SLCM 2003.1
Artifacts in a Nutshell
(as of 01/21/2005)

Project Development Phases – Pension Benefit Guaranty Corporation's (PBGC) System Life Cycle Methodology (SLCM) is comprised of five project development life cycle phases. These phases include, from development start to finish: Project Initiation, Definition & Analysis, Design & Planning, Development, and Operations & Refinement. All potential development projects participate in the Project Initiation phase. All approved development projects participate in the Definition & Analysis phase, Design & Planning phase and Development phase and the Operations & Refinement phase.

The specification of 5 delimited phases of PBGC's SLCM is not meant to imply that a system's life cycle ends with deployment to a production environment. There are of course, operational events that may cause a system to change following initial installation, including: bug discovery and fixes, database, network or other operational changes and system tuning and system de-commisioning. Further, project development from concept to implementation is not linear; often a project will move forward in a life cycle phase or from one phase to another only to find that a prior phase must be revisited before further progress can be made forward.

1. **Project Initiation** – Phase 1 of the SLCM defines the development, proposal and corporate approval of business cases from: strategic planning efforts or Enterprise (EA) efforts.
   - **1.1. Deliverables:**
     - 1.1.1. Business Case – A good business case will provide proof that a business concept, regardless of its source, is feasible, viable, provides a positive return on investment, and falls within an acceptable level of risk.
       - 1.1.1.1. Feasibility Study – A positive feasibility study result substantiates a good business case. An evaluation of a business concept to determine if the proposed change is reasonable can determine feasibility. Determine if the concept is possible given: the current technical environment, funding availability, skilled resource availability and schedule constraints.
       - 1.1.1.2. Viability Study – To propose an acceptable business case the viability of the idea should be substantiated. An evaluation of a business concept based on whether 5 critical success factors can be met can prove viability. The factors are: project sponsor with organization authority, business case linking project and objective, project manager with appropriate experience, key resource availability and major functional milestone identification.
       - 1.1.1.3. Cost Benefit Analysis – The proposed project cost, measured at intervals over some period of time (a good example is yearly estimate of cost and benefits over 5 years) should be a measured against the additional income it will provide or based on the money it will save the corporation following implementation. Corporate value for the project might be measured in a non-monetary method (such as an increase to the customer satisfaction increase), against established measurable corporate goals. A business concept should not be proposed unless it can provide a positive return on investment. (ROI) within an acceptable time frame.
       - 1.1.1.4. Risk Assessment – A risk is an uncertain event or condition that, if it occurs, positively or negatively affects a project's objectives. A business concept risk (business, cost, technical, schedule) should be measured and reported as tolerable prior to the proposing a business case.
     - 1.1.2. EA Plans – Enterprise Architecture may promote business cases required to further enterprise architecture system goals. In addition, all system projects, regardless of source, must map or be mapped to the planned enterprise architecture.
     - 1.1.3. Project Approval – Project approval is implied when budget is assigned to the project through the CPIC process.

2. **Definition & Analysis -** Phase 2 of the SLCM will kick off with an approved project and budget. During this phase the project manager will identify project participants and their roles, develop the approach and clearly identify business requirements.

2.1. Deliverables:

2.1.1. Project Charter – The project charter states the project objectives, the business problems to be solved and the benefits the proposed system will provide. In addition, it identifies project stakeholders or champions. It will also present a development and testing approach to the extent that it is known. The intent of this document is to present the challenge, approach and measurements of success to corporate executives and obtain their approval.

2.1.2. Schedule/Project Plan - The plan provides details on the functional units involved, required job tasks, cost and schedule performance measurement, schedule start and end dates, milestones, task dependencies, resources, and review scheduling. Project progress can then be measured by tracking actual task completion dates against scheduled dates in a baseline project plan, producing Percent Completed Reports and Earned Value Reports.

2.1.3. SLCM Tailoring Document—Produced by the PBGC SLCM methodologist after working with the project team; this document "tailors" the SLCM all-inclusive scope of deliverables to meet the need of individual projects. It determines whether an SLCM artifact is required for a specified project or if it can be excluded. Exclusions are documented with the date of agreement and the reason for the exclusion. In addition, this document identifies, for clarification, whether or not the artifact requires approval signatures. The specification of SLCM deliverables in this document as "required" is not meant to limit or exclude additional identified project deliverables. This document is only meant to clarify what specific SLCM artifacts are required to meet PBGC's SLCM compliance standards.

2.1.4. Communications Plan - Establishes the project communication goals and objectives: with users, the development team, stakeholders, interfacing systems, vendors, infrastructure support teams, and with IV&V. It outlines the communications approach, details how communication performance is measured. Further, it identifies stakeholders and involved organizations and resources, proposes key messages, and recommends corresponding communication tools for delivering information. In addition it describes the processes and procedures for communicating with the various teams involved in the implementation effort (e.g., PBGC, software vendor, other contractors).

2.1.5. Project Risk Plan - Defines the methodology, process, and procedures for identifying, analyzing, and responding to risks associated with a project. It enables program management to identify and track specific known risks for the project as well as impact, consequence, likelihood, mitigation strategies, and owners.

2.1.6. Issue Tracking Plan - Defines the methodology, process, and procedures for identifying, analyzing, and responding to issues associated with the PPS project. It enables program management to identify and track project issues, description, proposed resolution, and the member responsible for rectifying the issue.

2.1.7. Security Plan – A plan produced by the project team that details the types of computer security required for a system based on the type of information being processed and the degree of sensitivity. This plan also identifies whether the PBGC security office requires that a system be certified and accredited. This plan requires signoff.

2.1.8. Certification &Accreditation (C&A) Plan – A plan produced by the PBGC Security Office for all major business systems or mission critical general support systems. It describes the approach for obtaining certification and accreditation of the system before it becomes operational.

2.1.9. Current Business Process Flows – Diagram of existing business processes (activities and decisions) as they relate to each other.

2.1.10. Desired Business Process Flows - Diagram of improved and desired business processes (activities and decisions) as they relate to each other.

2.1.11. Business Gap Analysis – Identifies differences between the current or "As Is" business process and the desired or "To Be" business process.

2.1.12. Business Requirements – Identifies what the business "needs" to do to meet the process "gaps" identified in the gap analysis. This document will also identify business rules which define more of the "how" the business needs to do something. Business requirements should

be testable, in that a test plan can be written to address a business scenario. This document sets a major milestone point in the SLCM Definition and Analysis phase; it requires the signoff of the CTO and others. It is recommended that requirements be managed across the project via a requirements traceability matrix to provide a traceable method of matching design, testing, and software components to business requirements. Once final requirements are produced, a formal change management process should be in place to approve or deny requested changes to requirements and their impact to scope and planning throughout the project life cycle.

2.1.13. Phase Corrective Action Plan – Actions identified in an SLCM phase compliance review, peer review, or specific artifact review that are needed to resolve problems or discrepancies identified as a result of this review. These actions or issues should be captured and carried over into the issues log for resolution in the next phase.

2.1.14. SLCM Phase Compliance Report – At the end of every SLCM phase an SLCM methodologist will compile a compliance report that measures project team compliance against SLCM processes and artifacts. The compliance report results may feed into a Phase Corrective Action Plan.

2.2. Milestone Review and Signoff Points:

2.2.1. Business Requirements Signoff

2.2.2. Security Plan Signoff

2.2.3. SLCM Phase Compliance Report

3. Design & Planning – During Phase 3 of SLCM project teams will develop and submit plans for approval. Perform functional and technical design based on established requirements.

3.1. Deliverables:

3.1.1. SOW – Statement of Work may be required to obtain a product or software development services to build to the business requirements resulting from a business process re-engineering effort. This document requires IT signoff.

3.1.2. RFQ – Request for Quotation, may be required if a SOW is necessary to acquire a product or system development services.

3.1.3. Risk Assessment - Identifies and tracks specific known risks for the project as well as impact, consequence, likelihood, mitigation strategies, and owners.

3.1.4. Change Management Plan – The plan assesses the impact of the proposed technology on staff, business objectives, and the current operating environment. Defines how software updates, corrections, and/or changes are managed in development, test and production environments. Defines what tool will be used to track and manage software change (like PVCS). Defines update responsibility for software promotion through Service Center. In addition, the project team will create a change request record in Peregrine Service Center to identify upcoming changes required to hardware and/or software in support of a new product, system or system upgrades. As details of the change become known specific tasks may be identified and scheduled via Service Center. All changes to PBGC's software or infrastructure environment must be identified and managed through Service Center.

3.1.5. Logical Data Model – Database models that have not yet been normalized; these models are built strictly on the data as it relates to business entities.

3.1.6. System Design - Contextual models of the software system, data and functions that relate to the system.

3.1.7. Functional Specifications – A description of the system and its components by function.

3.1.8. Development Strategy –Describes the development methodology, such as waterfall, prototyping, Rapid Application Development (RAD), Rational Unified Process (RUP), or iterative. Includes a discussion of development tools, languages, object oriented approach, service based components, and use of COTS ancestor components. In addition this document should reference all established corporate software development standards. Discuss object coding standards such as naming conventions for classes, variables, and validations. Also includes naming conventions for software objects and naming conventions for storage of work products.

3.1.9. Common Services Use Plan – Identifies shared services that are used or built by this system. Identifies which existing software service components this system will use such as

document imaging, correspondence tracking, letter generation and workflow. Also identifies which service components this development effort will build that will be available for re-use by other systems.

3.1.10. Release Packaging Approach – Definition of the software release strategy. This artifact defines whether the results of this development effort will be delivered in a phased approach or as a single package. It should also define if the software staging from the development environment to test and/or production is dependent on any other system development efforts and describe the release dependencies.

3.1.11. Data Conversion Plan – If the new system is an enhancement or replacement to an existing software system that is responsible for data management, a data conversion plan is required. It should define what data needs to be moved by entity and data element, what data changes are required, and what are the data conversion steps and time-table. It might also define how the specialized software to convert the data is to be built or acquired and the requirements of that software. Includes the data conversion requirements, products/tools evaluated, and strategy. The software utilities, routines, or scripts used to load data from PAS and other sources into the production PPS environment.

3.1.12. Development System Environment Plan – Requirements of the development environment that must be met to built the project software. In a shared development environment these plans must be explicit and produced early. Identifies hardware requirements such as servers, physical data center location, universal power supply (UPS) and networking. It also defines software requirements such as compilers, DBMS's, and licensing requirements. In addition, it may specifically state data refresh and software back-up requirements necessary to meet development needs.

3.1.13. System Test System Environment Plan - Requirements of the system test environment that must be met to test the project software. In a shared testing environment, such as the ITC (Integrated Test Center), these plans must be explicit and produced early in the project life-cycle. It identifies hardware requirements such as servers, physical data center location, universal power supply (UPS) and networking. It also defines software requirements such as compilers, DBMS's, and licensing requirements. In addition, it may specifically state data refresh and software back-up requirements necessary to meet development needs.

3.1.14. Integration Test System Environment Plan - Requirements of the integration test environment that must be met to test the project software as it shares data or resources with other system software. Best practices recommend that this test should be done in an isolated system, mirrored from a production environment. Identifies hardware requirements such as servers, physical datacenter location, universal power supply (UPS) and networking. It also defines software requirements such as compilers, DBMS's, and licensing requirements. In addition, it may specifically state data refresh and software back-up requirements necessary to meet development needs.

3.1.15. User Acceptance Test System Environment Plan - Requirements of the user test environment that must be met to test and accept the project software. Identifies hardware requirements such as servers, physical datacenter location, universal power supply (UPS) and networking. It also defines software requirements such as compilers, DBMS's, and licensing requirements. In addition, it may specifically state data refresh and software back-up requirements necessary to meet development needs.

3.1.16. Production System Environment Plan - Requirements of the production environment that are necessary to support this system. This plan should define changes or upgrades required in the production environment necessary to support the new software. A common mistake a project team might make is to assume that the existing production environment has spare resources, sufficient to support new or expanded system functionality. Proper production environmental planning helps mitigate this risk by identifying early all required resources. Identifies hardware requirements such as servers, physical datacenter location, universal power supply (UPS) and networking. It also defines software requirements such as compilers, DBMS's, and licensing requirements. In addition, it may specifically state data refresh and software back-up requirements necessary to meet development needs.

3.1.17. Unit Test Plan - Each business requirement is translated into one or more test purposes, which in turn is expanded into test scenarios. A test plan for a "unit" is generally limited in

scope by the function performed in a unit of code, generally one program or functional piece of code that relates to a business function.

3.1.18. System Test Plan - Each business requirement is translated into one or more test purposes, which in turn are expanded into test scenarios. A "system" test plan draws business scenarios that usually cover multiple functions and cross the interfaces between multiple units of code.

3.1.19. Integration Test Plan - Each business requirement is translated into one or more test purposes, which in turn are expanded into test scenarios. An "integration" test plan draws business scenarios that usually cover numerous functions and crosses the interfaces between one or more business systems.

3.1.20. IV&V Test Plan – A testing approach defined by an independent verification and validation test team, organizationally and functionally separate and apart from the development team. In general, this team performs independent integration and regression testing. This plan defines their approach and business scenarios to be tested.

3.1.21. User Acceptance Test Plan – Defines the business scenarios and testing approach in enough detail that when scripted, a user will feel confident that the system has met their requirements as defined.

3.1.22. Production Rollout Plan (Deployment Plan)

3.1.23. Production Services Support Plan (DBA, Help Desk, User Administration)

3.1.24. Infrastructure (Configuration) Support Plan (DB, Servers, Memory, Other Hardware, Network, Footprint)

3.1.25. Configuration Management Plan – This plan identifies methods and tools for promoting deliverable items within development, testing, and production environments. Identifies the final location of all work products including software, documentation, plans, status reports, critical communications, action items, and meeting minutes.

3.1.26. User Training Plan – Defines the approach and steps necessary to ensure that all users are trained to use new system functionality. It must include Application User Manuals, Error Code and Recovery Procedures.

3.1.27. Disaster Recovery Plan (COOP) – A disaster recovery plan that is agreed upon by management, developers, and users. Includes a scenario that covers the scope of activities required to recover the proposed system in the event of a catastrophic failure. Describes, for example, how much of the system will have to be brought back on-line and in what time frame, the formal support organization, the maintenance activities at the disaster recovery site, and the maintenance records and reports. This plan should be updated on a regular quarterly or annual schedule.

3.1.28. Phase Corrective Action Plan – Actions identified in an SLCM phase compliance review, peer review, or specific artifact review that are needed to resolve problems or discrepancies identified as a result of this review. These actions or issues should be captured and carried over into the issues log for immediate resolution or resolution in the next phase.

3.1.29. SLCM Phase Compliance Report – At the end of every SLCM phase an SLCM methodologist will compile a compliance report that measures project team compliance against SLCM processes and artifacts. The compliance review results may feed into a Phase Corrective Action Plan.

3.2. Milestone Review and Signoff Points:

3.2.1. SOW Signoff

3.2.2. Design Signoff

3.2.3. Planning Documents Signoff

3.2.4. SLCM Phase Compliance Report

4. Development – Phase 4 of the SLCM applies to physical construction of all software, configuration of system environments, testing software in each staging environment and advancement of software to the next staged environment, from development through production.

4.1. Deliverables:

4.1.1. Detailed Functional Design Specifications - Provides a detailed description of the system design, as well as a detail breakout of the functional design. Addresses issues such as GUI, pseudo-code for processes, interfaces with other systems, and hardware considerations.

4.1.2. Component or Module Design - The Component/Module Design translates user-based scenarios to services-based components and technologies. Vendors are encouraged to use a tool such as UML to explain the classes, attributes, methods, and hierarchy of the software components. Components should be designed using a Model View Control (MVC).

4.1.3. Security Risk Analysis and Assessment – An analysis of the system design to assess the system security risk must be completed and provided to the OIT security team for review and approval.

4.1.4. Physical Data Base Design - The physical database design prepares and optimizes the data and database for the types of software being used to create, access, update, and delete data instances as well as its execution from clients, servers, and the web. It includes a mapping between the logical database design and the physical database design for the components.

4.1.5. Application Software – Custom code and/or COTS software necessary to meet business requirements.

4.1.6. Prototype Assessment – Review of a working prototype of a portion of the entire system. It might be: user interfaces, web functionality or database access that is prototyped. It may be one function prototyped. As such the assessment should measure the software's viability; it should identify limitations or approve completeness.

4.1.7. Unit Test Report – The development team will create a document reporting the results of each unit test. Unit level testing is the lowest level of software testing, usually done at the component level. IV&V will verify the results to ensure that development testing of the product meets a level of quality and that the final result is free of errors.

4.1.8. Functional Test Report - The development team will create a document reporting the results of each functional test. Functional level testing may cover multiple unit or interrelated components necessary to perform a system function. IV&V will verify the results to ensure that development testing of the product meets a pre-determined level of quality and that the final result is free of errors.

4.1.9. System Test Report - The development team will create a document reporting the results of a system test. System level testing may cover multiple system functions providing services within a single business area. IV&V will verify the results to ensure that development testing of the product meets a pre-determined level of quality and that the final result is free of errors.

4.1.10. Integration Test Report - The development team will create a document reporting the results of an integration test. Integration testing tests one or more system interfaces processes, where one system passes data or receives data from another system. This report may involve more than one development team. IV&V will verify the results to ensure that development testing of the product meets a level of quality and that the final result is free of errors.

4.1.11. IV&V (Independent Verification and Validation) Test Report – PBGC requires that all software undergo independent testing by an organization outside of the project development team. The IV&V test report measures the errors of software/product roll-out to an independent environment as well as reporting the results of the independently run tests in a production-like environment. With a clean test report and signoff, the product is eligible to be deployed to the production environment.

4.1.12. UAT Test Report – A report of bugs or success based on testing done by users to accept the software as provided as meeting their business needs.

4.1.13. Security Certification and Analysis Report - PBGC independently verifies the security components of a system against the Certification and Accreditation Plan. The assessor produces a report of any identified physical, network or data security vulnerabilities as well as compliance with stated security procedures.

4.1.14. Phase Corrective Action Plan – Actions identified in an SLCM phase compliance review, peer review, or specific artifact review that are needed to resolve problems or discrepancies identified as a result of this review. These actions or issues should be captured and carried over into the issues log for resolution in the next phase.

4.1.15. SLCM Phase Compliance Report – At the end of every SLCM phase an SLCM methodologist will compile a compliance report that measures project team compliance

against SLCM processes and artifacts. The compliance review results may feed into a Phase Corrective Action Plan.

4.2. Milestone Review and Signoff Points:

4.2.1. Technical Design Signoff – Technical design is composed of the physical data model, detailed functional specifications, security assessment and business and functional requirements.

4.2.2. SLCM Phase Compliance Report

5. Operations & Refinement - Phase 5 of the SLCM will conduct all post-implement tasks and formally close out the project.

5.1. Deliverables:

5.1.1. Post Implementation Review Report

5.1.2. SLCM Final Project Compliance Report

5.2. Close Out Management and Tracking Processes/Plans:

5.2.1. EA Architecture

5.2.2. Project Plan/Schedule

5.2.3. Communication Plan

5.2.4. Project Risk Identification and Risk Mitigation Plan

5.2.5. Issue Tracking

5.2.6. Systems Security Plan

5.2.7. Environment Configuration Management Plan

5.2.8. Change Management Plan

5.3. Milestone Review and Signoff Points:

5.3.1. Post Implementation Review - The purpose of this review is to document implementation experiences, to recommend system enhancements, provide guidance and lessons learned for future projects, ensure that the system functions as planned and expected, verify that the system cost is within the estimated amount, and to verify that the intended benefits are derived as projected.

5.3.2. SLCM Phase Compliance Report – At the end of every SLCM phase an SLCM methodologist will compile a compliance report that measures project team compliance with SLCM processes and artifacts. The report in this phase addresses operations and refinement activities compliance; it also summarizes compliance reports across all project phases.