



15/08

Software

¿Qué entendemos por Software?

- Es un conjunto de programas, procedimientos, reglas, documentación, datos.
- Es información que se presenta en distintos niveles de **abstracción**.
- Todos los resultados de las **actividades** del proceso de desarrollo de software son artefactos. Cada una de esas salidas es **conocimiento**.

abstracción → Está relacionado inversamente con el nivel de detalle. Los requerimientos de negocio o usuario son los que más nivel de abstracción tienen.

actividades → Workflow de análisis, diseño, prueba y despliegue.

Software en contexto



- Un **proceso** se instancia en un **proyecto** de desarrollo de software.
- Cada proyecto tiene como resultado un **producto** de software.
- Para lograrlo, se incorporan **personas** que trabajarán en conjunto para ello.
- Para lograr un trabajo más eficiente, se intenta automatizar los procesos a través de diversas herramientas.

Cambios en el Software

El Software está en continuo cambio y mejora. Esto se debe a:

1. Cambios en el negocio que pueden resultar en nuevos requerimientos.
 2. Defectos a corregir.
 3. Oportunidades de mejora.
 4. Cambios en productos asociados.
 5. Reorganización de las propiedades de la empresa.
 6. Cambios en el presupuesto.
-
- La evolución del Software puede traer desafíos y problemas (nuevas dependencias, **deuda técnica**, la posibilidad de aumentar costos y tiempo).
 - Para gestionar con éxito los cambios del software y mitigar los posibles impactos negativos se deben implementar **prácticas de Ingeniería de Software**.
-

deuda técnica → Se refiere al costo implícito del retrabajo adicional por elegir una solución fácil en lugar de utilizar un enfoque que llevaría más tiempo en su desarrollo e implementación.

SCM (Gestión de Configuración de Software)

Concepto

- Es una **disciplina** protectora del Software que vela por mantener la integridad de los **ítems de configuración** y del **repositorio**, o sea, del producto.
- Es una actividad **transversal** a todo el proyecto, relevante para el producto a lo largo de su ciclo de vida.
- La gestión de la configuración del Software se planifica antes de arrancar un proyecto.
- Esta disciplina se tiene que ocupar del hecho de que el Software es muy fácil de editar, por eso controla sistemáticamente sus cambios, registrando las diferentes versiones y estados del Software.

- **ítems de configuración** → Cada artefacto del producto (cada elemento del Software) que pueden sufrir cambios o necesitan del equipo y sobre los cuales se necesita conocer su estado y evolución. Vienen acompañados de una versión. Cada ítem debe tener un nombre único y representativo. Pueden ser documentos de diseño, código fuente, código ejecutable, etc.

repositorio → Contenedor de IC (ítems de configuración). Mantienen la historia de cada ítem. Se emplea un modelo check-in (pull) y check-out (push). Para asegurar calidad de código podemos hacer uso de revisiones de código de pares (pull request).

versión → Forma particular de un ítem en un instante o contexto dado. (Hoy se usan herramientas de control de versionado como Git).

variante → Versión de un IC que evoluciona por separado. Representan configuraciones alternativas.

configuración de software → La sumatoria de todas las versiones de los IC en un momento dado. Cuando un ítem cambia, se trata de otra configuración de software.

línea base → Contiene 1 o más ítems de configuración que han sido revisados y controlados y podemos decir que son estables. Cuando se da esto, lo tageamos. (Un tag por línea base). Es una versión estable a la que volver si ocurre un problema. Las líneas base no se modifican ni se eliminan.

ramas → Sirven para bifurcar el desarrollo. No son persistentes en el tiempo y eventualmente serán mergeadas.

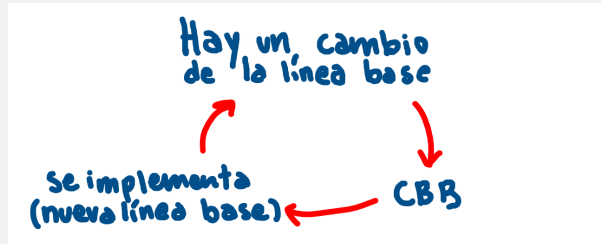
Las 4 actividades (patas) fundamentales del SCM

1. Identificación de ítems de configuración

Acá se identifican los ítems de manera unívoca y se especifican las reglas de nombrado, la estructura del repositorio, la ubicación de los ítems dentro de ella y cuándo se establecerán las líneas base.

2. Control de cambios

Una vez que la línea base se conformó, no es posible cambiarla sin pasar por un proceso formal, llevada a cabo por un comité de control de cambios (CCB)



CCB → Comité de Control de Cambios, formado por representantes de todas las áreas involucradas en el desarrollo. Analista, arquitecto y líder del proyecto es el staff mínimo. Si el cambio viene del cliente, él puede formar parte. El CCB realiza un **análisis de impacto**.

3. Auditorías de configuración de software

Revisión objetiva e **independiente** del estado de la configuración.

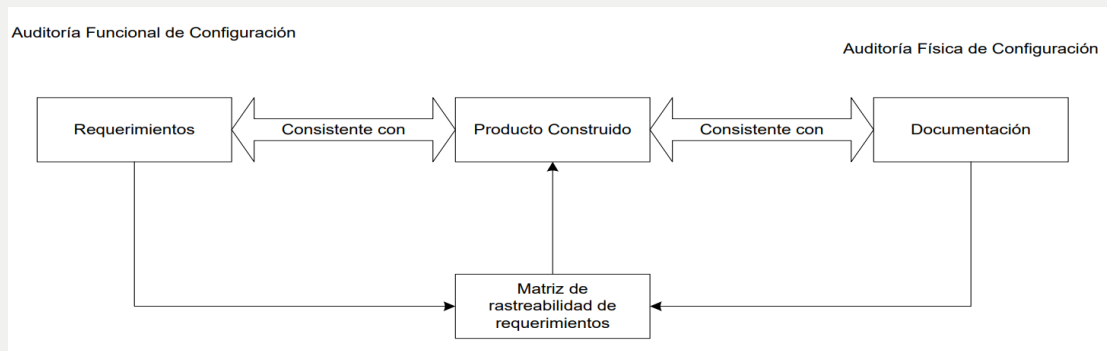
independiente → Porque no puede auditar alguien del equipo ni de la organización.

1. Auditoría de configuración física

Proceso de verificación. Vela por la integridad del repositorio (ubicación de los IC dentro del repo, permisos, reglas de nombrado, etc).

2. Auditoría de configuración funcional

Proceso de validación. Vela por que los requerimientos se vean reflejados y se puedan trazar.



4. Informes de estado

Se trata de mantener registros de la evolución del sistema. Incluye reportes de rastreabilidad de todos los cambios realizados a las líneas base durante el ciclo de vida.