Site-supplied materials

- Personnel and equipment requirements

Prepare a Testing Schedule – Prepare a testing schedule to reflect the unit, integration, and system acceptance tests and the time duration of each. This schedule should reflect the personnel involved in the test effort and the site location. In the test schedule, reflect the following information:

Table 4-1 Test Scheduling Requirements

Test Scheduling Requirements	
• Documentation review	• Data preparation
• Test execution	• Output review
• System certification	• System release
• Return of test site to pretest condition	• Identify Software Requirements

Identify any software requirements that will be needed for the tests, including software tools or utilities, such as tape compares. Include version numbers and media type for all required software.

Design

Identify Deliverable Materials – List all deliverable materials, including technical and documentation deliverables that will be needed for the tests. Version and release numbers should be included as well as the names of any supporting off-the-shelf software requirements.

Identify Any Security Considerations – Prepare a list of requirements necessary to ensure the integrity of the testing procedures, data, and site. Any special security considerations (e.g., passwords, classifications, security or monitoring software, or computer room badges) should be described in detail.

Identify Site-supplied Materials – Describe any materials required to perform the test that need to be supplied at the test site. These materials could include desks, chairs, special equipment, and office supplies, as necessary.

Identify Personnel Requirements – A listing of the personnel necessary to perform the test should be provided. For each of the personnel, this listing should provide the following information:

Name, title, current organization, grade (if known), and level of security background investigation

- Description of the required tasks to be performed
- Geographical location of the work to be performed
- Time required (dates needed)
- Whether the requirement is full time, part time, or as needed
- Any special skills required (i.e., programming language, machine familiarity)

Identify Equipment Requirements – Describe the equipment needed to perform the test. Include the type of equipment, the amount, the location, and the time periods needed.

Describe Testing Methods and Tools – Describe the testing methods and tools. First, describe the testing method to be used during the system test. Then, identify the testing tools to be used during the preparation for and execution of the test. Finally, describe the specific tasks and activities to be accomplished during the designated testing period.

Identify Testing Methods and Tools – Analyze the detailed design specifications to determine the appropriate methods to employ when performing the unit, integration, and system tests. In this analysis, include a reference to all tools that may be required in the performance of the test. Cross-reference the methodologies and tools selected with the design specifications to ensure that all areas of the system are thoroughly tested before their operational release.

Describe Testing Tasks and Activities – Based on the determinations made in the analysis of the detailed design specifications, prepare descriptions of the necessary tasks and activities that will be performed during the described tests. Include the methodology and tools to be employed during testing. Prepare a cross-reference to the system's detailed design documents to ensure that all tasks and activities necessary to perform a thorough test of the system are described.

Prepare Preliminary VV&T Plan – Document all information in the VV&T Plan in accordance with SBA SDM documentation standards.

Update the requirements traceability matrix to reflect the allocation of requirements to acceptance criteria.

Describe the overall activities involved with testing and the schedule for performing the activities. For each level of testing (unit, integration, and system), document the following information: organizations involved, hardware and software required for testing, location of tests, test data development, test scenarios, evaluation criteria, procedures for testing, and certification procedures.

4.3.7   Develop Initial Test Plan

The Test Plan (Unit and Integration) defines the testing techniques, procedures, and schedules throughout the Build and Evaluate System phases of development. The Test Plan also assigns testing responsibilities to project personnel and the methods of test execution and test cycle performance activities to identify and correct errors, then retest. The Test Plan has significant overlap with the Validation, Verification, and Testing Plan. The Test Plan (Unit and Integration) is a descriptive subset of the VV&T Plan, focusing on test definitions and execution. It might be used in place of the VV&T Plan for smaller projects.

Develop Preliminary User's Manual – Develop a User's Manual that provides system procedures necessary to allow user organizations to determine the software's applicability and when and how to use it. Update current office procedure documentation to reflect changes that will occur as a result of the system becoming operational.

Prepare System Overview – Develop an overview of the system detailing the intended purpose of the system and outlining how that purpose is met by the system. Include in this outline a description of the following:

- Intended purpose of system(s)

- Relationships among inputs, outputs, and system functions

- Equipment, communications, and networks

- System performance capabilities

- Contingencies and alternate modes of operation

- Method used to store and maintain data

- Data flows

Describe Input Procedures and Expected Output – Prepare a detailed series of instructions (in non-technical terms) describing the procedures the user will need to follow to use the system. The following information should be included in these instructions:

- Detailed procedures to initiate system operation, including identification of job request forms or control statements and the input's frequency, reason, origin, and medium for each type of output

- Illustrations of input formats

- Descriptions of input preparation rules

- Descriptions of output procedures identifying output formats and specifying the output's purpose, frequency, options, media, and location

- Identification of all codes and abbreviations used in the system's output

- Description of all recovery and error correction procedures, including error conditions that may be generated, corrective actions that may need to be taken, and any recovery and restart procedures

Describe File Query Procedures – Develop detailed descriptions of the procedures necessary for file query including the parameters of the query and the sequenced control instructions to extract query requests from the database.

Update Office Procedures – Update or create office procedure documentation to reflect new procedures to be instituted upon release of the system into operation.

4.3.8 Update System Audit Strategy

Notify the OIG staff about any work products completed during the current phase and informed of any changes that may have an impact on internal data processing controls as they were described in the specific phase documentation. Activities include coordinating with IG staff, providing input to IG staff, and updating the Internal Audit Plan.

Schedule meeting(s) with the Inspector General's staff to inform them of any developments that occurred during the specific phase. Determine the scope of the information required from the project manager in order for the IG staff to perform the necessary updates to the Internal Audit Plan and provide the OIG staff with the necessary information.

Finally, update the Internal Audit Plan.

4.3.9 Refine & Update Project Planning Documentation

The Project Plan is continually refined throughout the development process as new details become available. The Project Manager controls the project by monitoring phase activities, taking corrective action where necessary, and refining the project plan to account for changes attributable to actions taken in the current phase and new information for upcoming phases. The Project Manager also reviews the quality process planned by users, developers, testers, and quality assurance for the next phase. Activities include the following:

Update Project Plan – Update the Project Plan with cost, schedule, and budget data for the current phase to the level of detail necessary to reflect the status of the project. Review the plan and determine if any activities described for the next phase and subsequent ones are affected by the completion status of the current phase activities. Adjust the schedules and resource requirements for activities in the next phase, if necessary, and assign starting and ending dates for activities that have been affected. Take into account that the starting date for some activities may be dependent on the completion of other activities. Update milestones, schedules, and resource requirements for the remainder of the project.

Review Planned Quality Process & Plans – Project manager, users, developers, testers, and QA work closely together to determine the process that will be used to build the product with the desired quality. Activities include tailoring the planned quality process for the next phase and identifying the standards and procedures to be used. The project manager reviews the quality process for approval and execution in the next phase.

4.3.10 Update System Decision Paper

Update the System Decision Paper to include a summary of the progress made during the specific SDM phase and the schedule of events for SDM phase activities. Include the current status of the project. List in detail any changes to the original plan for the development of the project and the parties responsible for approving those changes. The updated System Decision Paper is then reviewed and approved by the appropriate management officials. Include the following activities:

Summarize Progress of System – Summarize the progress of the project through the specific phase, and update the System Decision Paper to include this summary.

Identify Development Alternative Selected – Briefly describe the selected development alternative, and provide a justification for the selection of this alternative. Update the System Decision Paper to include this information.

Identify Resources Expended, Needed, and Projected – Identify resources that were expended during the specific phase, and update the projected schedule of resources to be expended through the remaining system life cycle.

Identify Changes – Document any changes to the system development milestones and schedule or any modifications made to the project strategy. Update the System Decision Paper accordingly.

Describe Changes in Milestones, Schedules, and Activities – Provide a description of any SDM phase occurrences that may have had an impact on the projected milestones, schedules, and activities of the development effort. Provide a brief explanation of the nature of these impacts and describe any changes to the milestones or schedule that may occur as a result of these changes.

Describe Modifications to Project Strategy – Provide a description of any modifications to the project strategy that have occurred during the specific SDM phase of development. Provide a brief explanation of these modifications, indicating the reasons for the changes as well as any anticipated impacts.

Summarize Schedule of Events – Develop a brief summary of the events that took place during this phase of development and a description of the remaining schedule of development activities. Show the impact these events have had on the original schedule and any anticipated impacts on future development.

Summarize Status – Provide a status report of the development of the system. Include in the status report the percentage of the system that has been developed and successfully passed integration testing and its readiness for acceptance testing, as reported by the integration test team in the test analysis report. Indicate the components or subsystem of the system that may be causing delays or problems.

Document all information for this phase in the System Decision Paper.

4.3.11 Perform Change Control Activities

Change Control activities occur throughout the development process as proposed changes to the specification or any baseline document must be reviewed and authorized. This description of the Change Control process is the same for SDM Sections 1-5.

The CM function manages, implements, and reports on project CM activities. In supporting the project, CM performs the following activities:

- Verify changes made to product (document, software or system). Review the updated product to ensure that the changes have been made as described in supporting documentation. Supporting documentation may be comments received from document reviews. Supporting documentation for software may be software change requests or discrepancy reports generated during testing.

- Assign version number. The version number must follow conventions established by SBA and enable the project's CM to monitor updates to the product and assist in its distribution. For software programs, increment the version number each time a change is made to the software to correct deficiencies found during testing.

- Store approved version in central library. After each baseline product has been completed and approved, according to SBA procedures, store the approved version in the project's central library. The project's CM function controls access to the library. For software, the library can be a subdirectory or dataset where all baselined software configuration items (CIs) will be stored.

- Record product information in inventory log. Maintain an inventory log that includes the title of the product, release date, version number, name and version or model numbers of the software and hardware used in the development of the product, name of the organization responsible for development of the product (usually the sponsoring organization), and the product distribution list.

- Distribute copies of products as required. The project's CM function distributes copies of the products according to a distribution list maintained as part of the inventory log information, with distribution based on need and the security level of the product.

- Archive old versions of products. Archive and retain outdated versions of all products for the required period of time, in keeping with SBA standards and guidelines.

4.4   Performance & Governance Considerations

During this stage, the following factors should be considered:

Examine the Business Reference Model – Are the functions being developed similar or identical to existing BRM functionalities? If so, consider the possibility of reusing existing functionality. In the case where the business functionality being developed is new, consider whether this functionality might be usable by other business elements. If so, consider adding the functionality under development to the Agency's BRM and making it available to outside agencies.

Examine the current system development in the context of the Technical Reference Model – Does any element of the new development conflict with the standards, specifications, and technologies described therein? If so, what considerations justify overriding these requirements?

Examine the Service Component Reference Model – Are services available that satisfy the system requirements? Is it possible to reuse existing components by either making small changes to the requirements or developing small "translation" functions to allow this reuse? If so, consider how best to reuse these capabilities. Also, examine the new functional groupings being developed (see section 4.4). Are these functionalities applicable to other systems or existing systems? Would they be shareable with minor modifications? If so, consider making these services available to others by adding them to the SRM

Design

Review the Performance Reference Model – Consider whether the system under consideration is, in fact, improving strategic and daily decision-making and contributing to the needs of the business.

4.5   Exit Criteria

Prior to exiting the Design Stage, projects must execute the following activities:

- Complete the design of hardware and software configuration items.
- Develop test plans for each hardware and software configuration item.
- Ensure that every defined function is implemented.
- Ensure that every system or hardware configuration item function corresponds to a required system function.
- Conduct a successful Critical Design Review.
- Demonstrate to the BTIC conformance with any applicable guidance issued pursuant to an IT Modernization Blueprint.
- Demonstrate EA alignment.
- Obtain SBA approval to enter the Build and Test Stage.

Table 4-2 Design Stage Document Deliverables

SDM Design Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Project Plan (Updated)	X				
Requirements Traceability Matrix (Updated)	X				
System Audit Paper (updated)	X				
Test Plan	X				X
Risk Assessment (Updated)*	X		X		X
Critical Design Review Presentation Materials	X	X	X	X	X
Product Baseline	X				
Project Planning Documents	X				
Approved Design Stage SDM Decision Document	X	X			

5   Build

Presents the process used by the development team to build and test the system before releasing it to the test team.

5.1   Purpose

The Build Stage of system development is the period in which developers take the detailed logical information provided in the Design Stage, transform it into machine-executable form, and ensure that all of the individual components of the system function correctly and interface properly with other components within the system. Figure 4-1 highlights the process flow for the Build Stage.

The activities of this stage translate the system design produced in the Design Stage into a working information system capable of addressing the system requirements. The stage contains activities for building the system, testing it, and conducting further functional qualification testing to ensure the system satisfies the functional requirements in the functional requirements specification. At the end of this stage, the system will be ready for the Build and Test Stage.

Figure 5-1 Build Stage Process Steps

This stage is usually carried out in parallel with the development of user procedures and user documentation for the Implement Stage. Both of these will be required for module testing, upon the completion of the Build Stage. Coordination of the Build and Implement Stage activities is a key responsibility of the project manager at this time.

Any special procedures for data conversion and/or data warehousing are also developed and tested here. The process of developing and testing the data conversion and data warehousing modules is no different from that required for the system itself.

The purpose of integrating and testing is to ensure that the individual system components that successfully completed unit and integration testing during the Build Stage:

- Satisfy the functional requirements;
- Satisfy the business needs;
- Adhere to all mandates, physical constraints and service level agreements; and
- Operate as described in the User and Operator Manuals.

5.2   Entry Criteria

Prior to entering the Build Stage, the following areas must have been developed and documented in detail:

- Design specifications
- Test plans and procedures
- A CPIC review schedule

66

- Management approval to enter the Build and Test Stage

Before entering the Test part of this stage, the entire system must be ready for assembly and subsequent integration testing. This means:

- All system and hardware configuration item components have been acquired, assembled and successfully tested.

- All integration test plans are prepared.

- Transition approach of existing data and processes that will be reused is complete.

5.3 Process

After the system design is completed and approved in the Design System phase, it is used as a blueprint to develop the database and computer programs that will be used by the target hardware and in conjunction with any specified support software to automate the system. To aid in this effort, any additional hardware or support software required to develop, test, or operate the system is procured and installed.

During this phase of system development, supporting organizations assist the project team in performing the following activities:

- Compile and record in system documentation, e.g., users, operations, and maintenance manuals, information needed to use, operate, and maintain the system.

- Finalize and document in the Validation, Verification, and Test (VV&T) Plan the strategy for testing the system in the Evaluate System phase.

- Finalize in the Training Plan the strategy for training staff that will use and operate the system or will oversee users of the system.

- Provide staff from other affected SBA organizations, including the Inspector General's Office, with updates on project status and any other requested project information.

- Maintain the Project Plan. Document project expenses for this phase and any new or revised estimates for subsequent phases to reflect cost and schedule for completing the project.

- Track all hardware and software, whether developed or procured, as well as all documentation, using the project's CM function.

- Revise and review prior products to reflect information learned during the Build System phase and review those products.

- Obtain approval of products produced during the Build System phase through project staff reviews and appropriate SBA review boards. Table 4-1 lists the functions and products resulting from the Build System phase.

- Use information gathered during this phase to update the System Decision Paper. Present the paper to the appropriate review board at the end of this phase. Projects that receive review board approval will continue on to the Evaluate System phase of the development lifecycle.

During this stage, detailed testing is carried out to ensure that the developed system satisfies the requirements defined in the functional requirements specification. Several

types of tests will be conducted in this stage. First, system tests are executed and evaluated by the development team to prove that the program components integrate properly into the subsystems and that the subsystems integrate properly into an application. Next, the testing team conducts and evaluates system tests to ensure the developed system meets all technical requirements, including performance requirements. Next, the testing team and the security program manager conduct security tests to validate that the access and data security requirements are met. Finally, users participate in acceptance testing to confirm that the developed system meets all user requirements as stated in the functional requirements specification. Acceptance testing is then carried out in a simulated "real" user environment with the users interacting with a collection of simulated and real external systems, data and infrastructures.

5.3.1   Acquire and Install Development & Test Site System Components

Procure, install, and verify all hardware and software required for coding, physical implementation of the database, and testing the system at all locations where these activities are to take place.

Acquire System Hardware – This activity began during the Define System phase, when the procurement process was started, to allow for timely delivery of hardware. During this activity, the project manager ensures the timely arrival of the correct hardware as specified as part of the system requirements and recorded in the system support plan.

Execute the final steps in the procurement of any additional hardware at this point of system development to ensure that the hardware will be ready for system development and test personnel. Select hardware in accordance with SBA's hardware platform standard.

Acquire Workstation Hardware – Procure all hardware that will be used as the input and output devices by the system users and operators. Workstation hardware may include PCs, portable workstations, scanning devices, pointing devices, remote printers available for users' access, and display and security hardware not otherwise provided by SBA.

Acquire Operational System Hardware – Procure any additional operational system hardware that will be used as the central or front-end processing unit(s) and other equipment central to the operation of the system. Additional system hardware may include LAN, Web, Database and Application servers, Personal computers (PCs), Direct Access Storage Device (DASD) equipment, tape drives, security hardware, and central printing equipment not otherwise provided by SBA.

Acquire Communication Hardware – Procure any hardware not presently available at SBA to provide connectivity between remote workstations and the Central Processing Unit (CPU) or other computer systems. Communication hardware may include switching equipment, cabling, multiplexers, modems, gateways, protocol converters, communications boards, and security hardware not otherwise provided by SBA.

Acquire System Software – This activity began during the Define System phase, when the procurement process was started, to allow for the timely delivery of the software. During this activity, the project manager ensures the timely arrival of the correct software, and the correct version of the software, targeted for operation on the existing or newly acquired hardware.

Execute the final steps in the procurement of any additional software at this point of system development to ensure that the software will be ready for system development

and test personnel. Select all software selected in accordance with SBA's software and security standards.

Acquire System Support Software – Procure any commercially available system support software not presently available at SBA to provide support to the system under development. System software may include DBMS, compilers, database query software, debuggers, security software, operating system software, and test support software.

Acquire Communication Software – Procure any software not presently available at SBA to provide connectivity between remote workstations and the CPU(s) or other computer systems. Communication software may include Network Operating Systems (NOSs), security software, gateway software, and protocol support and conversion software.

Integrate and Install System Hardware & Software at Development & Test Site – After the system hardware and software have been procured and any modifications to the physical facilities are completed as outlined in the system support plan, install the procured hardware and software and verify each component individually for correct operation. Thoroughly test security software and hardware to ensure a safe operating environment. After the hardware is procured and tested, install and test the procured software to ensure correct operation with the integrated hardware environment. Coordinate all hardware and software installations and perform a checkout with the operations organization appropriate for the system platform.

5.3.2   Develop Database

After the development and test system hardware and software have been installed and verified, and before the application programs have been developed, define and verify the structure of the database specific to the system being developed, in accordance with the DBMS. Test the database using actual data developed for the test. Perform a formal review of the database with participation from all organizations that have an interest in the system development, data administration, or system and database operations. Update the database specifications developed during the Define System phase to reflect any modifications to the structure of the database necessary to resolve issues found during database testing and review.

Build Database – Using the DBMS design as described in the database specifications, implement the design in the selected DBMS. This effort entails creating the database on the hardware in the physical environment, performing a review to check for errors, and then making any necessary modifications to the database design.

Create and Load Data Into Database – Using the DBMS, and any software supported by the DBMS, build the physical database structure in which the data will reside. This effort includes the development of all datasets and files containing the correct record structure, associated keys, and the development of all index and validation files.

Upon establishing the database, the test data to be used during testing of the database and development of application programs should be uploaded. The data may be actual data from existing automated systems, with modifications to test for error checking, or may be developed specifically for the purposes of testing.

Review Database – Hold a review of the physically implemented database to ensure that the database is implemented as designed in the database specification. Coordinate the review of the database with the operations organization appropriate to the platform of the system and with SBA's Data Administrator to ensure that data standards and guidelines are followed. This review will focus on ensuring the following:

- Records defined in the database specifications exist and all fields are accounted for

- Required index files are present. Primary and secondary keys are present

- Files in the database are sized correctly

- Edit or validation tables have been defined

- Aliases or synonyms are present

- All applicable database and data standards and guidelines have been followed

Refine Database as Required – Modify and reload the database structure to correct any deficiencies found during review of the database. Reload test data and, if necessary, perform a review of the modifications to ensure correctness.

Test Database – Test the physically implemented database and verify it before any live data are added. Testing should be completed before development of application programs that access the database because changes to any part of the database will affect programs using that portion of the database. The test to verify that the design will operate as expected focuses on such items as the following:

- Security of the data

- Database performance

- Considerations for operating the database on the target platform

- File and database sizing

- Data backup and recovery

Create Database Tests – Develop a plan for testing the database before test execution, and identify the tests that must be performed to execute the plan. Develop and document the tests and the data prepared by adding new data to the database or modifying existing data. The test documentation can be in any format within SBA guidelines that is agreed upon by the project team, but it generally will contain the following information:

- Description of the test

- Steps to be taken to execute the test

- Data to be used to execute the test

- Expected results

- Perform a QA walkthrough of the tests before test execution to ensure that all necessary testing will be performed.

Execute Database Tests and Record Results – Once the database tests have been developed and approved, execute them following the steps described in the test documentation. Note any deviations from the original test and record them in the test documentation.

After the test is executed, record the results as part of the test documentation along with any supporting data, reports from the database, or reports generated by the system software during execution of the test.

Evaluate Test Results – Compare and validate the actual results from the execution of the tests to the expected results. Make decisions on the acceptability of the deviations.

Document acceptable deviations as part of the test documentation - For unacceptable deviations requiring modifications to the database, determine how to correct the error or problem.

Correct Database Errors – Modify the database to correct errors. Make modifications to correct performance issues. Additional security levels or data encryption may be imposed on certain data to correct security deficiencies. Resize files or the entire database, if necessary, to correct space deficiencies.

Update Database Documentation – After the database has been tested and any necessary modifications made, update the database specifications document to reflect those changes. If changes are significant, review the changes with the SBA Data Administrator for adherence to SBA data standards.

5.3.3   Develop Computer Programs

During this portion of the Build System phase, the system's computer program design is transformed into machine operating instructions. Along with the development of the programs, the effort to test and make any necessary program modifications, at both the unit and integration levels, is also performed. Record and store all test results in the project library. Document all modifications to program designs necessary as a result of errors found during testing by updating the appropriate Program Specifications or the System/Subsystem Specifications or both.

Prepare Structure Charts – Using the system's Program Specifications as a guide, develop the program structure chart(s), depicting program modules and hierarchical relationships between modules. A module can consist of one or more related software units. If a CASE tool is being used, develop the program structure using appropriate functions of the tool.

Identify Modules – To ease programming and eventually program maintenance, break the program down into manageable portions or modules. Each module should perform one unique function in the program, use all data that is passed to it, and have one entry and one exit point only.

Hierarchically Arrange Modules – Once all the modules that will perform the functions of the program have been identified, determine the data or flags that must be passed between modules and the sequence in which the modules must be executed. Additional modules may be needed to control the sequence in which the lower-level modules will be executed. More than one layer of control modules may be necessary. As modules that control the sequence of execution are added, the hierarchy of the program is refined.

Identify Data Access Strategy – Identify how the program will retrieve data from the database and update it. Make decisions about the following:

- Primary or secondary key(s) to be used

- Index files to be used

- Program input and output formats:

- Loading data into tables or arrays

- Use of leading dollar signs

- Decimal accuracy

Build

Develop Structure Charts – Document the hierarchical arrangement of the program in the form of program structure chart(s). Provide the name and number of each module, and indicate the modules that are called by each module higher in the structure of the program. Use project naming standards and configuration item identifiers.

Cross-Reference with Program Specifications – Provide a means by which the program specifications can be cross-referenced to the structure charts to provide program traceability.

Inspect Structure Charts – After the structure chart is developed and the modules are cross-referenced with the program specifications, perform a QA review of the structure chart to ensure that all logic documented in the program specifications is accounted for in the structure chart of the program under development. Modify the structure chart, as necessary, to resolve any discrepancies found during the review.

Develop Programs – Develop the programs in the programming language(s) selected for the project. Use the database specifications and the program specifications when developing each program. If a CASE tool is being used, use the tool to generate code from the specifications. Follow coding standards in accordance with SBA and project standards.

Code Programs – Write source code instructions in the selected programming language that will perform the logic documented in the program specification. For each program, follow the input and output layouts depicted in the data access portion of the program specification and follow the structure chart while developing the program. Use coding standards appropriate for the programming language. Add comments into the code at appropriate locations to provide clarification of the function and logic being performed. The types of software units and programs include the following:

- Utility Programs. Programs that are not available as part of the programming language's compiler or are not commercially available but that contain common logic that will be used by other programs in the system. Utility programs may include edit or sort routines, control of user access to functions, or error message handling routines.

- Input and Output Modules. Programs that handle the retrieval or output of data for the system. These programs may handle anticipatory retrieval and caching of data from the database for output, deferred storage, or data formatting.

- Application Programs. Programs that perform the actual logic or "mainline" of the system being developed.

- Command Language. Programs that are written in operating system command language or job control language and that are used to execute batch jobs, compile programs, execute programs, and control peripheral devices.

Compile Programs and Correct Errors – Use the control programs developed along with the source programs to submit the source code for compilation. Compile the program source code with error and syntax checking.

Using the source listing report, identify any error messages. Modify the source code to correct the problem. Recompile the program after the changes have been made, and recheck the source listing. Repeat the process until no error messages are generated. *Perform Code Desk Check* – Check the source listing of the program, using a list, checklist, or guide of common errors not typically found by the compiler.

Build

Inspect Code – Perform an inspection of code either one on one between a programmer and an inspector or in a team environment using a moderator and a team as inspectors. The moderator records all problems identified at the inspection and tracks the problems to successful resolution. Perform a one-on-one inspection on programs that are simple and noncritical to the system. Perform a team inspection on programs that are high risk, have security and safety implications, or are central to the successful implementation of the system under development. For team inspections, also inspect, optionally, the program specifications and structure charts with the program source listing. All programs should undergo some type of code inspection.

Unit-Test Programs – After a module has been coded and compiled without error and has undergone an inspection, it is ready to be tested as a standalone entity. This unit testing of software modules and programs uses both valid and invalid data developed specifically for the execution of the test. Unit testing employs dynamic testing techniques that execute the software module with both expected and erroneous data and compares actual with expected results. The goal of unit testing is to exercise all functions of the software module and all logic paths within the module.

Develop Unit Test Procedures – Document a procedure for each unit test to be executed for the program. Each unit test procedure should contain the following information:

- Name of the software module to be tested

- Description and objective of the test

- Any test stubs or drivers to be used in executing the test

- Test data to be used in the test

- Job control language to be executed

- Expected results

- Steps to be taken to execute the test

Document the unit test plan(s) in accordance with SBA SDM documentation standards for Test Plans.

Create Unit Test Data – Employ utilities provided by the DBMS, Structured Query Language (SQL), or other data manipulation tools to develop the data necessary to execute the unit tests by either adding new data to the database or modifying data that already exist in the database.

Execute Unit Tests – Once the test procedures and test data have been prepared, the software unit undergoes the test cycle shown below:

- Run Test. Follow the steps necessary to execute the test as documented in the unit test procedure. If any additional steps are found to be necessary during the test execution, perform the steps and update the test documentation to reflect the modification(s).

- Verify Results and Correct Errors. After the test has been executed, compare the actual tests results with the expected results documented in the unit test plan. For any unacceptable test results that require program modification, make the modification(s) to the program.

- Rerun Test. Repeat the test cycle until the program successfully passes all tests at the unit level, and then promote the program for integration testing.

- Inspect Unit Test. Perform a quality review of the results of unit testing to ensure that all tests were successfully conducted, all logic paths were exercised, and any known problems or errors were adequately documented.

5.3.4 Integrate Database and Programs

Develop tests to ensure that all software unit and program interfaces are present and function correctly. After a program has successfully completed unit testing, add it to a group by interface and test those programs in the group together. As groups of programs test successfully, integrate them with other groups and test all as a whole. Continue grouping and testing until the entire system has been integrated and tested. If any program fails, rework and unit test it; and then repeat each prior integration test. If integration testing requires access to the production environment, coordinate with Operations to minimize the impact and reserve test time.

Define Integration Test – The Integration Test Plan should provide the coordination necessary for developers to define, document, and execute each integration test identified and to develop the test data required to perform the test. The Integration Test Plan should identify the following:

- Each integration test to be executed

- System and program requirements to be tested

- Location of each test in the hierarchy of the testing to be performed

- Software units and programs that are to be included in the test

- Required test data, including database fields

- External (or existing) programs needed to support the test

- Procedures for reporting errors, test results, and reworking and retesting programs

Document the Integration Test Plan in accordance with SBA SDM documentation standards for Test Plans. Perform a quality review of the Integration Test Plan. Ensure the plan identifies the program functions and interfaces that require testing.

Create Test Data – Identify required test data, either converted or input from an existing system. Use utilities provided by the DBMS, SQL, or other data manipulation tools to modify test data, as necessary, to create erroneous data as well as "staged" data to test all program interfaces. In the absence of any converted data, use data manipulation tools to create the test data or "test bed." Coordinate with the appropriate operations organization to allow for the recovery of the original data via planned backups of the test bed before test execution. Use CM version control procedures to identify and maintain test data.

Execute Integration Tests – Once the test procedures are developed as identified in the Integration Test Plan and the test data have been prepared, the program undergoes the test cycle listed below:

- Run Test. Follow the steps necessary to execute each test as documented in the Integration Test Plan. If any additional steps are found necessary during test execution, perform the steps and update the test documentation to reflect the modification(s).

- Verify Results and Correct Errors. After the test has been executed, perform a review to compare the actual test results. Ensure that actual results conform to the expected results documented in the Integration Test Plan. For any unacceptable test results that require program modification, make the necessary modification(s) to the program(s).

- Rerun Test. Repeat the test cycle until the program successfully passes all tests at each level in the integration of the system. Once all of the programs in the system successfully pass all integration level tests, promote the system for system acceptance testing.

Document All Test Results – Project test personnel track the results of all tests that are identified in the Integration Test Plan and are executed during testing. At the completion of testing, record and summarize test results in a Test Analysis Report. Along with the test results, update the requirements matrix to indicate that a test has verified each functional requirement. Add the matrix to the Test Analysis Report. Provide a summary of the system's capabilities, identified deficiencies, and urgency for correcting each deficiency. Integration test personnel provide to project management an assessment of the system's readiness for continuing to the next phase.

5.3.5   Develop Operations and Maintenance Documentation

Develop, review, and approve all procedural documentation necessary for installation, conversion, operation, maintenance, and use of the system. Update existing office procedures or create new ones to accommodate any impact that may occur from operation of the developed system.

Develop Operations Procedures – Develop an Operations Manual to provide computer operations personnel with a description of the software, the environment in which this software will be run, and the procedures to be followed.

Prepare System Overview – Develop an overview of the system that details its intended purpose and outlines how that purpose is met by the system. Include the following in the outline:

- Description of the system operation

- Inventory of all software units

- List of all permanent files and databases referenced

- List of all reports

- Process overview that includes descriptions of system restrictions, any waivers of operational standards, and all interfaces with other systems

- Description of the communications network within the system

- Description of security considerations associated with the system

Describe System Runs – Develop descriptions for all system runs with accompanying run listings and operation schedules, and define the setup and diagnostic procedures for any software diagnostics. Define software units and jobs by run, with a list of all error messages and any corresponding correction procedures. Provide the following information in the run descriptions:

- Purpose of each run

- Runstream job control statements for job initiation

- Run management requirements

- Descriptions of all related files and databases

- Requirements and procedures for report generation and reproduction

- Any restart and recovery procedures

Prepare Operations Manual – Document all operations procedures in the system's Operations Manual, including the system description and system run procedures. Ensure that the manual is well indexed and cross-referenced and includes a glossary of jargon, special terms, and acronyms.

Prepare the Operations Manual in accordance with SBA SDM documentation standards.

Develop Office Procedures – Update or create Desk Procedures Manuals to provide detailed descriptions of the changes to office procedures made necessary by the introduction of the system. Include a full description of requirements related to security, privacy, and internal controls.

Review and Update Office Procedures – Review current office procedures to determine the impact that operation of the new system will have on the office environment. Consider the following when performing this review:

- Manual procedures that have been automated as a result of the new system

- Procedures that have become obsolete

- New procedures, both manual and automated, that are now required because of the operational release of the new system

Based on the results of the review, update or develop office procedures to meet the requirements of the new system.

Create and Update Desk Procedures Manuals – Document all information in desk procedures manuals concerning changed, updated, or newly created office procedures and their related interface with the system. Ensure that each manual is well indexed and cross-referenced and includes a glossary of jargon, special terms, and acronyms.

Prepare all documentation in accordance with SBA documentation standards.

Develop Maintenance Procedures – Develop a Maintenance Manual to provide the maintenance programmer the information and source code necessary to understand the programs, operating environment, maintenance procedures, and security and control requirements.

Prepare System Overview – Develop an overview of the system that details the purpose of the system and outlines how that purpose is met by the design of the system. Include the following in this outline:

- Explanation of the system's purpose and its functions

- Description of system, subsystems, and communications design

- Discussion of security considerations associated with the system

Describe System Environment – Prepare a detailed description of the environment in which the system is to be maintained. Include the following information in this description:

- Configuration of equipment

- Listing of support software identified by version and release number

- Purpose and content of system database

- Database characteristics, such as IDs and storage

- Identification of data structures, elements, and entities

Describe Software and Database Maintenance Procedures – Prepare a detailed description of procedures to be followed by the maintenance programmer. Include the following information in this description:

- Identification of all system software units

- Description of each software unit that includes:

    - Description of each function

    - Illustration of each input format

    - Detailed descriptions of processing

    - Description of each output produced

    - Amount and type of storage required

    - Interfaces

- Description of the database rules, schema, and conventions used

- Description of performance verification procedures

- Description of all error conditions and correction procedures

- References to location of software listings

Prepare Maintenance Manual – Document in a Maintenance Manual all information concerning the maintenance programmer's interface with the system, including system overview and software maintenance procedures. Ensure that the manual is well indexed and cross-referenced and includes a glossary of jargon, special terms, and acronyms.

Prepare the Maintenance Manual in accordance with SBA SDM documentation standards

Develop Preliminary Installation and Conversion Procedures – Produce the preliminary installation and conversion procedures to direct installation or implementation of an automated information system at locations other than the test site after testing has been completed.

Determine System Installation Procedures – Develop procedures that must be followed when the system is installed. Include an overview of the installation procedure, identifying the following information:

- Support materials needed

- Personnel requirements

- Security requirements

- Special training needs, if any

Provide in the overview additional details for pilot site and operations site installations. Include the following information in those details:

- Installation schedule

- Inventory of hardware and software to be installed

- Operation facility requirements

- Identification of installation team

- Procedures to be followed during installation

Determine Data Conversion Procedures – Develop conversion procedures describing the initiation procedure and output formats. Identify sequential steps to perform the conversion and describe all recovery and correction procedures.

Summarize Preliminary Installation and Conversion Strategy – Prepare a preliminary Installation and Conversion Strategy describing the system installation procedures and the data conversion procedures. Present in non-technical terms those portions of the document directed toward staff personnel; present in suitable terminology those parts written for operations personnel.

Prepare the Installation and Conversion Plan in accordance with SBA SDM documentation standards.

5.3.6 *Finalize Test Strategy*

Combine the initial test strategy developed during the Define Stage with all system and testing information developed during the Build Stage to prepare the final VV&T Plan. The plan is reviewed and approved by the appropriate areas within SBA.

Refine Test Strategy – Update the test strategy to include any revisions resulting from Build Stage activities. Ensure that the test database is updated to meet any changes in test requirements or procedures or both.

Revise Test Strategy – Update the description of the test strategy to reflect changes that may have occurred during the Build Stage.

Revise Test Data – Update the test database to reflect changes to the test strategy. Make sure that sufficient data have been developed to ensure the performance of all required tests. Ensure version control procedures are used to manage changes incorporated into test data.

Revise Test Procedures – Update the system acceptance test procedures developed during the Define Stage to reflect any impact on the testing strategy that may have occurred during the Build Stage.

Finalize Validation, Verification, and Test Plan – Perform final updates to the VV&T Plan to reflect any changes in system acceptance testing requirements that may have occurred during the Build Stage of development.

Update VV&T Plan – Revise the VV&T Plan to include changes to the testing schedule and equipment or personnel requirements. Verify that the VV&T Plan accurately reflects the materials needed to perform system acceptance testing.

Update Test Specifications – Revise the VV&T Plan to include any changes to the test specifications. Review the following to ensure that they accurately reflect the current specifications:

- Performance requirements
- Test methods
- Test conditions
- Test data criteria

Prepare the VV&T Plan.

5.3.7 Complete User's Manual

Prepare User's Manual – Document user procedures in the system User's Manual, including the system description, input and expected output procedures, user interface descriptions, file query, and office procedures. Describe this information to allow user organizations to determine the system's applicability and when and how to use it.

5.3.8 Develop Training Documentation

Establish a strategy for training the user community and all related system personnel (e.g., operators and support personnel). If a training guide is to be used to assist instructors in their training tasks, finalize the training guide as well. Tasks include identifying the training methods and tools, preparing a preliminary training schedule, developing an appropriate curriculum, and preparing the training plan.

Determine Training Objectives – Based on a review of the roles of the personnel to be instructed, determine the objectives that must be met for the students to achieve an adequate understanding of the system functions to perform their job functions with minimum guidance. Objectives may vary depending on the skill of the class participants and their levels of involvement with the system.

Identify Training Methods, Techniques, and Tools – Describe the training methods to be used for the project (e.g., whether the SBA staff will conduct the training or whether the services of a vendor will be acquired). Determine the training techniques to be used (e.g., computer-based instruction, self-paced written manual, peer training, hands-on practical sessions, classroom lectures, or any combination of these). Identify the tools needed for the training, such as Personal Computers (PCs), training manual and classroom.

Identify Training Required for Revised Office Procedures – Identify the training needs for the user's staff if the advent of the system under development will change the procedures of the user's office in any way. Because user acceptance of the system is paramount, consider all changes expected to the user's current processing, no matter how small. The project sponsor is the best qualified to identify projected changes to office procedures; however, cooperation from the developer's staff as well as the Agency Computer Security Program Manager (ACSPM) is necessary because they are most knowledgeable about the new system operation and security requirements.

Prepare Preliminary Training Schedule – Prepare a training schedule to include the following information:

- Planned training dates
- Names of students
- Roles/Names of instructors
- Location of the sessions

This schedule should be as comprehensive as possible; however, the schedule may be revised at later points in the project's lifecycle.

Develop Curriculum – Determine the job classifications of the individuals that will need to be trained on the use of the system. Using the system/subsystem specifications and other system development documents, determine the system functions that each job class must be familiar with to enable them to successfully interface with the system. Using this information, determine the different courses and the materials that must be presented in each course.

Prepare Training Plan – Document all information regarding the training strategy and approach in the Training Plan. Include the schedule of training courses, a description of each course, and list of the materials required.

Update Training Plan as Needed

Identify modifications and prepare a list of the changes that need to be made to the training approach to reflect the impact of any changes that may have occurred during the Build Stage of development. Consider the following when identifying modifications that must be made:

- Training schedules
- Student attendance lists
- Training requirements for office procedures
- Training materials

When complete, document modifications by preparing a finalized Training Plan that reflects any modifications made to the training strategy prior to the Implement Stage.

Develop Training Guide (Optional)

Develop a guide to assist in instructing various personnel, e.g., management, operations, programmers, and users, in the use of the system.

Prepare Training Guide (Optional)

Outline the procedures for instructing various types of personnel in the use of the system. Develop a training guide for each personnel group to include lesson plans, objectives to be met, equipment to be used ,e.g., overhead projectors and classrooms, procedures to be followed, e.g., Computer-Based Training (CBT), and a course book for the students to follow during their training modules, if necessary.

5.3.9   Develop Installation, Conversion, and Transition to Support Plan

During the Build Stage, establish the strategies for installing the new system, supporting conversion of required manual and automated files, identifying key roles and responsibilities in developing the system operational environment and its dependencies, and migrating operations from any related systems. This task is completed during the Evaluate Stage.

5.3.10   Update System Audit Strategy

Audit activities occur throughout the later stages of the development process. This description of the Update to the Audit Strategy process is the same for SDM Sections 3-5. When finished, return to the previous material by clicking on a section shown above or

at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Notify the IG staff about any work products completed during the current phase and informed of any changes that may have an impact on internal data processing controls as they were described in the specific phase documentation. Activities include coordinating with IG staff, providing input to IG staff, and updating the Internal Audit Plan.

Coordinate with IG Staff. Schedule meeting(s) with the Inspector General's staff to inform them of any developments that occurred during the specific phase. Determine the scope of the information required from the project manager in order for the IG staff to perform the necessary updates to the Internal Audit Plan.

Provide input to IG staff. Provide the IG staff with the necessary information.

Update Internal Audit Plan. Update the Internal Audit Plan to include information provided by the project manager.

5.3.11 Update System Decision Paper

This description of the Decision Paper Update process is the same for SDM Sections 2-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Update the System Decision Paper to include a summary of the progress made during the specific SDM stage and the schedule of events for SDM stage activities. Include the current status of the project. List in detail any changes to the original plan for the development of the project and the parties responsible for approving those changes. The updated System Decision Paper is then reviewed and approved by the appropriate management officials. Include the following activities:

Summarize Progress of System – Summarize the progress of the project through the specific stage, and update the System Decision Paper to include this summary.

Identify Resources Expended, Needed, and Projected – Identify resources that were expended during the specific stage, and update the projected schedule of resources to be expended through the remaining system life cycle.

Identify Changes – Document any changes to the system development milestones and schedule or any modifications made to the project strategy. Update the System Decision Paper accordingly.

Describe Changes in Milestones, Schedules, and Activities – Provide a description of any SDM stage occurrences that may have had an impact on the projected milestones, schedules, and activities of the development effort. Provide a brief explanation of the nature of these impacts and describe any changes to the milestones or schedule that may occur as a result of these changes.

Describe Modifications to Project Strategy – Provide a description of any modifications to the project strategy that have occurred during the specific SDM stage of development. Provide a brief explanation of these modifications, indicating the reasons for the changes as well as any anticipated impacts.

Summarize Schedule of Events – Develop a brief summary of the events that took place during this stage of development and a description of the remaining schedule of

development activities. Show the impact these events have had on the original schedule and any anticipated impacts on future development.

Summarize Status – Provide a status report of the development of the system. Include in the status report the percentage of the system that has been developed and successfully passed integration testing and its readiness for acceptance testing, as reported by the integration test team in the test analysis report. Indicate the components or subsystem of the system that may be causing delays or problems.

Document Results – Document all information for this phase in the System Decision Paper.

5.3.12   Refine Project Plan

The Project Plan is continually refined throughout the development process as new details become available. This description of the refinement process is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

The Project Manager controls the project by monitoring phase activities, taking corrective action where necessary, and refining the project plan to account for changes attributable to actions taken in the current phase and new information for upcoming phases. The Project Manager also reviews the quality process planned by users, developers, testers, and quality assurance for the next phase.

Update Project Plan – Update the Project Plan with cost, schedule, and budget data for the current phase to the level of detail necessary to reflect the status of the project. Review the plan and determine if any activities described for the next phase and subsequent ones are affected by the completion status of the current phase activities. Adjust the schedules and resource requirements for activities in the next phase, if necessary, and assign starting and ending dates for activities that have been affected. Take into account that the starting date for some activities may be dependent on the completion of other activities. Update milestones, schedules, and resource requirements for the remainder of the project.

Review Planned Quality Process – Project manager, users, developers, testers, and QA work closely together to determine the process that will be used to build the product with the desired quality. Activities include tailoring the planned quality process for the next phase and identifying the standards and procedures to be used. The project manager reviews the quality process for approval and execution in the next phase.

5.3.13   Review and Approve Documents and Deliverables

This description of the Review and Approval of Documents and Deliverables is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Review and approval of documents and deliverables is an iterative process as each stage provides more definition and as products are revised. In addition, the approval process offers senior management the capability to monitor the activities of the project.

Review Revised Documents from Prior Phases, as Required – Review the activities performed during the specific phase to determine if they have an impact on any

documents produced during the previous phases. If changes are required to prior lifecycle phase documents, update those documents to reflect current project developments. The revised documents are reviewed by project personnel similar to reviews for current phase documents to ensure that changes are within the scope of the project's requirements and are in compliance with SBA and project standards and procedures. Prepare a management summary for each revised document that summarizes the essential revisions. Submit the revised products to the appropriate review board for approval. File affected product change records with CM for appropriate version control updates.

Conduct Review for Phase Documents and Deliverables – Conduct a project review with project personnel and system stakeholders to ensure that the project documents and deliverables for the specific phase include the necessary level of detail, fulfill the system's requirements, and meet the appropriate SBA and project standards and guidelines.

A minimum of 10 working days before the scheduled review, notify the personnel required to attend and provide each with a copy of the product for their pre-review. Discuss all comments or objections that are raised during the review, and reach a consensus on one of the following before the review session terminates:

- Document is correct and complete, as is, without any further changes.

- Additional changes that need to be made are minor and do not require further review. In this case, the updates should be made and change pages should be distributed by an agreed-upon date.

- Required changes will have a major impact on the plan. A second review must be scheduled. The changes must be incorporated and the resulting change pages distributed at least 10 working days before the second review.

Prepare a management summary for each document that summarizes the essential data collected in the document, lists conclusions that may be drawn from the document, and describes the potential impacts on the project, if applicable. Submit the management summaries to the appropriate review board for approval. Templates for generating documents for approval by SDM phases are available on the SDM Document Templates page.

Obtain Approval of Project Documents and Deliverables – Present the project documents and deliverables to the chairperson of the appropriate review board for approval at least 10 days before the scheduled decision date. Include the management summary information, approval (sign-off) record, the phase specific document, and any other required or requested information. The review board chairperson coordinates review board comments, recommendations, and approval signature(s), and returns the approval record to the project manager. Recommendations on the approval record are addressed by the project and the document should be resubmitted to the board if requested. Approval will be assumed if there is no response or if the response is "no comment." Project approval records are maintained by the project's configuration management (CM) function. A copy is inserted into the central project library.

The project proceeds to the next phase after all project documents and deliverables are approved.

5.3.14 Perform Change Control Activities

Change Control activities occur throughout the development process as proposed changes to the specification or any baseline document must be reviewed and authorized. This description of the Change Control process is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Any products developed during a specific phase are baselined and subjected to version control. Products baselined during prior activities are assigned new version control numbers when they undergo change (e.g., an update or rewrite). Configuration management reports are provided to the project manager as requested or required.

The CM function manages, implements, and reports on project CM activities. In supporting the project, CM performs the following activities:

Verify changes made to product. Review the updated product to ensure that the changes have been made as described in supporting documentation. Supporting documentation may be comments received from document reviews. Supporting documentation for software may be software change requests or discrepancy reports generated during testing.

Assign version number. The version number must follow conventions established by SBA and enable the project's CM to monitor updates to the product and assist in its distribution. For software programs, increment the version number each time a change is made to the software to correct deficiencies found during testing.

Store approved version in central library. After each baseline product has been completed and approved, according to SBA procedures, store the approved version in the project's central library. The project's CM function controls access to the library. For software, the library can be a subdirectory or dataset where all baselined software configuration items (CIs) will be stored.

Record product information in inventory log. Maintain an inventory log that includes the title of the product, release date, version number, name and version or model numbers of the software and hardware used in the development of the product, name of the organization responsible for development of the product (usually the sponsoring organization), and the product distribution list.

Distribute copies of products as required. The project's CM function distributes copies of the products according to a distribution list maintained as part of the inventory log information, with distribution based on need and the security level of the product.

Archive old versions of products. Archive and retain outdated versions of all products for the required period of time, in keeping with SBA standards and guidelines.

5.4   Performance & Governance Considerations

Particular care should be given to the building and testing of any data structures or processes that will be shared by other systems. Additional testing should be performed on these elements to ensure that they perform in the exact manner specified in the DRM and the SRM. The TRM should be reviewed to ensure that all enterprise policies, procedures and mandates have been observed.

During the Integration and Test Stage problems with interfaces, data structure and functions are frequently discovered that require rework. It is critical to ensure that no changes are made to shared processes or data structures without changes to the reference models that describe them.

5.5 Exit Criteria

The project remains in the Build & Test Stage until the SBA BTIC or management grant authorization to proceed. This will be based on:

- A complete build and test of every software and hardware configuration item
- Test plans for every configuration item function
- Finalized documentation
- Finalized Transition Plan

To exit the Implementation and Test Stage, it is necessary that all major testing be complete to the satisfaction of the business, technical and management stakeholders. In particular, the following tests must complete successfully:

- System Test
- Integration Test
- User Acceptance Test

Table 5-1 Build & Test Stage Document Deliverables

SDM Build & Test Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Project Plan (Updated)					
Detailed Test Plans	X				
Transition Plan	X				
User Manual	X				
Operator Manual	X				
Maintenance Manual	X				
System Operations Manual	X				
Training Manual	X				
Contingency Plan	X				
Detailed Integration Test Plans	X				
Integration Test Results	X				

Build

SDM Build & Test Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
System Test Results	X				X
User Acceptance Test Results	X				
Validation & Verification	X				X
Audit Strategy (Updated)					
Accreditation & Certification	X				X
Systems Decision Paper (Updated)					
Approved Build & Test Stage SDM Decision Document	X	X			

6 Evaluate

This section provides requirements and guidelines for the evaluation of systems against their functional requirements before going operational.

6.1 Purpose

The Evaluate Stage is the period in which independent testers measure the system's ability to perform the functions that are required by the user and ensure an acceptable level of performance. After this phase of development is completed, a clear indication of the system's readiness for operation is evident.

6.2 Entry Criteria

The project must have successfully completed the system, integration and documentation from the Build and Test Stage and have been judged acceptable by project, technical and management review.

6.3 Process

In the Evaluate Stage, the system undergoes a complete and thorough System Acceptance Test conducted by a test team that is designated by the project sponsor and is independent of the solution development organization. Each of the system acceptance test procedures and scenarios, documented in the VV&T Plan, is executed. The results of the system acceptance test, as well as the testing methods employed, are documented for final approval.

A management summary is prepared for each product produced or revised during the Evaluate System phase. This one-page summary includes a summary of the essential data collected in a document product, conclusions that may be drawn from the document, and potential impacts on the project, if applicable.

Figure 6-1 Evaluate Stage Process Steps

6.3.1 Perform System Acceptance Test

Until this point, all testing has used procedures and scenarios generated by the project development team that has emphasized correct operation as specified in the analysis and design documentation. System acceptance testing emphasizes the proper functioning of the system from the user's point of view. System acceptance testing is the process of demonstrating whether the program meets the user's written set of measurable objectives. The primary focus of the test should be on translation errors, which are errors made in the process of transforming objectives and requirements of the system (documented in the Functional Requirements Document and the Data Requirements Document) into design specifications (System/Subsystem, Program and Database) and, finally, into an operational system. The classes of tests listed below provide a general description of the testing that is executed during this phase of testing.

Evaluate

Requirements Validation: This class of testing ensures that all functional requirements are implemented as originally envisioned by the users and that all requirements are accounted for in the system.

Functional Testing: Functional testing stresses the correct operation of the application; the correct addition, modification, and deletion of data from the system; and the correct operation of any audit trails in the system.

Performance/Volume/Stress Testing: Performance testing shows whether the program satisfies its performance or efficiency requirements. Volume testing is intended to demonstrate the system's capability to handle the volume of data specified in the requirement. Stress testing determines the maximum capacity of the system, given user requirements for response time and throughput.

Security Testing: Security testing checks the adequacy of security processes and procedures by trying to violate those countermeasures. Both authorized and unauthorized transactions and processes are attempted.

Ease of Use: Testing the system for ease of use ensures that the system is user friendly, the processes and messages are easy to understand, online documentation is available and useful, the use of function keys is standardized throughout the system, and the presentation of the screens and reports are the same as those approved by the user group during the Define System phase.

Operational Testing: Operational testing tests the procedures for installing and operating the systems software on its related hardware. It also tests the backup and recovery procedures.

Documentation Testing: Documentation testing evaluates documentation for content, clarity, and consistency. Content refers to the relevance and completeness of the documentation and its applicability to the computer system. Consistency refers to the maintenance of standards throughout the documentation, uniform terminology, and consistency with other documents. The instructions from the user documentation are checked for accuracy against actual operation of the system.

Procedure Testing: Procedure testing examines the interface between the programs (system) and any manual systems or human procedures, such as those followed by the system operator, database administrator, or terminal user.

Interface Testing: Interface testing evaluates the system's ability to perform required interfaces with other systems operated by SBA and outside organizations.

During this stage of the system's development, final preparations for performing system acceptance testing are undertaken, including finalizing the VV&T Plan and readying the test environment. The requirements matrix is updated to reflect the specific tests used to verify the requirements. Each test identified in the VV&T Plan is executed, and the results of the test are evaluated against expected results. After testing is completed, the Test Results and Evaluation Report is prepared, and the system is certified for production readiness. Reviews are held to determine the quality and thoroughness of the system acceptance testing as documented in the Test Results and Evaluation Report and the requirements matrix. Approval and concurrence for both documents is received from the appropriate review board, and the project sponsor organization certifies the system before its release into production. Activities include the following:

Prepare for System Acceptance Testing – Undertake final preparations that allow for execution of the VV&T Plan that was finalized and approved during the Build System

phase. Make final modifications to the test procedures, scenarios and test data developed during the Build System phase. Ensure that the objectives and requirements are correctly outlined by the project development team in the Functional Requirements Document and Data Requirements Document and the System/Subsystem, Program, and Database Specification documents. Update the requirements matrix to validate that all functional requirements are verified by one or more tests.

Finalize Test Procedures and Scenarios – Modify test procedures and scenarios to reflect any changes to the system that were made to correct errors found during integration testing. Make any necessary changes to the procedures in the following areas:

- Objectives of each test or test description

- Resources needed to execute the test

- Steps to be taken to execute the test

- Expected test results

Prepare Test Data – The test data or testbed that was created for the unit and integration tests performed during the Build System phase may prove sufficient to serve as the system acceptance testing testbed. Modify the data, if necessary, to allow all test conditions to be executed. Make any necessary final modifications to the test data by using utilities provided by the DBMS, SQL, or other data manipulation tools. Perform the tests as outlined in the test procedure or scenario. Coordinate with the project's Database Administrator (DBA) and with Operations as necessary to initialize the database and provide adequate space for the test data. Make any provisions at this time for restoring test data after test executions.

Finalize Test Environment – Before the start of the test, ensure that all necessary resource requirements have been met and are in place at the test site(s). Ensure that the hardware has been installed and that computer time has been made available for the test team (if the tests are to be run using production hardware).

Execute Tests and Verify Results – The standard test cycle includes the activities described in the following sections. Execute each test as stated in each test procedure or scenario, and document the tests in the approved VV&T Plan. After test execution, verify the results of the test against expected results, and formally record the test results. Repeat the test cycle, if needed, for a given program or portion of the system after errors are corrected by the development staff.

Execute Test – For each test, follow the steps necessary to execute each test procedure (or scenario) as documented in the VV&T Plan. If any additional steps are necessary to execute the test, record the additional steps in the test procedure documentation.

Record and Verify Test Results – Upon completion of each individual test identified in the VV&T Plan, compare the actual output generated by the test against the expected output documented in the test procedures and scenarios. If deviations from the expected results are discovered, review the predetermined results to ensure the results are correctly stated. Include the following information when recording the results of a test:

- Name and version number of the application or document that was tested

- Identification of the input data used in the test (e.g., reel number or file ID)

- Identification of the hardware and operating systems on which the test was run

- Time, date, and location of the test

- Names, work areas, and phone numbers of personnel involved in the test

- Detailed description of the nature of any deviations from predetermined results that were found during the test

- Identification of the output (e.g., reel number or file ID) in which the deviation was found

- Names, work areas, and phone numbers of development area personnel who were informed about the deviations

- Date the developers were informed about the potential problem

- Name and version number of the application or document that was issued to correct the error

- Date the new version was reissued

Record the information each time a test or retest is performed, and log the information in chronological order to serve as an historical document of the test.

If it is determined that the deviation is an error in the system's software, hardware, or documentation, document the error and notify the appropriate development area by using problem reporting procedures established for the project. Include with the documentation a detailed description and supporting documentation when describing the error. Supporting documentation may include images of the test data before and after the test was executed, series of screen images with narratives showing the exact sequence of events that led up to the error, and actual system reports with errors highlighted.

For large-scale development efforts, design a control system to allow one-to-one traceability from the test data to the predetermined result. Group predetermined results in the same order as the input data to allow for easier output review.

Evaluate Results – Compile the outcome of each individual test documented in the VV&T Plan in the Test Results and Evaluation Report along with the evaluation of test methods and test administration procedures. Record the results of each test execution, and compare the test results to the expected results recorded as part of the test documentation. Prepare the Test Results and Evaluation Report, and make a final determination regarding the readiness of the system. Submit the Test Results and Evaluation Report for approval. If the system is approved for production, formally record the decision and certify the system for release.

Compile Test Results – As system acceptance testing continues, maintain information on the execution of each test and the results of all tests in a central repository for the project. Once the execution of all tests and retests is complete, use this information to form the core of the Test Results and Evaluation Report.

Prepare Test Results and Evaluation Report – At the completion of system acceptance testing, prepare a Test Results and Evaluation Report to describe the test procedures and scenarios and the results that were found following their execution. Prepare the Test Results and Evaluation Report following SBA SDM documentation standards. To prepare the report, –

Describe purpose of report: Identify the project, by name and number, for which the Test Results and Evaluation Report was developed. Document all of the purposes of the

report. At a minimum, every Test Results and Evaluation Report will contain the purposes listed below:

To document the results of all system acceptance tests

- To identify deviations from the VV&T Plan

- To assist in assigning responsibility for resolving issues raised as a result of the system acceptance testing

- To report on the system's ability to fulfill its intended objectives

- To provide a basis for estimation of the project completion time

Provide summary of project references: List the high-level requirements the system was supposed to meet upon completion of its development. Provide summary information about the sponsoring organization and those organizations involved in developing, testing, using, and operating the system. Also, provide any additional information, such as terms, definitions, and acronyms that will provide clarification for the reader.

Describe security considerations: Provide a detailed description of the security requirements that have been built into the system and verified during system acceptance testing. Identify and describe security issues or weaknesses that were discovered as a result of testing.

Provide test analysis of each function tested: Provide summary information for each system acceptance test executed. Identify the test, the purpose of the test, the function or capability demonstrated by the test, an analysis of the system's ability to meet the requirements of the test, and any deviations from the original VV&T Plan that occurred during test execution.

Describe system deficiencies: Describe the deficiencies that remain after the system test is completed. Provide traceability to the problem report that was written when the deficiency was detected, and provide the status of the deficiency.

Recommend improvements as required: Provide a detailed description of any recommendation discovered during testing that could improve the system, its performance, or its related procedures. If additional functionality is seen as a potential improvement for the user, although not specified in the FRD, it should be included here. Provide a priority ranking of each recommended improvement relative to all suggested improvements for the system.

Summarize capability of the system to meet requirements: Provide a summary of the state of the system upon completion of system acceptance testing. This summary should support one of the following test recommendations:

1. The system is virtually error free and should be released into production.

2. Errors still exist that should be addressed, but a decision could be made to fix these errors in production and not delay release of the system.

3. The system has major shortcomings and should not be released into production at this time; instead, it should be returned for further development and retesting.

Perform Configuration Audits – The objective of a configuration audit is to assess whether a system, subsystem, or configuration item meets its technical requirements and if any unauthorized changes are scheduled for the delivery. There are two types of configuration audits: the functional configuration audit (FCA), and the physical configuration audit (PCA).

After acceptance testing for a CI release, perform an FCA to determine if the test results demonstrate that the CI meets its allocated requirements. Perform a PCA to determine if the CI's documentation is complete and consistent with the "as-built" CI.

If a configuration audit uncovers any deviations or discrepancies, or results in action items, the audit leader prepares an action item list and audit report. A satisfactory corrective action plan or deviation authorization must be prepared by the project manager and approved by the user and sponsoring organization before the product can be certified and delivered.

Determine Readiness – Certify the system based on its fitness for use in a production environment as determined by review and approval of the Test Results and Evaluation Report and the physical and functional configuration audits. If the developed system is certified for production, release it.

Certify System – Prepare a certification as to the readiness of the system to be released into the production environment. Summarize the current state of the system. Include the name and version numbers of the system software, hardware, and related documentation and guides that are covered under the certification. Certify the system under one of the following categories:

1. The system is virtually error free and should be released into production.

2. Errors still exist that should be addressed, but a decision could be made to fix these errors in production and not delay release of the system.

3. The system has major shortcomings and should not be released into production at this time; instead, it should be returned for further development and retesting.

Issue Release Request – After certification, issue a Release Request to Operations for approval to install the system in the production environment.

6.3.2   Finalize Installation and Conversion Plan

Complete the strategies and procedures for system installation and data conversion. Document in the installation strategy the procedures for installing the system application software, support software, security software and hardware, telecommunications equipment, and peripheral devices at all sites. Indicate in the conversion strategy the procedures for converting existing automated and manual files and for ensuring the correctness of the data after conversion. The appropriate organizations within SBA then review and approve the Installation and Conversion Plan.

Finalize Installation and Conversion Procedures – Complete development of strategies for installing the system hardware and software in the production environment and for converting existing data to the format required by the new system. Update the Installation and Conversion Plan to reflect the final strategy, and submit the plan for review and approval by the appropriate organizations before execution of the Installation and Conversion Plan.

Finalize System Installation and Conversion Procedures – Complete the final updates to the strategies for installing and converting the system to the operational environment begun during the Build System phase of development. Include any additional information discovered during the Evaluate System phase of development that may have an impact on these procedures. This is the final update of the plan before data conversion and hardware and software installation. In the procedures, include information on the following:

- Additional climate control

- Wiring and power conditioning

- Physical security and access controls

- Floor plan

- Required supplies

- Hardware and peripheral equipment

- Also, finalize decisions on the following:

- Organizations involved and how each organization is involved

- Procedures for installing the system for each site and the sequence of site installations

- Sequence and schedule for the complete (or incremental) installation of the system

- Identification of organizations involved with system installation, each organization's role, and methods, e.g., meetings and communications for keeping each organization informed about the status of the installation

- Account for the following in data conversion procedures:

- Converting existing automated files

- Converting existing manual files

- Staffing requirements and organization for conversion

- Checking accuracy of converted data

- Schedule for data conversion

- Organizations involved with data conversion effort and each organization's role in the effort

- Methods (e.g., meetings and communications) for keeping all organizations informed about the status of the conversion

Document Finalized Plan – Complete the Installation and Conversion Plan to include any updates as outlined above. Ensure that the finalized Installation and Conversion Plan has been developed to meet all applicable SBA SDM documentation standards. Also, ensure that the plan complies with operations and IT procedures appropriate for the system's target platform.

6.3.3   Update System Audit Strategy

Audit activities occur throughout the later stages of the development process. This description of the Update to the Audit Strategy process is the same for SDM Sections 3-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Notify the IG staff about any work products completed during the current phase and informed of any changes that may have an impact on internal data processing controls as they were described in the specific phase documentation. Activities include

coordinating with IG staff, providing input to IG staff, and updating the Internal Audit Plan as well as:

Coordinate with IG Staff: Schedule meeting(s) with the Inspector General's staff to inform them of any developments that occurred during the specific phase. Determine the scope of the information required from the project manager in order for the IG staff to perform the necessary updates to the Internal Audit Plan.

Provide input to IG staff: Provide the IG staff with the necessary information.

Update Internal Audit Plan: Update the Internal Audit Plan to include information provided by the project manager.

6.3.4   Refine Project Plan

The Project Plan is continually refined throughout the development process as new details become available. This description of the refinement process is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

The Project Manager controls the project by monitoring phase activities, taking corrective action where necessary, and refining the project plan to account for changes attributable to actions taken in the current phase and new information for upcoming phases. The Project Manager also reviews the quality process planned by users, developers, testers, and quality assurance for the next phase.

Update Project Plan – Update the Project Plan with cost, schedule, and budget data for the current phase to the level of detail necessary to reflect the status of the project. Review the plan and determine if any activities described for the next phase and subsequent ones are affected by the completion status of the current phase activities. Adjust the schedules and resource requirements for activities in the next phase, if necessary, and assign starting and ending dates for activities that have been affected. Take into account that the starting date for some activities may be dependent on the completion of other activities. Update milestones, schedules, and resource requirements for the remainder of the project.

Review Planned Quality Process – Project manager, users, developers, testers, and QA work closely together to determine the process that will be used to build the product with the desired quality. Activities include tailoring the planned quality process for the next phase and identifying the standards and procedures to be used. The project manager reviews the quality process for approval and execution in the next phase.

6.3.5   Review and Approve Documents and Deliverables

This description of the Review and Approval of Documents and Deliverables is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Review and approval of documents and deliverables is an iterative process as each phase provides more definition and as products are revised. In addition, the approval process offers senior management the capability to monitor the activities of the project.

Review Revised Documents from Prior Phases, as Required – Review the activities performed during the specific phase to determine if they have an impact on any

documents produced during the previous phases. If changes are required to prior lifecycle phase documents, update those documents to reflect current project developments. The revised documents are reviewed by project personnel similar to reviews for current phase documents to ensure that changes are within the scope of the project's requirements and are in compliance with SBA and project standards and procedures. Prepare a management summary for each revised document that summarizes the essential revisions. Submit the revised products to the appropriate review board for approval. File affected product change records with CM for appropriate version control updates.

Conduct Review for Phase Documents and Deliverables – Conduct a project review with project personnel and system stakeholders to ensure that the project documents and deliverables for the specific phase include the necessary level of detail, fulfill the system's requirements, and meet the appropriate SBA and project standards and guidelines.

A minimum of 10 working days before the scheduled review, notify the personnel required to attend and provide each with a copy of the product for their pre-review. Discuss all comments or objections that are raised during the review, and reach a consensus on one of the following before the review session terminates:

1. Document is correct and complete, as is, without any further changes.

2. Additional changes that need to be made are minor and do not require further review. In this case, the updates should be made and change pages should be distributed by an agreed-upon date.

3. Required changes will have a major impact on the plan. A second review must be scheduled. The changes must be incorporated and the resulting change pages distributed at least 10 working days before the second review.

Prepare a management summary for each document that summarizes the essential data collected in the document, lists conclusions that may be drawn from the document, and describes the potential impacts on the project, if applicable. Submit the management summaries to the appropriate review board for approval. Templates for generating documents for approval by SDM phases are available on the SDM Document Templates page.

Obtain Approval of Project Documents and Deliverables – Present the project documents and deliverables to the chairperson of the appropriate review board for approval at least 10 days before the scheduled decision date. Include the management summary information, approval (sign-off) record, the phase specific document, and any other required or requested information. The review board chairperson coordinates review board comments, recommendations, and approval signature(s), and returns the approval record to the project manager. Recommendations on the approval record are addressed by the project and the document should be resubmitted to the board if requested. Approval will be assumed if there is no response or if the response is "no comment." Project approval records are maintained by the project's configuration management (CM) function. A copy is inserted into the central project library.

The project proceeds to the next phase after all project documents and deliverables are approved.

6.3.6 *Perform Change Control Activities*

Change Control activities occur throughout the development process as proposed changes to the specification or any baseline document must be reviewed and authorized. This description of the Change Control process is the same for SDM Sections 1-5. When finished, return to the previous material by clicking on a section shown above or at the bottom of this page. In addition, you can use your browser 'Back' button to return to the previous page you visited.

Any products developed during a specific phase are baselined and subjected to version control. Products baselined during prior activities are assigned new version control numbers when they undergo change (e.g., an update or rewrite). Configuration management reports are provided to the project manager as requested or required.

The CM function manages, implements, and reports on project CM activities. In supporting the project, CM performs the following activities:

Verify changes made to product: Review the updated product to ensure that the changes have been made as described in supporting documentation. Supporting documentation may be comments received from document reviews. Supporting documentation for software may be software change requests or discrepancy reports generated during testing.

Assign version number: The version number must follow conventions established by SBA and enable the project's CM to monitor updates to the product and assist in its distribution. For software programs, increment the version number each time a change is made to the software to correct deficiencies found during testing.

Store approved version in central library: After each baseline product has been completed and approved, according to SBA procedures, store the approved version in the project's central library. The project's CM function controls access to the library. For software, the library can be a subdirectory or dataset where all baselined software configuration items (CIs) will be stored.

Record product information in inventory log: Maintain an inventory log that includes the title of the product, release date, version number, name and version or model numbers of the software and hardware used in the development of the product, name of the organization responsible for development of the product (usually the sponsoring organization), and the product distribution list.

Distribute copies of products as required: The project's CM function distributes copies of the products according to a distribution list maintained as part of the inventory log information, with distribution based on need and the security level of the product.

Archive old versions of products: Archive and retain outdated versions of all products for the required period of time, in keeping with SBA standards and guidelines.

6.4 *Performance & Governance Considerations*

Once the system is successfully operating, the Performance Reference Model should be revisited to evaluate the actual "line of sight" achieved; that is, how well the fielded system has (or has not) met the business goals assigned to it. The results of this examination should be made available for future projects.

6.5   Exit Criteria

The Implementation Stage completes when the system has been successfully made operational and demonstrates the satisfaction of all business and technical requirements.

Table 6-1 Implement Stage Document Deliverables

SDM Implement Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Project Plan (Updated	X				
Maintenance Plan (updated)	X				
Operator Manual (updated)	X				
User Manual (updated)	X				
Audit Strategy (Updated)	X				
Detailed Installation, Conversion and Transition Plan	X	X	X		X
System Introduction Presentation	X				
Training Manual (updated)	X				
Contingency & Disaster Recovery Plan	X				
System Decision Paper (Updated)	X				
Approved Implement Stage SDM Decision Document	X	X			

7　Operate & Maintain

The Operate and Maintain Stage section provides requirements and guidelines for system implementation, operation and maintenance activities and documentation, including system installation and operation and reporting of system deficiencies and enhancements.

7.1　Purpose

The purpose of the Operate and Maintain Stage is to put into production the certified system and to perform the activities required to keep a system operational and responsive to users' changing needs after the system is accepted and placed into production. The system is initially installed at a pilot site (optional) and/or eventually released, or into its full-scale production environment. All necessary training for using the system is accomplished, and the system's performance is monitored in a production environment. Figure 6-1 highlights the process flow for the Operate and Maintain Stage.

7.2　Entry Criteria

The system has completed evaluation and is successfully performing in the development environment.

7.3　Process

Operate System is the final phase of the development lifecycle, when the system is released, initially to a pilot site and then into the full-scale production environment. The piloting of the system at a specified pilot site ensures that the system will perform all of its functions to meet full-scale operational requirements and serves as the vehicle for verifying that the system is acceptable to the users. The minimum time for piloting a system should be long enough to complete one full processing cycle.

Figure 7-1 Operate & Maintain Stage Process Steps

During this Stage, supporting organizations perform the following activities:

- Install the system at the pilot site.

- Upon completion of the pilot tests, release the system into operation at all sites originally specified in the project plan.

- Perform installation and conversion procedures at both the pilot and operation sites in accordance with the Installation and Conversion Plan.

- Hold training classes for user and operations personnel in accordance with the Training Plan. Complete the training modules, acquire the necessary training resources, schedule the training sessions, and notify the system users about the schedule.

- Perform project peer reviews of deliverables listed in Table 6-1.

- Upon request, present the System Decision Paper and all required supporting documentation to the appropriate review boards for approval.

7.3.1 Install System at Pilot Site

Prepare the pilot site and install and/or convert the hardware, software, and databases by following the Installation and Conversion Plan and the SBA ADP Standard Release Procedures. Closely monitor the system during operation at the pilot site and document the results. Based on the documented results, project managers will make recommendations to the user and project sponsor organizations about the system's operational readiness in the production environment.

Ensure Pilot Environment Is Correctly Established – Review all equipment and site facilities to ensure that the pilot operation environment is ready for the newly developed hardware and software to be installed. Notify affected personnel and organizations about the upcoming installation and conversion, and schedule meetings to ensure that all affected personnel are aware of any procedural changes.

Ensure That Physical Facilities Are Ready. At the pilot site, perform a review of the equipment and the physical environment where the equipment will be located to ensure that all safety regulations have been followed and that the installation and conversion procedures were carried out in accordance with the Installation and Conversion Plan.

Ensure That Affected Organizations Are Notified. Contact the organizations that will be affected by the pilot of the new system to ensure they are aware of the cutover date for the pilot program. Make sure that these organizations are aware of their responsibilities and of any revised operating procedures required during pilot activities, that they have completed the necessary preparations, and that they are ready for the pilot to begin.

Execute Installation and Conversion Plan – Install the hardware and software for the new system at the pilot site in accordance with the Installation and Conversion Plan. Convert or install the necessary data and databases and initiate system performance monitoring functions.

Execute System Installation in Accordance with Plan: Install the system at the physical location where the pilot test is to be performed. Ensure that the installation is carried out in accordance with the procedures described in the Installation and Conversion Plan. Verify that only the required hardware and software are installed in the pilot test environment. Conduct initial test checkout to verify correct integration of the installed components.

Execute Data Conversion in Accordance with Plan: Convert the data to be used in the pilot system to the media and format required by the new system. Ensure that data conversion is carried out in accordance with the procedures described in the Installation and Conversion Plan. Verify, by inspection or test, that data conversion has been implemented correctly. If problems are encountered during data conversion, follow the problem resolution procedures defined in the Installation and Conversion Plan.

Control Pilot Environment: Ensure that the integrity of the system is preserved during installation. Review the list of equipment to ensure that all components necessary for the pilot site are installed. Prepare software installation package from the configuration controlled software library. Maintain all records associated with the installation (e.g., problem reports, audit reports, and installation reports).

Pilot Readiness Review: When installation is complete, notify the project sponsor and user organizations that piloting activities are ready to begin. Be prepared to discuss the steps of the installation process, the equipment and software installed, and tests to be

performed to show the users that the system is ready to be used in the pilot environment.

Conduct Pilot Site Training – Schedule and conduct training classes for all personnel affected at the pilot site. Perform pilot site training in accordance with the Training Plan. Monitor training activities to determine if the selected training techniques are achieving the desired results.

Analyze feedback received from personnel attending the pilot training sessions; and, based on this analysis, refine training procedures or materials to ensure that training objectives are met. Changes that occur during pilot system activities may require changing the Training Plan and related courses and materials.

Operate System in Pilot Environment – Operate the newly installed system in the pilot environment designed to mirror all aspects of the production environment. Closely monitor all aspects of system performance for compliance with relevant system documentation, and report and correct any deviations before a full-scale production release. Pilot operation should be run parallel to the existing system(s) so that the data output from the two systems can be compared to ensure that the new system is operating correctly.

Perform Day-to-Day Operation of Pilot System: During day-to-day operations at the pilot site, follow the procedures described in the User's and Operations Manual. Ensure the following:

- The system performs the transactions or functions for which it is designed.

- System performance meets or exceeds required capabilities.

- Necessary computer time has been scheduled for the systems.

- Resources necessary for the system to run correctly are available (e.g., disk space, tapes, and computer paper).

- Personnel are available to run the pilot system in parallel with the existing systems.

Maintain System Logs: Ensure that operations personnel maintain proper system logs that meet the requirements of operations. These system logs identify any problems found or changes implemented during pilot activities. The system logs are to be used to verify that all problems have been identified, corrected, or deferred to a future system release. Use existing SBA automated tools for maintaining system logs.

Maintain and Follow System Documentation: Ensure that all changes made to the system in the pilot environment are updated in the system documentation. Use configuration version control procedures for all system documentation throughout the life of the system, including pilot operation.

Resolve Problems Found While Operating Pilot System: If error messages are received during the pilot operation of the system, document them as required, and follow the response procedures as outlined in the User's Manual or Operations Manual. If the response procedures do not have the desired effect or if the error message is not listed in either guide, contact the system administrator for support in analyzing the cause of the message.

Report all errors detected during the pilot operation of the system to the project's development staff by using the agreed-upon discrepancy reporting procedures. Report problems determined to have an impact on system security to the ADP security staff.

7.3.2   Conduct Training

Finalize training materials, and schedule training classes for all affected personnel. Conduct all training sessions in accordance with the Training Plan, and monitor training activities to determine if the selected training techniques are achieving the desired results.

Finalize Training Materials – Complete all materials necessary to support the scheduled training, including training booklets, online exercises, and presentation materials. Review these materials to ensure compliance with the project Training Plan. Modify training materials to reflect any corrections or changes to the system that may have resulted from deficiencies found during systems acceptance and pilot tests.

Finalize Training Schedule – Complete scheduling of training sessions for all required personnel. Develop the training rosters and attendance sheets for each session. Notify all personnel to be trained about the time and location of their respective sessions. Ensure that the training facilities and equipment have been reserved in advance.

Conduct Training Sessions – Carry out the approved training sessions in accordance with the project Training Plan. At the completion of each training session, elicit feedback from personnel to ensure training objectives were met.

Evaluate Effectiveness of Training – Analyze all feedback received from personnel attending training sessions; and, based on this analysis, make recommendations for changes to training procedures or materials to ensure that training objectives are met, as described in the project Training Plan.

Modify Training Materials as Necessary – Respond to recommendations made as a result of analyzing feedback. Update or change training materials and course procedures to ensure compliance with all approved recommendations received. Changes to the system itself also may require updating the Training Plan and related courses and materials

Document Results and Make Recommendations: Document the results of pilot system operation in a Pilot Test Report. Use the system log to record any errors detected during the pilot operation and all feedback received from monitoring the system. Closely review the results and the actions taken to resolve errors, and develop and document any recommendations for improving system performance. Submit these recommendations to the appropriate management areas for consideration.

Document System Problems: Document problems that surface as a result of system diagnostic checks, system performance tests, and user or operations personnel complaints, or detected errors. At a minimum, this documentation should include the following:

- List of any error messages that may have been received either at the terminal or at the operator's console

- Name and number of the application that was in use at the time of the problem

- List of the data being input to the application when the problem occurred

- Time and date of the occurrence

- Detailed description of the nature of the problem

- Function being performed and segment of the system in use at the time of the error

Document Results of System Operation: Keep a continual record of the results of the system's performance in its pilot operational environment. At a minimum, include the following in this documentation:

Time and date that the system was in use

- Types of applications that were run on the system, if applicable

- Number of users during various predefined times during the day or night

- Problems that may have been encountered

- Detailed description of any measurable degradation in response time

This documentation can usually be created by automated techniques, such as using the operating system's ability to dump information concerning CPU time, disk input and output, number of lines printed, elapsed time, and number of jobs or steps processed.

Approve System: System approval is the ultimate objective of the system development project. At the completion of the pilot operation, review pilot test results with the project sponsor and user organizations. Highlight project requirements and functions performed in the pilot environment, results of pilot operation, and any outstanding system problems. Upon successful review of the test results, the user should formally sign off for acceptance of the system before the system is released into production.

Establish Production Baseline: Upon system approval, the production environment is baselined. Maintain strict version and change control over this baseline. Conduct periodic audits to verify that only approved requirements and changes are incorporated into the baseline. Use automated CM tools for processing approved changes and generating change status reports on demand.

7.3.3   Install System at Production Site

Perform inspections of all designated production sites. Detect and correct deficiencies in the production environment before installation of the new system. Perform all system installation and conversion procedures in compliance with the Installation and Conversion Plan.

Ensure Production Environment Is Correctly Established – Perform facility inspections of all areas designated as production sites for the new system. Document and correct any deficiencies detected during these inspections before the installation and conversion of the system at the production sites. Notify all personnel affected by the new system about the upcoming installation and conversion, and perform a review to ensure that all necessary staffing and training are complete.

Ensure That Physical Facilities Are Ready: Perform a review of the equipment and the physical environment where the equipment will be located to ensure that all safety regulations have been followed and that the installation and conversion procedures have been carried out in accordance with the Installation and Conversion Plan.

Ensure That Affected Organizations Are Notified: Contact the organizations that will be affected by operation of the new system to ensure they are aware of the cutover date for the production release. Make sure that they are aware of their responsibilities and any revised operating procedures affected by the new system, that they have completed the necessary preparations, and that they are ready for the system to become operational.

Execute Installation and Conversion Plan – Complete installation of system, hardware, and software. Complete all conversions of data and databases. Closely monitor installation and conversion procedures to ensure compliance with the project's Installation and Conversion Plan and the SBA ADP Standard Release Procedures.

Execute System Installation in Accordance with Plan: At each site where it is to be operated, install the system in production mode and establish user password and security authorizations when required. Ensure that the installation is carried out in accordance with the procedures described in the Installation and Conversion Plan. Include installation procedures for all system components, including hardware, software, networks, and workstations. At the completion of the system installation at each site, obtain a final signoff to signify that the system has been thoroughly checked.

Execute Data Conversion in Accordance with Plan: Convert the data to be used by the system, when it is operational, to a media and format acceptable to the new system. This conversion should be performed following the procedures described in the Installation and Conversion Plan, including the following:

Placing data files and their contents on the system

- Checking converted data with source data to ensure correctness

- Determining source data retention requirements should any problems arise during operation of the system

7.3.4 Run System in Production Environment

Schedule production runs for the new system. Perform all daily activities necessary to operate the system, including monitoring the system's performance to ensure adequate response time, system security, and problem-free operation. Take actions to correct any problems uncovered during system monitoring.

Schedule Production Runs – Schedule and execute the new system's runs in a production environment, and carry out all production personnel responsibilities in accordance with the requirements described in the Operations Manual. Ensure that records produced by the system are retained in the project library.

Monitor System Performance – Monitor the daily operation of the system to ensure the following:

- Capacity of the platform is not exceeded.

- System is operating with sufficient response time.

- Sufficient transaction throughput occurs.

- Downtime is monitored (mean time between failures [MTBF] and mean time to failure [MTTF]).

- System is operating efficiently.

Ensure System Is Responsive to Users – Perform periodic reviews to assess responsiveness of the system. Address all facets of the system, including response time and throughput, user satisfaction with functional and data requirements, technical performance, and system management. Project managers must document recommendations to ensure that the system continues to respond adequately to users' needs.

Conduct Review of System: Schedule and conduct periodic surveys of the system's users to determine their level of satisfaction with the current operation.

Identify Any New Requirements: Describe any new requirements that may be imposed on the system. Ensure that the description is of sufficient detail to enable maintenance programmers to make the necessary modifications.

Formulate Recommendations: Provide recommendations for updating the system to meet the new requirements or improve performance. Ensure that recommendations for change include changes to system documentation and to software and hardware.

Document Findings: Describe the recommended changes by providing as much detail as is available for developers to act on the recommendations. Document these recommendations and adhere to the applicable SBA standards and guidelines.

Ensure System Is Consistently Available – While the system is operational, ensure that all functions of the system are available to all users, as required, and that the users are interfacing with an error-free system. Measure availability based on the percentage of total time that the system is available to users.

Schedule for Maximum Use of System Resources: Ensure that operations personnel are aware of the user's requirements and that the operations schedule allows the user access to the system when needed. Schedule all activities that may result in an interruption of service to the user (e.g., system backup and updates) during the hours that will have the least impact on the user's needs.

Reduce Downtime for Performing Maintenance: Although periodic maintenance is required to keep the system responsive to the user's needs, perform such maintenance in a manner that will reduce the amount of time the system will be down. This may mean consolidating system updates so that they can be performed at one time, thus reducing the number of times the system needs to be brought down, or performing maintenance during off hours when the user has no need for the system.

Report Discrepancies for Problem Resolution – If system monitoring reveals areas of performance that need improvement, take the necessary steps to restore system performance to acceptable levels. Use the Problem Tracking and Reporting System (PTARS), which provides IT a centrally located repository for logging computer-related requests and problems and provides a facility to track completion of problem resolution. Describe the problem, on a PTAR form, in as much detail as possible based on information received during system monitoring. Provide as much information as possible to assist in resolving the problem.

Determine Potential System Modifications and Enhancements – Periodically assess the system to determine potential modifications and enhancements. Use findings and recommendations presented to management from the post-implementation review as input for the assessment. During these assessments, target the following areas:

- Parts of the system that did not meet user expectations or instances in which the implementation of a requirement is inefficient

- Parts of the system that do not function according to specifications

- Potential scheduling or process improvements

- Functions requested by the users as their familiarity with the system increases

Upgrade System as Required – Release upgrades of the system software and hardware into the production environment, when required, to ensure that the system remains responsive to the user's needs. Such upgrades may incorporate improved or additional functions or data, updated system hardware, increased capacity, or improved system performance.

Perform Software Releases and Obtain Release Approval: Perform software releases in accordance with SBA standards and guidelines. Forward all release requests to SBA operations for approval before software is released into production. SBA operations will verify completion of appropriate testing, appropriate user notification, and necessary documentation to support the release. SBA Operations then verifies that all necessary preparations for releasing the software and supporting the software in production have been satisfactorily completed and that the release schedule does not conflict with any other data center activities or processing.

Upgrade Hardware in Accordance With Operations Procedures: Perform periodic monitoring to determine if an upgrade or replacement of hardware is required. If a determination is made to replace or upgrade hardware, develop a plan detailing the type of upgrade, the impact on the system, and the date on which the upgrade is required. Before the actual upgrade, Operations must approve the upgrade plan. Perform all hardware upgrades or replacements in accordance with Operations procedures.

7.3.5   Maintain System

System maintenance is a shared responsibility between the OCIO (or the Programs IT organization) and the user community. During the Operate and Maintain Stage, performance and accuracy of the production system are monitored by Operations staff and the users. Any changes due to new needs or discrepancies are recorded through the appropriate change request reporting mechanism, e.g., Needs Statement [NS], Advanced Requirements Notice [ARN], or Problem Tracking and Reporting System [PTARS]. Needs Statements are forwarded by the user community to the BTIC for disposition; ARNs and PTARs (problem tracking reports) are reported to the appropriate IT organization for disposition.

When the IT organization and the user community determine that changes are needed, approval for a maintenance release is triggered by BTIC direction or by sponsor approval of an Evaluation of Request (EOR). These changes consist of corrections, insertions, deletions, extensions, and enhancements to the system hardware and software. Generally, these changes are made to keep the system functioning in an evolving, expanding user and operational environment.

System maintenance involves many of the same activities associated with system development, but it also has unique characteristics of its own. Maintenance activities are performed within the context of an existing system framework. The maintenance staff implements changes within the existing system architecture. The older the system, the more challenging and time-consuming the maintenance effort becomes, especially when minimal or no documentation exists for the system.

Maintenance activities usually are performed within shorter time frames than development efforts. A development effort may span one or more years, while a maintenance effort may be required within hours or in releases requiring from one to six months or longer to complete. Software maintenance efforts in the SBA environment, and their implementation approaches, are associated with one of the following three sizes:

Operate & Maintain

1. Small maintenance efforts (generally less than a staff month)
2. Medium maintenance efforts (generally between one and six staff months)
3. Large maintenance efforts (greater than six months)

Maintenance activities for a production system are performed by a development team of users and maintainers working together. To preserve the integrity of the production system, the team uses a separate maintenance environment so that modifications can be made and tested without affecting the production environment.

SBA uses three maintenance types to categorize and track changes to SBA applications, described below:

- Corrective maintenance encompasses modifications that fix application problems caused by design, logic, or coding errors. This type of maintenance is usually triggered by an explicit PTAR and usually involves errors that must be investigated immediately. Examples of corrective maintenance are the following:
  - Calculations that generate incorrect totals
  - Data screens that omit a required entry or store an entry in the improper location
  - Aborted programs

- Adaptive maintenance generally involves modifications that are mandated in response to laws or regulations that govern SBA programs; to hardware or software changes, and/or to upgrades within SBA's technical architecture. This type of maintenance also may involve activities aimed at adding or modifying functionalities in response to new or changed requirements.

- Adaptive maintenance usually includes those activities that have been thought out and planned for future implementation. This type is usually a modification required because of software changes or a modification required because of anticipated hardware changes. Examples of adaptive maintenance are:
  - Year 2000 activities required for most of the IT software
  - Changes in software that were promulgated by new laws or regulations
  - System software upgrades such as database management systems, CASE tools, and new communications protocols

- Perfective Maintenance generally involves modifications that satisfy a need to improve the application in some area that is a nonfunctional requirement or to resolve a problem before the user community is affected. These activities could also be aimed at improving the maintainability of the application software. Perfective maintenance changes could be thought of as those that are "nice to have." Examples of perfective maintenance are:

Optimization of execution time, memory size, and file management
  - Standardization of naming conventions and development methodology

Functions and Products – Functionally, system maintenance activities mirror all or part of the development phases. However, activities are often scaled down or done in combination with each other. The activities essentially consist of successive iterations of these combined phases of the development lifecycle. All products that are produced must pass appropriate CM and QA procedures before the affected software is released into production.

Consistency of documentation throughout the project supports objective, valid, and actionable observations

7.3.6   *Perform Maintenance Implementation*

Normally, system maintenance efforts are divided into the following three sizes:

Small maintenance efforts: A small maintenance effort consists of quick reaction assignments (PTARs) that can be implemented and installed with one staff month or less of effort. The assignments can address an error (corrective), a cosmetic change (perfective), or a mandated enhancement (adaptive). During a small maintenance effort, the programmer concentrates on addressing the problem and is not concerned with updating the documentation, unless the user needs updated documentation to use the software.

Small maintenance efforts require flexibility, and the project manager must be aware of this need during planning and be responsive to it. For example, management must decide on a quick-fix maintenance effort (which may require only one day to implement) and how to combine phases for effective implementation. One approach is to have the programmer combine the Initiate, Define, Design, and Build phases into one phase. After the combined phases are complete and the software changes are tested, validated, and delivered to the user, documentation changes could be made and approved. In this case, there is no iteration of phases. If user documentation did not change, the PTAR analysis becomes the documentation of the system change.

Medium maintenance efforts: A medium-sized maintenance effort is similar to the small-sized effort, but it is generally more complex and requires one to six staff months of effort to implement and install. A medium-sized maintenance effort consists of assignments (PTARs, ARNs) that address an error (corrective), a cosmetic change (perfective), or a mandated enhancement (adaptive). During a medium-sized maintenance effort, the programmer concentrates on addressing the problem and is more concerned about updating the documentation than on a small-sized effort.

A medium-sized maintenance effort generally follows a more disciplined approach because more planning and documentation are normally required. To ensure delivery with the software modifications, produce, review, and approve user documentation of the modifications before the maintenance product is put into operation.

Large maintenance efforts: A large-sized maintenance effort requires an implementation and installation effort of more than six staff months to address one or more errors (corrective); cosmetic changes (perfective), or mandated enhancements (adaptive).

A large-sized maintenance effort follows a disciplined approach because planning and documentation are required. It is recommended that the project manager and the development team follow the SDM development lifecycle phases, without combining them, for this type of maintenance.

Change Request Initiation – Initiation of the maintenance process is the receipt of one or more software change requests (Needs Statement, ARN, or PTAR) for a system. In general, the requests should contain the problem or need for change, type of maintenance requested (corrective, perfective, or adaptive), and priority level. Assign a project development team to analyze requests. If a change request is a problem identified by a PTAR, determine if the problem is due to a misunderstanding of procedures or an actual processing malfunction. Also, determine if the problem is

already being resolved in response to another PTAR. Provide a needs assessment, and generate a preliminary feasibility study for change requests.

Concurrently, the project manager defines a preliminary schedule, resource profile, WBS, and budget and documents the preliminary information in a maintenance plan.

Change Request Identification, Solution, and Impact Analysis – This step localizes the change request to the precise parts of the affected software. Develop various change alternatives to satisfy the change request. Perform an impact analysis to evaluate the consequences to the system of each alternative, and select a solution. The project manager updates the maintenance plan to reflect the process activities and schedule, based on the scale of the effort to implement the solution.

Change Request Implementation – The development team performs the unit modifications (design and code), generates unit and integration test plans, and performs unit and integration testing. Based on the scale of the effort, the development team updates the documentation specified in the maintenance plan. The acceptance test team updates the VV&T Plan with the new test procedures and expected results. QA monitors the implementation process for adherence to processes and standards.

Regression Testing – The development team tests the new or modified components for compatibility with the existing system. Use the VV&T Plan that was generated during the development of the system. Choose the tests from the VV&T Plan that validate the functionality being modified by, or being affected by, the maintenance change. Use a test environment that duplicates the operational environment. CM configures the test environment.

Validation and Verification – This step is an independent verification and validation process that ensures the quality of the modified components. The acceptance test team, independent of the project development team, executes the new test procedures that validate the modifications and verifies that all of the change request requirements have been met. Additionally, the acceptance test team performs independent regression testing to verify that the maintenance change has not affected existing functionality. Use the original VV&T Plan to choose a limited set of regression tests. Document and forward any discrepancies to the maintainers for resolution. QA monitors test results and test plans. Make modifications to the User's Manual, Operations Manual, and Installation and Conversion Plan, as necessary. Check all modified documentation for conformity to requested changes.

Solution Acceptance – The requester of the change performs solution acceptance. The requester checks the test results to verify that the solution fulfills and resolves the requested change. QA verifies that only authorized work was completed.

Solution Installation – This is the step during which the modified software is placed into a staging area or is installed in the production environment. A production control unit normally replaces the old version with the new one.

Statistics for Trend Analysis – Collect and analyze statistics for maintenance ARNs and PTARs. Focus the analysis on why a change was made, the type of change made (corrective, perfective, adaptive), and which areas were affected by the change. Reasons why a change was made could be a change in SBA regulations, hardware upgrade, support software upgrade, or new requirement on the system. Areas affected by a change could include the program, hardware, environment, database, data reports, documentation, and users. The data can be used to determine the impact on resources required to enhance the SBA system.

7.3.7   Perform Post-Implementation Review

After the system is installed and operational and initial problems or "bugs" have been resolved, conduct a post-implementation review. The level of effort and depth of this review can vary. For example, the data collection and analysis activity can be modified in relation to the size and scope of the maintenance effort.

Determine Time for Evaluation – Do not hold this review immediately after system implementation; postpone it until adequate and accurate data can be gathered regarding system qualities and characteristics. Schedule the review after all users are thoroughly familiar with the automated system and all of the software problems have been resolved. This review should take place approximately 90 days after implementation and again in six months.

Conduct Evaluation – Conduct the post-implementation evaluation using interviews of operations personnel and user representatives. Review project development and operation documentation. Determine the following:

- Does the system meet the stated needs of the user? Are those needs still applicable? How much longer will the system meet those needs?

- Were the final project staffing and cost estimates within the original estimates?

- Have requirements changed, making the hardware, system software, and application software obsolete?

- Have revisions to the system been made only as necessary, and are they being accomplished effectively and efficiently?

- Was the project well controlled and documented?

- Is the system responsive to users? Do the system functions meet user expectations? Are the system interfaces, screens, and documents user friendly?

- Is the data in the system accurate and nonredundant?

- Are the controls (e.g., access, data validation, run-to-run) in the operational system adequate?

- Can benefits be identified in specific categories (e.g., cost avoidance, cost reduction, opportunity yield, direct revenue yield, intangibles)?

- Are cost savings being met?

- Have regulations or laws changed? Will these changes impact the volume of work?

- Because of automation, have staff duties and procedures changed? Will these changes impact the benefit/cost ratio?

- Does the system receive adequate support from operations personnel?

Formulate Recommendations – The objective of the review is to determine if the system has achieved or surpassed the system's goals. If it is determined during the review that the desired return on investment has not been achieved, develop recommendations for achieving the desired return.

Document Evaluation and Recommendations – Document the review findings to reflect the pre- and post-automation results. Develop documentation as a standardized

means of capturing data and providing a comparison of anticipated and actual benefits derived from the automation effort.

Present Evaluation for Management Review – Forward copies of the review findings and recommendations to the management officials responsible for acquisition, development, and implementation of the system. While analyzing these results and recommendations, management should ask the following questions:

- Has workflow improved since the introduction of automation? For example, have unproductive or repetitive tasks been eliminated?

- Has the paperwork improved in such areas as appearance, format, usefulness, accuracy, and completeness?

- If processing time has been and continues to be tracked, has this time decreased?

- Is the new system achieving the estimated goals planned before the new system was implemented?

- Has feedback from users and clients been evaluated and incorporated to improve the effectiveness of the system?

7.3.8 Perform Change Control Activities

Perform change control activities whenever any products being baselined under the project's CM function are revised. Change control activities include verifying changes made to the products, assigning a new version number to the revision, updating the logbook with the change information, updating the central library with the new version, distributing copies of the new version, and archiving the old version.

Verify Changes Made to Product – Review the updated product to ensure that changes have been made as described in the supporting documentation. The supporting documentation may be comments received from document reviews. The supporting documentation for software may be software change requests or discrepancy reports generated during testing.

Assign New Version Number – Assign a new version number to each updated technical deliverable (hardware or software configuration item) and document deliverable. The version number must follow conventions established by the project's CM function, within SBA guidelines, and enable monitoring of the distribution of official copies and of the removal of outdated versions from the workplace.

Store Approved Product in Central Library – After updating a baselined product, store the updated version in the project's central library. The library may be a physical storage space for hard copy documentation and program listings or an on-line subdirectory or partitioned data set where programs are stored. The central repository must be maintained in a manner that facilitates auditing the new version and all prior versions of a baselined product. To protect the integrity of the project's products, the ability to move CIs into and out of the library should be limited to as few staff members as possible and strict procedures should be established in the CMP and adhered to. These procedures will prevent old versions of CIs from overwriting newer versions and the same product from being updated concurrently.

Update Inventory Log – Ensure that updated product information is recorded in the inventory log. This information should include the following details:

Title of the product

- Latest release date

- New version number

- Name and version or model numbers of the software and hardware used in development of the new version of the product

- Organization responsible for updating the product (usually the sponsoring organization)

- The product's distribution list

- Updates to the central library

Distribute Copies of Products as Required – After an approved, updated version of the product is stored in the project's central library, distribute copies of the updated product or change packages following the same distribution channels and procedures used for the original product. Institute a plan to ensure the prompt removal of outdated versions from the work area.

Archive Old Version of Products – Outdated versions of all products should be archived and retained for the period of time required to comply with SBA standards and guidelines. Old versions of software may be moved to tape or may be moved to a separate subdirectory or data set; store only the latest version in the main software library to prevent old versions from being tested, reworked, and released for production.

7.3.9   Prepare Project Review Document

The Project Review process serves two important functions. First, it measures project output based on the criteria established during early phase requirements capture. This information is critical is assessing the project management function and its methodology, pointing out process weaknesses and strengths. Second, it measures project outcomes through system-user feedback, allowing stakeholders to measure their initial expectations against benefits realized by the operational system.

Such information is critical to improving both output and outcomes success for future projects. The culmination of system documentation, the Project Review establishes ongoing maintenance and development expectations for the system, especially for those projects, which encountered output or outcomes shortfalls for any of a variety of reasons (technological challenges, lack of stakeholder participation, changing business goals, etc.).

7.4   Performance & Governance Considerations

The alignment of the system with the overall enterprise architecture should be measured at regular intervals to ensure that the system continues to contribute effectively to the achievement of enterprise strategic goals.

7.5   Exit Criteria

The system remains in the Steady State Stage until it is determined that its operation is no longer cost-effective.

Table 7-1 Operate & Maintain Stage Document Deliverables

SDM Operate & Maintain Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
Security Plan* (updated)					
Contingency Plan (updated) and tested					
Security Procedures (updated)					
Operations Manual (updated as needed)					
Post Implementation Review* (PIR)					
Approved Operate & Maintain Stage SDM Decision Document	X	X			

8 Retire

This section defines requirements and guidelines for project level system disposition activities and documentation.

8.1 Purpose

This stage represents the end of the system's life cycle. It provides for the deliberate termination of a system to ensure that vital information is preserved for potential future access and/or reactivation. The system, when placed in the Retire Stage has been declared no longer useful and/or obsolete and has been scheduled for shutdown. The emphasis of this phase is to ensure that the system, e.g., equipment, parts, software, data, procedures, documentation, is properly packaged and disposed.

Disposition activities ensure the orderly termination of the system and preserve the vital information about the system so that some or all of the information may be reactivated in the future if necessary. Particular emphasis is given to proper preservation of the data processed by the system, so that the data can be effectively migrated to another system or archived for potential future access in accordance with applicable records management regulations and policies. Each system should have an interface control document defining inputs and outputs and data exchange. Signatures should be required to verify that all dependent users and impacted systems are aware of disposition.

While the decision to retire or replace a system is a function of the CPIC process (See [insert CPIC reference]) that involves SBA Enterprise Architecture, there are project level activities that are required to fulfill and complete the process of retiring a system.

8.2 Entry Criteria

A documented decision by the BTIC to retire the system initiates the Retire Stage.

8.3 Process

The objectives for all tasks identified in this phase are to retire the system, software, hardware and data. The tasks and activities actually performed are dependent on the nature of the project. The disposition activities are performed at the end of the systems life cycle. The disposition activities ensure the orderly termination of the system and preserve vital information about the system so that some or all of it may be reactivated in the future if necessary. Particular emphasis shall be given to proper preservation of the data processed by the system, so that the data are effectively migrated to another system or disposed of in accordance with applicable records management and program area regulations and policies for potential future access. These activities may be expanded, combined or deleted, depending on the size of the system.

Figure 8-1 Retire Stage Process Steps

When notified to retire a system, the System Sponsor works with the OISS, etc. to prepare documentation and take action to remove the system from service. Because of the complexity of many systems, it is imperative that all stakeholders and SBA support offices and functions participate in the retirement process.

8.3.1   Develop Disposition Plan

The Disposition Plan must be developed and implemented. The Disposition Plan will identify how the termination of the system/data will be conducted, and when, as well as the system termination date, software components to be preserved, data to be preserved, disposition of remaining equipment, and archiving of life cycle products.

The objectives of the plan are to end the operation of the system in a planned, orderly manner and to ensure that system components and data are properly archived or incorporated into other systems. This will include removing the active support by the operations and maintenance organizations. The users will need to play an active role in the transition. All concerned groups will need to be kept informed of the progress and target dates. The decision to proceed with Disposition will be based on recommendations and approvals from an In-Process Review or based on a date (or time period) specified in the System Project Plan.

This plan will include a statement of why the application is no longer supported, a description of replacement / upgrade, list of tasks/activities (transition plan) with estimated dates of completion and the notification strategy. Additionally, it will include the responsibilities for future residual support issues: identifying media alternatives if technology changes; new software product transition plans; alternative support issues (once the application is removed); parallel operations of retiring and the new software product; archiving of the software product, associated documentation, movement of logs, code; and accessibility of archive, data protection identification, and audit applicability.

The Sponsor with participating SBA offices and functions work together to develop the System Disposition Plan for all system components including, but not limited to, software, hardware, manual and records. The plan should define the roles, responsibilities and activities of all key system stakeholders. Through its development and when it is complete, the plan is reviewed by all key stakeholders and signed off on by, at a minimum, the Sponsor and CIO.

Record Disposition Schedule – Federal regulations require that all records no longer needed for the conduct of the regular business of the agency be disposed of, retired, or preserved in a manner consistent with official Records Disposition Schedules. The decisions concerning the disposition criteria, including when and how records are to be disposed, and the coordination with SBA Records Management representatives to prepare the Records Disposition Schedule for the system, shall be the responsibilities of the Project Manager and done in accordance with SBA policies.

It is critical at the time of retirement or rollover of the system that the system Sponsor and participating SBA offices ensure:

- All records are preserved, retained, and fully accessible for the full retentions in accordance with appropriate dispositions

- Any temporary records are destroyed in accordance with appropriate dispositions

- All permanent records are transferred to NARA in accordance with the appropriate dispositions

8.3.2   Archive or Transfer Software Components

Similar to the data that is archived or transferred, the software components will need to be transferred to the new system, or if that is not feasible, disposed of.

8.3.3 Archive Life Cycle Deliverables

A lot of documentation went into developing the application or system. This documentation needs to be archived, where it can be referenced if needed at a later date.

8.3.4 End the System in an Orderly Manner

Follow the plan in the Disposition Plan for the orderly breakdown of the system, its components and the data within.

8.3.5 Dispose of Equipment

If the equipment can be used elsewhere in the organization, recycle. If it is obsolete, notify the appropriate SBA office to excess all hardware components.

8.3.6 Conduct Post-Termination Review

To complete the Retire process, the project team conducts the Post-Termination Review and a report that details the findings of the Retire Stage review. It includes details of where to find all products and documentation that has been archived. It is reviewed by all key stakeholders and signed off on by, at a minimum, the Sponsor and CIO.

8.4 Performance & Governance Considerations

The disposition of the system must comply with CPIC and Enterprise Architecture requirements.

8.5 Exit Criteria

The Retire Stage is complete when all documents and associated requirements are complete and sign off on.

Table 8-1 Retire Stage Document Deliverables

SDM Retire Stage Deliverable	Project File	Sponsor / CIO Approval	OCIO		
			EA Review	CPIC Review	Security Review
System Disposition Plan*	X	X			
Post-termination Review Report	X	X			
System Decision Paper (Updated)	X				
Approved Retire Stage SDM Decision Document	X	X			