The Neuma Unified Process

Neuma Technology Inc.

6 November, 2003

Copyright Information

The Neuma Unified Process (Release 6.0 Edition ah03)

Neuma Technology provides this manual “as is” without warranty of any kind, either expressed or implied, including, but not limited to the implied warranties of merchantability and fitness for a particular purpose. Neuma reserves the right to make changes to the product described in this manual at any time and without prior notice.

The Neuma Unified Process

by: Neuma Technology Inc.
#51 - 5450 Canotek Rd.
Ottawa, ON K1J 9G3
Canada

Tel: (613) 749-9450

http://www.neuma.com
email: sales@neuma.com

Printed in Canada

Table of Contents

List of Tables

List of Figures

1 The Neuma Unified Process - Overview

The Neuma Unified Process (**NUP**) gives you a framework to model your software development process in CM+. With NUP, you can rapidly configure CM+ so that your team follows industry best practices for software quality and rapid development.

The NUP provides process building blocks consisting of Phases, Disciplines, and Activities. With these building blocks, you establish a development lifecycle model where your team executes the activities, generating Deliverables, and completing Milestones along a project schedule.

The Activities, Deliverables and Milestones are mapped to key CM+ configuration settings resulting in a tailored system. This gives you an efficient tool to track and manage all aspects of the project. As well, it gives each of your team members a tool configured for their role, clearly laying out the workflow activities that they are responsible for.

1.1 CM+ -- The Foundation for NUP

The Neuma Unified Process is designed to work with CM+. NUP adds a process layer to the integrated set of software project support applications already built in to CM+. These applications include:

- Project planning and project activity tracking
- Requirements management
- Feature and design element management
- Change request management
- Document management
- Software change packaging and control
- Source code revision control and parallel stream development
- Build and release automation and tracking
- Test case management
- Problem tracking

The CM+ architecture employs an extendable data schema, a powerful command and scripting language and a user configurable GUI. All of the CM+ applications work within the common architecture and share a common repository. The application data are also fully integrated, enabling a high degree of process automation. The NUP relies on this tight, application integration in order to deploy a process that is ready to use, requiring no additional applications or “process-glue”.

1.2 Phases, Disciplines and Activities

The NUP first breaks down your development organization into industry best practice disciplines allowing for efficient grouping of activities. The NUP then breaks down the development cycle into key phases, again following industry best practice. During a phase, the people in each discipline are responsible for executing the activities necessary to move the project along. At the end of a phase (and during), important decisions are made based on the completion (and review) of project deliverables. These decisions (or “gates”) determine if the project moves forward.

1.2.1 Disciplines

Disciplines allow you to organize your workflow activities and visualize how the different project team members will interact. Each team member falls into a discipline that matches his or her skill set and overall contribution to the project. On small projects, individuals may fall into multiple disciplines. The available disciplines represent industry best practice workflows. They include:

Business Planning (b)

Establishing a business case for the project by aligning corporate business objectives, engineering constraints, core capabilities and market or customer requirements.

Requirements Management (rm)

Eliciting, reviewing and documenting product requirements at all levels from the business objectives down to the detailed specification.

Analysis And Design (ad)

Determining the concepts, architecture, use-cases, objects and function points for the software product to be developed.

Implementation (i)

Coding and unit testing software and other solution work products.

Testing and Verification (tv)

Verifying integrated product modules against a pre-defined test plan, originating requirements and solution design.

Delivery (d)

Preparing the product for final release and shipment to both internal and external customers.

Operations and Support (os)

Providing customer support for delivered solutions and identifying and tracking new problems and requirements originating from the customer base.

Configuration and Change Management (cm)

Identifying the configuration of the product in terms of the organization of its component parts, tracking and managing all changes to the product modules and product configuration, and preparing builds or releases of the product.

Project Management (pm)

Managing the people, scope, cost and schedule for the development of the software product.

Environment (e)

Administering the software development environment so that each user is working within the established process and has all the tools and access required to work effectively.

Infrastructure Management (im)

Establishing and administering a core IT infrastructure to support the development environment and all its users.

Disciplines vs. Roles in CM+

Disciplines are a process building block to group activities and to demonstrate workflow interactions. CM+ “roles” are user attributes that control access to commands, data and menu functions. Each user is configured with a set of roles in CM+. Discipline on the other hand is an activity attribute in CM+, and is used primarily for reporting purposes.

1.2.2 Project Phases

The NUP breaks the project timeline into consecutive phases, each representing a major completion step in the project. Activities are grouped into phases based on when they should occur in the project. Some activities are repeated in successive phases (e.g. project re-planning).

The NUP phases are derived from industry best practices and regardless of which lifecycle model you apply, the overall goal of each phase remains the same. The NUP phases are:

Inception

The inception phase produces the vision and business case for the project. The requirements, cost and potential benefit are examined in sufficient detail to decide whether or not to proceed with the project.

Elaboration

In the elaboration phase the analysis, design and planning activities are completed, producing a requirements document, high level design (or architecture) and software development plan. This allows stakeholders to validate both the product and project scope before development starts.

Construction

In the construction phase, the detailed design, development and testing activities are completed. The process model you follow depends on the type of software you are developing and the standards in place in your organization. It can be iterative, evolutionary or a single stage development style.

Assurance

In the assurance phase, the product is prepared for general release to your customer through system testing followed by limited operation by end users. Limited operation could include alpha and beta testing, or parallel operation with a legacy system. A certain amount of product rework is expected before the product is deemed ready for formal release.

Delivery

The delivery phase includes all the activities necessary to formally release the product to your customer (whether that is a specific customer or the general market). It includes the transition activities required to bring sales, training and product support organizations on line.

Maintenance

During the maintenance phase, the users are supported, changes are initiated, reviewed and developed and new product versions or patches are released as necessary.

1.2.3 Activities

Each phase of a development project requires project contributors from the various disciplines to perform certain activities, generating a specific deliverable or taking the project through a decision gate. The activities they perform depend entirely on the process model you establish for your organization.

NUP provides a comprehensive set of activities from which you can tailor a project plan. By organizing these activities into phases, and tailoring them to suit your process, you create not only a blueprint WBS but also a workflow model for each of your disciplines. This workflow model prescribes, in each phase, step by step instructions to follow. Project team members can navigate these instructions through a workflow assistant. The activities themselves are stored in the CM+ Activity Tracking module, which supports a hierarchical breakdown structure and a suite of project scheduling functions.

The available activities, by discipline are as follows:

Discipline	Activities
Business Planning	Business case analysis
Requirements Management	Product visioning
	Requirements definition
	Updating Requirements
Analysis and Design	Architecture definition
	Test planning
	Product design
Implementation	Development
Testing and Verification	Test case design
	Verification
	System testing
Delivery	Alpha release
	Beta release
	General release
	Maintenance Release
Operations and Support	Alpha change management
	Beta change management
	Solution Delivery Services
	In Service Support
Configuration Management	Release planning
	Component Design
	Building
Project Management	Project risk assessment
	High level project planning
	Software development planning
	Cycle Planning
	Cycle assessment
	Project closing

1.2.4 Supporting Parallel Release Streams with NUP

Most software products undergo several major releases in their life span. Alternatively, a single major project may be planned at the outset to generate multiple releases of a product. There are typically many information elements (e.g. requirements, problems, code) and resources (people, computers) shared across these release streams.

CM+ gives you all the tools you need to manage multiple release streams with appropriate sharing of resources and information elements. The NUP gives you the ability to focus activities to specific release streams while maintaining visibility to the wider product picture.

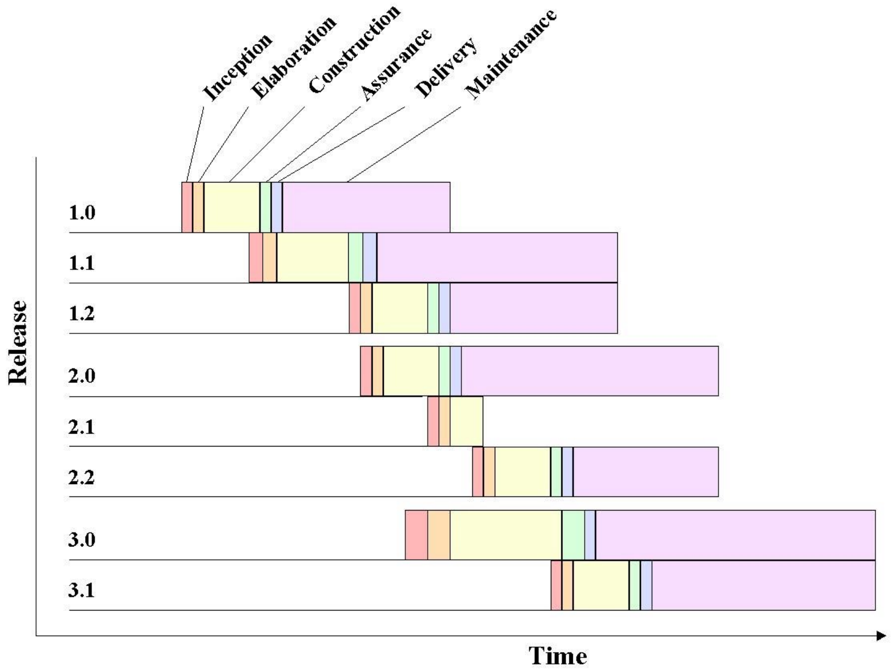

Figure 1 Parallel Release Streams

1.3 Deliverables

Every software project generates new or changed deliverables. These include product specific items such as executable code, installation media, training materials, as well as project specific items such as project plans, design specifications etc. The activities you place in your project plan will result in specific deliverables. Typically these deliverables are required in order to advance a project from one phase to the next.

Many deliverables can be generated from information elements tracked directly in the CM+ applications. For example a project schedule can be generated from the activities defined for that project. Other deliverables are tracked in the CM+ Document Tracking module (Enterprise only) as managed documents, undergoing formal change, review and release cycles. A full list of the NUP deliverables is provided in Section 2.1.

1.4 Milestones

The NUP framework establishes project milestones used to identify major completion steps in the project. Most of these formally mark a transition between phases, but some exist to mark important steps within a phase.

The NUP aligns project milestones with key information element states (which are in turn tied to workflow steps), so that the milestone can be completed as soon as all its linked information elements are promoted to the correct status. The linking is accomplished through a report or query that consolidates the tracked information for review. For each milestone, NUP also inserts a CM+ activity into your WBS so that project-scheduling information can be tracked for the milestone.

The NUP includes the following best practice milestones. Your development cycle may require additional milestones (and process workflow steps) or may merge milestones.

Project Start (PS)

The project start milestone marks the start of the project.

Business Case (BC)

The business case milestone marks the approval of the business case for the project and the start of the elaboration phase.

Analysis Close (AC)

The analysis close milestone marks the approval of the requirements by the project customer and the approval of the architecture by the technical stakeholders (e.g. senior architect). This may include the review of one or more prototypes.

Software Development Plan Approval (SDP)

The software development plan approval milestone marks the start of the construction phase of the project with all analysis and planning having been completed.

Design Close (DC)

The design close milestone marks the completion of a critical or detailed design for a planned release stream including the definition of a complete set of test cases.

Alpha

The alpha milestone marks the completion of all internal test cycles (for which other milestones may be set as well) and the approval of an external release for alpha testing (first round of limited external testing).

Beta

The beta milestone marks the completion of alpha testing and the approval of an external release for beta testing (general external testing or parallel operations).

General Release (GR)

The general release milestone marks the completion of beta testing and the general release of the product to the customer or market (e.g. launch, go-live, activation). The GR milestone also marks the start of the Maintenance phase.

Maintenance Release (MR)

The maintenance release milestone marks the general release of a maintenance upgrade or patch to the product after completing a maintenance development cycle.

Project Close (PC)

The project close milestone marks the end of the project and the migration of all customers to an alternative product. Support and maintenance ends at project close.

2 The NUP Configuration Framework

This section presents the framework for configuring and loading the NUP into CM+. CM+ arrives out of the box with applications, schema, and user interface functions pre-set to enable all of the process elements in NUP (see Section 3 – CM+ Standard Development Lifecycle). The NUP, when loaded, adds pre-set activities, roles and other data for managing your process and initiating your project. In CM+ Enterprise, the NUP also adds requirements, test cases and document templates. In addition, the NUP creates a specialized workflow assistant to help all users execute activities according to the process.

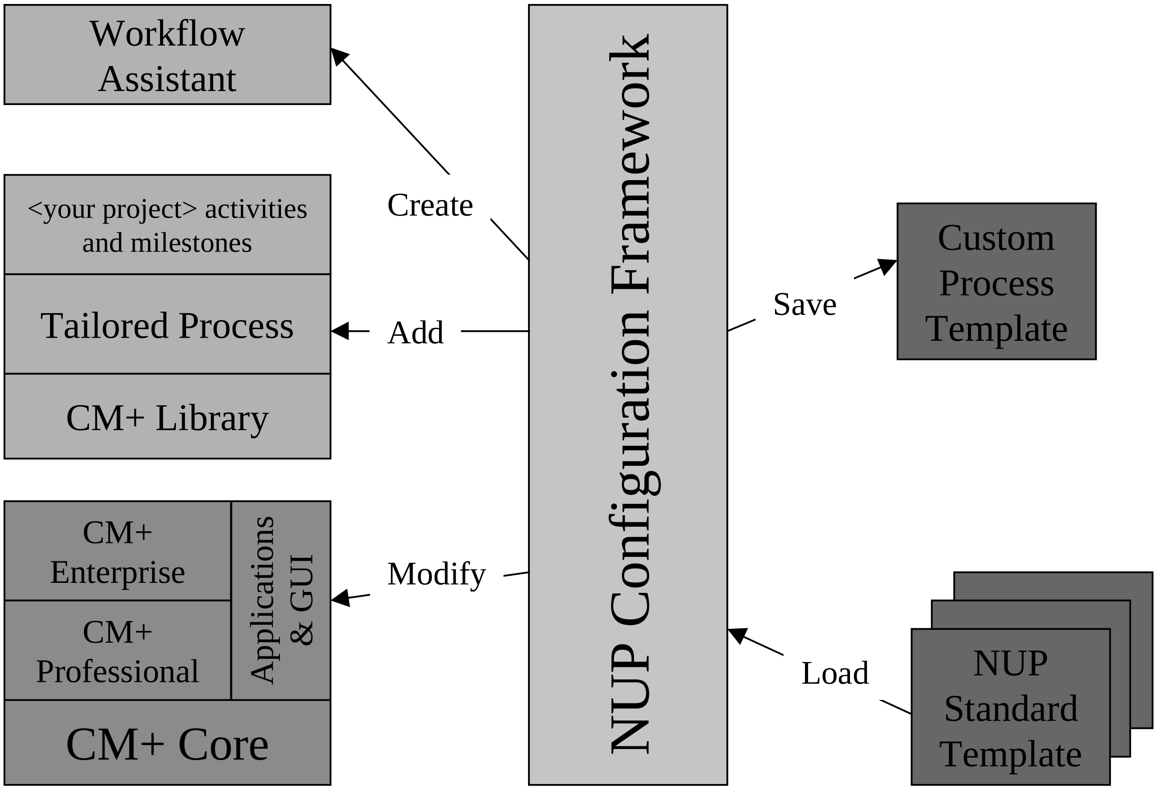

Figure 2 The NUP Configuration Framework

The degree of automation available depends on whether you have installed CM+ Professional or CM+ Enterprise. In Professional, some NUP deliverables (e.g. requirements, documents) must be tracked outside of CM+. In Enterprise, the information is tracked within CM+ itself and the deliverables are generated.

Your business process is mapped using the NUP Configuration Framework. Starting with the NUP Standard Template (or another off-the-shelf template), you configure the activities, deliverables, milestones and other data to produce a tailored process. This process is then added to your current CM+ library. Modifications to the application schema and GUI may also be made. You then save the custom process as a new template for later refinement and redeployment.

Also with the NUP Configuration Framework you create a workflow assistant that documents the custom workflow for each of your user roles. This gives individual contributors a focused set of HTML help screens to guide them through each of the activity sequences.

2.1 The NUP Standard Template

The NUP standard template contains default process configuration settings. Combined with the CM+ default installation, the NUP standard template results in the CM+ Standard Development Lifecycle presented in Section 3.

2.1.1 Standard Template Roles

The following roles are part of the NUP Standard Template.

Table 1 Standard Template Roles

Role Code	Role
admin	Administration
cmmgr	Configuration Management
designer	Software Design
prjmgr	Project Management
verif	Verification
prob	Problem Tracking
support	Customer Support

2.1.2 Standard Template Activities

The NUP Standard Template organizes the NUP activities described in Section 1.2.3 into a template WBS. Each of the phases is modeled as an activity of type subproject. A single stream activity is inserted as well as a single development cycle in the construction phase. After the NUP standard template is loaded, the project manager can rename and restructure the activities to match the actual project plan.

A subset of the standard template WBS is shown below (the inception phase). The full WBS template can be found in the appendix.

Table 2 Example Standard Template Activities

WBS	Activity Title	Discipline	Activity Type
0	<Your project title>	pm	project
1	Inception Phase	pm	subproject
1.1	Product visioning	rm	task
1.2	Project risk assessment	pm	task
1.3	High level project planning	pm	task
1.4	Business case analysis	b	task
1.5	Business Case	b	milestone

2.1.3 Generated Deliverables

Generated deliverables are process deliverables created by CM+ from tracked information elements. They are typically generated by running a report that consolidates the information for review. The NUP standard template provides the following generated deliverables.

Project Plan

The project plan is generated from the Project Definition report. The Project Definition report takes as input an activity of type project and produces an activity hierarchy with total planned effort.

Requirements Document

The requirements document is a generated from Requirements report. The Requirements report takes as input a root level requirement and produces the requirements hierarchy under this root.

Release Plan

The release plan is generated from two reports, the Streams report and the Products report. The Streams report takes as input an activity of type project (as in the Project Definition report) and produces a hierarchy of activities of type stream or design. The Products report takes as input a list of products (or defaults to all tracked products) and produces the product, sub-product hierarchy.

Release Delta Report

The release delta report is a summary of the changes in a new build or product baseline. The report compiles information based on the changes (updates) between any two releases, baselines or builds. It produces lists of updates, activities and problems.

Test Cases

Test cases are internally tracked as test case records in CM+. They can be compiled into a report to drive the execution of any planned testing cycle.

Source Code

Source code is internally tracked as files in CM+. Files can be shared across products and directories as necessary and changed in parallel across development streams.

Change Requests

Change requests are internally tracked as request records in CM+. They can be compiled into a report that drives the cycle planning for a new development cycle.

2.1.4 Tracked Documents

Some process deliverables are prepared separately from CM+ and checked in as tracked documents. The NUP standard template provides placeholder records (with title and description) for tracking these deliverables as documents.

Architectural Model

The Architectural Model describes the broad architectural design of the product.

Business Case

The business case describes the justification for the project, linking the requirements to enterprise goals, and projecting the return on investment.

Business Plan

The business plan contains the overall strategic plan for your organization.

Change Management Plan

The change management plan describes your organization’s process for initiating, tracking, prioritizing and approving product changes that originate as requests from your end-users.

Customer Requirements

Customer requirements are a compilation of contractual or specific customer requirements that are driving your product development.

Engineering Requirements

Technical requirements or constraints imposed by

Enterprise Architecture

The Enterprise Architecture describes the integrated framework of business, IT and data architectures, which your organization uses to ensure meeting its objectives.

Market Requirements

Market requirements are a compilation of perceived requirements from the overall market.

Product Vision

The product vision is the highest-level design description for the product identifying the key requirements or business objectives it is to meet.

Project Close Release Document

The project close release document compiles project close out information, identifying the migration plan for existing customers and changes being deferred to new projects or release streams.

Release Documents

Describes and defines the contents of the product. These documents take the form of release notes; on-line documentation and help notes; installation notes; installation guide.

Release Status Report

A release status report compiles the results of a development cycle in terms of the changes or design elements that were completed.

Risk Assessment

A project risk assessment compiles the current set of project and product risks with updates to all risk attributes and summary of completed mitigation or contingency activities.

Test Plan

A test plan establishes the overall plan for all testing activities in a project in terms of the resources, types of testing, phases of testing, metrics, objectives etc.

Test report

A test report compiles the results of a testing cycle.

2.1.5 Standard Template Milestones

The NUP standard template employs all of the NUP framework milestones, with each specifying gate conditions. These milestones are described in Section 1.4 above and again as activities in Section 2.1.2. For a description of the gate conditions, please review the CM+ Standard Development Lifecycle in Section 3.

2.1.6 Information Element State Transitions

The NUP standard template establishes default states and process transitions for each of the main information elements tracked in CM+: Activities, Documents, Problems, Requests (change requests), Requirements and Updates. Through these process definitions, CM+ enforces process transition rules and executes transition triggers to enable workflow automation.

An example state transition process is provided below (for the CM+ Activity Tracking module). The full set of definitions can be found in the appendix.

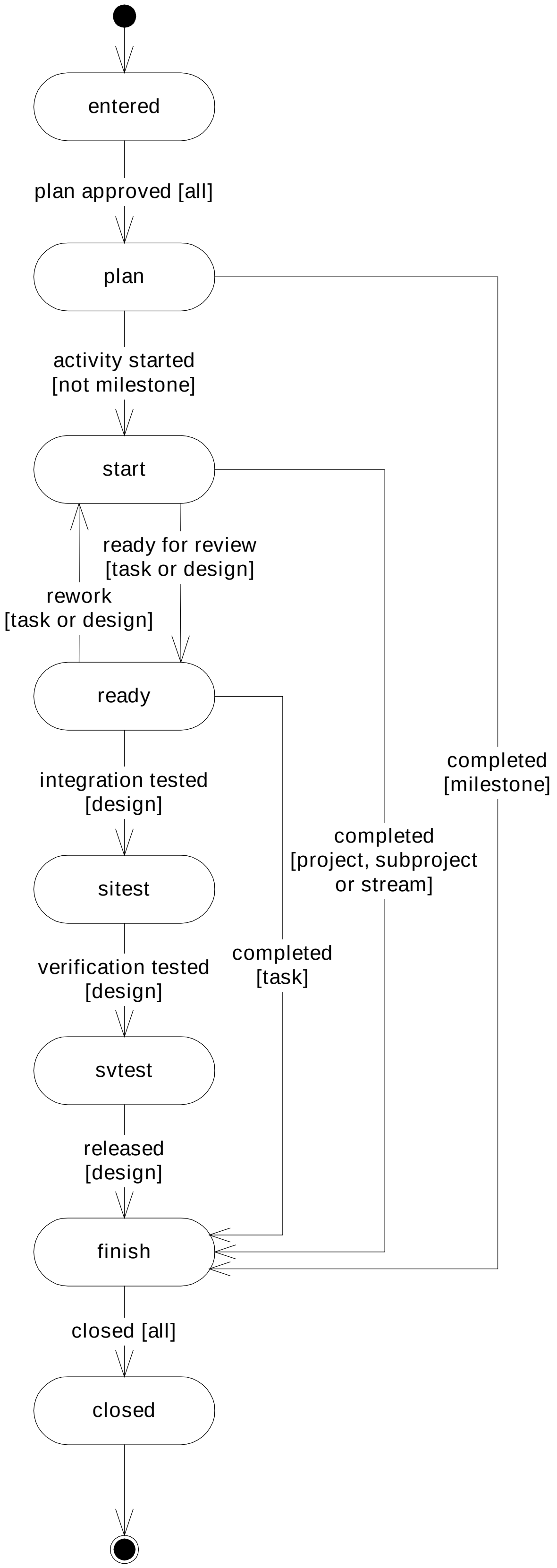

Figure 3 Activities State Transitions

2.2 NUP Configuration Options

The configuration framework allows you to tailor most aspects of the NUP. By tailoring the NUP to map to your development lifecycle, you leverage the investment already made in process improvement and training. You can also seamlessly integrate CM+ into your quality management system by building in the milestones and deliverables required to ensure compliance.

2.2.1 Roles

You may rename any of the standard template roles or add to the list of roles. Roles can then be mapped to specific activities and GUI menu options to tailor the workflow and the use of CM+ for that role.

2.2.2 Phases and Activities

You may rename, reorder, delete from or add to the set of phases and activities in the standard template. This affects the project WBS that gets generated as well as the workflow assistant that appears for each user.

Each template activity is mapped to both a role and a set of workflow instructions (how-to). By organizing activities into phases, you create a custom workflow assistant for each user (by role) in each phase. The workflow assistant compiles the instructions in logical order and presents them to the user as a set of web pages.

2.2.3 Workflow Instructions (How-To)

The workflow instructions are designed to guide the user through the steps required to complete the activity sequences in any phase. You may augment the workflow instructions with any internal documents or web pages that document your processes. You may also modify any of the existing instructions, to match CM+ customizations you may have made. Finally, you can change the mapping between workflow instructions and activities as well as the order that the instructions are presented for each activity.

2.2.4 Milestones

Milestones are represented both as activities in the activity hierarchy (of type milestone) and as custom reports. The milestone activity records schedule information for the milestone in your project plan. The milestone custom report presents, at a glance, the milestone gate conditions.

You may rename, delete from or add to the set of milestones that are provided in the standard template, and for each milestone you may tailor the following report parameters:

- selection of tracked documents to report on (title, status, description)
- queries of any CM+ tracked information (e.g. activities of type project or subproject with duration and planned dates)
- summary counts of any CM+ tracked information (e.g. number of open, high priority problems in current stream)

2.2.5 Deliverables: generated

Generated deliverables are represented as custom reports. As with milestones you may rename, delete from or add to the set of generated deliverables that are provided in the standard template. You may also tailor each report to present any tracked information in CM+

2.2.6 Deliverables: tracked documents

The standard template provides a name and description for each tracked document. You may update any of the existing document templates or add to and remove from the set of documents.

2.2.7 Information Element State Transitions

The available information element states and the permitted transitions can all be changed. Roles and permissions associated with each transition can also be adjusted. Pre and post transition triggers can be written or changed as necessary.

2.3 Saving A Custom Process Template

A custom process template can be saved as a set of configuration files and reused to initialize new projects or CM+ libraries.

3 CM+ Standard Development Lifecycle

CM+ installs by default with a set of applications, schema and user interface functions that enable all of the NUP elements to be loaded and activated for use on a new project. The NUP Standard Template adds to this a package of activities, roles, document templates and other data, allowing users to easily execute the Activity Sequences along each of the project phases.

This section describes the out-of-the-box development lifecycle model that is produced when the NUP Standard Template is loaded into CM+.

3.1 Inception Phase

The inception phase produces the vision and business case for the project. The requirements, cost and potential benefit are examined in sufficient detail to decide whether or not to proceed with the project (Business Case milestone). The activity sequence for the inception phase is as follows.

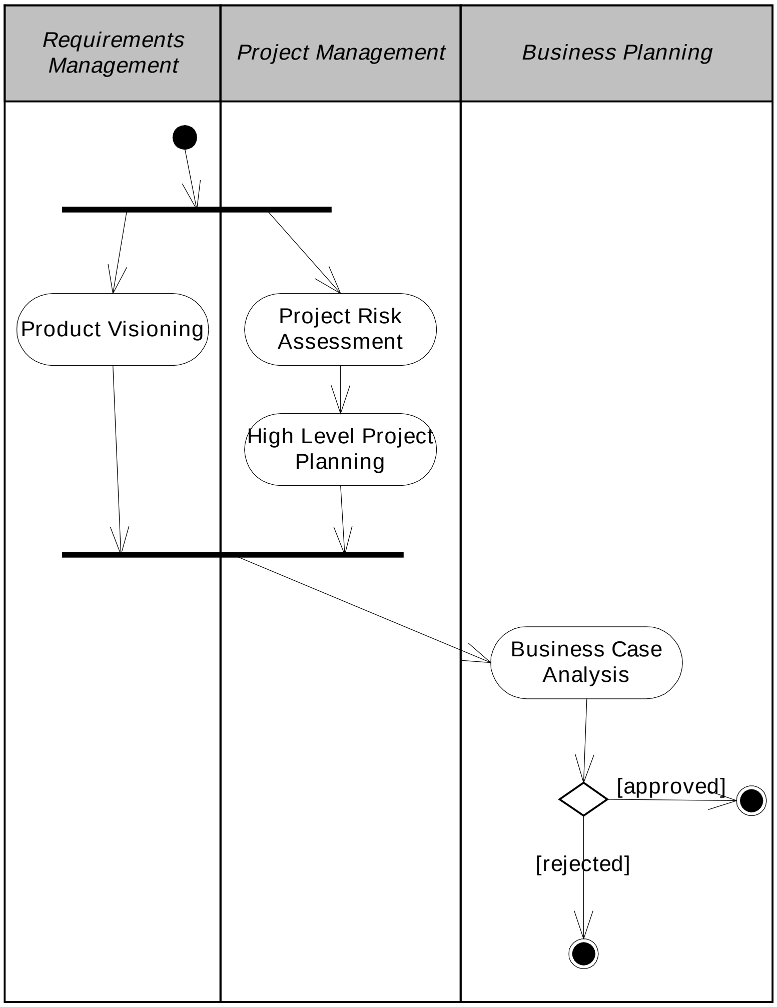

Figure 4 Inception Phase Activity Diagram

3.1.1 Inception Phase Activities

Product Visioning

Product visioning is the process of describing a product or product concepts in terms of features, use-cases, constraints and other attributes, identifying how the market or customer requirements will be addressed. Usually a vision paper or concept paper is produced. Sometimes mockups or prototypes are built.

Role: prjmgr

Workflow Task: Adding documents (Enterprise)

Input Deliverables: Market Requirements, Customer Requirements, Engineering Requirements

Output Deliverable: Product Vision

Project Risk Assessment

Project risk assessment is the formal identification, classification, and prioritization of risks associated with the project. These include product risks, e.g. risk of not meeting certain requirements, and project risks, e.g. schedule risks. The risk assessment, usually summarized in a document, is a critical input to both the high level project planning and the business case analysis activities.

Role: prjmgr

Workflow Task: Adding documents (Enterprise)

Input Deliverables: Market Requirements, Customer Requirements, Engineering Requirements

Output Deliverable: Risk Assessment

High Level Project Planning

High level project planning is the generation of a top-level work breakdown structure for the project with ROM scope and schedule estimates. This WBS is tracked as a set of Activities in CM+. At this early planning stage activities of one of the following types are used, project, subproject or stream.

Role: prjmgr

Workflow Tasks:

1. Adding and modifying streams
2. Adding a project.
3. Adding activities.
4. Producing a project definition document.

Input Deliverable: Risk Assessment

Output Deliverable: Project Plan

Business Case Analysis

Business case analysis is the formal review of the product vision, high level plan and project risks, taking into consideration any enterprise level constraints such as the business plan or enterprise architecture. The business case analysis produces a summary document explaining the net benefit of undertaking the project. This business case is reviewed and a formal decision is made whether or not to proceed with the project.

Role: prjmgr

Workflow Task:

1. Adding documents (Enterprise)
2. Approving a document (Enterprise)
3. Assigning an activity.

Input Deliverables: Risk Assessment, Project Plan, Product Vision, Business Plan, Enterprise Architecture

Output Deliverable: Business Case

3.1.2 Inception Phase Deliverables

External Deliverables

The following deliverables are external inputs to the development lifecycle and are typically in existence prior to the start of a project. These documents can be tracked in CM+ (Enterprise version only). See Section 2.1 for a description of each deliverable.

Market Requirements

Customer Requiremens

Engineering Requirements

Business Plan

Enterprise Architecture

Workflow Input Deliverables

As the inception phase is the first phase of the CM+ standard development lifecycle, there are no workflow deliverables input to the phase.

Workflow Produced Deliverables

The following deliverables are produced as part of the workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See Section 2.1 for a description of each deliverable.

Product Vision: Tracked document

Risk Assessment: Tracked document

Business Case: Tracked document

Project Plan: generated from project activities using the Project Definition report.

3.1.3 Inception Phase Milestones

Business Case (BC)

The business case milestone marks the approval of the business case for the project and the start of the elaboration phase. The gate conditions for the BC milestone are:

1. Completion of the Product Vision, Risk Assessment and Business Case documents (document status: reviewed)

2. Completion of a high-level work breakdown using activities, with planned efforts and forecast dates entered (activity status: planned).
3. Review and approval of the Business Case and referenced documents (review notes added).

Upon completion of the business case milestone, the deliverables are promoted to status: released, with the project activities promoted to status: start.

3.2 Elaboration Phase

Once a project is deemed to be viable, the Elaboration Phase determines how it will get done. The analysis, design and planning activities are completed, producing a requirements document, high level design (or architecture) and software development plan. This allows stakeholders to validate both the product and project scope before development starts. The activity sequence for the elaboration phase is as follows.

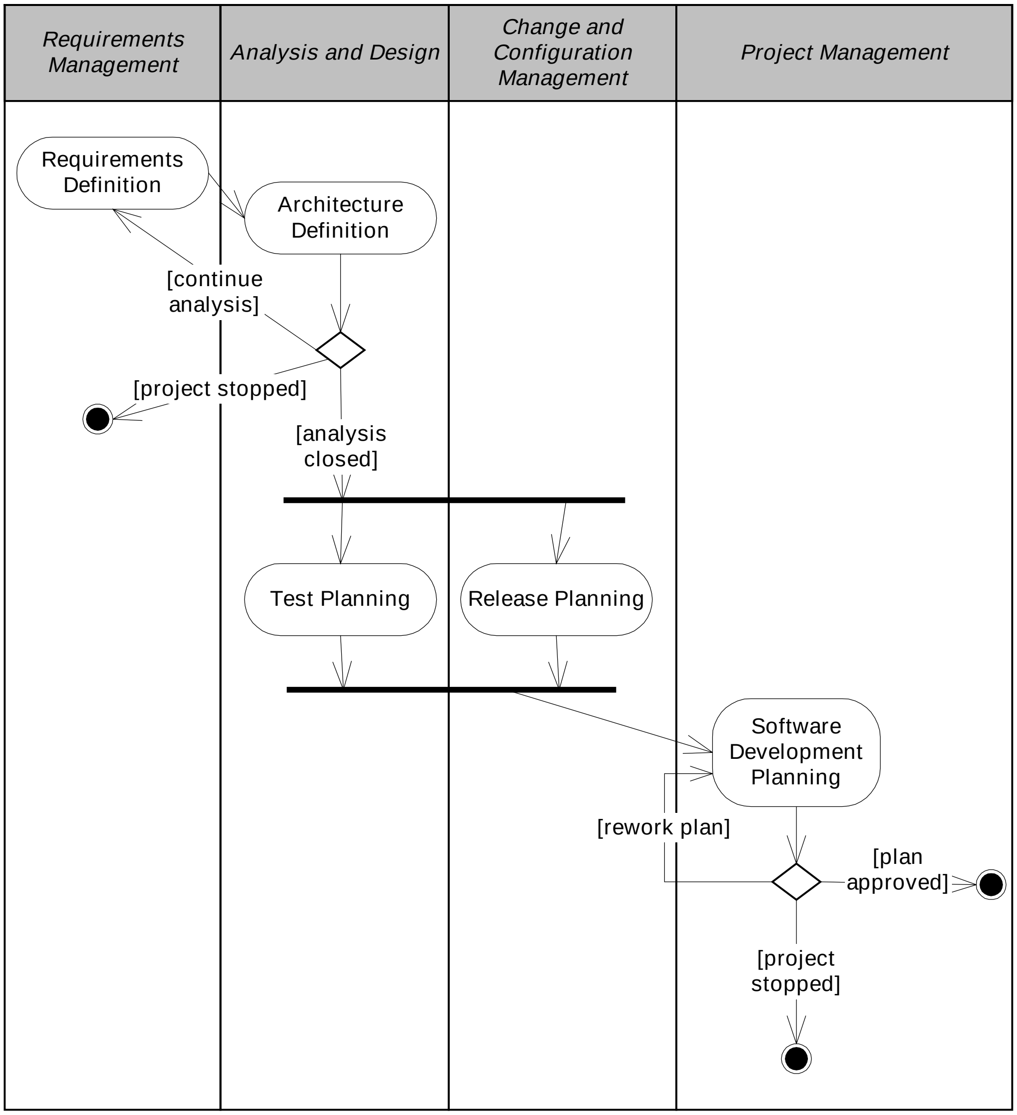

Figure 5 Elaboration Phase Activity Diagram

3.2.1 Elaboration Phase Activities

Requirements Definition

Requirements definition is the process of eliciting, documenting and reviewing the product requirements. These requirements can be categorized into various types (e.g. glossary, technical, features, use cases etc.) according to standard practice for requirements definition. The product is a requirements document, which is reviewed and approved by project stakeholders (typically the product customer). The requirements definition activity is only available in CM+ Enterprise.

Role: prjmgr

Workflow Tasks:

1. Adding requirements (Enterprise).
2. Linking requirements (Enterprise)
3. Creating a requirements baseline (Enterprise)
4. Producing a requirements document (Enterprise).

Input Deliverable: Product Vision

Output Deliverable: Requirements Document

Architecture Definition

Architecture Definition is a high level design activity, describing your product in terms that can drive the planning and development activities (e.g. functions, classes, major components, application tiers, use cases etc.). Architecture Definition may include the development of prototypes or research into component technologies. The product is an architectural model document.

Role: designer

Workflow Task: Adding documents (Enterprise)

Input Deliverables: Requirements Document

Output Deliverable: Architectural Model

Test Planning

Test planning is the definition of the testing scope, the identification of key testing scenarios and the setting of quality criteria. Detailed test cases do not have to be included. The product is a test plan document that feeds the software development planning activity.

Role: designer

Workflow Task: Adding documents (Enterprise)

Input Deliverables: Architectural Model, Requirements Document

Output Deliverable: Test Plan

Release Planning

Release planning is the definition of the top-level configuration items for your product, the initial set of development streams, and the major design elements to be included in each stream. The top-level configuration items are defined as products and sub-products in CM+. Each maps to a set of streams in which software will be developed (with possible overlap in the development schedules – see Supporting Parallel Release Streams with NUP in Section 1). Within each stream, key design elements (features, use cases, components etc.) will be developed and integrated.

The CM manager, working together with the lead designer or architect produces a release plan by defining products, streams (as activities) and design elements (also as activities) in CM+. Each design element is linked to one or more requirements from the current approved requirements baseline. The release plan is essentially a roadmap for your product within the boundaries of the current project plan.

Role: cmmgr

Workflow Tasks:

1. Adding and editing streams.
2. Adding and editing products.
3. Adding an activity.
4. Moving an activity.

Input Deliverable: Architectural Model

Output Deliverable: Release Plan

Software Development Planning

Software development planning is the detailed planning for your development project. It includes defining project teams, creating and assigning project activities and planning the effort and schedule. Activities are added as members of the project, sub-project hierarchy that was developed during the inception phase and organized into development cycles. Scope and schedule estimates are refined and set at the lowest levels in the activity hierarchy.

The project manager, working together with the lead designer, organizes the major design elements into development cycles, based on the current risk assessment (addressing the highest priority elements first). Then each development cycle is rounded out with build, test and cycle assessment activities. See Section 1 for an example breakdown of a project plan.

At a minimum the detailed activities for the first development cycle (Construction phase) should be added. Subsequent phases can be planned in less detail.

Role: prjmgr

Workflow Tasks:

5. Creating and modifying a staff tree
6. Adding an activity.
7. Moving an activity.
8. Assigning an activity

Input Deliverables: Requirements Document, Project Plan (from inception phase), Test Plan, Release Plan

Output Deliverable: Project Plan

3.2.2 Elaboration Phase Deliverables

External Deliverables

There are no external deliverables to the elaboration phase. All inputs come from the inception phase.

Workflow Input Deliverables

The following deliverables are required inputs from the inception phase.

Product Vision

Project Plan

Workflow Produced Deliverables

The following deliverables are produced as part of the elaboration phase workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See the Section 2.1 for a description of each deliverable.

Requirements Document: generated from requirements tracked in CM+ (Enterprise version only) using the Requirements report.

Architectural Model: Tracked document

Test Plan: Tracked document

Release Plan: generated from products and streams tracked in CM+.

Project Plan: generated from project activities using the Project Definition report.

3.2.3 Elaboration Phase Milestones

Analysis Close (AC)

The analysis close milestone marks the approval of the requirements by the project customer and the approval of the architecture by the technical stakeholders (e.g. senior architect). The AC milestone is only tracked in CM+ Enterprise. The gate conditions for the AC milestone are:

1. All changes and additions (updates) to the product requirements completed, submitted and marked as status: ready so that a complete requirements document can be generated from CM+.
2. Completion of the Architectural Model document (document status: reviewed).
3. Review and approval of the product requirements by the customer and other project stakeholders (requirement change status set to select and review notes added).
4. Review and approval of the Architectural Model by the technical stakeholders (review notes added).

Upon completion of the AC milestone, the product requirements are baselined and frozen and the Architectural Model document is promoted to status: released.

If the gate conditions are not met, the requirements definition and Architecture Definition activities can be repeated. The Architectural Model document status is rolled back to rework and updates are added to change the requirements.

Software Development Plan Approval (SDP)

The software development plan approval milestone marks the start of the construction phase of the project with all analysis and planning having been completed. The gate conditions for the SDP milestone are:

1. Project team defined as staff of the project manager. Correct roles and protections assigned to all team members.
2. High level activities for all project phases complete with planned efforts and forecast schedules entered (activity status: planned).

Upon completion of the SDP milestone, the project activities promoted to status: start. If the gate conditions are not met the software development planning activity can be repeated.

3.3 Construction Phase

The Construction Phase is where requirements and architecture are transformed into a design that is implemented by programmers – this is where coding takes place. The activity sequence for the elaboration phase is as follows.

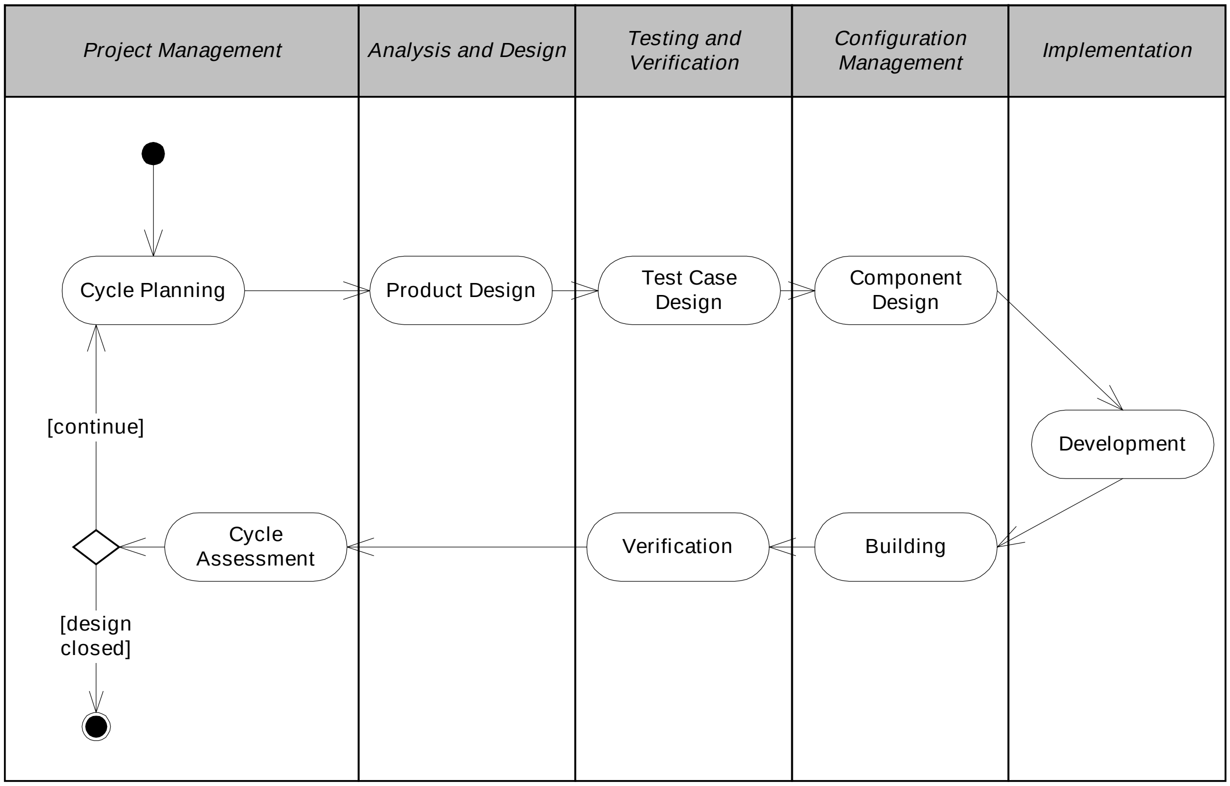

Figure 6 Construction Phase Activity Diagram

3.3.1 Construction Phase Activities

Cycle Planning

During Software Development Planning (Elaboration phase), the cycles for each release stream would have been planned out at a high level. During Cycle Planning in the construction phase, a detailed plan based on the current risk assessment is produced.

This plan further refines the set of design elements to be addressed in the current cycle as well as the planned efforts and durations. It also incorporates any problems or changes that arose in the last development cycle, which were reviewed during a cycle assessment.

Role: prjmgr

Workflow Tasks:

1. Approving change requests
2. Adding and modifying an activity.
3. Moving an activity.
4. Assigning an activity
5. Assigning a problem.

Input Deliverables: Project Plan (from elaboration phase), Test Plan, Release Plan

Output Deliverable: Project Plan

Product Design

Product design is the detailed design of the elements (e.g. functions, classes, major components, application tiers, use cases etc.) identified during the elaboration phase, as well as any new design elements added during the cycle planning.

Design elements are meant to be incremental, promoting a change based, iterative development model. As such, some design elements may simply be extensions of previous elements. Taken together they comprise the entire software specification.

The design description is usually written directly into the design element activity by the assignee. Otherwise a separate document may be submitted and linked to the design element.

Role: designer

Workflow Tasks:

1. Modifying an activity.
2. Adding a document.

Input Deliverables: Architectural Model

Output Deliverable: Release Plan

Test Case Design

Test case design is the elaboration of the test plan into specific test cases that must be executed in the current development cycle.

Role: verif

Workflow Tasks:

1. Adding and modifying test cases.

Input Deliverables: Requirements Document, Test Plan, Release Plan

Output Deliverable: Test Cases

Component Design

Component design is the addition of new software components and the organization of these components in a configuration hierarchy. Lowest level components are source code files. Their hierarchy is usually a logical organization of directories. In CM+, component design is change managed through updates.

In the first few development cycles, the CM manager may be bulk loading legacy code or sub-trees of legacy code. In later development cycles the CM manager or individual designers may be adding or removing individual components.

Role: cmmgr

Workflow Tasks:

1. Loading in your product code.
2. Adding a sub-tree to your product.
3. Adding a file.
4. Adding a directory.
5. Removing a file or directory.

6. Moving a file or directory

Input Deliverables: Release Plan

Output Deliverable: Source Code

Development

Development is the changing or addition of source code including unit testing and code review.

Role: designer

Workflow Tasks:

1. Setting your workspace
2. Populating your workspace
3. Workspace code differences
4. Synchronizing your workspace
5. Adding an update
6. Adding a sub-tree to your product
7. Checking out a file
8. Checking in an update

Input Deliverables: Project Plan

Output Deliverable: Source Code

Building

The building activity is split into two sub-activities, Build Definition and Build Production. Build Definition is the selection of updates for a new product baseline. Build Production is the production of an executable product for testing or deployment purposes. The updates checked in during the development activity are selected for a new product baseline and a new build is generated. After the build is complete, the CM manager produces a Release Delta Document that lists all the completed changes, completed activities (design elements) and fixed problems that went into the build.

Role: cmmgr

Workflow Tasks: Build Definition

1. Identifying and selecting updates for promotion
2. Rolling back an update
3. Aligning a product baseline
4. Freezing a baseline definition
5. Registering a build

Workflow Tasks: Build Production

1. Deploying a build
2. Defining a make environment
3. Editing your make environment
4. Selecting a make environment
5. Identifying a make target

6. Generating a make file
7. Running a make file

Input Deliverables: Source Code

Output Deliverable: Release Delta Document

Verification

Verification is an integration testing activity focusing on the changes introduced during the development cycle rather than a complete system test of all features.

Role: verif

Workflow Tasks:

1. Replying to a problem report
2. Adding and modifying a problem report.

Input Deliverables: Release Delta Document

Output Deliverable: Test Report

Cycle Assessment

Cycle assessment is the formal review of the results of a development cycle. The project manager along with the lead designer or architect review the changes completed in the development cycle as well as the test results and determine if an additional cycle is required or if the design can be closed. Any information elements ready for system testing are promoted to the status of sitest. This includes updates, activities (design elements) and problems and builds.

Role: prjmgr, cmmgr

Workflow Tasks:

1. Identifying and selecting updates for promotion
2. Rolling back an update
3. Adding and modifying a problem report
4. Adding and modifying an activity.
5. Adding and modifying change requests

Input Deliverables: Release Delta Document, Test Report

Output Deliverable: Release Status Report

3.3.2 Construction Phase Deliverables

External Deliverables

There are no external deliverables to the construction phase. All inputs come from the elaboration phase.

Workflow Input Deliverables

The following deliverables are required inputs from the elaboration phase.

Project Plan,

Test Plan,

Release Plan

Workflow Produced Deliverables

The following deliverables are produced as part of the construction phase workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See Section 2.1 for a description of each deliverable.

Project Plan: generated from project activities using the Project Definition report.

Test Cases: internally tracked as test case records in CM+

Source Code: internally tracked as files in CM+

Release Delta Report: generated from updates, problems and activities associated with a build

Release Status Report: Tracked document per development cycle

Test report: Tracked document per development cycle

3.3.3 Construction Phase Milestones

Design Close (DC)

The design close milestone marks the completion of all planned development activities in preparation for a release. Essentially, the design is functionally complete. The gate conditions for the design close milestone are as follows:

1. Review and approval of the completion status of development: all activities, problems and updates in the planned release are at status sitest. This applies to all products and sub-products originally planned for this release.
2. Review and approval of the planned requirements coverage. All requirements planned for this release are covered by design element activities at status sitest, and test cases at status approved.
3. Review and approval of the test plan.

Upon completion of the DC milestone, the build being planned for release is promoted to sitest and the associated product baseline is frozen.

If the gate conditions are not met, another development cycle is initiated.

3.4 Assurance Phase

In the assurance phase, the product is prepared for general release to your customer through system testing followed by limited operation by end users. Limited operation could include alpha and beta testing, or parallel operation with a legacy system. A certain amount of product rework is expected before the product is deemed ready for formal release.

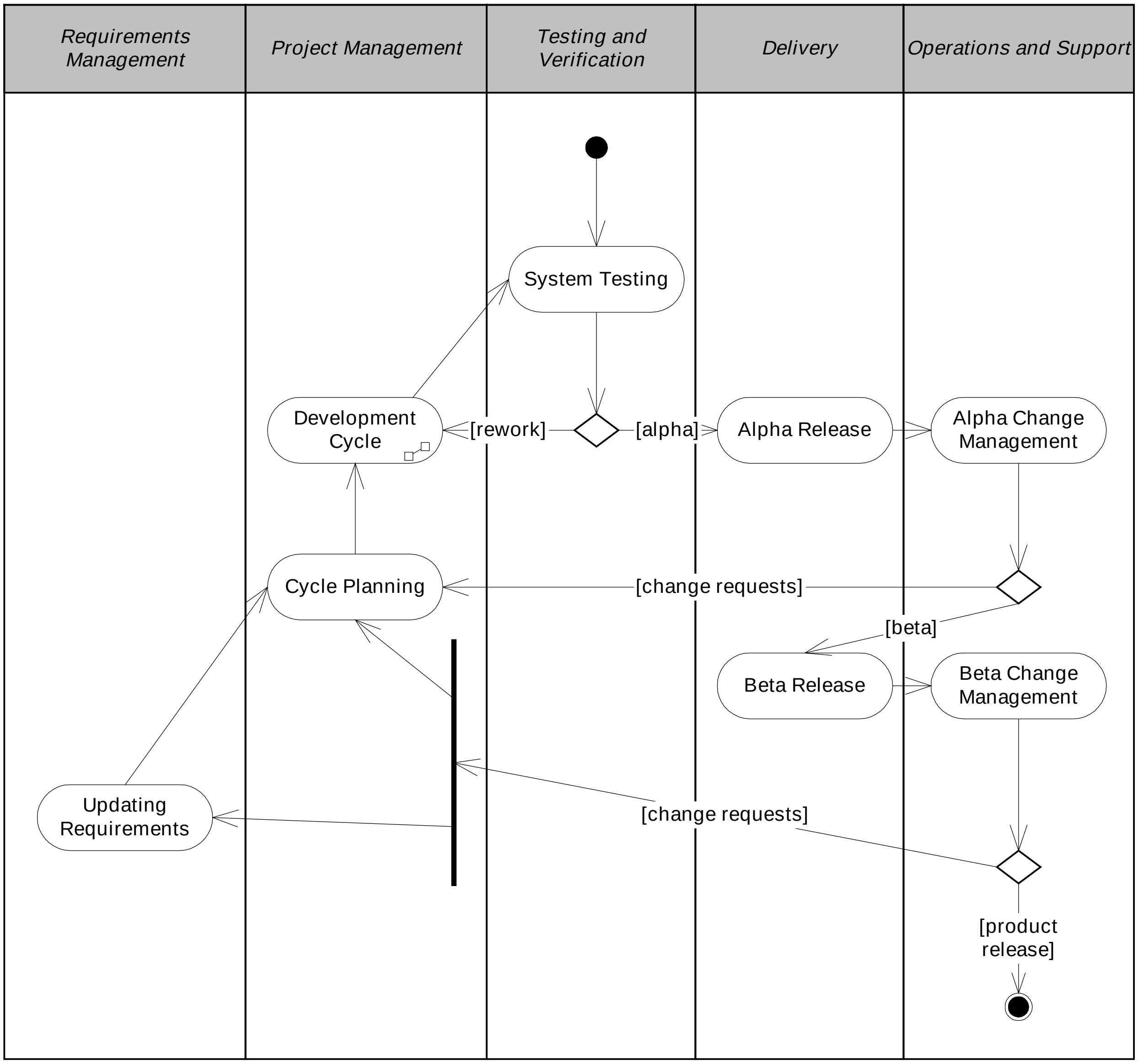

Figure 7 Assurance Phase Activity Diagram

3.4.1 Assurance Phase Activities

System Testing

System testing is the end to end software testing activity required to verify all features. System testing typically follows the test plan and results in a test report and a number of problem reports being raised. At the completion of system testing, updates, activities and problem reports that pass are promoted to the status of svtest.

Role: verif

Workflow Tasks:

1. Adding documents
2. Adding and modifying a problem report
3. Replying to a problem report

4. Promoting problem reports
5. Identifying and selecting updates for promotion

Input Deliverables: Test Plan, Test Cases, Release Delta Report

Output Deliverable: Test Report

Development Cycle

The development cycle is a set of activities that are completed in order to rework and re-release the software. A development cycle will include some or all of the following Construction Phase activities, but typically requires minimal re-design.

Software Design

Test Case Design

Component Design

Development

Building

Verification

Alpha Release, Beta Release

The release activities are the preparation of the software and any accompanying documentation, training, licensing materials etc. to the testing users (alpha, beta, migration team, parallel operation etc.)

Role: cmmgr

Workflow Tasks:

1. Adding documents
2. Approving a document

Input Deliverables: Test Report, Release Delta Report

Output Deliverable: Release Documents, Installed Software

Alpha Change Management, Beta Change Management

Change management is the process of supporting the users and collecting change requests and problem reports for input to a new development cycle. After a predetermined period of time, the change requests are reviewed (typically by a Change Control Board) and selected for input to a rework cycle.

Role: support

Workflow Tasks:

1. Adding and modifying change requests
2. Approving change requests

Input Deliverables: Installed Software, Change Management Plan

Output Deliverable: Change Requests

Updating Requirements

This is a requirements management activity to ensure that the current set of requirements match the changes being applied through the change requests. Additions or changes to the requirements may necessitate additional design and test case definition activities in the subsequent development cycle.

Role: prjmgr

Workflow Tasks:

1. Adding requirements (Enterprise).
2. Linking requirements (Enterprise)
3. Changing requirements (Enterprise)
4. Creating a requirements baseline (Enterprise).
5. Producing a requirements document (Enterprise).

Input Deliverable: Change Requests

Output Deliverable: Requirements Document

Cycle Planning

Cycle planning in the assurance phase is primarily the assignment of specific change requests to designers for completion in the target release stream. Depending on the scope of the changes, additional design work and test cases may be required, necessitating the addition of activities to the project plan.

Role: prjmgr

Workflow Tasks:

1. Assigning change requests
2. Adding and modifying an activity.
3. Moving an activity.
4. Assigning an activity
5. Assigning a problem

Input Deliverables: Change Requests

Output Deliverable: Project Plan

3.4.2 Assurance Phase Deliverables

External Deliverables

The following deliverables are external inputs to the development lifecycle and are typically in existence prior to the start of a project. They can be tracked in CM+ (Enterprise version only); however no templates are provided for them. See Section 2.1 for a description of each deliverable.

Change Management Plan

Workflow Input Deliverables

The following deliverables are required inputs from the construction phase.

Release Delta Report,

Test Cases.

Workflow Produced Deliverables

The following deliverables are produced as part of the assurance phase workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See Section 2.1 for a description of each deliverable.

Requirements Document: generated from requirements tracked in CM+ (Enterprise version only) using the Requirements report.

Project Plan: generated from project activities using the Project Definition report.

Test Cases: internally tracked as test case records in CM+

Source Code: internally tracked as files in CM+

Change Requests: internally tracked as change request records in CM+

Test report: Tracked document per system test cycle

Release Documents: Tracked documents containing release notes and associated information.

3.4.3 Assurance Phase Milestones

Alpha Release (alpha),

The alpha release milestone represents the completion of a full cycle of system testing and the readiness of the product for release to a limited number of users. The gate conditions for the alpha release are as follows:

1. Review and approval of the open problem reports remaining in release stream: no high priority problems.
2. Review and approval of the completion status of rework: all activities, and updates added during the rework cycles are at status svtest. This applies to all products and sub-products originally planned for this release.

Beta Release (beta), General Release (GR)

The beta and general release milestones represent product readiness gates. Achieving these milestones requires a subjective evaluation of the quality, validity and completeness of the product being released. Every project has different criteria. The following gate conditions apply:

1. Review and approval of the open problem reports remaining in release stream: no high or medium priority problems.
2. Review and approval of the change requests open against product release: for beta, no new, high priority change requests; for general release, no high or medium priority change requests.
3. Review and approval of the planned requirements coverage. All new requirements added during alpha or beta testing are covered by design element activities at status svtest, and test cases at status approved.
4. Review and approval of the project activities. All planned system testing activities are at status finished and all open design activities have been moved to a post release construction cycle.

3.5 Delivery Phase

This phase focuses on the activities required to place the software into the hands of the users. Considerable effort is expended in developing user-oriented documentation, training users, supporting users in their initial product use, and reacting to user feedback. In some projects, a predefined set of services (integration, data migration etc.) coincides with the product delivery. The completion of these services, where the software is being set to work, may trigger a number of change requests, requiring a maintenance release.

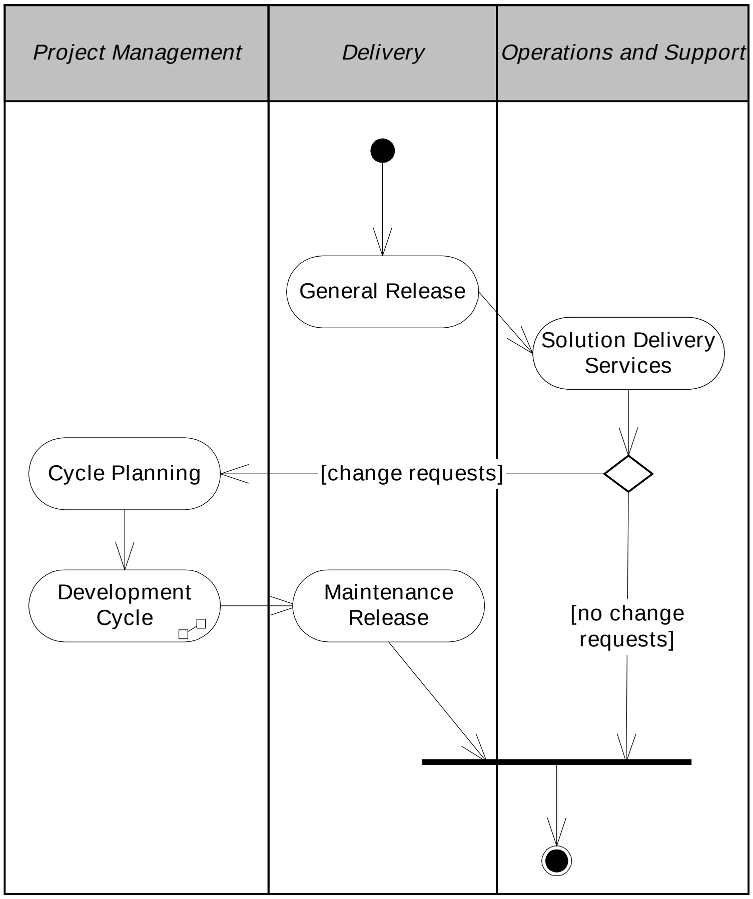

Figure 8 Delivery Phase Activity Diagram

3.5.1 Delivery Phase Activities

General Release

The general release is the preparation, packaging and delivery of the software and any accompanying documentation, training, licensing materials etc. to the general end users (specific customer, solution delivery team, direct sales team, distributers etc.)

Role: cmmgr

Workflow Tasks:

1. Adding documents
2. Approving a document

Input Deliverables: Release Delta Report

Output Deliverable: Release Documents, Installed Software

Solution Delivery Services

These are the professional services activities that may or may not required with every product delivery. The activities themselves are tracked outside of CM+; however, any new change requests are added and managed within CM+. The change requests are reviewed (typically by a Change Control Board) and selected for input to the maintenance release cycle. Major changes are deferred to the inception or elaboration phases of a new project or to a separate release stream within the current project.

Role: support

Workflow Tasks:

1. Adding and modifying change requests
2. Approving change requests

Input Deliverables: Installed Software, Change Management Plan

Output Deliverable: Change Requests

Cycle Planning

Cycle planning in the delivery phase deals exclusively with the assignment of change requests and the planning of a short development cycle for a maintenance release.

Role: prjmgr

Workflow Tasks:

1. Assigning change requests

Input Deliverables: Change Requests

Output Deliverable: Change Requests

Development Cycle

A development cycle during the delivery phase is a set of activities that are completed in order to rework the software and release a maintenance release. Typically only the development, building and verification activities from the construction phase are included.

Maintenance Release

The maintenance release activity is the preparation, packaging and delivery of of the software typically with minimal additional documentation.

Role: cmmgr

Workflow Tasks:

1. Adding documents
2. Approving a document

Input Deliverables: Release Delta Report

Output Deliverable: Release Documents, Installed Software

3.5.2 Delivery Phase Deliverables

External Deliverables

The following deliverables are external inputs to the development lifecycle and are typically in existence prior to the start of a project. They can be tracked in CM+ (Enterprise version only); however no templates are provided for them. See Section 2.1 for a description of each deliverable.

Change Management Plan

Workflow Input Deliverables

The following deliverables are required inputs from the assurance phase.

Release Delta Report,

Workflow Produced Deliverables

The following deliverables are produced as part of the assurance phase workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See Section 2.1 for a description of each deliverable.

Source Code: internally tracked as files in CM+

Change Requests: internally tracked as change request records in CM+

Release Documents: Tracked documents containing release notes and associated information.

3.5.3 Delivery Phase Milestones

Maintenance Release (MR)

The maintenance release milestone is triggered only if a maintenance release was required during the delivery phase. The following gate conditions apply:

5. Review and approval of the change requests open against product release: no high or medium priority change requests.

3.6 Maintenance Phase

During the Maintenance Phase, the finished product is in the field, being supported. Depending on the change requests received from the end users, there may be several development cycles resulting in maintenance release milestones. The changes approved for these development cycles are typically bug fixes and minor enhancements. Major changes are deferred to the inception or elaboration phases of a new project or to a separate release stream within the current project.

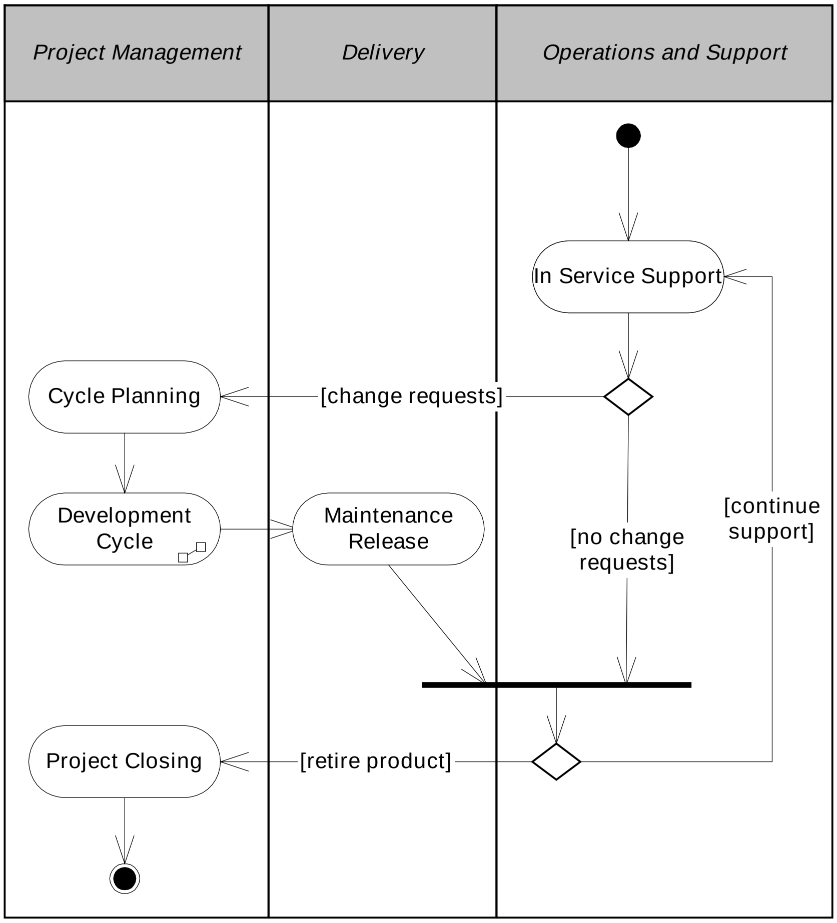

Figure 9 Maintenance Phase Activity Diagram

3.6.1 Maintenance Phase Activities

In Service Support

This is the ongoing support activity for the product in the field, with the initiation, tracking and management of change requests to the product.

Role: support

Workflow Tasks:

1. Adding and modifying change requests
2. Approving change requests

Input Deliverables: Installed Software, Change Management Plan

Output Deliverable: Change Requests

Cycle Planning, Development Cycle, Maintenance Release

These activities are identical to those in the delivery phase.

Project Closing

This activity is the final wrap-up of the project activities and the production of documents and reports that formally close the project. The project closing activity is typically repeated for each release stream until all the streams originally planned during the elaboration phase are closed. The project manager promotes all activities to closed, and ensures that all open updates, problem reports and change requests are deferred to later release streams.

Role: prjmgr

Workflow Tasks:

1. Adding and modifying change requests
2. Adding and modifying an activity
3. Adding and modifying a problem report
4. Adding documents
5. Approving documents

Input Deliverables: none

Output Deliverable: Project Close Release Documents

3.6.2 Maintenance Phase Deliverables

External Deliverables

The following deliverables are external inputs to the development lifecycle and are typically in existence prior to the start of a project. They can be tracked in CM+ (Enterprise version only); however no templates are provided for them. See Section 2.1 for a description of each deliverable.

Change Management Plan

Workflow Input Deliverables

The following deliverables are required inputs from the assurance phase.

Release Delta Report,

Project Plan

Workflow Produced Deliverables

The following deliverables are produced as part of the assurance phase workflow and are either tracked as documents (CM+ Enterprise) or generated from internally tracked information.

In the case of tracked documents, placeholder records are provided so all you have to do is check in the document when it is ready. If you are running CM+ Professional, you may track the documents externally or add and check them in as part of your source tree. See Section 2.1 for a description of each deliverable.

Source Code: internally tracked as files in CM+

Change Requests: internally tracked as change request records in CM+

Release Documents: Tracked documents containing release notes and associated information.

Project Close Release Document: Tracked document summarizing project closure information.

3.6.3 Maintenance Phase Milestones

Maintenance Release (MR)

The maintenance release milestone is triggered only if a release was required during the maintenance phase. The following gate conditions apply:

1. Review and approval of the change requests open against product release: no high or medium priority change requests.

Project Close (PC)

The project close milestone signifies the termination of the project, which usually means that one or more release streams of your product are also closed. The following gate conditions apply:

1. The Project Close Release Document is complete and approved.
2. Verify no pending software changes: All updates in the streams to be closed are at svtest.
3. Verify no pending design changes: All activities in the streams to be closed are at status closed. Pending design changes have been allocated to different streams or products.
4. Verify no pending change requests: All open change requests against the current product and stream have been either closed or allocated to different products or streams.

Appendix

Table 3 NUP Standard Template Activities

WBS	Activity Title	Discipline	Activity Type
0	<Your project title>	pm	project
1	Inception Phase	pm	subproject
1.1	Product visioning	rm	task
1.2	Project risk assessment	pm	task
1.3	High level project planning	pm	task
1.4	Business case analysis	b	task
1.5	Business Case	b	milestone
2	Elaboration Phase	pm	subproject
2.1	Requirements definition	rm	task
2.2	Architecture Definition	ad	task
2.3	Analysis Close	ad	milestone
2.4	Release planning	cm	task
2.5	Test planning	tv	task
2.6	Software development planning	pm	task
2.7	Software development plan approval	pm	milestone
3	<Stream A>	pm	stream
3.1	Construction	cm	subproject
3.1.1	Cycle Planning	pm	task
3.1.2	<Cycle 1>	pm	subproject
3.1.2.1	<Design Element 1>	ad	design
3.1.2.1.1	Software Design	ad	task
3.1.2.1.2	Test Case Design	tv	task
3.1.2.1.3	Component Design	cm	task
3.1.2.1.4	Development	i	task
3.1.2.2	<Cycle 1> Building	cm	task
3.1.2.3	<Cycle 1> Verification	tv	task
3.1.2.4	<Cycle 1> Assessment	pm	task
3.1.3	Design Close	ad	milestone
3.2	Assurance	pm	subproject
3.2.1	System testing	tv	task
3.2.2	Alpha release	d	task

WBS	Activity Title	Discipline	Activity Type
3.2.3	Alpha release (milestone)	pm	milestone
3.2.4	Alpha change management	os	task
3.2.5	Beta release	d	task
3.2.6	Beta release (milestone)	d	milestone
3.2.7	Beta change management	os	task
3.2.8	Cycle Planning (rework)	pm	task
3.2.9	Development Cycle (rework)	ad	task
3.2.10	General Release (milestone)	pm	milestone
3.3	Delivery	pm	subproject
3.3.1	General Release	d	task
3.3.2	Solution Delivery Services	os	task
3.3.3	Cycle planning (maintenance release 1)	pm	task
3.3.4	Development cycle (maintenance release 1)	ad	task
3.3.5	Maintenance release 1	d	task
3.3.6	Maintenance release 1 (milestone)	pm	milestone
3.4	Maintenance	pm	subproject
3.4.1	In service support	os	task
3.4.2	Cycle planning (maintenance release x)	pm	task
3.4.3	Development cycle (maintenance release x)	ad	task
3.4.4	Maintenance release x	d	task
3.4.5	Maintenance release x (milestone)	pm	milestone
3.4.6	Project closing <Stream A>	pm	task
3.4.7	Project closing <Stream A> (milestone)	pm	milestone

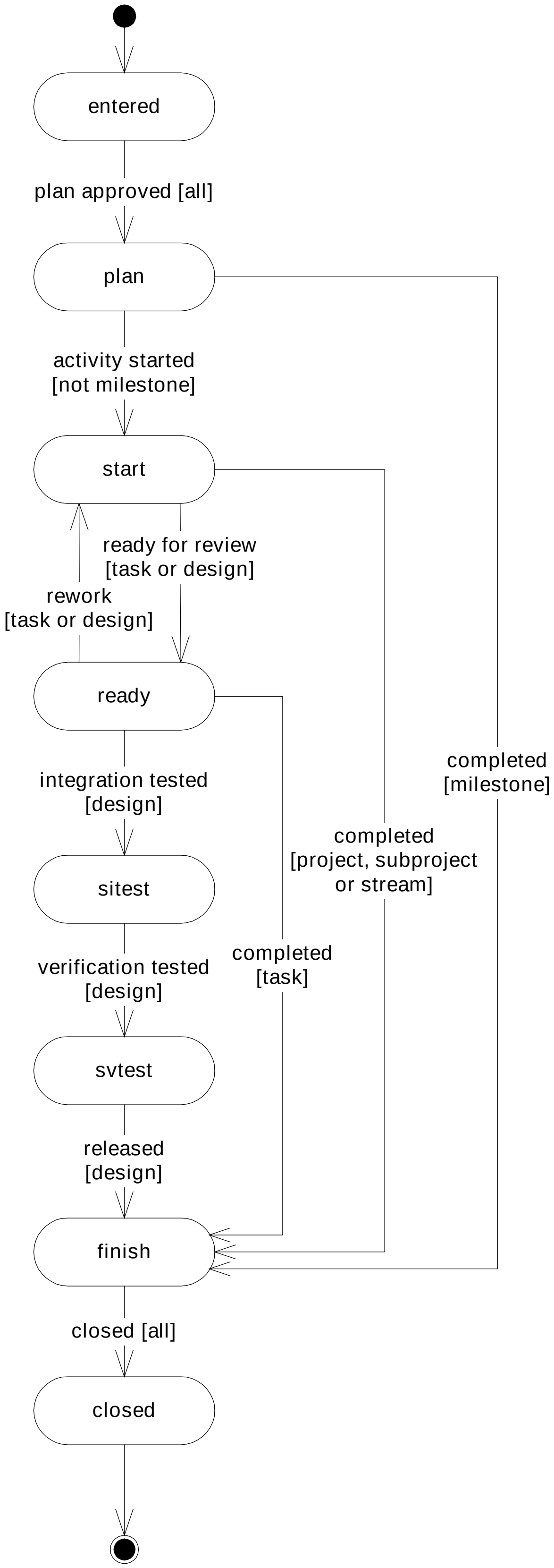

Figure 10 Activities State Transitions

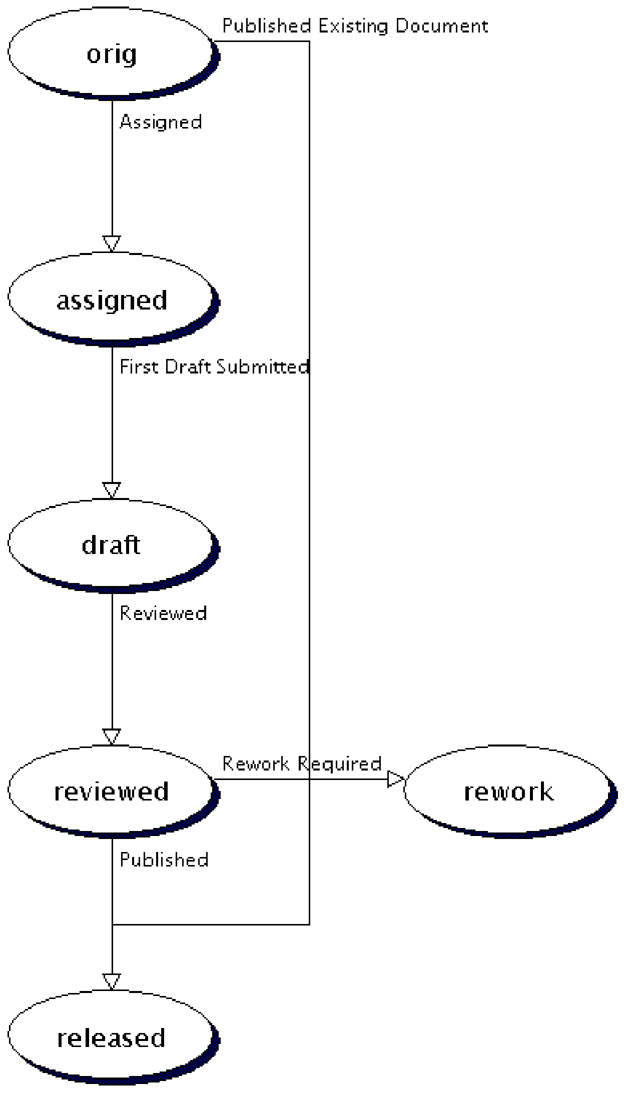

Figure 11 Documents State Transitions

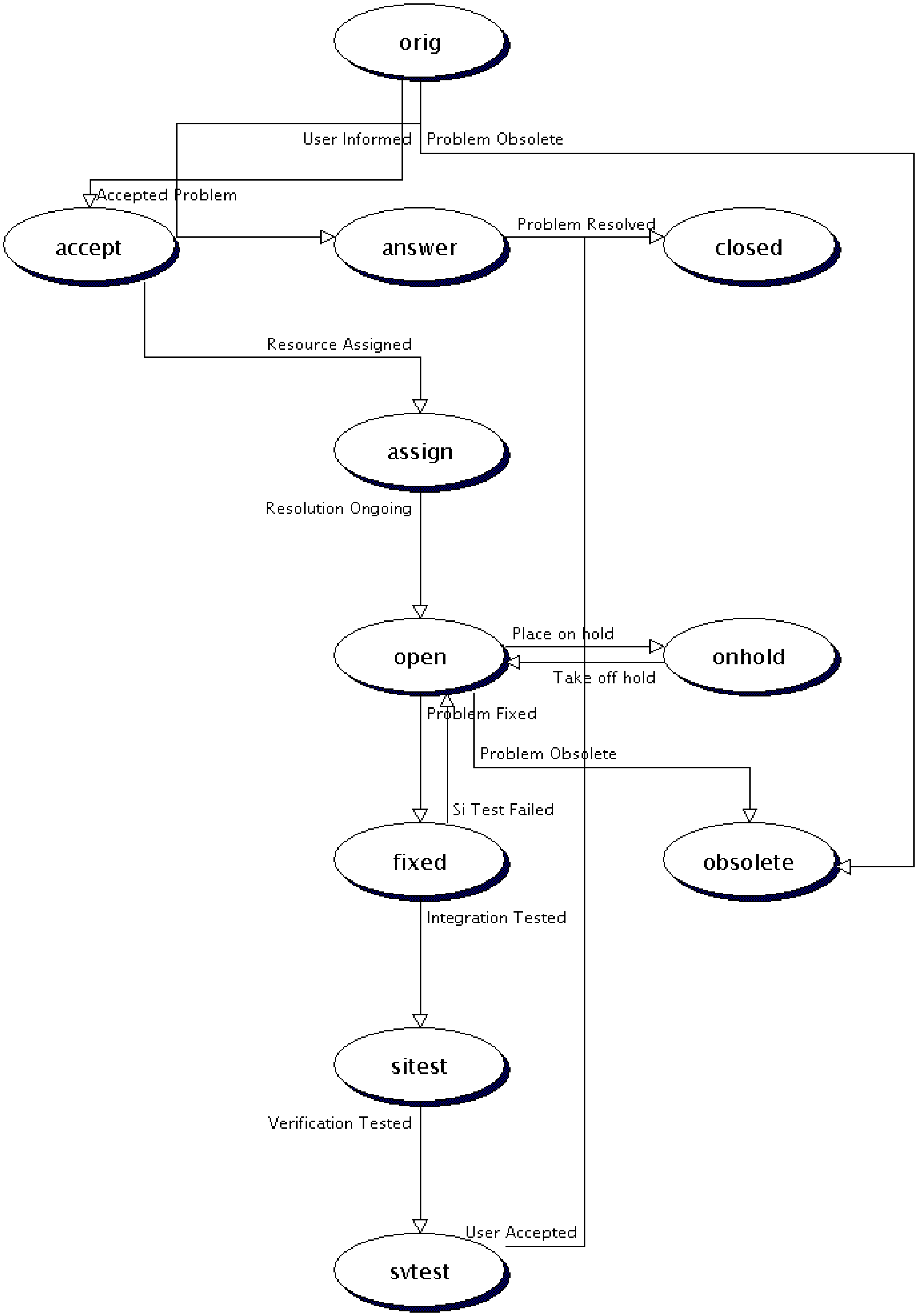

Figure 12 Problems State Transitions

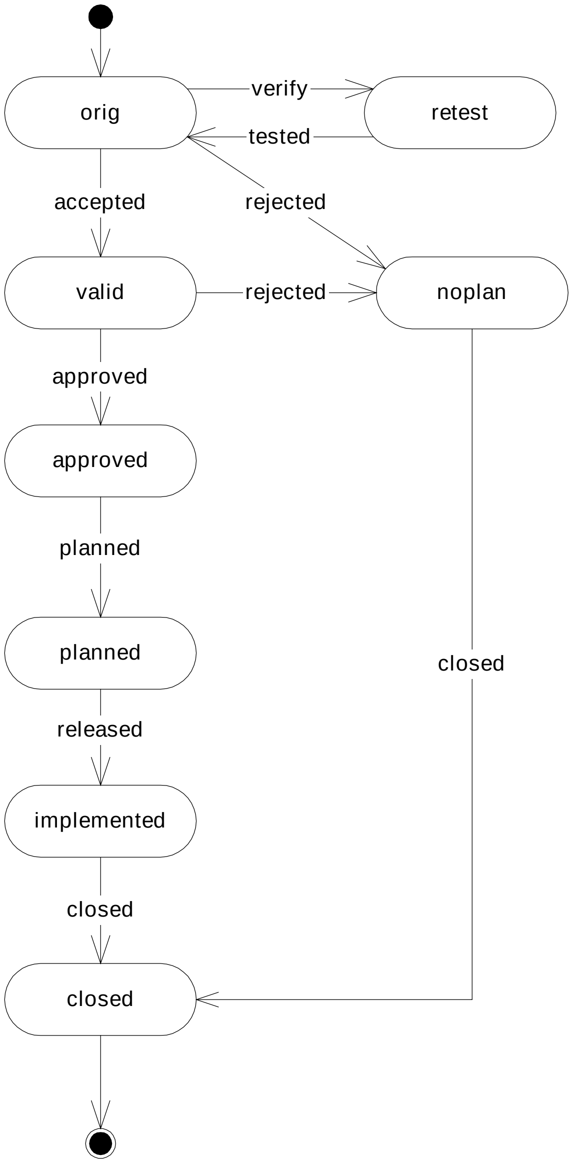

Figure 13 Requests State Transitions

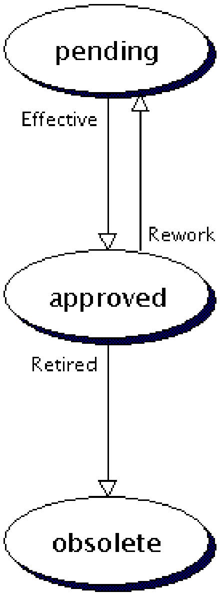

Figure 14 Requirements State Transitions

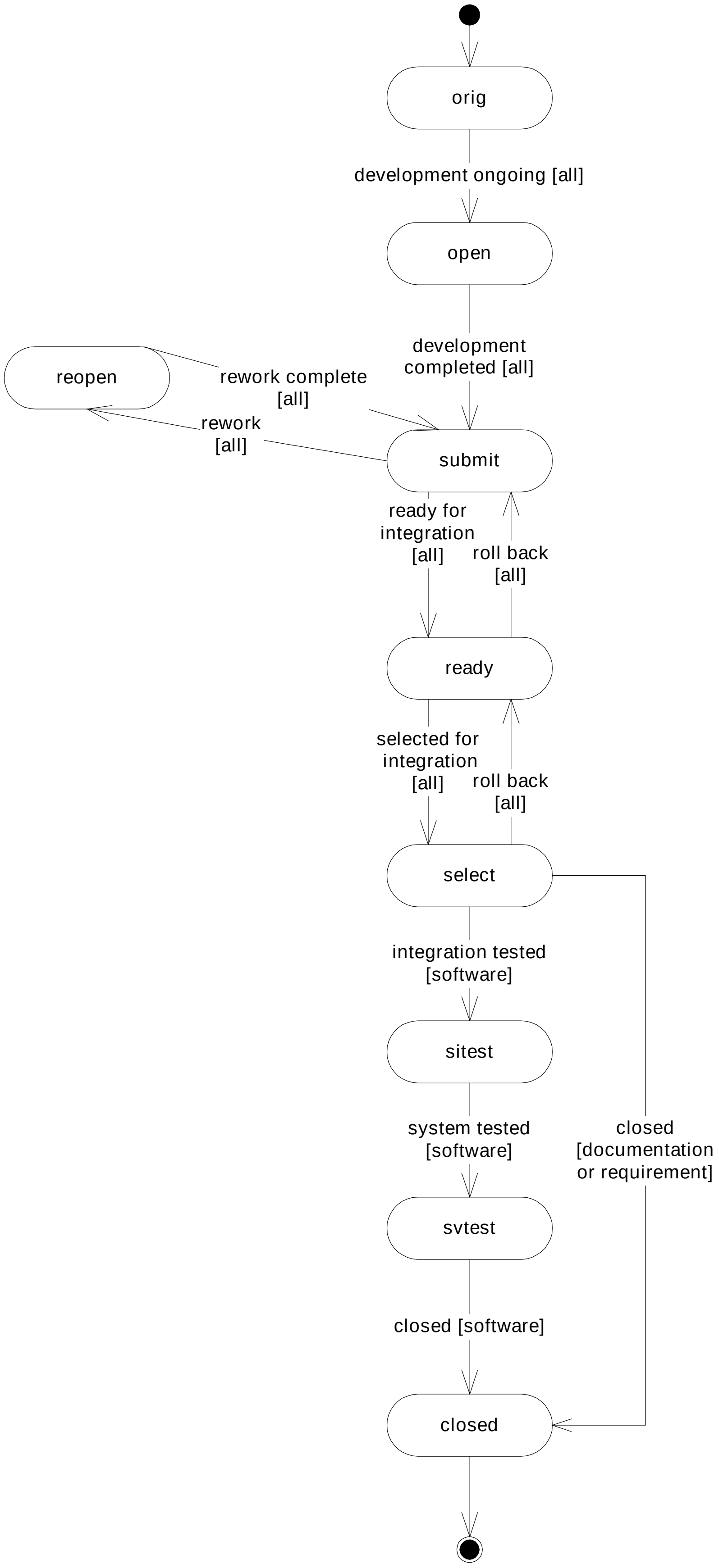
orig
development ongoing [all]
open
development completed [all]
reopen
rework complete [all]
rework [all]
submit
ready for integration [all]
roll back [all]
ready
selected for integration [all]
roll back [all]
select
integration tested [software]
sitest
system tested [software]
svtest
closed [documentation or requirement]
closed [software]
closed

Figure 15 Updates State Transitions