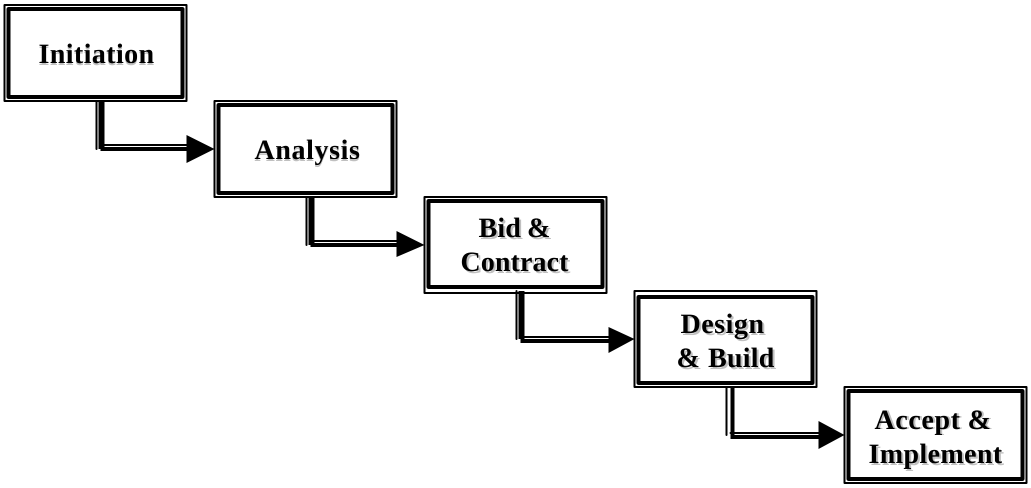
Data-Oriented Quality Solutions
PROJECT LIFECYCLE HANDBOOK



Data-Oriented Quality Solutions

PROJECT LIFECYCLE HANDBOOK

Data-Oriented Quality Solutions
2105 Whitfield Lane
Orlando, FL 32835-5940 USA

407.296.6900 (Voice & Fax)

www.doqs.com

rbiehl@doqs.com

Copyright © 1999, Data-Oriented Quality Solutions. All rights reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of Data-Oriented Quality Solutions.

Trademarks

CMM and *Capability Maturity Model* are registered
trademarks of Carnegie Mellon University.

TABLE OF CONTENTS

Copyright © 1999, D.O.Q.S. All rights reserved.

LIST OF FIGURES

HOW TO USE THIS HANDBOOK

This *CIS Project Lifecycle Handbook* is intended to be read and used by experienced information technology professionals. It presumes background experience in the use of structured project methodologies.

All readers should become familiar with the Lifecycle Overview chapter in order to understand the basis for the lifecycle, and its 3-dimensional architecture. This architecture is key to establishing a correct flow through this document.

This lifecycle is a high-level document. Detail procedures describing portions of the activities required are described in a separate document entitled *DOQS Project Lifecycle Procedures*. Readers should also obtain, or identify sources for, the various materials listed in the Supporting Materials (p. 39). Less experienced readers will need to rely on these additional materials in order to meet the goals and deliverables of this Handbook.

Uses for this Handbook will fall into two broad approaches: a) planning a project phase for a project that is moving forward through the lifecycle from beginning to end, and b) reviewing a project that is in progress but has not used this lifecycle. Other uses might involve combinations of these two approaches.

Planning a Project Phase

For the project phase being planned:

1. Review the goals, activities, and quality assurance criteria for the phase in the *Project Phase Flow* chapter.
2. Review the noted components from any cited industry standards.
3. For each KPA in the *Key Process Areas* chapter:

   3a. Review the expected impact statement for the phase.

   3b. Review the noted components of the industry standard models.
4. For each deliverable in the *Key Lifecycle Products* chapter:

   4a. Review the contents, form, and success factor descriptions for general requirements and any phase-specific criteria.

   4b. Review the noted components of the industry standard models.
5. Review the items in the *SW-CMM Levels 1 to 2* reference, noting any descriptive items that might apply specifically to the phase.

Copyright © 1999, D.O.Q.S. All rights reserved.

Reviewing a Project

Identify the lifecycle project phase just completed. The project review can be done starting with this current phase and working backward to the Initiation Phase, or it can start with the Initiation Phase and work forward to the current phase.

1. Review the items in the *SW-CMM Levels 1 to 2* reference, noting any descriptive items that apply specifically to the project being reviewed.
2. Repeating for each project phase:

   2a. For each deliverable in the *Key Lifecycle Products* chapter, assure that the criteria for content, form, and success factors have been met by the project's deliverables. Pay particular attention to any phase-specific criteria.

   2b. For each KPA in the *Key Process Areas* chapter, assure that the project has experienced the impact described in the expected impact statement for the phase.

   2c. Using the phase description in the *Project Phase Flow* chapter, assure that the goals, activities, and quality assurance criteria for the phase have been satisfied by the project.

Whether working forward or backward through the lifecycle phases, the results of the review should be the same because all criteria in the lifecycle will be assessed. The order in which the review will be accomplished is really only relevant if there is a high enough perception of risk that the review might be cut-off at some point because of excessive problems being identified. In that case, effort will be saved if the direction of the review starts at the riskiest end of the lifecycle for the project.

LIFECYCLE OVERVIEW

This Project Lifecycle is comprised of three major and interrelated parts:

- **Project Process Flow**, defining the five major phases of activity associated with each information technology project. All projects pass through all five phases, albeit at different paces and levels of formality. Even very small projects that are accomplished in a day or two, or field support activities that are on-going, must conceptually meet the goals of all five phases.
- **Key Process Areas**, defining the core process competencies that are needed across all five phases of the process flow in order to assure high quality and continual improvement of project outcomes. Again, the rigor with which each process is applied to a project will be a function of the size, duration, and risk associated each project or support activity.
- **Key Lifecycle Products**, defining the four major project deliverables that need to be produced throughout the process flow and across all process areas on each project or support activity. Short small projects will produce deliverables that appear very different from longer and larger ones, but all projects must produce all deliverables in some form to be successful.

These three dimensions define the lifecycle tools for executing effective projects and support activities. At a macro level, successful projects and activities will execute all five phases using all seven key processes to produce and manage all four deliverables. At a micro level, project leaders and support staff will use this lifecycle to manage each *permutation*[1] of lifecycle components for continual improvement and risk reduction.

[1] For example, one project might concentrate on improving the *Analysis Phase - Project Planning - Statement* of Work triad, while another focuses on the *Bid & Contract - Requirements - Contractual Agreement* triad. Each permutation of the three dimensions of this lifecycle offers an opportunity to focus attention and improve outcomes related to that specific aspect of the project.

Copyright © 1999, D.O.Q.S. All rights reserved.

Lifecycle CMM Basis

The scope, complexity, and level of detail of this lifecycle is largely driven by the adoption of the *Capability Maturity Model® for Software* (SW-CMM) from the Software Engineering Institute.

This Handbook is designed to promote process maturity at the Repeatable Level (Level 2). The focus is on establishing core project management controls such that projects are planned and executed consistently, using a common framework of procedures and deliverables. This results in highly *repeatable* project processes*.*

The fundamental goal of the Capability Maturity Model for Software is to enable and accelerate organizational learning. The SW-CMM is a vehicle for communicating visions of improving practice, not an end in itself. Compliance to SW-CMM requirements is not the objective. Rather, continually improving processes and organizational effectiveness are the objective.

Leveraging the SW-CMM in this lifecycle is focused on accelerating adoption of the guiding principles implied therein; taking advantage of the countless person-hours devoted by industry in the exploration and definition of key process maturity factors for information technology.

Simplifying Assumptions

To be used effectively in this lifecycle, several aspects of the SW-CMM must be tailored to meet unique characteristics of smaller, less mature, information technology organizations.[2] The SW-CMM is applicable, and was originally intended for, very large and complex software projects involving potentially hundreds of specialized people working over a period of years. These characteristics result in a SW-CMM description that can be overwhelming to a smaller or less mature organization.

Considering this, the following assumptions were considered in the creation of the portions of this lifecycle that map to requirements of the SW-CMM:

- The nature of the organization is such that *all* requirements can be treated as *allocated* requirements. The organization doesn't conduct major *systems engineering* initiatives that would result in a diversity of hardware, software, and interface requirements. For the purpose of this lifecycle, the phrases *requirement, software requirement*, and *allocated requirement* can be considered synonymous.

[2] *See* Process Tailoring and the Capability Maturity Model, Technical Report CMU/SEI-94-TR-024. Considerations for Tailoring with the SW-CMM (Section 2.2) discusses several areas that require tailoring, including similarities and differences between organizational structures, degree of formality or scope of the organization in general, specific business goals to be achieved through CMM adoption, and the current process capability of the organization.

- The customers for lifecycle projects are *not* presumed to be particularly sophisticated with respect to information technology. As a result, certain deliverables that the CMM attributes to customers (e.g. Statement of Work) are actually to be developed by the IT project team and *approved* by the customers.
- The organization is made up mostly of generalist who carry out a broad range of activity on each project. This is particularly true of the on-site field support activities. The various specialty groups discussed in the CMM (e.g. SQA, SCM) will generally be software project team members carrying out these functions as part of their regular duties.

Other Industry Models

Although driven by the adoption of the SW-CMM, this lifecycle is not limited to that single model. An additional CMM from SEI, the *Software Acquisition Capability Maturity Model*, also played a role in the definition of the lifecycle details; as did other various standards and guidelines from ISO, IEEE, and the Project Management Institute. Cross-references are provided throughout this handbook to aid readers in finding additional details to support and extend this material.

Organizational Roles

The distinctions attributed by the CMM to different levels of management; from senior managers, to project managers, to software project managers, to team leads, is *not* a major distinction in this lifecycle. Because lines of authority within the organization are short, the general bias of this lifecycle is to keep everyone informed and in-the-loop.

The principle roles required by this lifecycle include:

- **Senior Managers** - Those individuals concerned with the long-term viability of the organization, and the strategic direction of its services.
- **Managers** - Those individuals concerned with resourcing and providing specific key functions and areas of service under their responsibility.
- **Project/Service Managers** - Those individuals concerned with specific tactical project activities and service levels on a day-to-day basis.
- **Task Leaders** - Those individuals concerned with the planning and coordination of specific tasks and activities in the workplace.
- **Staff Individuals** - Those individuals concerned with carrying out specific work task assignments.

Copyright © 1999, D.O.Q.S. All rights reserved.

In most situations, many of these roles are played by a single individual or a small team. Lines of communication remain small even when other managers are required for decision making or setting policy. Distinctions between levels of management take on a more important role when discussing cross-location or multi-location issues and services.

At times, individuals working autonomously at field locations are required to take direct managerial responsibility to keep local issues from being overcome by events. In such circumstances, field staff are expected to make their own judgements regarding the appropriate blend of personal self-empowerment, outreach to central management, and the use of local management as proxy for central management.

The on-going emphasis should be on effectively meeting the goals of this lifecycle over the administrative emphasis on specific roles and responsibilities.

PROJECT PROCESS FLOW

The process flow for this project lifecycle includes five major phases of activity:

- **Initiation** – Receiving, clarifying, prioritizing, and allocating user service requests to assigned project activity; and the maintenance of the overall requirements repository that results from those projects.
- **Analysis** – Clarifying the problem and business case associated with assigned projects, and the specification of the full functional and technical requirements to be met to solve the associated business problems.
- **Bid & Contract** – Identifying, costing, scheduling, and contracting the resources necessary to implement the defined requirements.
- **Design & Build** – The designing and building of the application systems and components necessary to implement the defined and contracted requirements.
- **Accept & Implement** – The validation acceptance and implementation of provided software solutions, and the resulting shut-down of associated projects and contracts.

Waterfall Interpretation

The five phases represent a conceptual *waterfall* project lifecycle that can be thought of as a linear flow from project initiation to the acceptance and implementation of any resulting software products and procedures.

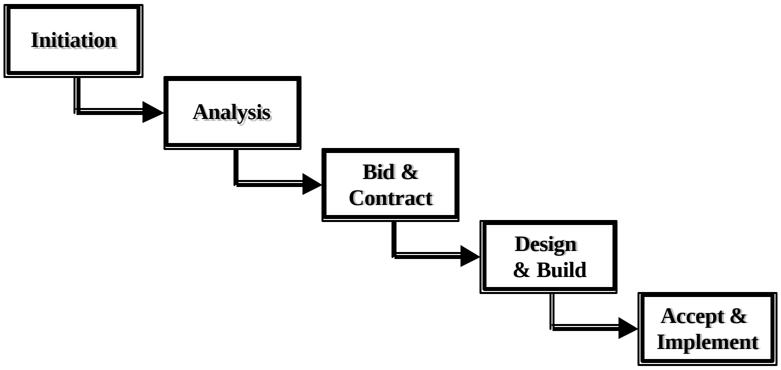


Figure 1 - Waterfall Perspective of Lifecycle

Copyright © 1999, D.O.Q.S. All rights reserved.

The waterfall interpretation of this lifecycle is conceptually valid for all projects, and most closely aligns with actual project plans for projects that are larger, longer, or higher risk than the typical project in the portfolio.

Spiral Interpretation

In fact, these five phases will be executed as an informal *spiral* lifecycle model; each phase cycling and active much of the time, with one phase perspective dominating based upon current project activities and requirements. The dominant phase at any time would be the one used to report project status in the sense of a waterfall lifecycle.

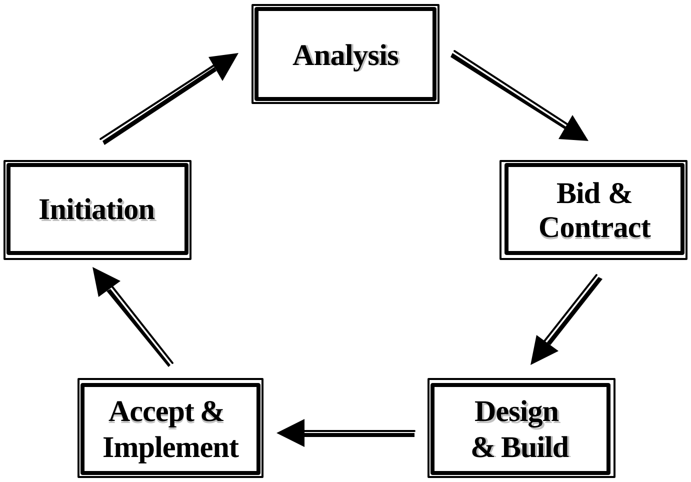


Figure 2 - Spiral Perspective of Lifecycle

The spiral interpretation is most likely to be used in situations where on-going activities are highly reactionary, driven by real-time requests that need to be incorporated into existing projects and support activities.

Examples include support or system maintenance projects where small teams or preferred vendors perform activities from multiple lifecycle phases at the same time in response to on-going requests from users. Other examples include larger projects where changes of scope or new requirements necessitate revisiting previously completed project phases in order to control and implement the changes.

INITIATION PHASE

The Initiation Phase is a spin-off of on-going organization activities to receive and respond to customer and supplier-driven service requests and needs.

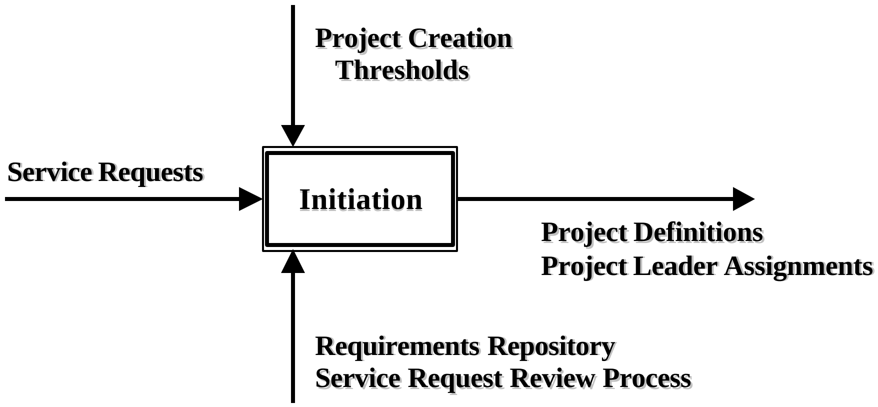


Figure 3 - Initiation Phase Overview

Goals include:

- Definition of the scope of activity.
- Identification of resources, users, or vendors that will be involved.
- Creation of a high-level project plan; including timeframes and allocations of assignments necessary to conduct work.

This Initiation Phase description includes components derived from the following industry models:

Model	Aspect
ISO 12207	Acquisition process initiation (Clause 5.1.1)

Project Activities

Significant activities in the Initiation Phase include:

- Review of each service request by the relevant support staff for the application to determine whether the change(s) requested warrant further initiation planning and consideration.
- Determination of the funding strategy to be applied to the request:
  - Determination that the request is for a service covered under an existing support/maintenance agreement with the requester, or

Copyright © 1999, D.O.Q.S. All rights reserved.

  - Negotiation with the requesting organization to obtain customer-based funding for the project.
- Assessment of the priority of the resulting effort; particularly with respect to identifying:
  - Emergency fix requirements vs. longer-term development.
  - Potential conflicts with other projects already in progress.
  - Opportunity costs associated with potential delay.
  - Likelihood of availability of funding and resources.
- Assignment of a project lead for the project.
- Determination of the system or application components that will be impacted by the project and whether or not those components are available under configuration management (i.e. not checked-out for other purposes).
- Allocation of resources to begin preliminary requirements, and
- Creation of an initial Development Plan by the project lead.

Quality Control

Interim review criteria include:

- Initial project plan is sufficient to assure that it conforms to overall objectives of the assigned service requests prior to detail project planning.
- Project leader can handle addition of this project to workload without creating an excessive burden.

Exit review criteria include:

- Project plan includes detail requirements activity plans, and at least high-level activities through to project completion.
- Project plan baselined and placed under configuration management.
- Needed system and application components confirmed to be available for check-out from configuration management for expected duration of project.

ANALYSIS PHASE

The Analysis Phase seeks and documents the true business problems to be solved by the project based upon the collection of service requests that were grouped and initiated in the prior phase.

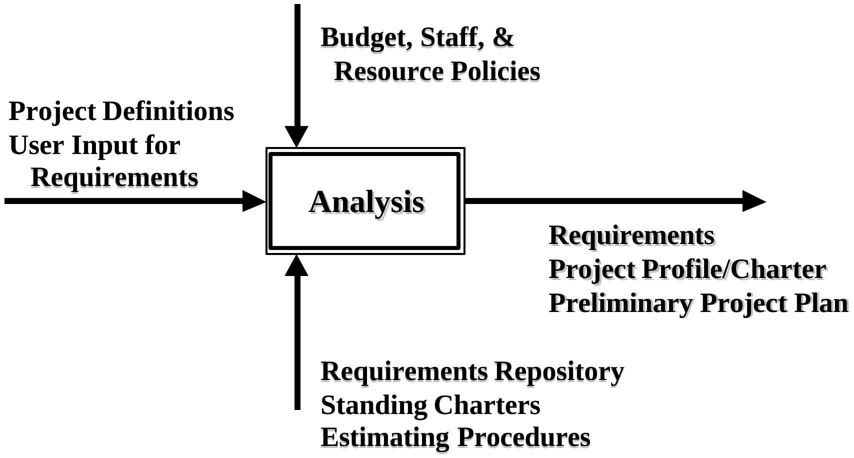


Figure 4 - Analysis Phase Overview

Goals include:

- Defining the *business* problem to be solved and the rationale that determines the importance and value of the solution to the business.
- Describing *what* needs to happen in order to solve the defined problem; and the criteria that should be used to determine if the project reaches successful implementation.
- Developing a more detailed description of the approach of the project team for continuing; and the plan for carrying out the remaining work.

This Analysis Phase description includes components derived from the following industry models:

Model	Aspect
IEEE 830	Recommended Practice for Software Requirements Specification
IEEE 1062	Defining the software requirements. (Clause 5.3)
IEEE 1233	Guide to Preparing System Requirements Specifications

Project Activities

Significant activities in the Analysis Phase include:

- Interpreting all of the service request materials in order to clarify the scope and range of issues to be addressed during requirements analysis.
- Interviewing, or otherwise communicating with, the requester and other stakeholders to identify all of the business issues that are involved in the problem to be solved. Such issues include, but are not limited to:
    - Data requirements, including new data elements, changes to data definitions, new or revised relationships among data, etc.
    - Process requirements, including revision to the triggers that cause events to happen within the business, new steps or sequences of steps that are required in the business, etc.
    - Interface requirements, including human interfaces and system-to-system interface.
    - Human-factors engineering (ergonomics), including those related to manual operations, human-equipment interactions, constraints on personnel, and areas needing concentrated human attention, that are sensitive to human errors and training.
    - Security specifications, including those related to compromise of sensitive information.
    - Reporting requirements, including new or revised reports across a variety of media.
    - Safety specifications, including those related to methods of operation and maintenance, environmental influences, and personnel injury.
    - Functional and capability specifications, including performance, physical characteristics, and environmental conditions under which the software item is to perform.
- Reconciling differences among requirements that create dependencies or decision points in resolving a design direction for the requirements.
- Prioritizing individual requirements as a basis for deciding among design alternatives.
- Documenting all requirements in the Requirements deliverable.
- Planning subsequent project activities to address requirements.

Quality Control

Interim review criteria include:

- Customer is actively involved in the requirements process, understanding the *definition* of high-level requirements prior to the *specification* of more detailed requirements.
- Requirements focus on the *business* problem, even if the technical direction of the solution is readily apparent.

Exit review criteria include:

- Implementing the requirements, as specified, will solve the problem(s) identified in the allocated service requests.
- Requirements deliverable baselined and placed under configuration management.
- Updated Development Plan includes detail bid and contract activity plans, and at least high-level activities through to project completion.
- Updated Development Plan baselined and placed under configuration management.

Copyright © 1999, D.O.Q.S. All rights reserved.

BID & CONTRACT PHASE

The Bid & Contract Phase seeks a resource solution to the requirements defined and specified in the prior phase.

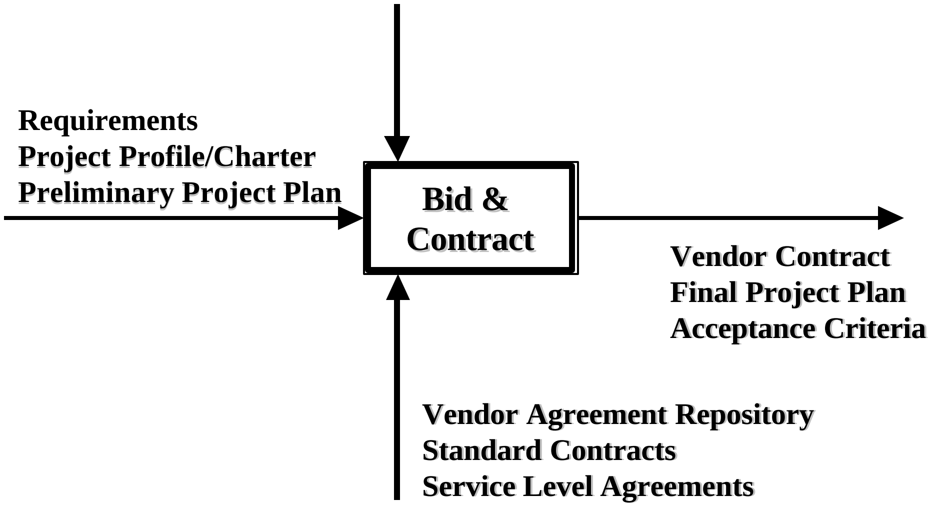


Figure 5 - Bid & Contract Phase Overview

Goals include:

- Assuring that the most effective means are used to complete activity on projects and related activities; particularly with respect to choosing and assigning external vendors to conduct work.
- Assuring that the terms under which vendors conduct activity for projects are under reasonable and favorable terms and conditions.
- Managing vendor relationships in order to reduce project risk across all projects.

This Bid & Contract Phase description includes components derived from the following industry models:

Model	Aspect
ISO 12207	Contract preparation and update (Clause 5.1.3)
IEEE 1062	Preparing contract requirements (Clause 5.5)

Project Activities

Significant activities in the Bid & Contract Phase include:

- Identifying appropriate vendors to carry out detail work on projects.
- Finalizing expected project scope through iterative communication and negotiation with selected vendors and users.
- Determining performance criteria for those vendors; particularly with respect to cost and schedule.
- Defining project acceptance criteria; particularly in terms of acceptance testing plans and product standards.
- Kicking-off vendor activities and subprojects.

Quality Control

Interim review criteria include:

- The request for proposal sent to any vendor shall be subject to internal review and approval before transmission.
- Acceptance plan is mapped to requirements to assure that all requirements are covered in the criteria.

Exit review criteria include:

- Project acceptance criteria include objective criteria for assessing whether or not delivered solutions actually satisfy requirements.
- Project acceptance criteria baselined and placed under configuration management.
- Contractual agreement is in place based on the Development Plan and Requirements scope.
- Updated Development Plan includes detail design and build activity plans, and at least high-level activities through to project completion.
- Updated Development Plan baselined and placed under configuration management.

Copyright © 1999, D.O.Q.S. All rights reserved.

DESIGN & BUILD PHASE

The Design & Build Phase seeks a technical solution that satisfies the defined requirements within the constraint of all contracts; and implements that solution.

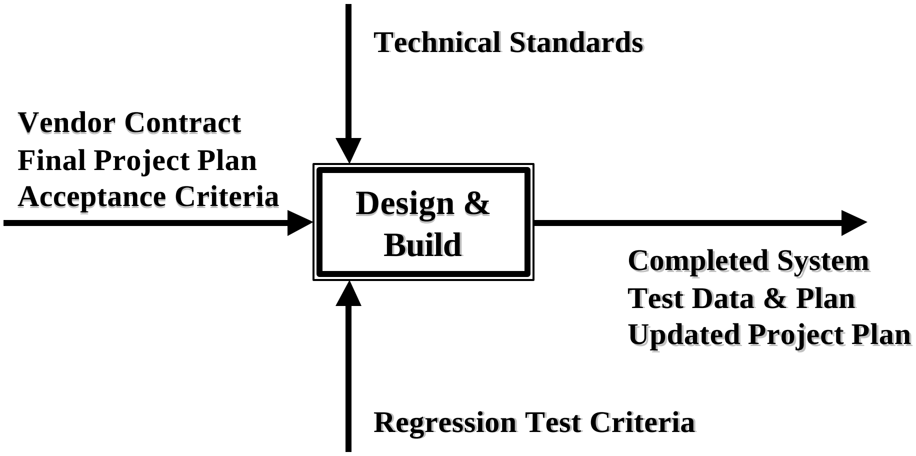


Figure 6 - Design & Build Overview

Goals include:

- Design of the most appropriate solution to meet the requirements.
- Build the designed solution within acceptable quality and cost constraints.
- Assure the built system passes system level testing.
- Assure the solution is properly and completely documented.
- Assure effective turnover to users for acceptance into production.

This Design & Build Phase description includes components derived from the following industry models:

Model	Aspect
ISO 12207	Development process (Clause 5.3)
ISO 9000-3	Contract review (Clause 5.2)
ISO 9000-3	Design and implementation (Clause 5.6)
IEEE 1219	Standard for Software Maintenance

Project Activities

Significant activities in the Design & Build Phase include:

- Coding and unit testing of applicable system components.
- Integration and system testing of applicable system components.
- Documentation of activities and results.
- Coordination of training and conversion planning.

Quality Control

Interim review criteria include:

- Artifacts conform to all applicable standards.
- Appropriateness of the design standards and methods used.
- Feasibility of the solution’s design.
- Feasibility of the operation and maintenance of that design.
- Test strategy is visible in the development process.
- Appropriateness of the test standards and methods used.
- Design is internally consistent with architecture of existing components.
- Design is traceable back to stated business requirements.

Exit review criteria include:

- Adherence to all contract terms for development can be demonstrated.
- If commercial software is involved, all proprietary, usage, ownership, warranty and licensing rights are satisfied.
- Completeness of test coverage during development can be demonstrated.
- Delivered system components baselined and placed under configuration management.
- Delivered test plans and data baselined and placed under configuration management.
- Updated Development Plan includes detail acceptance & implementation activity plans through to project completion.
- Updated Development Plan baselined and placed under configuration management.

ACCEPT & IMPLEMENT PHASE

The Accept & Implement Phase seeks to bring externally developed solutions in-house, assure that those solutions conform to requirements, and implement those solutions in production.

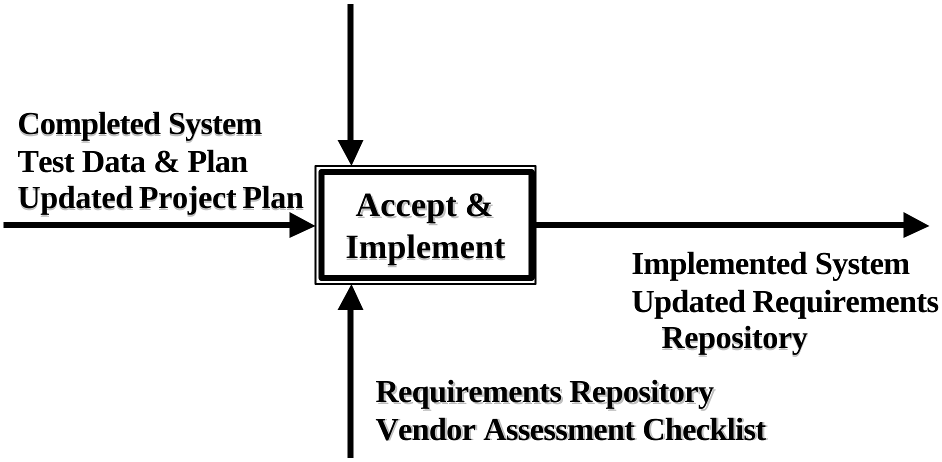


Figure 7 - Accept & Implement Phase Overview

Goals include:

- Assure that the completed solution effectively solves the original service request(s) and requirements.
- Assure that the completed solution is fit for use in user environment.
- Assure that implementation of the completed solution doesn't negatively impact the production environment.
- Assure that knowledge gained in this project is incorporated into the organization's requirements repository.

This Accept & Implement Phase description includes components derived from the following industry models:

Model	Aspect
ISO 12207	Software acceptance support (Clause 5.3.13)
ISO 9000-3	Acceptance (Clause 5.8)
IEEE 1062	Accepting the software (Clause 5.8)

Project Activities

Significant activities in the Accept & Implement Phase include:

- Final integration and system testing of applicable system components.
- User training and documentation integration.
- Acceptance and regression testing of applicable system components.
- Production turnover.
- Project shutdown; particularly update of requirements repository and lessons learned.

Quality Control

Interim review criteria include:

- All completed components work as specified.
- Test coverage is complete and conforms to expected results.
- Use of production implementation demonstrates solution to problems.
- Production implementation operates defect free.

Exit review criteria include:

- Final Requirements version is updated with lessons learned and implementation variances.
- Final Requirements is baselined and placed under configuration management.
- Final Development Plan is completed with all project actuals.
- Final completed Development Plan baselined and placed under configuration management.
- Project-related Contractual Agreements are closed, as appropriate.

KEY PROCESS AREAS

This Project Lifecycle Handbook is specifically tailored to meet the project needs of the organization and its customers, while simultaneously embedding the practices that are considered necessary to drive continuous improvement.

This edition of the lifecycle targets process maturity at the Repeatable Level. It adopts key processes from two distinct Capability Maturity Models® from the Software Engineering Institute:

1. **Capability Maturity Model for Software** (SW-CMM) – The SW-CMM focuses on the process maturity of information systems organizations that manage requirements, run projects, and design and build systems. Its primary focus is on the organization as a *producer* of software.
2. **Software Acquisition Capability Maturity Model** (SA-CMM) – The SA-CMM focuses on many of the same aspects as the SW-CMM except that it focuses on the organization as an *acquirer* of software. Because most of the projects using this lifecycle actually contract out development work; this CMM is more applicable to the core portion of this lifecycle that focuses on vendor contracting, monitoring, and control.

A Key Process Area (KPA) is one of the broad-brush processes that cross the entire lifecycle that are needed to drive such improvement. As such, the list of key processes will change and grow over time as organizational process maturity changes.

REQUIREMENTS ENGINEERING

The purpose of requirements engineering is to establish a common understanding between the customer and the software project of the requirements being addressed.

It is our policy to manage requirements formally through effective elicitation and specification, and to control changes to requirements throughout the project lifecycle and subsequent implementation.

This Requirements Engineering KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Requirements Management KPA
SA-CMM	Requirements Development & Management KPA
ISO 9000-3	Purchaser's requirements specification (Clause 5.3)

Requirements Engineering impacts every phase of this lifecycle:

Phase	Impact
Initiation	• Analysis of impact on previously implemented requirements. • Clustering and prioritizing of newly identified or emerging requirements.
Analysis	• Definition and specification of requirements. • Defining project scope using allocation and deferment of requirements.
Bid & Contract	• Establishment of acceptance criteria for incorporation into project plans and contracts. • Determination of cost and schedule against detailed requirements.
Design & Build	• Control of changes to requirements software engineering. • Verification of traceability and coverage of requirements.
Accept & Implement	• Validation that requirements have been met by implemented solutions. • Incorporation of requirements into organizational memory for on-going re-initiation and prioritization of future service requests.

Copyright © 1999, D.O.Q.S. All rights reserved.

PROJECT PLANNING

The purpose of project planning is to establish reasonable plans for performing software engineering activities and for managing the software project.

It is our policy to formally plan all projects and activities for the most effective utilization of resources and maximization of customer value and satisfaction.

This Project Planning KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Software Project Planning KPA
SA-CMM	Software Acquisition Planning KPA
SA-CMM	Solicitation KPA
PMBOK	Project Management Processes

Project Planning impacts every phase of this lifecycle:

Phase	Impact
Initiation	• Scoping of plan using service requests and expected requirements allocation. • Planning of requirements activities and approaches.
Analysis	• Planning of design alternatives and directions. • Planning of vendor bid and contract activities.
Bid & Contract	• Planning of selected design alternatives and approaches. • Planning of interdependencies between tasks, resources, and organizations.
Design & Build	• Planning of implementation activities.
Accept & Implement	• Closeout of project plans and identification of lessons learned.

PROJECT TRACKING & OVERSIGHT

The purpose of software project tracking and oversight is to establish adequate visibility into actual progress so that management can take effective actions when a software project's performance deviates significantly from the software plans.

It is our policy to formally monitor and control the expenditure of resources on all projects against project delivery targets and plans in order to proactively manage risk and improve delivery.

This Project Tracking & Oversight KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Software Tracking & Oversight KPA
SA-CMM	Project Management KPA

Project Tracking & Oversight impacts every phase of this lifecycle:

Phase	Impact
Initiation	• Evaluate the risk levels associated with the project in order to determine the tolerances that can be carried on this project before requiring management action.
Analysis	• Control of gap between actual and previously expected requirements according to determined tolerances.
Bid & Contract	• Control of gaps between bid and expected costs, schedules, and resources.
Design & Build	• Measuring vendor performance against contract and plan. • Providing visibility into vendor processes and activities for proactive risk and quality management.
Accept & Implement	• Assess final variances to original plan in order to draw conclusions about lessons that can be learned for future projects.

Copyright © 1999, D.O.Q.S. All rights reserved.

SOLICITATION

The purpose of solicitation is to manage the selection of vendors to be used in the completion of project activities, and to initiate the relationship that allows selected vendors to be managed as subcontractors.

It is our policy to identify and form relationships with a variety of external vendors through a rigorous and controlled process of identifying and evaluating candidates against requirements.

This Solicitation KPA is derived from the following industry models:

Model	Aspect
SA-CMM	Solicitation KPA

Solicitation impacts most of the phases of this lifecycle:

Phase	Impact
Initiation	• Recognition of a need for new or changed vendor to fulfill service request(s).
Analysis	• Conducting request-for-information activities, if needed, to further explore new application or technology areas. • Establishing evaluation criteria that will be used to assess vendor inputs.
Bid & Contract	• Issuing requests-for-proposal to candidate vendors. • Evaluating vendor responses. • Negotiating contracts with selected vendors. • Awarding and executing contracts.
Accept & Implement	• Continually assessing vendor performance on projects; including responsiveness of unselected vendors.

SUBCONTRACT MANAGEMENT

The purpose of subcontract management is to select qualified software subcontractors and manage them efficiently.

It is our policy to formally manage all external organization vendors and relationships in order to minimize miscommunication, maximize delivery, and promote win-win interactions among all stakeholders.

This Subcontract Management KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Software Subcontract Management KPA
SA-CMM	Contract Tracking & Oversight KPA
IEEE1062	Managing for supplier performance (Clause 5.7)
ISO 9000-3	Purchaser's management responsibility (Clause 4.1.2)

Subcontract Management impacts most of the phases of this lifecycle:

Phase	Impact
Analysis	• Identification of project stakeholders needed for project continuation. • Definition of needed service levels.
Bid & Contract	• Selection and assignment of vendor and other stakeholders.
Design & Build	• Monitor vendor performance against contracts. • Renegotiate activities and service levels as requirements change.

Copyright © 1999, D.O.Q.S. All rights reserved.

QUALITY ASSURANCE

The purpose of quality assurance is to provide management with appropriate visibility into the process being used by the software project and of the products being built.

It is our policy to maintain maximum visibility into project activities and processes by continually assessing products and services against process and product standards and guidelines, and acting on variances proactively to improve product quality and organization process effectiveness.

This Quality Assurance KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Software Quality Assurance KPA
SA-CMM	Evaluation KPA
SA-CMM	Transition to Support KPA
ISO 9000-3	Quality system (Clause 4.2)
IEEE 1062	Checklists to assist organizations in establishing their own software acquisition process (Annex A)

Quality Assurance impacts every phase of this lifecycle:

Phase	Impact
Initiation	• Assessing proper prioritization and stakeholder buy-in to projects.
Analysis	• Assessing the definition of the problem to be solved and its importance to the business.
Bid & Contract	• Assessing the ability of the project to verify and validate that requirements are being met by the final products. • Assessing the agreement exchanged between project management and vendors for fulfillment of project goals.
Design & Build	• Verifying deliverables against standards.
Accept & Implement	• Validating deliverables against requirements.

CONFIGURATION MANAGEMENT

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

It is our policy to manage and control all artifacts created and maintained by projects in order to assure optimal use of software components, increase project effectiveness, and decrease risk associated with uncontrolled access of different versions of the software asset.

This Configuration Management KPA is derived from the following industry models:

Model	Aspect
SW-CMM	Software Configuration Management KPA

Configuration Management impacts every phase of this lifecycle:

Phase	Impact
Initiation	• Assessing the availability of artifacts needed to address service requests.
Analysis	• Controlling the versioning of requirements documents throughout the project in order to control scope creep.
Bid & Contract	• Checking-out configuration objects for use by the project team.
Design & Build	• Version control over works-in-progress.
Accept & Implement	• Check-in of configuration objects when completed by projects. • Release control on application components going into production.

Artifacts under CM control shall include:

- Requirement definitions and specifications
- Project charters and statements of work
- Project plans
- Service level agreements and contracts
- Analysis and design models/documents
- Program code and parameters
- Test plans and data
- Issue logs and resolution notes

Copyright © 1999, D.O.Q.S. All rights reserved.

KEY LIFECYCLE PRODUCTS

There are myriad deliverables and work products that can, and should, be created by an information technology project over the course of a full project lifecycle. This section details the four key lifecycle products that are considered most critical in achieving a repeatable process capability.

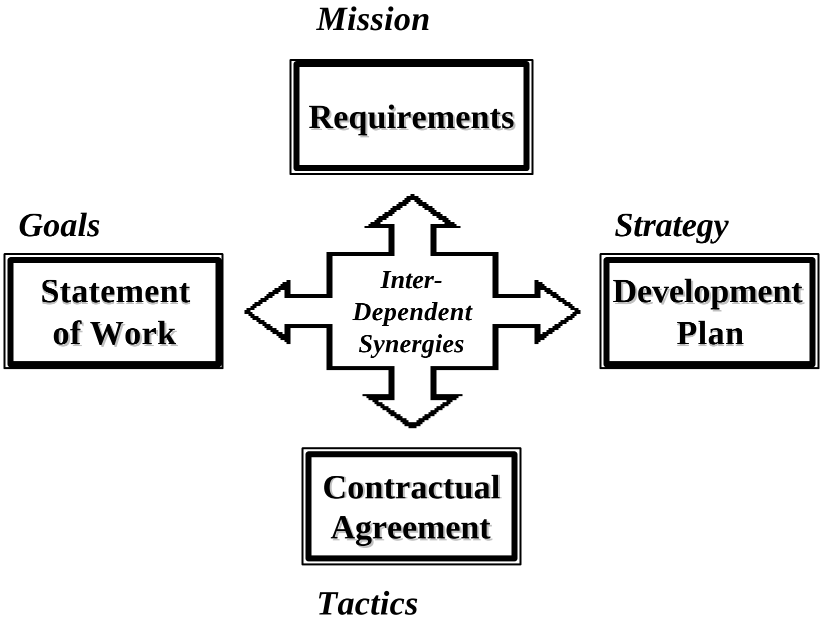


Figure 8 - Relationships Among Key Deliverables

REQUIREMENTS

A deliverable of every project is a set of requirements that are used to manage the project effort, and to incrementally improve and enlarge the production repository of requirements maintained by the organization over time.

This Requirements product is derived from the following industry models:

Model	Aspect
SW-CMM	Requirements Management KPA
SA-CMM	Requirements Development & Management KPA

Contents

The content of a requirements product shall include:

	Completed by the end of….
• A description of the overall business problem or condition that is driving the scope of the project.	Initiation
• A definition of the business requirements, primarily functional and technical, that constitute the scope of the project effort.	Analysis
• A specification of the software requirements that must be satisfied in order to implement the defined requirements.	Analysis
• A description of the non-technical requirements (i.e. agreements, conditions, and/or contractual terms) that affect the project.	Analysis (draft) Bid & Contract (final)
• A description of the acceptance criteria that will be used to validate that requirements have been met prior to implementation of the software products.	Bid & Contract
• A description of the design for the software, including data, algorithms, system features, design patterns, and implementation details.	Design & Build
• A description of the test plan and data for unit, integration, and system testing.	Design & Build
• A description of the information needed to install and support the software in production.	Design & Build (draft) Accept & Impl. (final)

Copyright © 1999, D.O.Q.S. All rights reserved.

Deliverable Form

The physical make-up of a requirements deliverable will vary across projects depending upon the scope of any particular effort, the number of stakeholders involved, and the perceived level of project risk associated with the project.

For typical sized projects within the organization, the requirements deliverable shall be a collection of the following artifacts:

- Service requests that were used to organize and initiate the project,
- Relevant minutes or notes from conversations or communications with the users/customers that initiated the project,
- Copies of e-mail correspondence that was used to communicate requirements to vendors and users,
- Working papers and models developed during interviews, research, and discussions of user needs, and
- Other correspondence and notes among project stakeholders related to project requirements.

Project leaders may choose to consolidate these artifacts into consolidated documents to aid in communication and retention. Larger projects, and small projects operating at higher risk levels, should definitely consolidate these written materials into a single requirements document for review and approval.

Success Factors

A successful Requirements product shall exhibit the following characteristics:

- **Requirements shall be documented.** A reference to a requirement should be to some specific artifact regardless of form, and not to the personal knowledge of one or more project team members or users.
- **Requirements shall be accessible.** The physical consolidation of materials should be sufficient such that all project stakeholders can find and access requirements as needed to perform their responsibilities.
- **Requirements shall be reviewable.** Documentation shall be clear and concise enough that project quality assurance and risk management activities can access and review all requirements in whatever form they are provided.
- **Requirements shall be traceable.** Each discrete requirement shall be documented such that it can be planned and evaluated separately; and can then be tracked through the lifecycle to completion.

STATEMENT OF WORK

A deliverable of every project is a clearly agreed statement of the work to be performed and the resources to be consumed along with documentation of the constraints under which such work is to be performed.

This Statement of Work product is derived from the following industry models:

Model	Aspect
SW-CMM	Software Project Planning KPA
SA-CMM	Software Acquisition Planning KPA

Contents

The content of a Statement of Work shall include:

- Scope of the work to be performed,
- Technical goals and objectives,
- Identification of customers and end users for the project,
- Imposed standards (if any),
- Assigned roles and responsibilities,
- Cost and schedule constraints and goals,
- Dependencies between the project and other projects and stakeholder activities,
- Resource constraints and goals, and
- Other constraints and goals for development and/or maintenance.

Deliverable Form

The form for the Statement of Work will vary based upon the type of work being described and the relationship that exists between and the parties that will conduct work. Statements describing work to be performed by external vendors under existing support contracts shall be treated as addendum to those contracts; and need to be formalized according to the contract terms.

Most project efforts documented in Statements of Work are the result of an iteration of communications between the project leader and the vendor contact. These communications are typically conducted via e-mail, with copies circulated to managers and users as appropriate.

Copyright © 1999, D.O.Q.S. All rights reserved.

Success Factors

A successful Statement of Work shall exhibit the following characteristics:

- **Statements of Work shall be documented.** Verbal or implied agreements regarding the scope of work to be carried out are not acceptable under any circumstances.
- **Statements of Work shall be reciprocal.** The work effort and responsibilities of all project stakeholders should be included in the work description. It should *not* simply be a list of what a software producer, internal or external, should do. Organizational activities and constraints should be included as well.
- **Statements of Work shall be clear and concise.** All project stakeholders should agree with the Statement of Work. Ambiguities shall be resolved and corrected before any project work is performed.

DEVELOPMENT PLAN

A deliverable of every project is a development plan that clearly lays out both the strategic and tactical approach that the project is taking to performing the work described in the statement of work.

This Development Plan product is derived from the following industry models:

Model	Aspect
SW-CMM	Software Project Planning KPA
SA-CMM	Software Acquisition Planning KPA
SA-CMM	Project Management KPA
ISO 9000-3	Development plan (Clause 5.4.2)

Contents

The content of a Development Plan shall include:

- Project's purpose, scope, goals, and objectives,
- Description of how this lifecycle is being tailored to meet project needs,
- Identification of the software work products that will be developed or maintained by the project,
- Estimates of the work products and effort in terms of size, effort, cost, critical resources (if any).
- Identification of the project schedule, including milestones and reviews,
- Identification of the project's risks and planned mitigation strategies, and
- Plan's for the project's working environment.

Deliverable Form

The form for a Development Plan will vary based on the expected duration of the project, and the perceived risk levels associated with the activity. Acceptable plans include simple task lists with resource assignments on small projects, up to much more detailed and thorough plans on on-going or major projects. For more detailed projects, automated tool support (e.g. Microsoft Project) is preferred.

Copyright © 1999, D.O.Q.S. All rights reserved.

Success Factors

A successful Development Plan shall exhibit the following characteristics:

- **Development Plans shall be documented.** A written plan facilitates management and peer review, and allows for proper configuration management. Documenting the plan also forces the project leader to think through all details to completion, avoiding the pitfalls of conducting only short-term planning.
- **Development Plans shall be appropriately detailed.** The plan breaks down the phases, tasks, and activities of the project. The lowest level of this hierarchy can then be planned with different resources, priorities, and risk mitigation strategies. Planned activities need to be detailed enough to support such differences. Plans need not be more detailed.

CONTRACTUAL AGREEMENT

A deliverable of every project that uses resources outside of its direct control is a contractual agreement that clearly delineates the mutual responsibilities and obligations of parties to the project effort.

This Contractual Agreement product is derived from the following industry models:

Model	Aspect
SW-CMM	Software Subcontract Management KPA
SA-CMM	Solicitation KPA
SA-CMM	Contract Tracking & Oversight KPA

Contents

The content of the Contractual Agreement shall include:

- The terms and conditions for the work on the project,
- The Statement of Work (can be incorporated by reference),
- The Requirements (can be incorporated by reference),
- A list of dependencies between the vendor and contracting project.
- The products to be delivered by the vendor.
- The conditions under which any revisions to the requirements or products are to be submitted to/from the project.
- The acceptance provisions and criteria to be used in evaluating the products and services delivered under the terms, and
- The evaluation criteria that will be applied to the vendor in assessing compliance and satisfaction with contracted services.

Copyright © 1999, D.O.Q.S. All rights reserved.

Deliverable Form

The form that a Contractual Agreement shall take depends upon the vendor selected to perform project work, and on the organization's relationship with that vendor.

The majority of vendor engagements that fall under this project lifecycle represent preferred vendor circumstances where the vendor that provided a software package or application is retained in order to perform some enhancement or maintenance effort. In those cases, a standard service level agreement or contract exists under which work is performed. Individual projects are submitted to the vendor for detail work plan and cost estimates as though they were to be treated as an addendum to the existing relationships. These interactions are typically conducted directly between the project leader and vendor contact via e-mail, with organization management copied as appropriate. Completed/confirmed estimates become part of the project documentation under the control of the project manager.

For larger projects, or where the choice of a vendor is less automatic, a standard contracting process should be used, including appropriate legal support and guidance.

Success Factors

A successful Contractual Agreement shall exhibit the following characteristics:

- **Contractual Agreements shall be documented.** All terms and expectations of the contract should be written clauses of the contract itself, or the documentation and addendum incorporated therein. Avoid unwritten verbal agreements, or agreements based upon the knowledge of specific persons involved in the contracting process.
- **Contractual Agreements shall be authorized.** Contracts of any kind place liabilities and obligations on all parties. Project managers should take care to avoid incurring unexpected or accidental liabilities and obligations. All contracts should be reviewed and executed by management under visible and informed circumstances.
- **Contractual Agreements shall be controlled.** Because the contract defines a relationship with an external party, the management of versions of the contract can not be overemphasized. The contract itself, as well as all correspondence that might be construed as modifying the contract, shall be placed under strict configuration management.

QUALITY CONTROL

Each phase of this lifecycle has a quality control checklist that can be used to quantify the level of compliance to the goals and quality criteria contained in this Handbook. Results of each phase checklist, along with a general project-level checklist, are combined into a scorecard for the project.

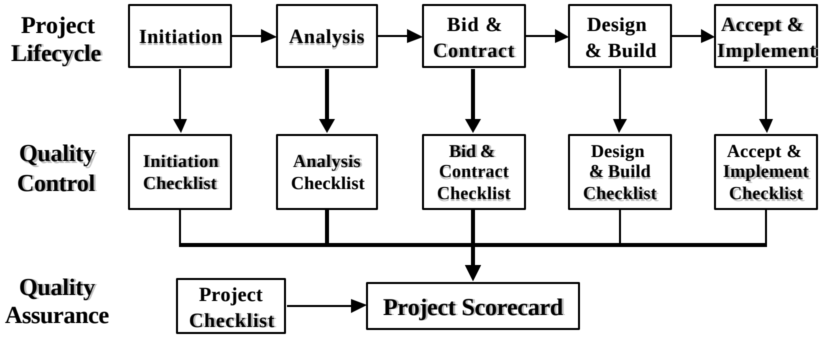


Figure 9 - Quality Control/Assurance Interactions

Project scorecard results can be used to compare multiple projects and establish a baseline and trend for improvement actions. At the project level, scorecard results indicate relative performance to other projects. At the organization level, scorecard trends indicate process capability and maturity.

Copyright © 1999, D.O.Q.S. All rights reserved.

PROJECT SCORECARD

Project Name: ________________________________ Date: ________________

This scorecard can be completed any time a project assessment is conducted using any or all of the project lifecycle phase checklists. Trend analysis of these scorecards over the life of the project should show continuously increasing values.

PROJECT CHECKLIST	TOTAL SCORE (0-100)	LOWEST SCORE (0-10)	HIGHEST SCORE (0-10)
Overall			
Initiation Phase			
Analysis Phase			
Bid & Contract Phase			
Design & Build Phase			
Accept & Implement Phase			
Total:			

Notes:

PROJECT CHECKLISTS

Overall Project Checklist

Project Name: ______________________________________ Date: ________________

This checklist can be completed at any point in the project, particularly when a major milestone has been reached. The questions presume that traditional project management activities are taking place on the project, and that some form of Project Plan is being produced and maintained.

Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1. Does the project have a project plan that includes details for near-term activities, and at least high-level activities through to the end of the project?	
2. Is the project keeping all project stakeholders appropriately informed and involved in the project?	
3. Is the project basing its estimation and commitments on the requirements and work to be performed, without allowing desired target dates to be the sole determining criteria?	
4. Can the project demonstrate that all appropriate terms and conditions are being met in all project contracts and agreements?	
5. Has the project assessed all reasonably possible risks and developed mitigation strategies for minimizing or avoiding their affects?	
6. Has the project received, clarified, and prioritized all user requests related to the proposed project scope; and planned project activity based on those factors? *(if not, see Initiation Phase)*	
7. Has the project clarified the problem and business case for the project, and provided for the specification of the functional and technical requirements to be met? *(if not, see Analysis Phase)*	
8. Has the project identified, costed, scheduled, and contracted the resources necessary to implement the defined requirements? *(if not, see Bid & Contract Phase)*	
9. Has the project designed and built the application systems and components necessary to implement the defined and contracted requirements? *(if not, see Design & Build Phase)*	
10. Has the project conducted validation and acceptance of all work products, and implemented provided software solutions into production? *(if not, see Accept & Implement Phase)*	
Total Score:	

Notes:

Initiation Phase Checklist

Project Name: ______________________________________ Date: ________________

The Initiation Phase clarifies the scope and business case for the project, and identifies the resources and planning issues associated with conducting the project.

This checklist can be completed at any point in the project, particularly at the end of the Initiation Phase, or else any major checkpoint where significant replanning activity has been completed. The questions presume that project organization is taking place on the project, and that some form of Project Charter/Plan is being produced, even if informally.

Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1. Has the project identified and clarified the scope of the intended effort to a sufficient level of detail to identify any overlap with other projects and project high level resource needs?	
2. Has the project identified all intended project stakeholders, both customer and suppliers, and involved them in the initial planning of all efforts?	
3. Has initiation determined the funding strategy for the project and negotiated with the organization to obtain the necessary funding for the project when initiated?	
4. Has the project developed a project plan that includes detail requirements activity plans, and at least high-level activities through to project completion?	
5. Has the project determined that all system components needed are available, and that such availability doesn't interfere with any other active projects?	
6. Has the project developed a project plan sufficient to assure conformance to all objectives of the anticipated scope, without interfering with other projects already underway?	
7. Has the project determined that the project leader and staff can handle addition of this project to their workload without creating an excessive burden?	
8. Has the project evaluated the risk levels associated with the project in order to determine the tolerances that can be carried on this project before requiring management action?	
9. Has the project assessed the priority of the work effort, taking into account any known or anticipated dependencies between this and other projects?	
10. Has the projects clearly identified and documented any known cost, schedule, resource, or standards constraints on the project?	
Total Score:	

Notes:

Analysis Phase Checklist

Project Name: ______________________________________ Date: ________________

The Analysis Phase seeks and documents the business problems to be solved by the project based upon the collection of requests that were grouped and initiated in the prior phase.

This checklist can be completed at any point in the project, particularly at the end of the Analysis Phase, or else any major checkpoint where significant analysis activity has been completed. The questions presume that analysis is taking place on the project, and that some form of Requirements Deliverable is being produced, even if informally.

Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1. Has analysis defined and clarified the *business* problem to be solved and the rationale that determines the importance and value of the solution to the business?	
2. Has analysis described *what* needs to happen in order to solve the defined problem; and the criteria that should be used to determine if the project reaches successful implementation?	
3. Has analysis included data requirements, including new data elements, changes to data definitions, or new or revised relationships among data?	
4. Has analysis included process requirements, including revision to the triggers that cause events to happen, or new steps or sequences of steps that are required in the business?	
5. Has analysis included interface requirements, including human interfaces and system-to-system interfaces among new, revised, and legacy systems?	
6. Has analysis included security specifications, including those related to any potential compromise of sensitive information and end-user privacy?	
7. Has analysis included reconciling differences among requirements that create dependencies or decision points in resolving a design direction for the requirements?	
8. Has analysis included prioritizing or clustering requirements as a basis for deciding among possible design alternatives?	
9. Has analysis been included of any impact this project may have on previously implemented requirements?	
10. Has the project documented and versioned the requirements to a sufficient level of detail that changes to requirements or expansion of scope will be quickly visible?	
Total Score:	

Notes:

Copyright © 1999, D.O.Q.S. All rights reserved.

Bid & Contract Phase Checklist

Project Name: ______________________________________ Date: ________________

The Bid & Contract Phase seeks a resource solution to the defined requirements, managing vendor relationships in order to reduce project risk.

This checklist can be completed at any point in the project, particularly at the end of the Bid & Contract Phase, or else any major checkpoint where significant analysis solicitation or sourcing has been completed. The questions presume that contracting is taking place on the project, and that some form of Statement of Work is being produced.

Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1. Has the project identified appropriate vendors to carry out work on the project so that all planned work is accounted for?	
2. Has the project determining performance criteria for all vendors; particularly with respect to cost, quality, and schedule?	
3. Has the project defining project acceptance criteria; particularly in terms of acceptance testing plans and product technical standards?	
4. Has the project mapped the acceptance plan to the documented requirements to assure that all requirements are covered in the criteria?	
5. Has the project assessed the agreement exchanged between project management and any vendors to assure the fulfillment of all project goals?	
6. Has the project clearly identified and planned all products to be delivered by any vendor?	
7. Has the project clearly identified and prioritized a list of dependencies between any vendor and other contracting projects?	
8. Has the project clearly defined the conditions under which any revisions to the requirements or products are to be submitted, either to or from any vendors?	
9. Has the project clearly defined the acceptance provisions and criteria to be used in evaluating the products and services delivered under any agreements?	
10. Has the project clearly defined the evaluation criteria that will be applied to the vendor in assessing compliance and satisfaction with contracted services?	
Total Score:	

Notes:

Design & Build Phase Checklist

Project Name: ______________________________________ Date: ________________

The Design & Build Phase seeks a technical solution that satisfies the defined requirements within the constraint of all contracts; and implements that solutions.

This checklist can be completed at any point in the project, particularly at the end of the Design & Build Phase, or else any major checkpoint where significant design/build activity has been completed. The questions presume that design and build are taking place on the project, and that some form of Specification and System is being produced.

	Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1.	Has the project developed multiple design scenarios and alternatives in order to select the most appropriate design for the project?	
2.	Has the project team used the most appropriate design standards and methods?	
3.	Has the project embedded a visible and integrated test strategy into the design and development process?	
4.	Has the project developed a design that is consistent with the architecture and design of related products on other projects and in the environment?	
5.	Has the project traced all design functions and features back to the requirements documented for the project, and does it adhere to the prioritization of those requirements?	
6.	Has the project built the software solutions within acceptable cost and quality constraints?	
7.	Has the project demonstrated effective version control over all components of any systems produced as part of the project?	
8.	Has the project demonstrated the completeness of the test coverage used during development?	
9.	Has the project developed an appropriate training and conversion plan to be used when the solution systems are implemented into production?	
10.	Has the project satisfied any proprietary, usage, ownership, warranty, or licensing rights for all software or work products included in the project?	
	Total Score:	

Notes:

Accept & Implement Phase Checklist

Project Name: ______________________________________ Date: ________________

The Accept & Implement Phase seeks to bring developed solutions in-house, assure that those solutions conform to requirements, and implement those solutions in production.

This checklist can be completed at any point in the project, particularly at the end of the Accept & Implement Phase, or else any major checkpoint where significant turnover activity has been completed. The questions presume that acceptance is taking place on the project, and that some form of verification and validation is being conducted.

Score should range from 0 (No) to 10 (Yes).	Score (0-10)
1. Has the project assured that the completed solution effectively solves the original business problems and satisfies all allocated requirements?	
2. Has the project assured that the completed solution is fit for use in the typically anticipated user/customer environment?	
3. Has the project assured that implementation of the completed solution doesn't negatively impact any production environment or other operational system?	
4. Has the project completed all final integration and system testing of applicable system components?	
5. Has the project completed all needed training and documentation integration?	
6. Has the project completed all acceptance and regression testing of applicable system components?	
7. Has the project assured that test coverage is complete and conforms to all expected results?	
8. Has the project assured that the production implementation operates defect free?	
9. Has the project demonstrated that use of production implementation actually demonstrates a solution to the initial problems?	
10. Has the project demonstrated that all contract related issues have been satisfied and that all vendors may be released from the project?	
Total Score:	

Notes:

SUPPORTING MATERIALS

SEI Capability Maturity Models

SA-CMM	Software Acquisition Capability Maturity Model (SA-CMM) Version 1.01; Technical Report CMU/SEI-96-TR-020.
SW-CMM	Key Practices of the Capability Maturity Model, Version 1.1; Technical Report CMU/SEI-93-TR-25.

Quality Management Criteria

ISO 9001	Quality Systems - Model for Quality Assurance in Design, Development, Production, Installation, and Servicing; ANSI Q9001-1994.
ISO 9000-3	Quality Management and Quality Assurance Standards – Guidelines for the Application of ANSI-Q9001 to the Development, Supply, and Maintenance of Software; ANSI Q9000-3-1991.

Project Management Criteria

PMBOK	A Guide to the Project Management Body of Knowledge, Project Management Institute Standards Committee.
IEEE 1058	IEEE Standard for Software Project Management Plans, Std 1058-1994.

Software Engineering Criteria

ISO 12207	Industry Implementation of International Standard ISO/IEC 12207:1995; (ISO/IEC 12207) Standard for Information Technology – Software Lifecycle processes; IEEE/EIA 12207.0-1996
	Supported by…
IEEE 830	IEEE Recommended Practice for Software Requirements Specifications, Std 830-1996.
IEEE 1062	IEEE Recommended Practice for Software Acquisition, Std 1062-1993.
IEEE 1219	IEEE Standard for Software Maintenance; Std 1219-1993.
IEEE 1233	IEEE Guide to Preparing System Requirements Specifications, Std 1233-1995.

SW-CMM LEVELS 1 TO 2[3]

INITIAL (LEVEL 1)	REPEATABLE (LEVEL 2)
When this column rings true, work to get to Level 2…	This column should ring true when using this Handbook…
Characteristics of Organizations at Level	
The software process is characterized as ad hoc, and occasionally even chaotic. Few processes are defined, and success depends on individual effort.	Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.
Behaviors of Organizations at Level	
The organization typically does not provide a stable environment for developing and maintaining software. The benefits of good software engineering practices are undermined by ineffective planning and reaction-driven commitment systems. During a crisis, projects typically abandon planned procedures and revert to coding and testing. Success depends entirely on having an exceptional manager and a seasoned and effective software team. Occasionally, capable and forceful software managers can withstand the pressures to take shortcuts in the software process; but when they leave the project, their stabilizing influence leaves with them. Even a strong engineering process cannot overcome the instability created by the absence of sound management practices.	Policies for managing a software project and procedures to implement those policies are established. An objective is to institutionalize effective management processes for software projects, which allow organizations to repeat successful practices developed on earlier projects, although the specific processes implemented by the projects may differ. An effective process can be characterized as practiced, documented, enforced, trained, measured, and able to improve. Realistic project commitments are based on the results observed on previous projects and on the requirements of the current project. The software managers for a project track software costs, schedules, and functionality; problems in meeting commitments are identified when they arise. Software project standards are defined, and the organization ensures they are faithfully followed.
Summary of Organizations at Level	
Process capability is unpredictable because the software process is constantly changed or modified as the work progresses (i.e., the process is *ad hoc*). Schedules, budgets, functionality, and product quality are generally unpredictable. Performance depends on the capabilities of individuals and varies with their innate skills, knowledge, and motivations. There are few stable software processes in evidence, and performance can be predicted only by individual rather than organizational capability.	The software process capability of Level 2 organizations can be summarized as disciplined because planning and tracking of the software project is stable and earlier successes can be repeated. The project's process is under the effective control of a project management system, following realistic plans based on the performance of previous projects.

[3] This table contains excerpts taken directly from various places in *Key Practices of the Capability Maturity Model, Version 1.1*; (Technical Report CMU/SEI-93-TR-25).

SW-CMM LEVEL 2 KPA GOALS

Requirements Management

System requirements allocated to software are controlled to establish a baseline for software engineering and management use.

Software plans, products, and activities are kept consistent with the system requirements allocated to software.

Software Project Planning

Software estimates are documented for use in planning and tracking the software project.

Software project activities and commitments are planned and documented.

Affected groups and individuals agree to their commitments related to the software project.

Software Project Tracking and Oversight

Actual results and performances are tracked against the software plans.

Corrective actions are taken and managed to closure when actual results and performance deviate significantly from the software plans.

Changes to software commitments are agreed to by the affected groups and individuals.

Software Subcontract Management

The prime contractor selects qualified software subcontractors.

The prime contractor and the software subcontractor agree to their commitments to each other.

The prime contractor and the software subcontractor maintain ongoing communications.

The prime contractor tracks the software subcontractor's actual results and performance against its commitments.

Software Quality Assurance

Software quality assurance activities are planned.

Adherence of software products and activities to the applicable standards, procedures, and requirements is verified objectively.

Affected groups and individuals are informed of software quality assurance activities and results.

Noncompliance issues that cannot be resolved within the software project are addressed by senior management.

Software Configuration Management

Software configuration management activities are planned.

Selected software work products are identified, controlled, and available.

Changes to identified software work products are controlled.

Affected groups and individuals are informed of the status and content of software baselines.

Copyright © 1999, D.O.Q.S. All rights reserved.

INDEX