

PROJECT MANAGEMENT

CXONE STANDARD

CXSTAND_PROJECTMANAGEMENT.DOC

NOVEMBER 3, 2002

Construx
SOFTWARE



Advancing the Art and
Science of Commercial
Software Engineering

CONTENTS

1 INTRODUCTION	1
1.1 OVERVIEW	1
1.1.1 <i>What is a Project?</i>	1
1.1.2 <i>CxOne Project Management</i>	1
1.2 BACKGROUND	2
1.3 SUPPORTING OTHER STANDARDS.....	2
2 CKA ORGANIZATION.....	3
2.1 PROJECT	3
2.2 CHARTERING	3
2.3 PLANNING	3
2.4 ESTIMATION	3
2.5 TRACKING AND CONTROL	3
2.6 ISSUE MANAGEMENT	3
2.7 RISK MANAGEMENT.....	3
2.8 TEAM STRUCTURE.....	3
3 PROJECT CHARTER	4
3.1 PURPOSE	4
3.2 OVERVIEW	4
3.3 RELATIONSHIP TO OTHER PROJECT ARTIFACTS	4
4 PROJECT PLAN.....	5
4.1 PURPOSE	5
4.2 OVERVIEW	5
4.2.1 <i>Creating the Project Plan</i>	5
4.2.2 <i>Software Lifecycles and Project Planning</i>	5
4.2.3 <i>The Project Plan as a Model</i>	5
4.3 CONTENT.....	6
4.3.1 <i>Project Plan Packaging</i>	7
4.3.2 <i>Groups of Project Plans</i>	7
4.4 RELATIONSHIP TO OTHER ARTIFACTS	7
4.4.1 <i>Project Charter</i>	7
4.4.2 <i>Project Artifacts</i>	7
4.5 REUSE AND THE PROJECT PLAN	7
5 ESTIMATION.....	8
5.1 PURPOSE	8
5.2 OVERVIEW	8

5.2.1 Estimates, Targets, and Control	8
5.2.2 Uncertainty in Estimates.....	9
5.2.3 Macro and Micro Estimation Techniques	9
5.2.4 Counting, Computing, Judging.....	9
5.2.5 Use of Multiple Techniques	9
5.2.6 Phased Estimation	9
6 CORRECTIVE ACTIVITY MANAGEMENT	10
6.1 PURPOSE	10
6.2 OVERVIEW	10
6.2.1 CAM Items	10
6.2.2 A PMBOK view of CAM	11
6.2.3 Using CAM to Manage the Project.....	11
6.2.4 Using Database Tools with CAM	11
6.3 CXONE CAM MATERIALS	12
7 TEAM STRUCTURE.....	13
7.1 PURPOSE	13
7.2 OVERVIEW	13
7.2.1 Standard Project Positions	14
7.2.2 Custom Project Positions.....	14
7.3 EXTERNAL PROJECT ROLES.....	15

Copyright Notice

© 2000-2002 Construx Software Builders, Inc. All Rights Reserved.

For further information or support visit www.construx.com.

1 INTRODUCTION

This standard documents CxOne’s organization of project management practices, knowledge, artifacts, and materials.

1.1 Overview

Project management includes the inception, estimation, and planning of a project along with tracking, controlling, and coordinating the project’s execution. CxOne’s goal for project management is to strike an optimal balance between planning and execution.

Chartering starts the project. Planning defines the lifecycle and provides blueprints for executing the project. Estimation ensures work plans and targets are feasible. Tracking and control practices manage the planned work. Corrective activity management handles work not explicitly planned. Risk management occurs both explicitly and implicitly as part of all project decisions and activities.

1.1.1 What is a Project?

CxOne adopts the PMBOK’s description of a project as:

A temporary endeavor undertaken to create a unique product or service.

CxOne supports projects that include software development. Some projects will consist almost exclusively of software development, while others will produce systems in which software plays one or more roles. Project size may span from a few staff-hours to hundreds of staff-years. Software projects may support products or services to a wide variety of domains.

1.1.2 CxOne Project Management

CxOne defines project management as the ongoing planning and directing of work on a software project. As the diagram below illustrates, CxOne makes a broad simplifying distinction between planned and unplanned work, and assumes both must be actively managed for projects to succeed. Unplanned work represents unforeseen work or work that is not precisely or explicitly planned ahead of time (see section 6, *Corrective Activity Management*).

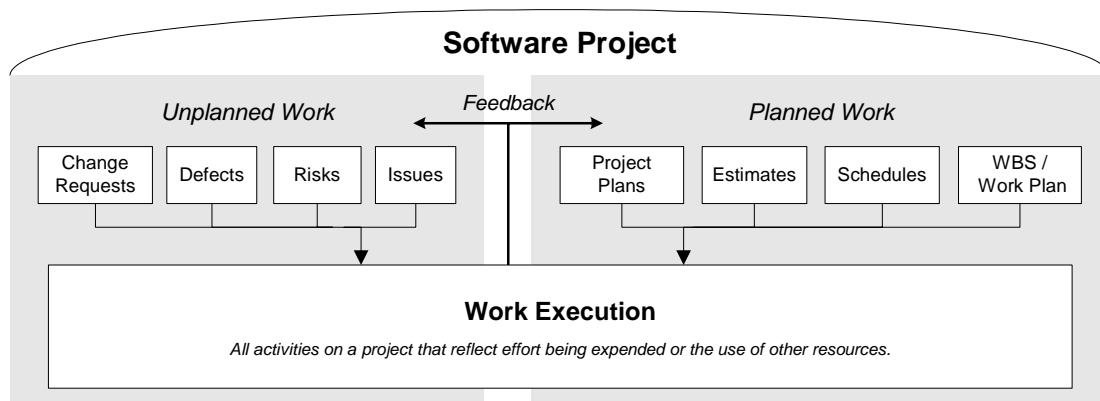


Figure 1-1: CxOne Project Work Model

1.2 Background

CxOne project management materials are synthesized from many sources including the Steve McConnell's works, Construx project experience, PMBOK, SWEBOK, IEEE standards, other industry sources and literature. CxOne organization, terminology, and packaging of project management knowledge and deliverables provide a complete, readily scalable, and easy to use distillation of project management concepts.

1.3 Supporting Other Standards

CxOne project management materials support the following standards:

Steve McConnell's Books

CxOne's project management materials support the concepts of McConnell's *Software Project Survival Guide* and *Rapid Development*.

Rational Unified Process (RUP)

CxOne project management materials are compatible with the project lifecycle, concepts, and methods defined by Rational's Unified Process.

Agile Movement (including Extreme Programming - XP)

CxOne principles and materials may be employed to bolster projects utilizing Extreme Programming or other techniques typically associated with "agile methodologies".

CMM and CMMI

CxOne project management materials provide value to organizations at any level of the CMM and CMMI (staged). If you are at CMM levels 1 or 2, these materials provide a framework for getting projects to CMM level 3 and higher. If using CMMI (continuous) see 15504.

ISO/IEC 15504 (SPICE)

CxOne project management materials provide value to organizations at any level of 15504 management practices. CxOne provides a framework for level 3+ performance in the management process category (MAN).

PMBOK

CxTemp_ProjectPlan supports the knowledge areas and processes of the PMBOK.

SWEBOK

CxTemp_ProjectPlan and *CxTemp_ProjectCharter* cover the SWEBOK's project management areas.

IEEE Software Standards

CxTemp_ProjectPlan supports IEEE 1058 Software Project Management Plan and IEEE/EIA 12207 (also ISO/IEC 12207) Software Project Lifecycle standards.

DOD-178B

CxOne materials support management of projects for creation of level A software.

2 CKA ORGANIZATION

Project management is a deep area of CxOne. This section provides a brief overview of the project management folders in the project management sub-CKA.

2.1 Project

This is the root folder for all engineering management materials focused on managing projects, as opposed to materials focused on organizational or external management.

2.2 Chartering

Starting a project, focused on the project charter. See section 3 for more details.

2.3 Planning

Includes project plan materials along with business schedule, WBS, and other top-level planning support. See section 4 for more details.

2.4 Estimation

Materials supporting estimation activities and artifacts.

2.5 Tracking And Control

Tracking and control of a project's planned work. Includes work planning, detailed scheduling, earned value management, and status reporting materials.

2.6 Issue Management

Materials to support management of unplanned project issues. See section 5 for more details.

2.7 Risk Management

Risk management is CxOne's central project management activity. Most risk management in CxOne is *intrinsic*, i.e., performed as part of other processes. Materials to support *extrinsic* project risk management are contained here. See section 5 for more details.

2.8 Team Structure

A defined team structure provides efficient staff collaboration. CxOne provides a template for project team structure and detailed checklists for different project roles. See section 7 for more details.

3 PROJECT CHARTER

The project charter is used to inception the project, define the project, control project inception activities, and delegate authority to execute the project.

This section provides an overview of CxOne project charter concepts and terminology. For more information, see the CxOne chartering materials and *CxOneTermsAndAcronyms*.

3.1 Purpose

Inception can be a long, poorly defined, and inefficient phase of a project. McConnell coined the phrase “fuzzy front end” to describe the period between a project being an idea in someone’s head, to being truly staffed and underway. Time and other resources lost at the front of a project are just as valuable as those lost at the end. Refining project inception can be a huge win for reducing overall project schedules.

The project charter provides a vehicle to move a project through the fuzzy front end as efficiently as possible.

3.2 Overview

CxOne defines the start of a project as the creation of the project charter. The creation of the charter is usually an iterative process that involves starting some project work to investigate opportunities, explore risks, test feasibility, and create initial plans. Although the charter is a living artifact that should be updated as necessary during the project, there should be a clear point at which the charter phase is considered complete and the actual project begins.

See *CxGuide_ProjectCharter* and other CxOne project charter materials for more details.

3.3 Relationship to other Project Artifacts

During the charter phase, work will often occur to start drafting a project plan along with work on requirements and feasibility prototyping. Often several refinements of a project charter are necessary to refine the business case, build consensus among stakeholders, and win approval for the project.

4 PROJECT PLAN

The project plan is the top-level controlling artifact for a project. CxOne provides materials for creating project plans and supporting artifacts such as estimates, work plans, schedules, and other specific plans.

This section provides an overview of CxOne project planning concepts and terminology. For more information, see the CxOne planning materials and *CxOneTermsAndAcronyms*.

4.1 Purpose

The project plan defines how a project will accomplish its goals. This includes defining the details of those goals (i.e., feature, schedule, budget, quality, etc.), mechanisms for planning, tracking, and controlling project work, as well as details of how the work will be carried out (i.e., lifecycle, staffing, technical processes, etc.).

4.2 Overview

The term “project plan” refers to the root artifact from which all other project planning flows. A project plan is often a document, and will often delegate detailed descriptions to other artifacts (e.g., a project plan referring to a CM plan, quality plan, business schedule, etc.).

4.2.1 Creating the Project Plan

CxOne requires that a project plan be created, that it is a living plan, and that the plan matches the reality of the project. There is no cookie-cutter for creating project plans; the precise nature and content of project plans can vary widely between projects. CxOne defines the areas a project plan needs to cover to ensure project success. The CxOne project plan template and supporting materials guide the creation of your project’s planning blueprint.

4.2.2 Software Lifecycles and Project Planning

Software development lifecycles are closely intertwined with project planning. CxOne defines standard project lifecycles in the process CKA. The selection and customization of lifecycles for projects is part of project management.

4.2.3 The Project Plan as a Model

A model is an abstract representation of a problem space. Modeling is used implicitly in many software activities, and often used explicitly in requirements and design. CxOne encourages explicit consideration of modeling when addressing project management challenges. One of the ways to think of project planning is the creation of models that will be used to understand and control the project.

4.3 Content

The key elements of a project plan are shown below. “Software Development Plan” is often used as a synonym for “Project Plan” on software focused projects.

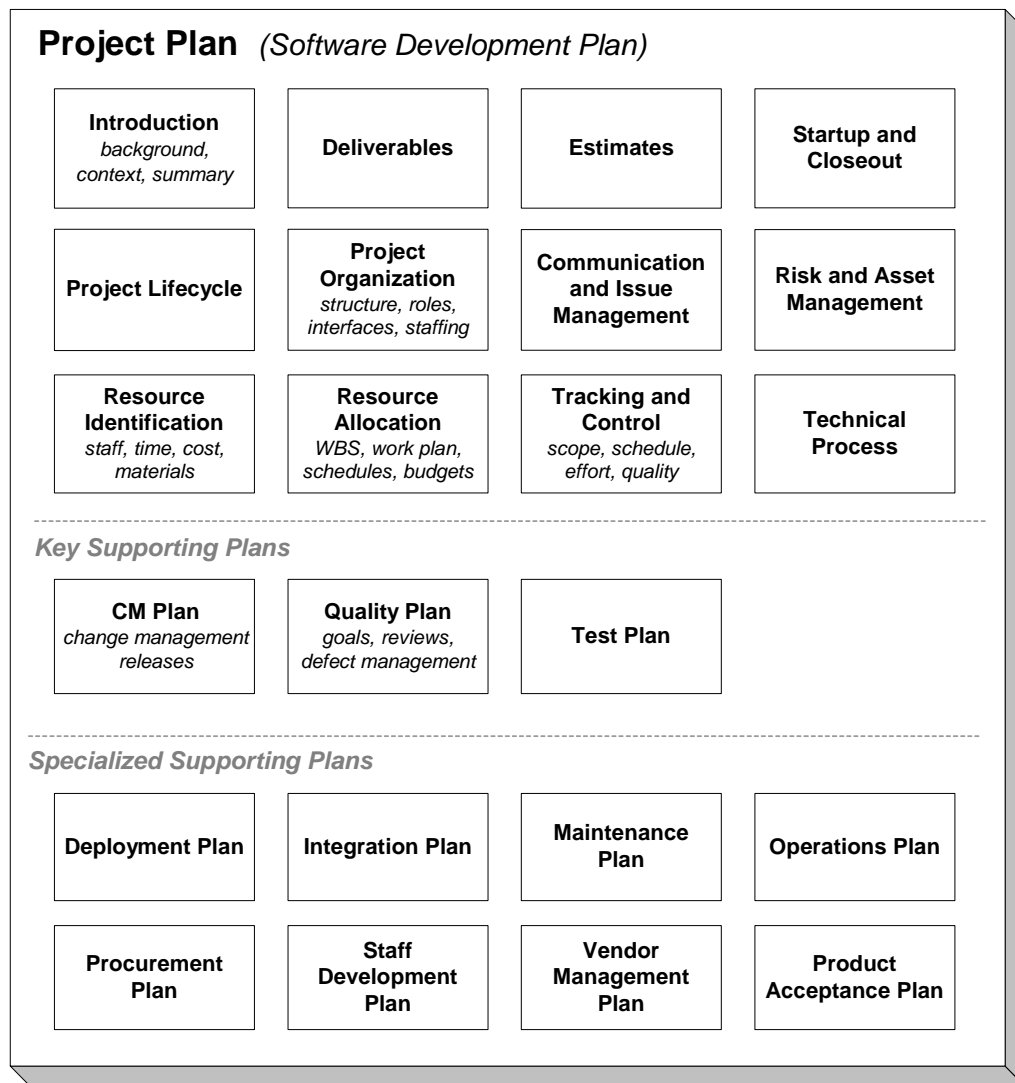


Figure 4-1: CxOne Project Plan Content

CxOne requires that the areas shown in Figure 4-1 be addressed by project planning. Due to the varied needs of projects, the precise content, presentation, packaging, and organization of a project plan is not dictated by CxOne. Rather a set of tools are provided that support creation and execution of project plans. Projects should aggregate, repackage, adapt, and extend CxOne materials to best meet their needs.

4.3.1 Project Plan Packaging

Depending on the nature and scope of a project, the content of a project plan may be contained in one or many documents. The top half of the project plan structure diagram indicates elements of a project that are usually part of the project plan itself. The bottom half (boxes with the word “plan” in them) indicate subjects that normally have stand-alone plans created.

The smallest projects (several staff days or less) may have a single page document covering all aspects of a project. Very large projects (20 staff years or more) may have planning captured in many artifacts. Most projects will have a balance between these two extremes.

Every project or group of projects should have a singular root artifact from which all project planning flows.

4.3.2 Groups of Project Plans

Projects may be decomposed into a hierarchy or web of subprojects, creating a hierarchy or web of project plans. A project plan should clearly describe its relationship to any other peer-, parent-, or child-projects.

Plans should avoid content duplication and must be consistent with each other. Groups of projects are often organized into a hierarchy. When traversing down a hierarchy the detail should increase. Higher level plans should summarize the plans below them. In a web structure, links between projects should be made clear.

4.4 Relationship to Other Artifacts

4.4.1 Project Charter

The project plan flows directly from the project charter. Iterations for a project plan are often begun as part of the charter process. The charter ultimately defines the scope, goals, and resources for a project. The project plan is a detailed description of how the charter is to be executed. The project plan and charter should always stay in sync. If the project plan doesn't have detail to add beyond what the charter has captured, the project plan should refer to the charter instead of duplicating information in the charter.

4.4.2 Project Artifacts

All project artifacts should be traceable to and in sync with the project plan. Project artifacts should not duplicate information, but will often elaborate details summarized in the plan.

4.5 Reuse and the Project Plan

In organizations with mature, stable, and well understood processes, the project plan can leverage significant reuse when defining how work will be executed. As an example, in organizations utilizing CxOne, project plans can refer to CxOne materials for descriptions of technical and managerial processes, focusing on detailing the unique aspects of the project.

5 ESTIMATION

Estimation is predicting future characteristics or outcomes from current information. Estimation capability may be improved by improving estimation techniques, improving current information, and improving how estimates are used. CxOne provides support for all three of these improvement paths.

This section outlines some key CxOne estimation concepts and terminology. For more information, see the CxOne estimation materials and *CxOneTermsAndAcronyms*.

5.1 Purpose

Software estimation is applied most often to determine resource expenditure (usually effort and time) necessary to produce a given outcome (usually functionality). The predictions provided by estimates allow plans to be created, feasibility of existing plans to be validated, and baselines to be set for tracking and control of execution. Solid estimates are critical to successful management of all project phases. Mature estimation practices allow organizations to become very sophisticated at managing their scope, budget, schedule, and quality.

5.2 Overview

Estimation touches many aspects of software projects, from the most detailed technical level (e.g., how much code will do X) to the broadest managerial level (e.g., how to best convey a realistic estimate to the CEO). There are numerous ideas and practices related to estimation, and many work well. But some of the most commonly used practices are ineffective, and accurate estimation is a challenge for many projects.

Creating estimates involves probabilistic modeling. Thinking of estimates as models reinforces that estimates are approximations, that multiple views of an estimate increases confidence in its accuracy, and that an estimate is only as good as the information used to create it.

CxOne defines several fundamental ideas for estimation. Estimates must be differentiated from targets. Estimates will always involve uncertainty. CxOne separates software estimation practices into two broad categories, project estimates and task estimates. Within these two categories there are a set of overlapping practices that CxOne recommends using for various estimation challenges. To apply these practices, CxOne advocates phased estimation.

5.2.1 Estimates, Targets, and Control

Projects usually have both estimates and targets, and the two are often confused. An estimate is a prediction of what is possible or likely. A target is a management goal used as the basis for control. The statement “project A must cost no more than X and be delivered by Y” may be a valid business target, but is not an estimate; it is a constraint on the project. Estimates should be used to set realistic targets and are part of the process of managing to targets, but they need to be clearly differentiated from targets.

Scheduling is time-focused control activity that **is often confused with estimation**. Schedules are a detailed plan for controlling project work to meet targets. Good schedules are derived from estimates, but are not themselves estimates.

5.2.2 Uncertainty in Estimates

Estimates seek to predict the future. By definition the future is unknown. CxOne requires that estimation practices must account for uncertainty. Handling uncertainty is integrated into CxOne and includes practices such as, treating estimate values as ranges or associating values with probabilities, differentiating between precision and accuracy,^{*} using multiple estimation techniques, and modifying techniques as context changes (see *Phased Estimation* below).

5.2.3 Macro and Micro Estimation Techniques

Macro estimates are aggregate size, effort, time, budget, and quality predictions dealing with an entire project or significant part of a project (also called project estimates or top-down estimates). Micro estimates are size, effort, and/or time predictions for tasks that an individual or small group will execute (also called task estimates or bottom-up estimates). Macro estimates are normally created at the whole-project level, whereas micro estimates are normally created at the task level. Macro estimates provide the foundation upon which lifecycle, managerial, and technical processes chosen in the project plan are based. Micro estimates support the management of detailed project work, and can be rolled up into macro estimates.

5.2.4 Counting, Computing, Judging

The less an estimator actually has to *estimate* (or judge), the more accurate the estimate is likely to be. The Construx estimation approaches emphasize *counting* work products that can be counted (such as lines of code, function points, use cases, etc.), *computing* from calibrated data for items that cannot be counted, and using the estimator's *judgment* only as a last resort.

5.2.5 Use of Multiple Techniques

Expert estimators typically use multiple techniques to estimate the same item, and then look for convergence or spread among the results of the different techniques.

5.2.6 Phased Estimation

Phased estimation is an estimation strategy that mitigates many risks caused by the cone of uncertainty. Phased estimation utilizes different estimation techniques during different phases of the project. This allows projects to optimize their estimates to their realities, i.e., that less detail is available early in the project than is available later in the project. Typically, macro techniques are used early in a project, and micro techniques are used later. On small projects, micro techniques may be used exclusively. See *CxBest_PhasedEstimation* for more details.

^{*} Precision refers to how many significant digits a value has. Accuracy refers to how close to the mark a value is. In estimation, false precision is the enemy of accuracy, and accuracy is what makes estimates high-quality. An estimate of "2-3 weeks" appears less impressive than an estimate of "11 days." However, depending on the situation the former estimate is often a more accurate representation of what is known than the latter.

6 CORRECTIVE ACTIVITY MANAGEMENT

Not all work on a project can be foreseen or precisely planned. Corrective Activity Management (CAM) captures the common elements of managing work activities resulting from addressing change requests, defects, risks, and issues. By using a single lightweight model to manage work arising from these different sources, CAM maximizes reuse and efficiency.

This section provides an overview of CxOne CAM concepts and terminology. For more information, see the CxOne CAM materials and *CxOneTermsAndAcronyms*.

6.1 Purpose

Corrective activities are the actions required to assure alignment of project planning with project performance and project goals. Corrective activity management synchronizes actual performance with planned performance or vice versa. It also ensures that the project activities and planning are in sync with stakeholder goals.

As shown previously in Figure 1-1, corrective activities represent work (or potential work) not defined in the current plan. We use the term “unplanned” as a convenient way to refer to this work, but that term is a simplification. The key point is that the work must be addressed for the project to be successful, and that details necessary to track and control the work aren’t explicitly defined in the work plan or schedule other than as a buffer or placeholder.

Where line between planned work (using work plan and schedules) and unplanned work (CAM) is drawn is part of project planning. This decision is heavily influenced by lifecycle, project size, and management style.

6.2 Overview

Change management, defect management, issue management, and risk management are corrective activities. CAM is CxOne’s model for managing project work arising from change requests, defects, issues, and risks.

The processes and tools for managing these four corrective activities share significant commonality, which is why CAM was developed. CAM does not imply that these four items are the same or that all aspects of managing them can or should be handled through CAM. Rather, CAM is a useful abstraction for tracking and controlling identified items.

6.2.1 CAM Items

CxOne models corrective activities using the following common classifications:

Classification	Description
Change Requests	Potential modification of plan (scope, schedule, budget)
Defects	Product or artifact not matching requirements or goals. Rework.
Issues	Unplanned work that will impact the project if not addressed.
Risks	Something that could happen that will negatively impact the project.

6.2.2 A PMBOK view of CAM

The PMBOK discusses the five project management process groups shown below:

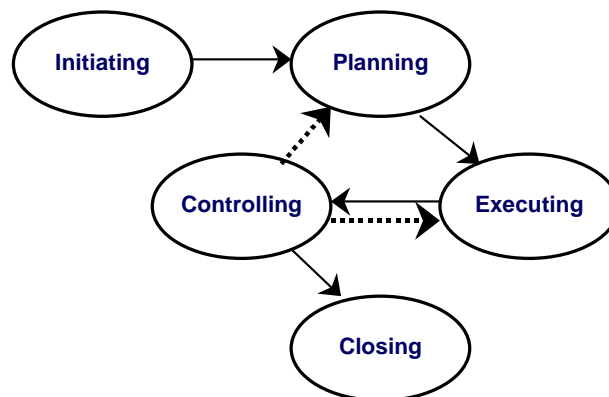


Figure 5-1: PMBOK Project Process Model

Executing processes carry out the plan defined by *planning* processes. *Controlling* processes monitor and measure progress, identify any variances or risks (either with the plan or with the stakeholders view of whether project goals are being met), and initiate corrective action.

Corrective activity management is represented by the dotted lines flowing out of the *controlling* process group. To align the goals and performance, corrective activities can either update the plan or affect execution. CAM consists of ensuring that all corrective activities are efficiently defined, communicated, tracked, executed, and verified.

6.2.3 Using CAM to Manage the Project

The amount of unplanned project work is controlled by the lifecycle and project management practices. Depending on the breadth and depth of planning, much of a project's work may be managed through CAM mechanisms. In general the larger and more complex a project, the more detailed planning should be. Small informal projects with evolutionary prototyping lifecycles may utilize CAM for the majority of their project management. Larger, more formal projects will utilize CAM to handle exceptions to planning.

The depth of CAM detail on a project should be balanced by diminishing returns from excess overhead. For example, defects that are self-identified and fixed immediately would normally not be captured as CAM items. Many day-to-day issues that crop up and are resolved do not need to be tracked by CAM mechanisms.

6.2.4 Using Database Tools with CAM

Due to the number of details to be tracked, database tools often facilitate the management of CAM items. As an example, most projects use defect tracking databases, which often expand to handle items that aren't really defects but have a similar management lifecycle. CxOne abstracts commonality from the different CAM activities into a single parent pattern, *CxPattern_CamDatabase*. Specifics related to the management or database configuration for each of the four CAM types are provided in specialized patterns. CAM is designed to allow a

combination of one or more databases to track items and assumes very basic tool support, so projects can optimize processes to their needs and available tools.

6.3 CxOne CAM Materials

CxPattern_CamDatabase provides general details about CAM implementation. This pattern provides the core details about setting up a database tool and surrounding processes to manage identified change requests, defects, risks, and issues.

Each type of CAM activity has unique specialization of the general CAM database pattern and other materials associated with it. These can be found in the following locations:

Knowledge Area	CKA Location
Change Management	ConfigurationManagement/ChangeManagement
Defect Management	Quality/DefectManagement
Issue Management	EngineeringManagement/Project/IssueManagement
Risk Management	EngineeringManagement/Project/RiskManagement

7 TEAM STRUCTURE

The staffing structure of a project, defined by roles and responsibilities, provides guidance for how people will carry out project work. This section provides an overview of CxOne roles and responsibilities concepts and terminology. See the materials in the *roles* folder for more details on specific roles.

7.1 Purpose

Ultimately software projects boil down to people working together to complete tasks. When groups of people work together on complex endeavors it is useful to decompose work and ensure there are clear points of responsibility for critical aspects of work.

7.2 Overview

The diagram below describes CxOne positions for a typical software development project. These positions are commonly referred to as “hats”. On smaller projects, engineers wear multiple hats and often will perform work as an engineer for other leads even if they are wearing a lead hat. Even on a two (or even one!) person project, calling out these responsibilities and using them to force thinking from different perspectives can be useful.

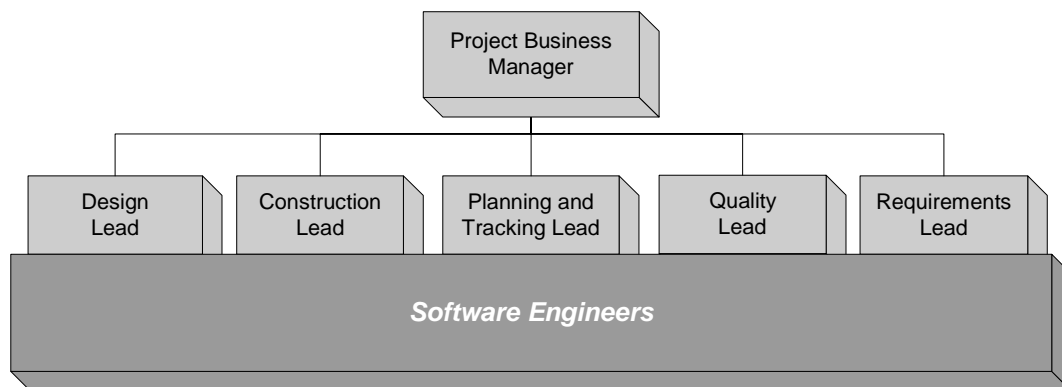


Figure 6-1: Project Team Structure

The responsibilities for managing work are split between a group of technical leads and the project business manager (PBM). Using an analogy with the movie industry, the PBM can be thought of as the producer (handles business issues), and the group of technical leads can be thought of as sharing the duties of a director (handles technical execution).

Front-line work on the project is carried out by a pool of engineers. Depending on the project and organization, this pool may be a group of cross-trained people that share responsibilities, or specialists may be provided from different areas to work under specific leads.

7.2.1 Standard Project Positions

The following roles are considered the nominal oversight structure for a project. If a project has specific needs for technical leadership in other areas, project business managers should create specific lead positions for the project's needs (see next section).

Standard Project Role Description

Position	Responsibility
Project Business Manager	Responsible for successful business outcome of the project. In charge of project staffing, acquiring resources for the project, personnel issues, top-level work assignments, and stakeholder interaction. Is the top decision maker on the project, but normally defers technical decisions to the appropriate technical lead. Resolves disputes between project participants. Responsible for coordinating the activities with the stakeholders, project sponsor, and project reviewers.
Planning and Tracking Lead	Directs overall flow of technical work on the project. Directly responsible for project planning and overseeing the execution of work breakdown, estimation, scheduling, and tracking.
Requirements Lead	In charge of eliciting, defining, maintaining, and tracing detailed product requirements. If necessary, separate sub-leads for UI design, documentation, and other areas may be assigned.
Quality Lead	Plans and directs all quality assurance and quality control activities including reviews and testing. Sub-leads may be assigned as appropriate.
Design Lead	Responsible for the system architecture and overseeing design activities. As appropriate may assign sub-leads for functional areas of the product or for technical specialties such as database, graphics, driver, distributed processing, communications protocol, etc.
Construction Lead	Responsible for construction, integration, product builds, development environment, and deployment. As appropriate, may assign sub-leads for functional areas, environment, integration, deployment, etc.
Software Engineers	A pool of software engineers that can be assigned various tasks on a project based on ability and interest. On many projects engineers will perform activities for more than one lead, and leads of one area may work as an engineers in a different area.

7.2.2 Custom Project Positions

Creating additional specialized lead positions on a software project allows for specific project needs to be more effectively met, because each areas has its own visible lead position. This structure allows distribution of lead responsibilities to take the best advantage of staff capabilities while providing many opportunities for professional development.

The project business manager should be careful to ensure that the splitting of responsibilities is balanced against efficiency of decision making. Requiring every lead to be involved in every decision will not likely be the most efficient way to run the project.

7.3 External Project Roles

The diagram below describes the CxOne model for external interfaces on a typical software development project. The project business manager is the focal point for communication in and out of the project, although it is quite likely that many other communication channels will develop between team members and stakeholders outside the project.

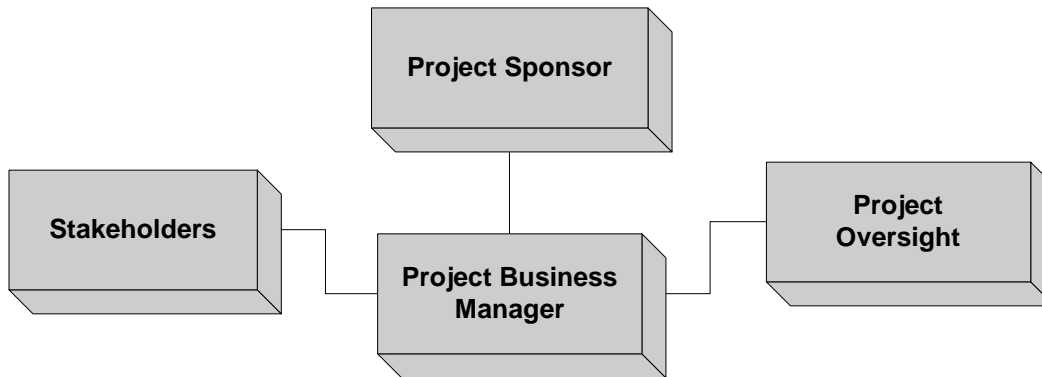


Figure 6-2: External Project Interfaces

External Project Positions

Position	Responsibility
Project Sponsor	The project sponsor is part of inception, approves resources for the project, directly oversees the project business manager, and is responsible for the project meeting the goals of all stakeholders.
Project Oversight	Experienced and objective technical and management oversight from individuals who are not directly involved in the project. Examples include a SEPG, a project reviewer, external consultant, etc.
Stakeholders	Any individual or organization that is explicitly recognized as being directly or indirectly affected by the project outcome.