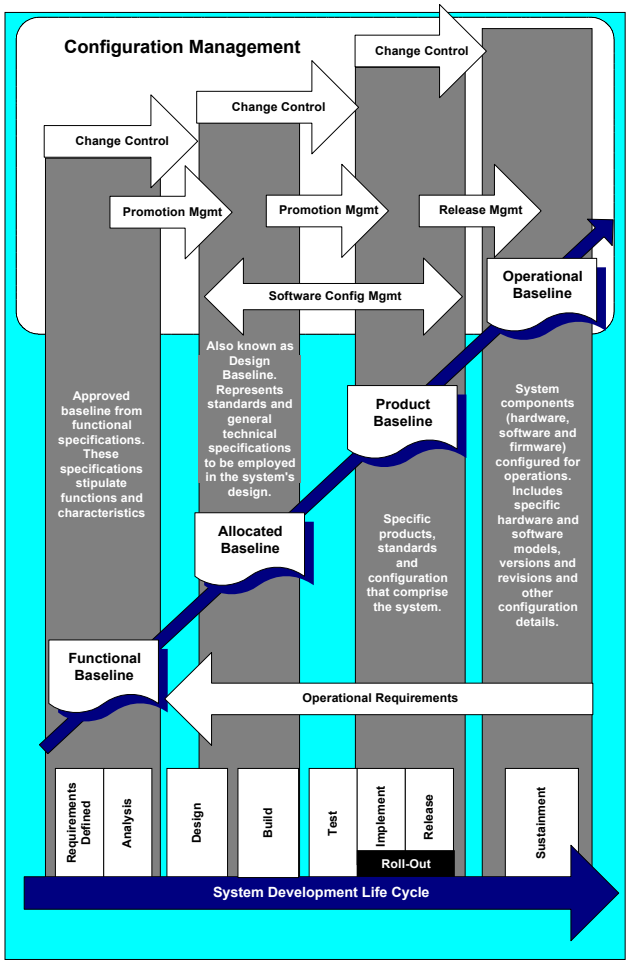
Configuration Management - The Big Picture

What

Consists of:

1. Product Baselines
   - aligned to system development life cycle
   - comprised of hardware and software configuration items
   - described by specifications, design documentation, change orders and modifications
2. Key components:
   - requirements, functional requirements specifications, technical specifications, service level objectives and other inputs into design
   - operational requirements
   - design documentation
   - hardware and software configuration items
   - system, technical and user documentation
3. Policies, processes and procedures for:
   - managing and controlling changes to key components
   - assuring the integrity of the product at each applicable stage of the system development life cycle
   - tracking the configuration of each baseline
   - tracing source of baseline configurations (requirements, design documentation, change requests, release notes, etc.)
   - defining how changes are tested, certified and released into production
4. Auditing and reconciliation:
   - status accounting - operations baseline compared to product baseline + approved changes
   - root cause determination for differences that cannot be traced back to approved change requests
   - operational baseline and product baseline reconciled

Overview

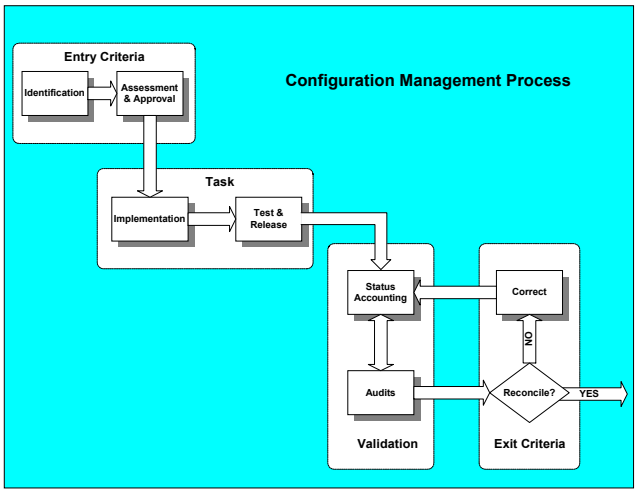


How

Process (Entry-Task-Validation-Exit Model):

1. Entry Criteria
   - *identify* changes that need to be made (new features, code changes to correct defects, changes to business rules, etc.)
   - *assess* impact of change and *approve* change for implementation and release to operational baseline (production environment)
2. Tasks
   - *Control* promotion from baseline-to-baseline and release into production
   - *manage* deliverables and artifacts - documentation and code
   - *implement* in pre-production or staging environment
   - *test* to ensure that all *release* criteria are met
3. Validation
   - perform *status accounting* by *auditing* operational baseline and comparing it to the product baseline + approved changes
   - ensure that all changes are traceable and approved
4. Exit Criteria
   - Does the operational baseline reconcile with the product baseline?
   - No - determine why, then eliminate root cause of discrepancy/discrepancies and synchronize the operational and product baselines (all configuration items)
   - Yes - exit criteria for current status accounting action, to recur when the next scheduled status accounting milestone occurs

ETVX Model of the Configuration Management Process



Change Management

What

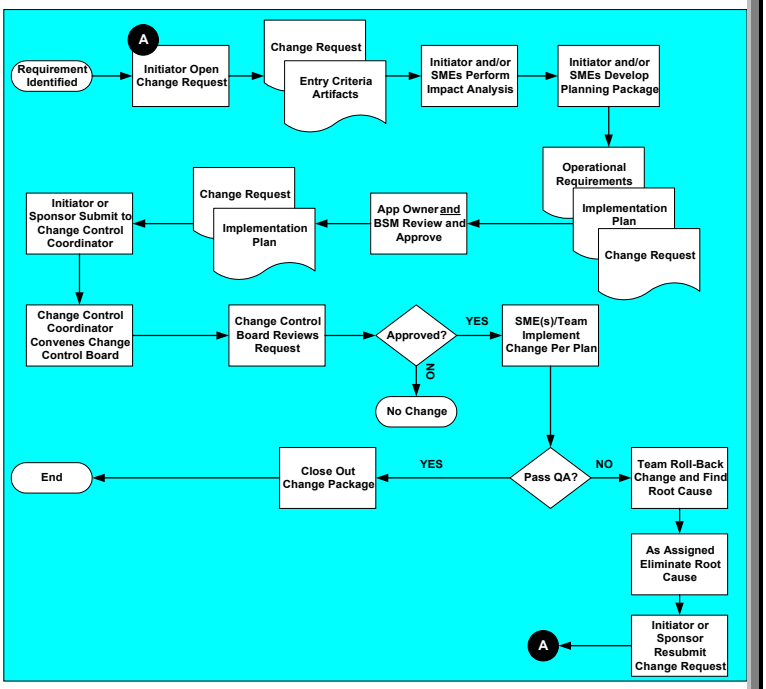
Change management is the exercise of control over any changes to specifications and requirements, documentation, and hardware and software configuration items of a system, subsystem or component.

The most critical control point for changes is the production system (known as the operational baseline). This control point encompasses the hardware platform, operating system, database management system interfaces and applications that comprise a specific system. Any changes to a specific system configuration are either a modification, a patch or a version promotion. Modifications are changes to the base system, such as an enhancement or configuration change. A patch is an interim change to the code base to correct issues or add features to software that are too minor to be considered as a version promotion.

Change control from product baseline (pre-production/ staging) to the operational baseline (production system) is accomplished through the change control process that is managed by the change control coordinator and overseen by the change control board. The objective is to support release management.

Change control from the functional and allocated baselines to the product baseline is managed by the project and/or product manager. This aspect of change management employs software configuration and document management to assure the integrity of the system or product. The objective is to support promotion management.

Process - Production Environment Focus



How

Sequence of Events:

1. Identified requirement - new system or version upgrade, modifications and patches, backend or infrastructure changes, maintenance window or service level objective changes, etc.
2. Change Request
3. Impact analysis - benefit of the change, how the change affects service levels, business operations and users (when the change is being implemented and after the change has been released to production), inter- and intra-system dependencies, interfaces, and other special considerations (including adherence to enterprise standards, maintenance and support requirements, etc.)
4. Planning package - entry criteria (release notes, installation manual, operational requirements, etc.), pre-implementation testing and validation, implementation plan (timeline, key personnel, problem escalation, roll-back plan and maintenance window requirements, quality checkpoints, communications/notification checkpoints and post implementation verification plan)
5. Key stakeholder approval - application owner and business systems manager sign-off of planning package
6. Submission of change control package to change control coordinator
7. Review and approval by change control board
8. Implementation
9. Change request close-out

Benefits

Business

1. All changes will be evaluated for how the implementation affects business operations, systems availability and attainment of service level objectives.
2. No change will be made to production systems unless approved by application owner(s).
3. Proposed changes will be clearly articulated to the application owner and users with respect to why the change is being made and the benefits that will be realized by making the change.

Technical

1. The process requires that any proposed change be analyzed for how it will affect upstream and downstream systems (inter- and intra-system dependencies), interfaces and other special requirements (maintenance and support needs going forward after the change, technical training, etc.).
2. System integrity will be safeguarded by examining how the proposed change conforms to organizational standards (technical and administrative).
3. All stakeholders and subject matter experts/technical domain managers will have an opportunity to determine if the change will affect their areas of cognizance (occurs when presented to change control board).
4. Changes will be made in a controlled, methodical manner (implementation plan, QA plan, roll-back plan, etc.)

Software Configuration Management

What

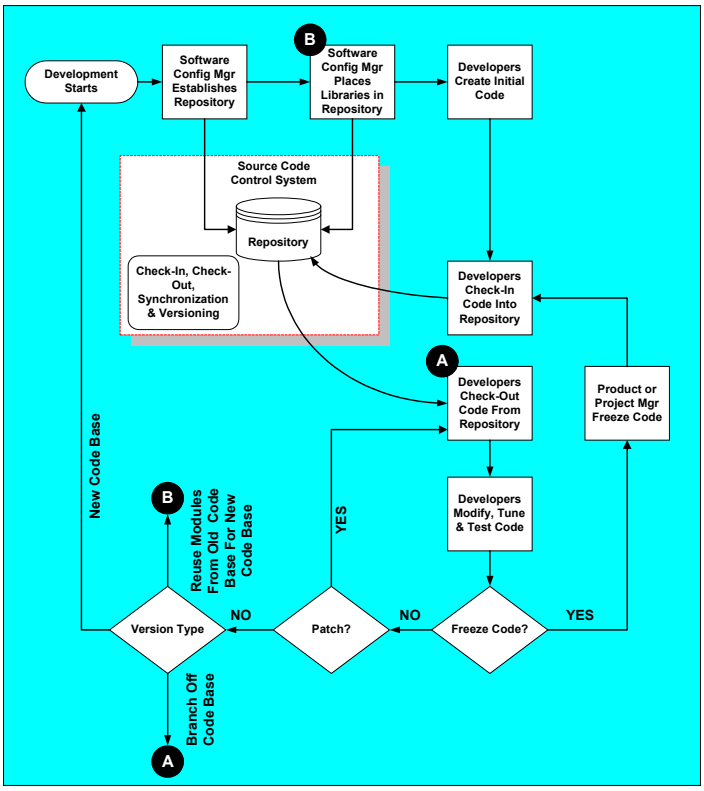
Software configuration management is the control of all software configuration items, off-the-shelf or developed by internal development. The objective is to assure product integrity. This is accomplished through change control at the software configuration item level. The primary control mechanism is code check-in/ check-out processes and procedures, versioning and tracking deltas of all software configuration items in the code base.

Key elements:

- source code control system repository
- code check-in and check-out management
- synchronization of changes
- version management
- build tracking (what source code versions are in object code, which object code versions are linked into the final product)

Software configuration management also supports the control of promotion of code from one baseline to the next, as well as release to production of the finished product by providing identification of every component that comprises the software configuration items, including version, and code base branches of related versions and patches.

Process



How

Sequence of Events:

1. Code base is established - as source code is developed it is placed in repository
2. Source code is checked-out for further work, unit and integration testing
3. Source code is checked-in and the check-in/check-out cycle continues until the code is frozen - changes are tracked from one revision to the next (at the source code module level)
4. After all source code is ready for a build (all source code modules are compiled and linked) a revision is assigned to the build and tracked against the configuration items that make up the build
5. Changes made to this code base are performed as a reiteration of steps 2-4 and the version is either incremented along the same product tree or branched into a different product tree until the code base is frozen and ready for promotion to the product baseline (product test/UAT and/or pre-production/staging)
6. Additional changes to the code base to rectify issues (defects) that arise in the product baseline are resolved using steps 2-5.
7. When the code base is ready for release to the operational baseline (production environment) the code base is frozen.
8. Issues that require patches to the code base to resolve severity 1 or 2 problems will employ steps 2-6 to manage and assure product integrity.
9. Product evolution to include new features or requested enhancements and/or resolve severity 3 and 4 issues will either (1) use the existing code base or modules from existing code base per steps 2-8, or (2) new product from scratch per steps 1-8.

Benefits

Business

1. Product reliability - assurance that the product has been built from the correct versions of each component, which eliminates a source of failure that can occur if an older component is inadvertently used in a build that is released into production
2. Quicker turnaround on fixes because components are in a single repository - also supports more accurate build analysis and identification and elimination of dependencies that could cause a cascade of failures

Technical

1. Systematic tracking of all component versions reduces re-work that could occur if the wrong component is included in a build
2. Reduces troubleshooting and backtracking to determine root cause of problems introduced with an out-of-sync component
3. Positive control over revisions by enforcing check-in and check-out
4. Supports accurate build analysis
5. Allows version branching into different family trees of the same base product
6. Provides for orderly promotion and release of product from one baseline to another
7. Tracks patches to an existing code base, which provides hooks into operational requirements and issue management

Promotion and Release Management

What

Promotion management is the orderly migration of a product from functional-to-allocated-to-product baseline.

Release management is a collection of safeguards (exit criteria) for releasing a software configuration item from the product baseline (pre-production/staging) to the operational baseline (production environment).

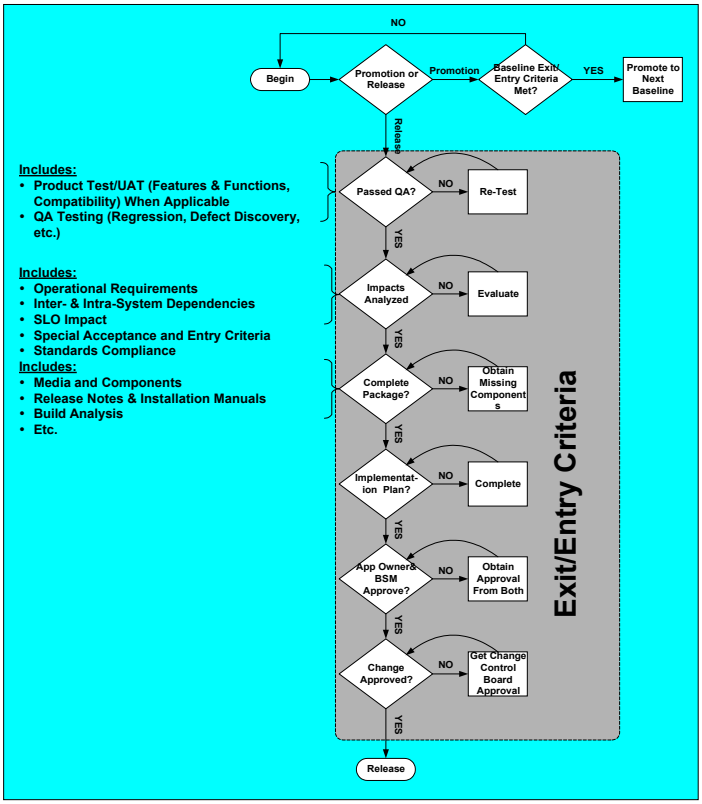
Promotion and release management employs a defined set of entry and exit criteria that must be met for any type of configuration item (hardware or software) to be promoted up the baseline chain.

Release management crosses project/product management, production support and business process domains. The critical point is the release of any change into production environment.

This safeguard is to ensure that any change is controlled and does not adversely affect business operations or compromise the ability to meet service level objectives.

Promotion of changes into baselines below the production environment is contained within the project and product management domain.

Process - Release Management Focus



How

Promotion Management:

1. Processes and procedures that are an integral part of the management of development projects as a product evolves from function-to-allocated-to-product baselines
2. Provides control through the use of document management, traceability techniques, quality assurance and software configuration management
3. Uses entry and exit criteria that set conditions that must be met before any hardware or software configuration item can be promoted from one baseline to the other

Release Management:

1. Processes and procedures to assure that hardware and software configuration items have been: thoroughly tested, evaluated for impacts to service level objectives and business operations supported by the system to which the release is being made, and inspected for completeness (all components are present, release and installation notes provided, etc.)
2. An implementation and roll-back plan is in place
3. All operational requirements and acceptance and/or entry/exit criteria have been met
4. The impacts to the production system and any interfaces to other systems or infrastructure components are understood and approved by domain experts (network and system administrators, DBAs, etc.)
5. Both the application owner(s) and business systems manager(s) have approved the release of the change into the production environment
6. The change to be released has been approved by the change control board

Benefits

Business

1. Less defects will escape into the production environment, providing a more stable production system
2. Implementations will be planned and executed within the context of meeting service level objectives
3. Releases of any change into the production environment will not occur until application owner(s) have been provided with an impact assessment of the change and the application owner agrees to implementing the change
4. Application owners control when changes are made

Technical

1. Framework to manage existing production baselines (software and hardware configuration items) with respect to patches and fixes, and versions and upgrades
2. Defined criteria and processes for promoting from one life cycle milestone to the next and release into production
3. IT and business shares the risks associated with releasing a change into the production environment

Operational Requirements

What

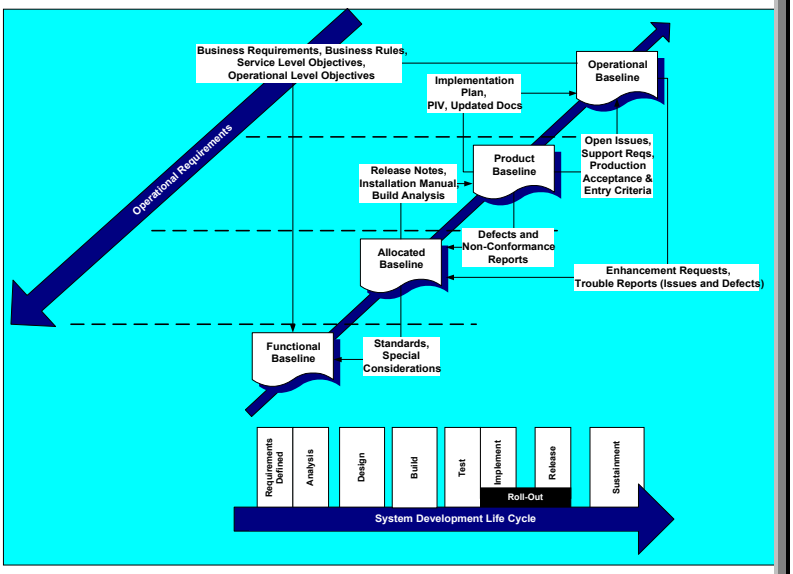
Operational requirements are a set of needs that define the product from a service level, business requirements and rules and operational level. The needs flow down from the production environment (business process domains) and supporting domains, and are expressed as:

1. Service and operational level objectives, each of which are incorporated into service and operational level agreements
2. Business rules and enhancement requests with which to align or re-align the product to business processes and strategic directions
3. Standards to which the product must conform (i.e., enterprise technical standards such as operating system or platform compatibility, development standards such as languages to be used, and business standards such as accounting or computation rules
4. Defects that need to be removed from the product
5. Special considerations, such as how the system will be instrumented for enterprise management, interfaced to other systems, etc.

Some operational requirements flow up from the development, delivery and support domains, and comprise entry criteria from one life cycle milestone to the next. These Include:

1. Release notes, installation manuals, updated documentation
2. Test results and open issues
3. Acceptance criteria (application acceptance, user acceptance, help desk entry criteria, post implementation validation [PIV],etc.)
4. Change control, implementation plans, etc.

Process



How

The management of operational requirements requires:

1. A formal requirements elicitation process that examines requirements as a holistic model and captures and manages the requirements throughout the life cycle
2. A "Product Roadmap" which will guide the development and evolution of a specific product (typically an application) to ensure that the product is, and remains, aligned to the business process(es) which the product supports. The initial roadmap is the design specifications derived from requirements. The roadmap is iterative and is modified to reflect enhancements approved to support emergent business requirements based on enterprise strategies and evolving business process methods and procedures
3. The ability to trace back all product characteristics to functional and operational requirements, and the ability to trace back functional and operational requirements to business objectives
4. A formal change management process with which to promote from one life cycle milestone to the next and to release into the production environment all fixes and enhancements using entry and exit criteria and validation checkpoints
5. Methods and mechanisms to assure the integrity of the product; i.e., software configuration management, requirements management, issue management, etc.

Benefits

Business	Technical
1. Business drives the technical solution vs. adapting business processes to technology	1. The technical solution (product) is viewed as a complete architecture with defined dependencies and interfaces at the earliest possible life cycle milestone
2. Assurance that products developed, enhanced or fixed meet all requirements to support business processes and objectives	2. Support requirements are factored into the product during the requirements milestone, allowing accurate total costs of ownership to be developed early into the life cycle
3. Requirements upon which the product design and construction is directly traceable to business objectives	3. Well defined entry and exit criteria for each life cycle milestone
4. Processes exist to ensure that the product evolves to support emergent business requirements and strategies, which directly support competitive advantage goals and objectives	4. Entry/exit criteria support task/deliverable approach for planning and estimating, scheduling and controlling the development project
	5. Entry/exit criteria provides a foundation for a repeatable service delivery and support process