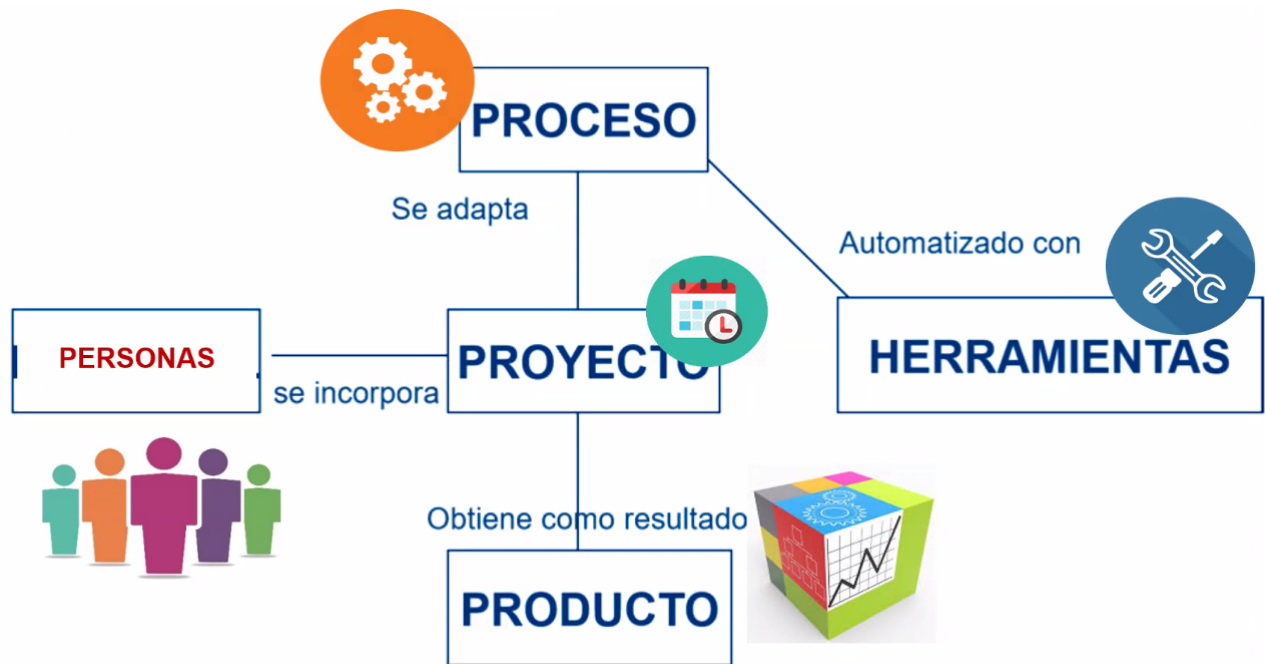


ISW 16-8

Los procesos empíricos tienen forma de trabajar diferentes.

Software Configuration Management (SCM)

Software en contexto



El proceso se instancia o cobra vida a través de un proyecto. Una vez instanciado en uno o n proyectos (cada uno es único). Cada proyecto es llevado a cabo o intervienen personas. La corriente lleva a automatizar estas tareas (proceso) con el fin de mejorar los tiempos. Hoy en día el proceso no existiría o sería poco rentable si no contamos con las herramientas de automatización.

Cuando pensamos en software

Son todos los artefactos que producimos en el contexto de un proceso de desarrollo de software. Conjunto de:

- Programas.
- Procedimientos
- Reglas.
- Documentación.

- Datos.

No es solo código fuente.

- Información:
 - estructurada con propiedades lógicas y funcionales.
 - creada y mantenida en varias formas y representaciones.
 - confeccionada para ser procesada por computadora en su estado más desarrollado

La evolución del software

La gestión de configuración surge para mantener la integridad del proceso de desarrollo. (integridad entre las herramientas y elementos del proceso).

Por qué cambia?

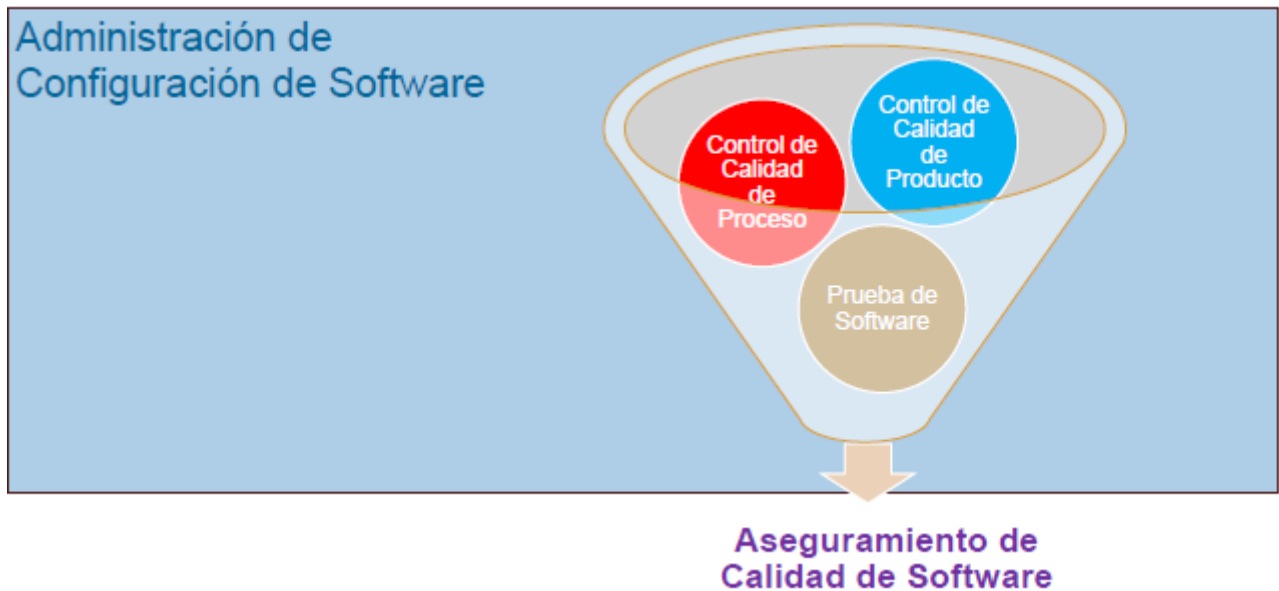
Tienen su origen en:

- ❖ Cambios del negocio y nuevos requerimientos
- ❖ Soporte de cambios de productos asociados
- ❖ Reorganización de las prioridades de la empresa por crecimiento
- ❖ Cambios en el presupuesto
- ❖ Defectos encontrados a corregir
- ❖ Oportunidades de mejora

Cambian todas las salidas de las etapas del proceso.

Esta disciplina, es una disciplina de soporte, transversal a todo el proyecto, relevante para el producto a lo largo de su ciclo de vida. Se lleva adelante desde que el proyecto comienza hasta que termina.

Otras disciplinas son:



Historia:

Nace con CM a mediados de 1950. utilizado para el desarrollo de hardware y control de producción, fue utilizado en el desarrollo de software.

Definición

Una disciplina que aplica dirección y monitoreo administrativo y técnico a: identificar y documentar las características funcionales y técnicas de los ítems de configuración, controlar los cambios de esas características, registrar y reportar los cambios y su estado de implementación y verificar correspondencia con los requerimientos

(ANSI/IEEE 828, 1990)

Estas son las 4 actividades fundamentales de SCM. El fin es mantener la

integridad del producto de software a lo largo del ciclo de vida.

Su propósito es establecer y mantener la integridad de los productos de software a lo largo de su ciclo de vida.

Involucra para la configuración:

- ❖ Identificarla en un momento dado
- ❖ Controlar sistemáticamente sus cambios
- ❖ Mantener su integridad y origen

Integridad del producto

- satisface las necesidades del usuario
- puede ser fácil y completamente rastreado durante su ciclo de vida
- satisface criterios de performance
- cumple con sus expectativas de costo

El software: un blanco

Tiene que ver con diversos aspectos:

- Que satisface las necesidades del usuario (no nos desviamos de lo que nos piden). Correspondencia del producto entre los requerimientos implementados y lo que pidió el usuario (Trazabilidad).
- Fácil y completamente rastreado durante su ciclo de vida.
- Satisface criterios de performance
- Cumple con sus expectativas de costo, no nos desviamos del presupuesto.

Problemas con la integridad

- ❖ Pérdida de un componente
- ❖ Pérdida de cambios (el componente que tengo no es el último)
- ❖ Sincronía fuente - objeto – ejecutable
- ❖ Regresión de fallas
- ❖ Doble mantenimiento
- ❖ Superposición de cambios
- ❖ Cambios no validados

Conceptos clave

Ítem de configuración

Documentos de diseño, código fuente, código ejecutable, etc.

*Se llama **ítem de configuración (IC)** a todos y cada uno de los artefactos que forman parte del producto o del proyecto, que pueden sufrir cambios o necesitan ser compartidos entre los miembros del equipo y sobre los cuales necesitamos conocer su estado y evolución.*

- ❖ Plan de CM
- ❖ Propuestas de Cambio
- ❖ Visión
- ❖ Riesgos
- ❖ Plan de desarrollo
- ❖ Prototipo de Interfaz
- ❖ Guía de Estilo de IHM
- ❖ Manual de Usuario
- ❖ Requerimientos
- ❖ Plan de Calidad
- ❖ Arquitectura del Software
- ❖ Plan de Integración
- ❖ Planes de Iteración
- ❖ Estándares de codificación
- ❖ Casos de prueba
- ❖ Código fuente
- ❖ Gráficos, iconos, ...
- ❖ Instructivo de ensamble
- ❖ Programa de instalación
- ❖ Documento de despliegue
- ❖ Lista de Control de entrega
- ❖ Formulario de aceptación
- ❖ Registro del proyecto

Es todo aquello que se decide someter, debido a que sufre cambios (mas peso).

Versión

Debido a que los items de configuración evolucionan en el tiempo, deberíamos tener historial de cambios. Es así como nace la versión, que nos muestra como evoluciona en el tiempo los diversos items de configuración.

Una versión se define, desde el punto de vista de la evolución, como la forma particular de un artefacto en un instante o contexto dado.

El control de versiones se refiere a la evolución de un único ítem de configuración (IC), o de cada IC por separado.

La evolución puede representarse gráficamente en forma de grafo.

Variante

Una versión, pero que evoluciona por separado. Surge fundamentalmente cuando tengo componentes que necesitan de diversas configuraciones.

Permanece en el tiempo

- Una variante es una versión de un ítem de configuración (o de la configuración) que evoluciona por separado.
- Las variantes representan configuraciones alternativas.
- Un producto de software puede adoptar distintas formas (configuraciones) dependiendo del lugar donde se instale.
- Por ejemplo, dependiendo de la plataforma (máquina + S.O.) que la soporta, o de las funciones opcionales que haya de realizar o no.

Configuración

Es un conjunto de items con su correspondiente versión en un momento determinado.

Repositorio

Lugar donde almaceno los items de configuración. Mantienen la historia de cada IC.

- Un repositorio de información conteniendo los ítems de configuración (ICs)
- Mantiene la historia de cada IC con sus atributos y relaciones.
- Usado para hacer evaluaciones de impacto de los cambios propuestos.
- Pueden ser una o varias bases de datos

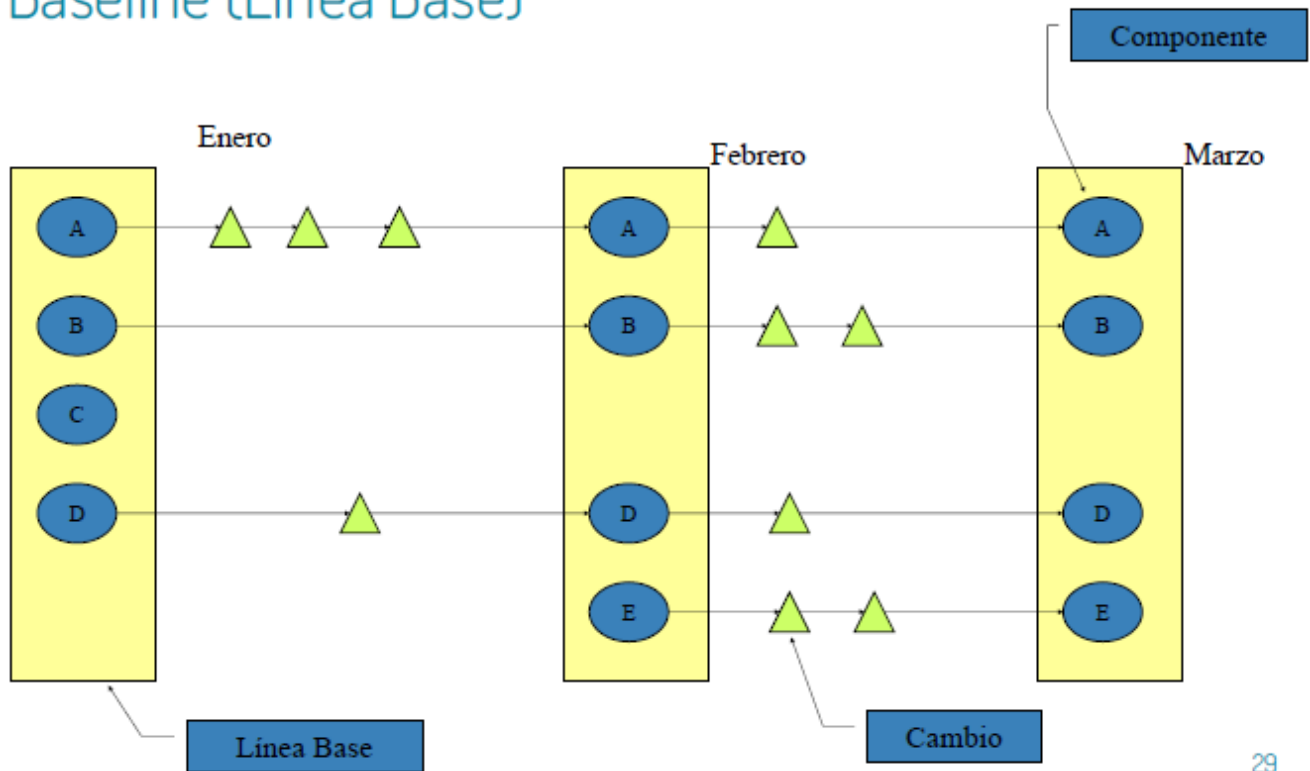
Identificación de la linea Base

Tomo una determinada configuración, y la etiqueto como linea base. Es una etiqueta que nos indica que la configuración marcada es especial, es decir, que es configuración ha sido revisada por el equipo y llevada a un acuerdo de que es una versión estable.

- ❖ Una configuración que ha sido revisada formalmente y sobre la que se ha llegado a un acuerdo
- ❖ Sirve como base para desarrollos posteriores y puede cambiarse sólo a través de un procedimiento formal de control de cambios
- ❖ Permiten ir atrás en el tiempo y reproducir el entorno de desarrollo en un momento dado del proyecto

Ante cambios que provocan problemas, esto nos permite volver atrás a esa versión estable.

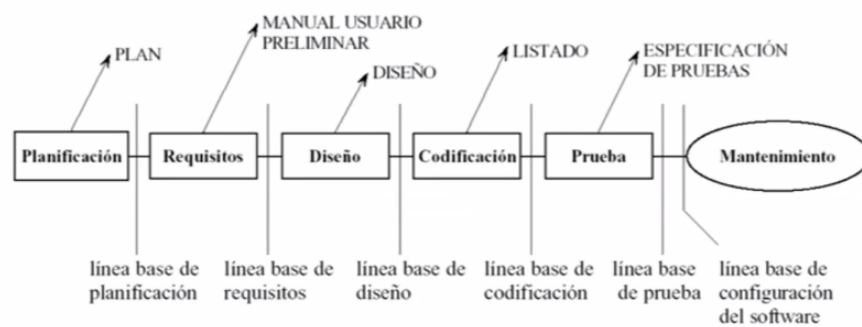
Baseline (Línea Base)



29

Pueden ser:

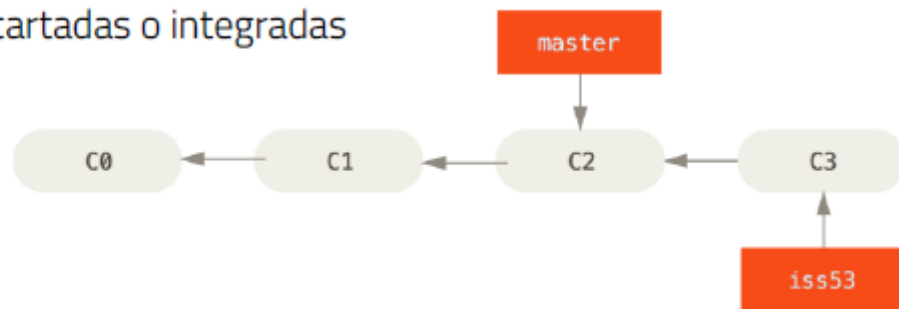
- De especificación (Requerimientos, Diseño)
- De productos que han pasado por un control de calidad definido previamente



Ramas

Permite bifurcar el desarrollo. Existe una rama principal (master)

- ❖ Existe una rama principal (trunk, master)
- ❖ Sirven para bifurcar el desarrollo
- ❖ Pueden tener razones de creación con semántica
- ❖ Permiten la experimentación
- ❖ Pueden ser descartadas o integradas

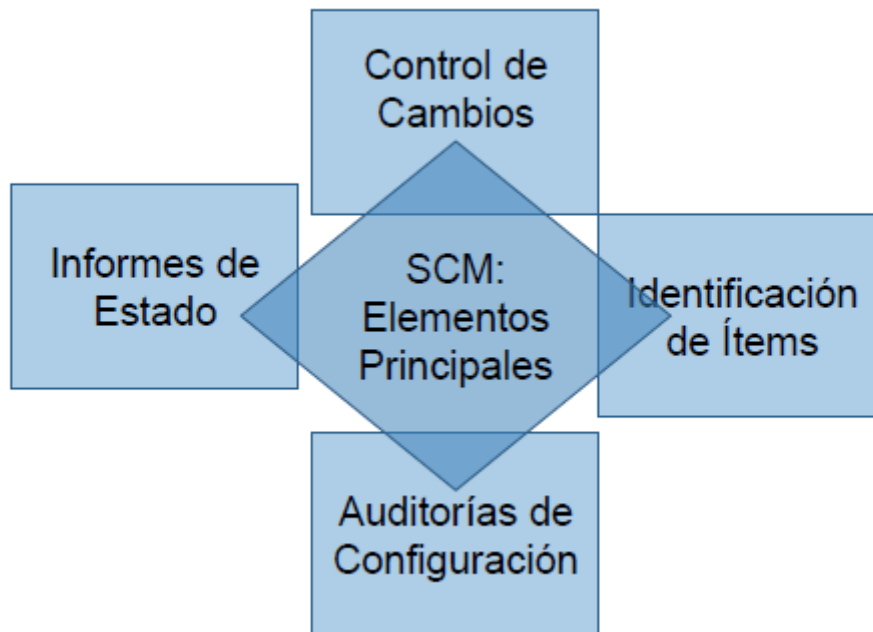


Tienen vida finita o se descarta o se integra a otra rama



- ❖ La operación se llama merge
- ❖ Lleva los cambios a la rama principal
- ❖ Pueden surgir conflictos (resolvemos con diff)
- ❖ Todas las ramas deberían eventualmente integrarse a la principal o ser descartadas

Actividades Fundamentales de SCM



Tienen validez en los procesos definidos, la gestión tradicional

1. Identificación de ítems

Todos los ítems, definir cómo los vamos a nombrar, de una manera única, que los permita diferenciar de otros.

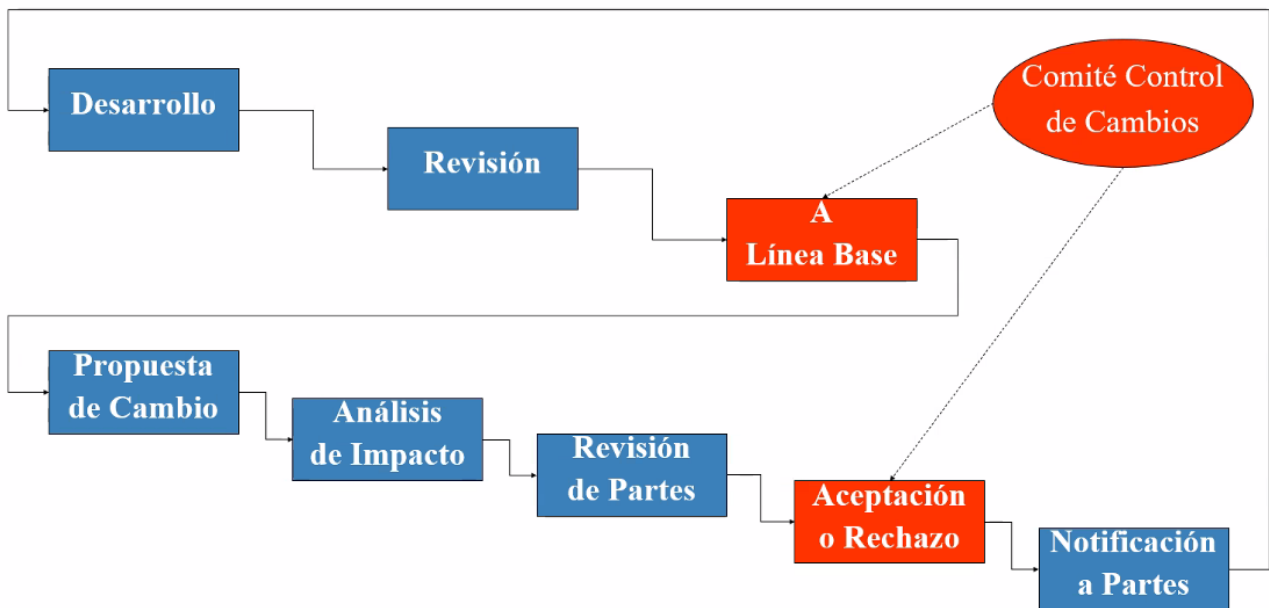
- ❖ Identificación unívoca de cada ítem de configuración
- ❖ Convenciones y reglas de nombrado
- ❖ Definición de la Estructura del Repositorio
- ❖ Ubicación dentro de la estructura del repositorio

Cuales son los ítems! Definirlos. Ejemplos:



2. Control de cambios

Cuando quiero hacer cambios en la línea base.



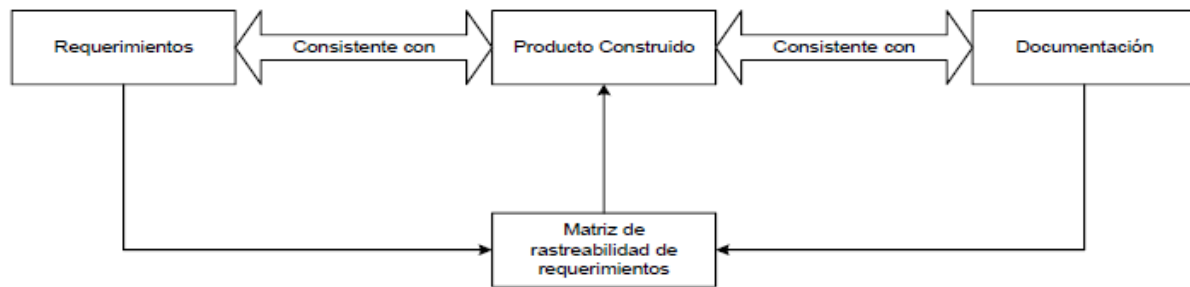
- ❖ Tiene su origen en un Requerimiento de Cambio a uno o varios ítems de configuración que se encuentran en una **línea base**.
- ❖ Es un Procedimiento formal que involucra diferentes actores y una evaluación del **impacto** del cambio

Y está formado por representantes de todas las áreas involucradas en el

- ❖ Análisis, Diseño
- ❖ Implementación
- ❖ Testing

desarrollo: ❖ Otros interesados

3. Auditorias



- Auditoría física: Verifica que todo lo que se definida, realmente se haga. Esto se define en el plan de gestión de configuración. Verificación
- Auditoria funcional: Valida que lo realizado sea consistente con lo que se definido. Validación

Auditoría física de configuración (PCA)

Asegura que lo que está indicado para cada ICS en la línea base o en la actualización se ha alcanzado realmente.

Auditoría funcional de configuración (FCA)

Evaluación independiente de los productos de software, controlando que la funcionalidad y performance reales de cada ítem de configuración sean consistentes con la especificación de requerimientos.

Sirve a dos procesos básicos: la validación y la verificación

- ❖ **Validación:** el problema es resuelto de manera apropiada que el usuario obtenga el producto correcto.
- ❖ **Verificación:** asegura que un producto cumple con los objetivos preestablecidos, definidos en la documentación de líneas base (línea base). Todas la funciones son llevadas a cabo con éxito y los test cases tengan status "ok" o bien consten como "problemas reportados" en la nota de release.

4. Informe de estado

Nos permite rastrear los cambios, y realizar reportes sobre los mismos.

- ❖ Se ocupa de mantener los registros de la evolución del sistema.
- ❖ Maneja mucha información y salidas por lo que se suele implementar dentro de procesos automáticos.
- ❖ Incluye reportes de rastreabilidad de todos los cambios realizados a las líneas base durante el ciclo de vida.

Plan

Plan de Gestión de Configuración

¡También se planifica! ¿Qué debería incluir el plan?

- ❖ Reglas de nombrado de los CI
- ❖ Herramientas a utilizar para SCM
- ❖ Roles e integrantes del Comité
- ❖ Procedimiento formal de cambios
- ❖ Plantillas de formularios
- ❖ Procesos de Auditoría

SCM en AGILE

HACER ACORDAR A LAU DE VER ESTO AL VER EL MANIFIESTO AGIL

clase que viene.

La premisa es que le sirva al equipo, hay ciertas actividades como las auditorias que no tienen razón de ser en la gestión ágil del proyecto.

Tips

- ❖ Es responsabilidad de todo el equipo.
- ❖ Automatizar lo más posible.
- ❖ Educar al equipo.
- ❖ Tareas de SCM embebidas en las demás tareas requeridas para alcanzar el objetivo del Sprint.

DEBATIR:

- ❖ ¿Qué pasa con el Comité de Control de Cambios?
- ❖ ¿Qué ítems de configuración podemos tener?
- ❖ ¿Qué pasa con las auditorías?
- ❖ ¿Qué pasa con los reportes de estado?