

**Trabajo de tesis
Framework de mejora
de procesos de desarrollo de software**

1^{ra} Sección: Cuerpo principal

Universidad Nacional de La Plata
Facultad de informática
Carrera: Magíster en Ingeniería
de Software



Alumno: Ing. Sebastián Barbieri
Director: Lic. Alejandro Bianchi
Codirector: Dr. Gustavo Rossi

Índice

Capítulo 1 – Introducción al trabajo de tesis	1
1.1 Introducción al capítulo	2
1.2 Objetivos del trabajo	2
1.3 Alcance	3
1.4 Estructura del trabajo de tesis	3
1.5 Estado del arte, trabajos relacionados y contribución del trabajo	4
1.6 Posibles modelos de referencia utilizables por el framework	4
1.7 Consideraciones de la metodología	5
1.8 Framework de implementación	7
Capítulo 2 - Modelos de Referencia	8
2.1 Introducción al capítulo	9
2.2 SW-CMM	10
2.3 ISO/IEC 15504 (SPICE)	13
2.4 BOOTSTRAP	16
2.5 CMMI-SW staged	18
2.6 Trillium	20
Capítulo 3 - Tarea 1 - Definir objetivos de la organización sobre la implementación y definir criticidad	22
3.1 Identificación	23
3.2 Introducción al capítulo	23
3.3 Estructura de la tarea	23
3.4 Consideraciones a la hora de plantear los objetivos de la organización	23
3.5 Consideraciones sobre los posibles objetivos a seleccionar	24
3.5.1 Reducir el time to market	24
3.5.2 Mejorar la calidad	25
3.5.3 Evitar costos	26
3.5.4 Incrementar la productividad	26
Capítulo 4 - Tarea 2 - Definir / identificar marco de proceso utilizado	27
4.1 Identificación	28
4.2 Introducción al capítulo	28
4.3 Estructura de la tarea	28
4.4 Marcos de procesos a seleccionar	29
4.4.1 Marcos de procesos pesados	29
4.4.1.1 Ejemplos de marcos de proceso pesados	29
4.4.1.1.1 PSP (Personal Software Process)	29
4.4.1.1.2 RUP (Rational Unified Process)	32
4.4.1.2 Otros marcos de proceso pesados	34
4.4.2 Marcos de proceso livianos	34
4.4.2.1 Ejemplos de marcos de proceso livianos	34
4.4.2.1.1 XP (Extreme Programming)	34
4.4.2.1.2 FDD (Feature Driven Development)	37
4.4.2.2 Otros marcos de proceso livianos	40
4.4.3 Software Product Line Framework	40
4.5 Notas y conclusiones	43

Capítulo 5 - Tarea 3 - Assessment	44
5.1 Identificación	45
5.2 Introducción al capítulo	45
5.3 Estructura de la tarea	46
5.3.1 Tarea 3-1: Identificar el ámbito a evaluar	46
5.3.2 Tarea 3-2: Administrar equipo de assessment	47
5.3.3 Tarea 3-3: Desarrollar plan de assessment y notificar involucrados	48
5.3.4 Tarea 3-4: Capacitación SPI, assessment y refuerzo de aspectos culturales	49
5.3.5 Tarea 3-5: Administrar cuestionarios	49
5.3.6 Tarea 3-6: Recolección de datos y entrevistas	50
5.3.7 Tarea 3-7: Analizar métricas por proyectos	57
5.3.8 Tarea 3-8: Consolidar información	59
5.3.9 Tarea 3-9: Preparar y presentar conclusiones	61
Capítulo 6 - Tarea 4 - Plan de acción	62
6.1 Identificación	63
6.2 Introducción al capítulo	63
6.3 Estructura de la tarea	64
6.3.1 Tarea 4-1: Analizar y construir lista de acciones para la mejora de procesos	64
6.3.2 Tarea 4-2: Desarrollar listas de issues RRHH y culturales	66
6.3.3 Tarea 4-3: Ordenar y agrupar acciones para la mejora de procesos	70
6.3.4 Tarea 4-4: Definir plan de mediciones para acciones agrupadas	71
6.3.5 Tarea 4-5: Estimación aproximada de recursos x acción y herramientas necesarias por Work Package	73
6.3.6 Tarea 4-6: Análisis de riesgos	75
6.3.7 Tarea 4-7: Definición de etapas del plan táctico y estrategia de implementación del plan táctico	78
6.3.8 Tarea 4-8: Obtener compromiso del sponsor	81
6.3.9 Tarea 4-9: Desarrollar SCMP preliminar	81
Capítulo 7 - Tarea 5 - Desarrollo del Business Case	83
7.1 Identificación	84
7.2 Introducción al capítulo	84
7.3 Perfiles necesarios para desarrollar el Business Case	84
7.4 Principios para el desarrollo de un Business Case	85
7.5 Estructura de la tarea	85
7.5.1 Tarea 5-1: Análisis de costo, beneficio y evaluación económica del proyecto	86
7.5.2 Tarea 5-2: Documentación del Business Case	94
7.5.3 Tarea 5-3: Presentación y aprobación del Business Case	97
Capítulo 8 - Tarea 6 - Implementación	98
8.1 Identificación	99
8.2 Introducción al capítulo	99
8.3 Estructura de la tarea	99
8.3.1 Tarea 6.1: Presentación pública del proyecto	101
8.3.2 Tarea 6-2: Instalación de única vez	102
8.3.3 Tarea 6A (abstracta): Implementar etapa	103
8.3.3.1 Tarea 6A - Sección de instalación de la etapa	103
8.3.3.1.1 Tarea 6A1: Arranque instalación	104
8.3.3.1.2 Tarea 6A2: Desarrollar / actualizar plan operacional del proyecto	105
8.3.3.1.3 Tarea 6A3: Instalar mecanismos para registrar mediciones	105
8.3.3.1.4 Tarea 6A4: Instalar herramientas necesarias para etapa	106
8.3.3.1.5 Tarea 6A5: Desarrollar activos de proceso de la etapa	107
8.3.3.1.6 Tarea 6A6: Recolección de mediciones instalación	107
8.3.3.1.7 Tarea 6A7: Monitoreo cumplimiento del plan	108
8.3.3.2 Tarea 6A8 – Refinar plan de implementación y Business Case	110
8.3.3.3 Tarea 6A - Sección de apalancamiento de la etapa	111
8.3.3.3.1 Tarea 6A9: Capacitar usuarios de activos de proceso de la etapa	111
8.3.3.3.2 Tarea 6A10: Apalancar etapas que están siendo implementadas	112
8.3.4 Tarea 6-5: Clausura de etapa y notificación de resultados	116
8.3.5 Tarea 3: Assessment	116
8.3.6 Tarea 6-6: Clausura de la implementación	117

Conclusiones	118
Trabajos futuros	118
Vocabulario	119
Referencias	120

Lista de Figuras

Figura 0.1 Factores que determinan una implementación particular	5
Figura 0.2 Objetivos perseguidos como intersección de variables	6
Figura 0.3 Implementación en particular como intersección de objetivos y mejores prácticas	7
Figura 0.4 Tareas Framework de mejora de procesos de software - Visión abstracta	7
Figura 1.1 Niveles SW-CMM	11
Figura 1.2 Estructura del SW-CMM	12
Figura 1.3 Diagrama de posibles usos de SPICE	13
Figura 1.4 Estructura de SPICE	15
Figura 1.5 Categorías de procesos en Bootstrap	17
Figura 1.6 Componentes del modelo CMMI-SW	18
Figura 1.7 Estructura de Trillium	21
Figura 2.1 Ciclo Time to market	24
Figura 3.1 Niveles PSP	30
Figura 3.2 Estructura de procesos del RUP	32
Figura 3.3 Flujos de trabajo de RUP	34
Figura 3.4 Formato de un proyecto XP	36
Figura 3.5 Procesos del FDD	39
Figura 3.6 Las 3 actividades elementales del SPL	41
Figura 3.7 Desarrollo de activos base para SPL	41
Figura 3.8 Procesos adjuntos del SPL	41
Figura 3.9 Desarrollo de productos en SPL	41
Figura 4.1 Estructura tarea 3	46
Figura 4.2 Porción de cuestionario a utilizar en entrevistas	51
Figura 4.3 Porción de planilla de herramientas case	53
Figura 4.4 Porción de cuestionario de RRHH	54
Figura 4.5 Modelo cultural de 3 capas para organizaciones	55
Figura 4.6 Porción de cuestionario de aspectos culturales	56
Figura 4.7 Porción posibles métricas usadas en proyectos	58
Figura 4.8 Porción de planilla para consolidar assessment SW-CMM	60
Figura 5.1 Estructura tarea 4	64
Figura 5.2 Estructura tarea para soporte a mejora de procesos	65
Figura 5.3 Porción template de acciones correctivas	66
Figura 5.4 Ejemplo de lista de issues RRHH y culturales a derivar	68
Figura 5.5 Ejemplo de lista de issues RRHH y culturales a tener en cuenta en la mejora de procesos	69
Figura 5.6 Ejemplo / template de agrupación de acciones correctivas	71
Figura 5.7 Porción de ejemplo de herramientas disponibles en el mercado	74
Figura 5.8 Porción de ejemplo / template de resultado tarea 4-5	74
Figura 5.9 Porción de ejemplo / template de análisis de riesgo	77
Figura 5.10 Estructura del SCMP	82
Figura 6.1 Estructura tarea 5	85
Figura 6.2 Porción de template para análisis y evaluación económica	93
Figura 7.1 Estructura de la tarea implementación a nivel de abstracción 0	100
Figura 7.2 Estructura de explosión tarea implementar etapa	100
Figura 7.3 Estructura tarea para soporte a mejora de procesos	102
Figura 7.4 Tabla para seguimiento de issues de implementación	109
Figura 7.5 Escenarios de cambios según estado de de implementación	111
Figura 7.6 Ciclo de operaciones de apalancamiento	113

Lista de Tablas

Tabla 1.1 Arquitectura de modelos de referencia	10
Tabla 1.2 Relación Nivel x Área de proceso x Categoría en CMMI Staged	19
Tabla 5.1 Composición de plan táctico	80
Tabla 6.1 Acciones a tomar según evaluación del BC por parte del sponsor	94
Tabla 6.2 Acciones a tomar según evaluación del BC a nivel organización	97
Tabla 7.1 Tabla de análisis de mediciones registradas	114
Tabla 7.2 Cantidad de meses para avanzar de nivel en SW-CMM	115

Este trabajo esta dedicado a mis dos Felipes.

Capítulo 1

Introducción al trabajo de tesis

1.1 Introducción al capítulo

El trabajo desarrolla un framework de mejora de procesos sobre organizaciones que realicen desarrollo o mantenimiento de software independientemente del tamaño de la organización.

Este framework no está atado a un modelo de referencia en particular, proveyendo los lineamientos para guiar a las organizaciones de software en la elección de estrategias de mejoras de procesos mediante la determinación de la madurez de sus procesos actuales y la identificación de algunos puntos críticos que afectan a la calidad del software y a la mejora de procesos.

Las carencias más frecuentes dentro de las compañías antes de la aplicación de un framework de mejora de procesos son [refs. 10, 11, 13]:

- Los proyectos se desvían en costo y plazo. La calidad es baja.
- El software resultante tiene numerosos defectos y no satisface las expectativas de los clientes.
- Poca documentación de las aplicaciones y los proyectos.
- No se puede validar la calidad del software en puntos intermedios.
- No existe control sobre los cambios en los requisitos.
- No existe uniformidad entre el trabajo desarrollado por distintas personas.
- El éxito de los proyectos depende de las capacidades individuales.
- Si personas clave dejan de participar en el proyecto se para la "fabrica".

Algunos de los beneficios de la aplicación de un framework de mejora de procesos son [ref. 23, 82]:

- Mayor efectividad en la detección de errores a lo largo del ciclo de vida, reduciendo drásticamente el número de errores que afecta directamente a los clientes y usuarios.
- Reducción de las desviaciones en plazo de los proyectos.
- Mayor tolerancia al cambio e incremento de la capacidad de adopción y adaptación de nuevas tecnologías.
- Mejora en la rapidez y efectividad de respuesta ante exigencias del negocio (Reducción del Time to Market).
- Mejora en la colaboración y comunicación efectiva con implicados internos y externos.
- Resultados predecibles en los proyectos.
- Implementar técnicas proactivas de gestión, mitigando los riesgos que afectan los proyectos. Todos los beneficios derivan finalmente en un incremento de la productividad en la realización de software y en una considerable mejora de la calidad del producto terminado.

¿Porque desarrollar un nuevo framework de mejora de procesos?

- Existe una necesidad real y actual del mercado de TI en Argentina ante la posibilidad de conseguir mercados externos solamente accesibles con la certificación de estándares internacionales. [refs. 4, 8]
- Este nuevo framework permite:
 - Alinear la mejora de procesos con los objetivos organizacionales de empresa.
 - Permite configurar una implementación de acuerdo a los objetivos de la organización y al marco de desarrollo que utiliza la empresa.
 - Utilizar alguno de los modelos de referencia, sin estar limitado a trabajar únicamente con alguno en particular.
 - Potenciar el framework complementándolo con la visión económica de la implementación a través del desarrollo de Business Cases.
 - Complementar los aspectos técnicos del framework con aspectos culturales, organizacionales y de RRHH.

1.2 Objetivos del trabajo

Desarrollar un trabajo de tesis que permita definir conceptualmente un framework de mejora de procesos de desarrollo de software.

El trabajo se compone del framework en sí modelado como un conjunto de procesos que guíen a la empresa o sector para poder certificar o evaluar estándares internacionales.

El desarrollo del framework de mejora de procesos es complementado con un conjunto de herramientas para

realizar la mejora, y un caso práctico concreto. Esto se detalla en la sección de alcance.

El framework se basa en el modelo de mejora de procesos de software IDEAL del SEI, pero acotado respecto:

1. Nivel de capacidad que pretende alcanzar la organización
2. Tipo de software desarrollado y objetivos perseguidos por la organización (ver sección 0.7).
3. Proveer un conjunto de prácticas alternativas que permitan alcanzar los objetivos de cada área clave de proceso.
4. Complementar el framework con prácticas del ámbito de RRHH y aspectos culturales.

1.3 Alcance

El alcance del framework de mejora de procesos viene impuesto por las clasificaciones de software consideradas, por los objetivos contemplados para las organizaciones, y por la capacidad a alcanzar.

El framework es un proceso iterativo que despliega un esquema disciplinado para realizar mejoras, que se compone de una serie de procesos propuestos como guía para que la organización pueda utilizarla y así poder certificar o evaluar estándares internacionales.

Cada una de las decisiones adoptadas para la construcción de la metodología están debidamente fundamentadas y justificadas.

Hay que aclarar que el presente framework no es "una bala de plata", ni tiene los mismos resultados en tiempo y esfuerzo entre distintas organizaciones.

El framework es libre de operar con distintos modelos de referencia, ya que el modelo de referencia es simplemente un facilitador. Entre los posibles modelos se encuentran:

- SW-CMM
- SPICE
- BOOTSTRAP
- CMMI staged
- Trillium

Por una cuestión de bajar en nivel de abstracción del framework, en algunos puntos concretos el desarrollo del trabajo de tesis hace referencia al SW-CMM o "Modelo de Madurez de Capacidad del Software" del SEI (Software Engineering Institute) como modelo de referencia.

1.4 Estructura del trabajo de tesis

El trabajo se compone de tres secciones:

- 1) **Sección Cuerpo principal:** Describe el framework en sí desplegando las tareas que lo componen.
Al finalizar este capítulo se presenta el framework de manera abstracta mostrando gráficamente las tareas que lo componen, para luego ir desglosando las tareas que lo componen en cada uno de los capítulos que componen esta sección.
 - **Capítulo 2** - Modelos de referencia
 - **Capítulo 3** - Tarea 1- Definir objetivos de la organización sobre la implementación
 - **Capítulo 4** - Tarea 2 - Marcos de procesos
 - **Capítulo 5** - Tarea 3 - Assessment
 - **Capítulo 6** - Tarea 4 - Plan de acción
 - **Capítulo 7** - Tarea 5 - Business Case
 - **Capítulo 8** - Tarea 6 - Implementación
 - **Conclusiones, futuros trabajos y vocabulario**
- 2) **Sección Caso práctico:** El framework es complementado con un caso práctico concreto, basado en una empresa real que es una software factory PyME. Por una cuestión de costos y tiempo, el alcance del desarrollo de este caso práctico llega hasta el nivel de la elaboración del plan táctico, dejando como posible futura extensión al trabajo el desarrollo del resto del caso práctico para las demás tareas del framework.
- 3) **Sección Anexos:** Provee un conjunto de herramientas del tipo templates, planillas, formularios o ejemplos de para registrar información clave al hacer uso del framework.

1.5 Estado del arte, trabajos relacionados y contribución del trabajo

Existe una amplia variedad de publicaciones y presentaciones sobre el tema que aborda este trabajo de tesis. Los mismos se pueden agrupar por:

- 1) Publicaciones que presentan un framework o guía independiente del modelo de mejora:
 - Software Process improvement Guidebook, Software Engineering Laboratory Series, NASA [ref. 6]
 - Tesis "A Software Process Improvement Framework for Small Organizations [ref 103]
 - An Investigation into Software Process Improvement in the Small. [ref 102]
 - IDEAL: A User's Guide for Software Process Improvement [ref. 5]
- 2) Publicaciones que presentan una guía o metodología dependiente de un modelo de mejora:
 - Tesis de magíster en ingeniería del software “Certificación de procesos de desarrollo de software” Basado en el Estándar ISO 9001:2000. [ref. 108]
 - Software Process Improvement With CMM [ref. 104]
 - CMM Implementation Guide - Choreographing Software Process Improvement [ref. 105]
 - CMM in Practice - Process for Executing Software Projects at Infosys. [ref. 1]

Algunas de las debilidades detectadas en las publicaciones relevadas son:

- Alto nivel de abstracción en los frameworks presentados (a nivel tareas que lo componen, roles, deliverables)
- Estar orientados hacia modelos de evaluación y mejora de procesos puntuales.
- No consideración de aspectos culturales o de RRHH.
- No consideración de aspectos económicos como parte del procesos de aceptación de la propuesta de mejora
- Implementaciones de mejoras de procesos orientadas a los drivers del negocio

La contribución que realiza este trabajo es aprovechar la oportunidad de integrar estos aspectos consolidándolos en un framework que se hace fuerte en aquellos puntos débiles detectados

Nota: Hay que destacar que existen más publicaciones al respecto, solo se pone una muestra del universo.

1.6 Posibles modelos de referencia utilizables por el framework

Dentro de los posibles modelos de referencia se encuentran los siguientes:

- SW-CMM
- ISO/IEC 15504 (SPICE)
- BOOTSTRAP
- CMMI-SW staged
- Trillium

Sin embargo, esta lista no es restrictiva y pueden utilizarse otros modelos no incluidos en la lista. En el capítulo 2 se hace una presentación de cada uno de los modelos antes mencionados.

1.7 Consideraciones de la metodología

Más allá de los pasos que componen el framework en sí, en la figura 1.1 se presentan en primer lugar las bases sobre las cuales se apoya la framework:

- El conjunto de mejores prácticas de la metodología, no es uniforme a todos los tipos de organizaciones, objetivos de las mismas y marcos de procesos que utilizan para el desarrollo del software.
- Abstrayéndose de las variables que determinan las mejores prácticas, una implementación en particular resulta de la participación de los siguientes factores:

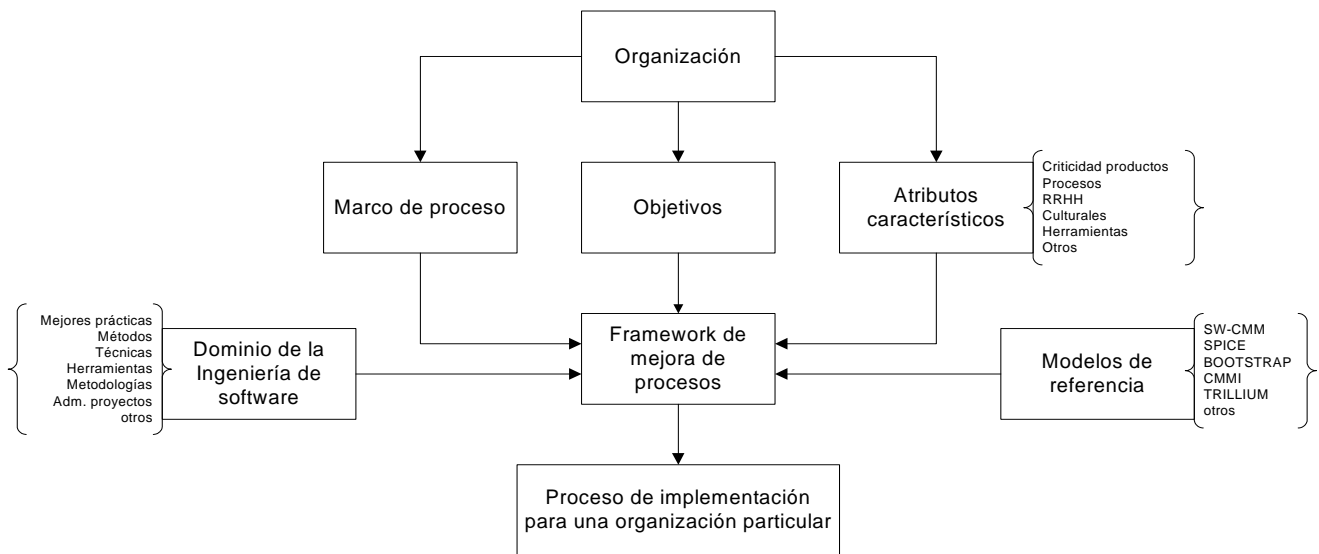


Figura 1.1 Factores que determinan una implementación particular

A continuación se presentan las variables para determinar el conjunto de mejores prácticas a aplicar en la implementación:

- **Tipo de objetivo de la organización** es el objetivo que persigue la organización, el cual es único para todo el proyecto de implementación de las prácticas. Sus valores pueden ser:
 - Reducción del Time to market
 - Incrementar la productividad
 - Evitar / reducir costos
 - Mejorar la calidad

La selección de los objetivos de la organización, se corresponde con la tarea 1 de la implementación.

- **Marco de proceso utilizado para el desarrollo de software** es el marco de proceso que utiliza la organización como estrategia a la hora de hacer sus desarrollos de software.

Ejemplos de marco de procesos pueden ser:

- PSP
- RUP
- XP
- FDD
- Otros

La selección del marco de proceso de la organización, se corresponde con la tarea 2 de la implementación.

- **Tipo de criticidad del software construido** es el riesgo ocasionado por el uso del software desarrollado. Sus posibles valores son:
 - **Software de alto riesgo:** El objetivo de la organización es producir software que ante una falla en el producto en uso podría producir un riesgo significativo de muerte o daño de la propiedad mucho más costoso que el desarrollo de software en sí.
 - **Riesgo medio:** El objetivo de la organización es producir software en el cual los riesgos de falla implican daños de la propiedad o un impacto negativo en la imagen de la organización.
 - **Riesgo bajo:** El objetivo de la organización es producir software en el cual los riesgos de falla del software implican pérdida de tiempo del usuario o exceso en el tiempo de proceso. Puede ser característico de esta clase que el software sea difícil de reparar.
 - **Riesgo insignificante:** El objetivo de la organización es producir software en el cual los riesgos de falla del software implican una pérdida menor de tiempo del usuario, o un incremento menor en el tiempo del proceso.

El valor de esta variable esta definida por la organización en sí y o se considera como una tarea de selección.

La intersección de estas tres variables, se corresponde con un número de objetivos perseguidos (figura 1.2).

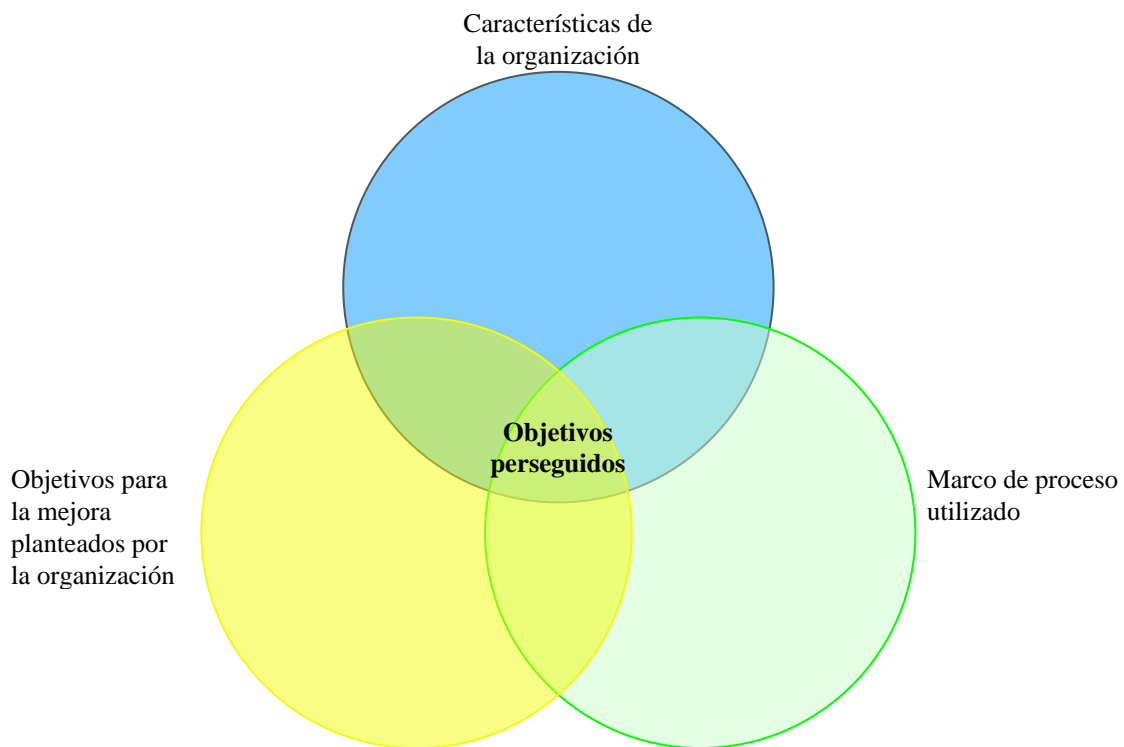


Figura 1.2 Objetivos perseguidos como intersección de variables

Estos objetivos perseguidos permitirán:

- Seleccionar el conjunto de mejores prácticas a implementar en cada ciclo de la metodología en base a las características de la organización, tipo de software y marco de proceso.
- Forma de controlar el estado de la implementación en un punto del tiempo en base a los objetivos de la organización y el marco de proceso, ya que de esta forma existe un mapeo entre las áreas de proceso de la implementación y las características de la organización.
- Realizar una implementación centrada en los drivers del negocio.

Luego, una implementación particular sale de la intersección mostrada en figura 1.3

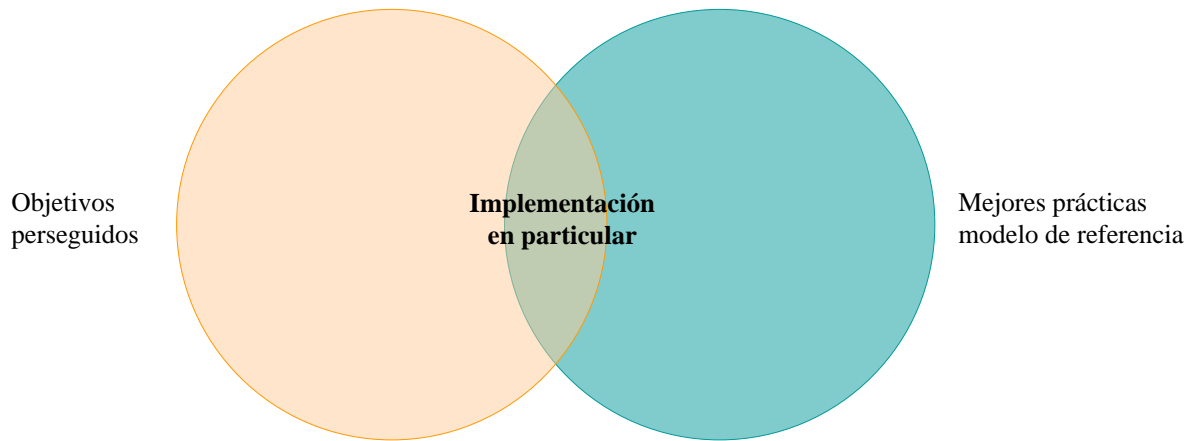


Figura 1.3 Implementación en particular como intersección de objetivos y mejores prácticas

1.8 Framework de implementación

En la figura 1.4 se presenta con un alto grado de abstracción cuales serán las tareas que componen la metodología de implementación.

En los capítulos siguientes se descomponen en detalle cada una de las tareas mostradas utilizando la notación ETVX y comentarios relacionados para cada caso.

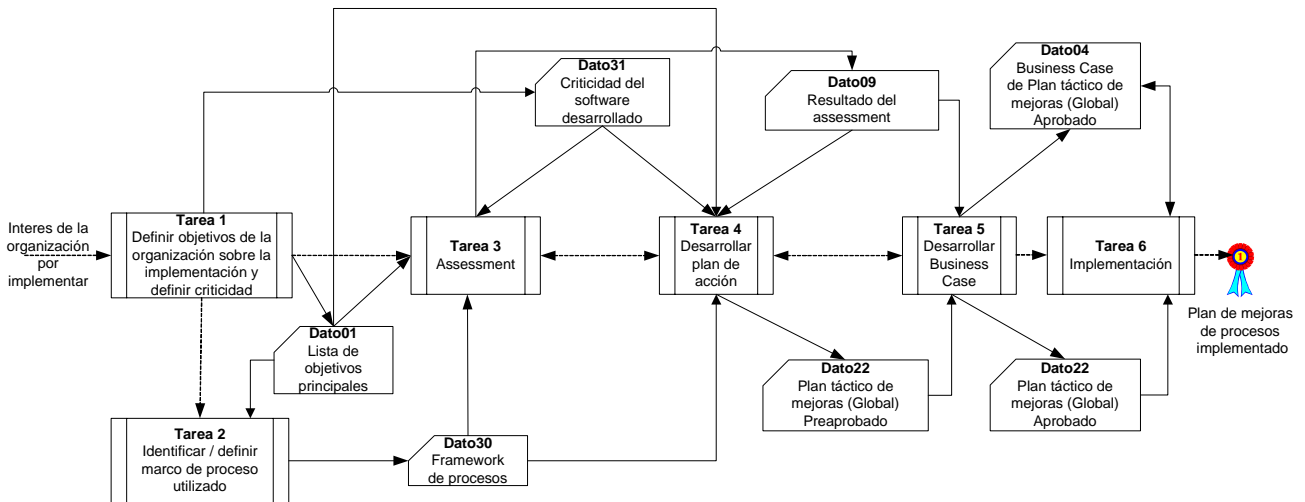


Figura 1.4 Tareas Framework de mejora de procesos de software - Visión abstracta

Capítulo 2

Modelos de Referencia

2.1 Introducción al capítulo

Por definición un modelo de referencia es un conjunto de procesos que guían a la organización a determinar su nivel de capacidad y a mejorar sus procesos de software.

Un modelo de referencia brinda:

- Un lenguaje y conjunto de constructores que facilitan la comunicación y el entendimiento para la mejora de procesos
- Un estándar de comparación y referencia para evaluar la efectividad de los procesos
- Una guía para la mejora de procesos
- Proveer una fuente de buenas prácticas o ideas

Estructura

La estructura de los modelos usados para la mejora de procesos incluye:

- **Áreas de proceso:** Conjunto de actividades relacionadas que sirven para alcanzar un conjunto de objetivos que son importantes para incrementar la capacidad del proceso
- **Objetivos:** Un aspecto simple del área de proceso cuyo cumplimiento puede ser objetivamente determinado.
- **Prácticas:** Describen las actividades e infraestructura que contribuyen a la efectiva implementación e institucionalización de objetivos y áreas de proceso. Las prácticas generalmente son formas recomendadas de alcanzar los objetivos. No son obligatorias ya que cada organización puede tener su conjunto de prácticas que cumplen efectivamente con los objetivos.
- **Material explicativo:** Material que ayuda a la organización a comprender algunos de los elementos importantes de las prácticas.

Existen dos características que determinan el modelo a utilizar para la mejora de procesos:

- **Dominio del modelo:** Se refiere a los sistemas cuya efectividad se quiere mejorar. Hay varios modelos creados para enfocar aspectos críticos de varios dominios, como por ejemplo:
 - Software
 - Ingeniería de software
 - Adquisición de sistemas
 - Personal
 - Integridad del software
 - otros

Los posibles modelos de referencia a tener en cuenta para el framework de mejora de proceso son aquellos cuyo dominio es el de desarrollo o mantenimiento de software.

- **Arquitectura:** Se refiere a la estructura subyacente del modelo y a la relación entre sus niveles de madurez y las áreas de proceso. Existen dos tipos de arquitecturas:
 - **Arquitectura por etapas,** características:
 - Tiene áreas de proceso específicas que están asociadas con distintos niveles de madurez.
 - La madurez es interpretada como una serie de etapas; donde un nivel de madurez es alcanzado mediante la implementación e institucionalización de áreas de proceso específicas de una etapa dada. De esta forma la madurez esta definida por las áreas de proceso del nivel.
 - Todas las áreas de proceso deben estar al mismo nivel antes que la organización pueda avanzar al próximo nivel de madurez.
 - Cada nivel de madurez tiene áreas de proceso específicas asociados al nivel.
 - La ruta de mejora organizacional esta bien definida respecto de cual área de proceso necesita ser atendida primero.
 - **Arquitectura continua,** características:
 - Tiene los niveles de capacidad dentro de las áreas de proceso
 - Provee un adecuado y bien definido camino de implementación para un área de proceso específica.
 - La madurez esta definida por la naturaleza de la implementación e institucionalización de alguna de las áreas de proceso.

- Las arquitecturas continuas proveen flexibilidad de implementación permitiendo a la organización seleccionar su ruta de implementación.
- No existe orientación respecto de cual área de proceso tiene mayor o menor importancia.
- El nivel de capacidad puede variar de un área de proceso a otra.
- Hace foco en áreas de proceso específicas, y cada área de proceso puede ser rankeada en un nivel de capacidad que va de 0 a 5 (estos niveles de capacidad son idénticos a los niveles de madurez de los modelos por etapas, solo que son aplicados a nivel área de proceso).
- Si la organización posee un gran número de áreas de proceso será más difícil orientar a la organización, ya que estaría intentando asignar recursos de mejora limitados a través de las áreas de proceso a lo largo de la organización.

Modelo	Arquitectura
SW-CMM	Por etapas
ISO/IEC 15504 (SPICE)	Continua
BOOTSTRAP	Continua
CMMI-SW staged	Por etapas
Trillium	Por etapas

Tabla 2.1 Arquitectura de modelos de referencia

Nota: A los efectos del framework desarrollado, utilizar un modelo de referencia que sea continuo o por etapas es indistinto, debido a que se planean e implementan paquetes de áreas de procesos que van subiendo de capacidad a medida que se avanza en el plan.

2.2 SW-CMM

El modelo de madurez de capacidad (SW-CMM) surge como iniciativa de SEI, a pedido del Gobierno Federal de los Estados Unidos de América.

Se considera que una organización ha alcanzado un nivel de madurez si ha institucionalizado todas las prácticas incluidas en ese nivel y niveles inferiores.

CMM propone alcanzar los resultados mediante pequeños cambios evolutivos. El framework de maduración de proceso de software ordena los cambios por etapas de manera tal que el mejoramiento de cada una de ellas brinde una base sobre la cual encarar la siguiente etapa de mejora. De esta manera, el framework describe una estrategia para mejoras continuas, guiando el avance e identificando deficiencias en la organización; sin embargo CMM no provee mejoras rápidas.

El CMM fue diseñado para guiar a la organización de software en el proceso de selección de estrategias de mejoras para determinar el nivel actual de madurez e identificar algunos problemas críticos en la calidad del software y en la mejora del proceso. La forma de lograr los objetivos propuesta es concentrarse en un conjunto de actividades y trabajar agresivamente para lograrlas de manera que estas provean logros continuos y duraderos en el proceso de software.

La mejora continua de procesos esta basada en realizar pequeños cambios evolutivos en lugar de una innovación revolucionaria. El CMM brinda a las organizaciones un marco para organizar los pasos de la mejora en cinco niveles de madurez (figura 2.1), estableciendo fundamentos para la mejora continua del proceso. Estos cinco niveles, definen una escala ordenada para evaluar la madurez del proceso de software. Además, estos niveles sirven para priorizar sus esfuerzos para la mejora.

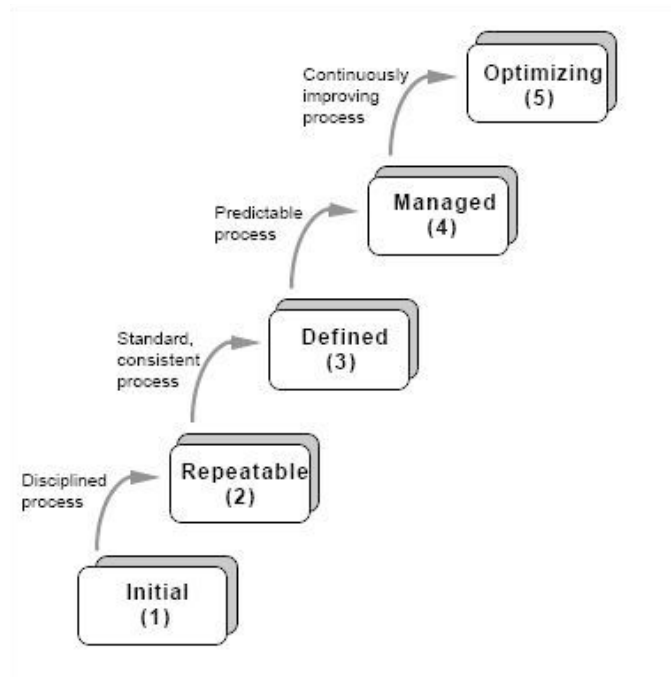


Figura 2.1 Niveles SW-CMM

La descripción de cada uno de los niveles es:

1. **Inicial:** el proceso de software está caracterizado como ad hoc y ocasionalmente caótico. Algunos procesos están definidos, y el éxito depende del esfuerzo individual y no de la organización.
2. **Repetible:** existen procesos administrativos básicos en los proyectos para el seguimiento de costo, planeamiento y funcionalidad. La disciplina necesaria en los procesos es acorde para repetir éxitos anteriores de proyectos con aplicaciones similares.
3. **Definido:** el proceso para la administración y la ingeniería está documentado, estandarizado e integrado a un proceso estándar para la organización. Todos los proyectos usan una versión del proceso estándar de la organización aprobada y ajustada para el desarrollo y mantenimiento del software.
4. **Administrado:** se detallan y recolectan medidas para que el proceso de software y la calidad del producto. Ambos son entendidos y controlados cuantitativamente.
5. **Optimizado:** existe un feedback cuantitativo del proceso, lo que permite una mejora continua del mismo. También se manejan ideas y tecnologías innovadoras.

El CMM es un modelo descriptivo en el sentido que describe los atributos esenciales que se esperan para caracterizar a una organización en un nivel de maduración en particular.

El CMM no es prescriptivo, es decir no dice a la organización cómo debe mejorar. El CMM dice en qué nivel se encuentra una organización sin decir una manera específica de cómo llegar a él.

Estructura

Cada nivel de madurez ha sido descompuesto en distintas partes que se componen de varias áreas de proceso clave (ver figura 2.2).

Un área de proceso clave es un abanico de actividades interrelacionadas, que cuando son llevadas a cabo, logran un conjunto de objetivos considerados importantes para establecer la capacidad de un proceso.

Cada una de estas, es organizada en cinco grupos llamadas características comunes. Estas son atributos que indican que tanto la implementación como la institucionalización, es efectiva, repetible y perdurable.

Los cinco grupos son:

- Objetivos a realizar
- Capacidad para realizar
- Actividades desarrolladas
- Medición y análisis
- Verificar Implementación

Donde Actividades desarrolladas describe lo que debe ser implementado para lograr capacidad del proceso, mientras que las demás prácticas institucionalizan las prácticas descritas en la primera.

Todos los objetivos del área clave deben ser satisfechos para cumplimentar un área clave de proceso. En esta situación se dice que la organización ha institucionalizado su capacidad de proceso caracterizado por el área clave.

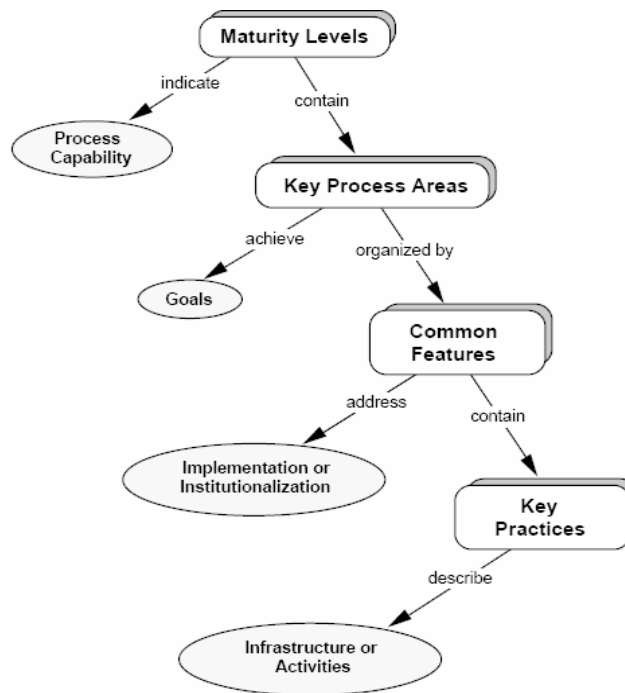


Figura 2.2 Estructura del SW-CMM

Las áreas clave de proceso pueden ser consideradas los requerimientos para alcanzar un nivel de madurez. Para que esto se cumpla, las áreas claves para ese nivel (y sus niveles inferiores) deben satisfacerse y los procesos deben institucionalizarse.

Áreas claves de proceso en el Nivel 2

- Administración de requerimientos (RM)
- Planeamiento del proyecto de software (PP)
- Control de proyectos de software (PTO)
- Administración de subcontratos (SM)
- Aseguramiento de la calidad del software (QA)
- Administración de configuración de software (CM)

Áreas claves de proceso en el Nivel 3

- Concentración del proceso organizacional (PF)
- Definición de procesos organizacionales (PD)
- Programa de capacitación (TP)
- Administración integral de software (IM)
- Ingeniería de productos de software (PE)

- Coordinación entre grupos (IC)
- Revisiones por pares (PR)

Áreas claves de proceso en el Nivel 4

- Administración cuantitativa de procesos (QPM)
- Administración de calidad de software (SQM)

Áreas claves de proceso en el Nivel 5

- Prevención de Defecto (DP)
- Administración de Cambio de Tecnología (TC)
- Administración del Cambio del Proceso (PC)

Nota: Contenido desarrollado en base a las referencias: 2, 3, 24

2.3 ISO/IEC 15504 (SPICE)

SPICE (Software Process Improvement and Capability dEtermination) es un framework estándar internacional para la evaluación de procesos de software que incorpora ideas de:

- SW-CMM
- BOOTSTRAP
- TRILLIUM
- Estándares de compañías (AT&T, HP, etc.)

Esta evaluación de procesos puede utilizarse tanto para propósitos de mejora de procesos como para la determinación de la capacidad (ver figura 2.3).

Describe la madurez de un proceso de software desde la perspectiva de procesos individuales, no como una mejora a nivel organización.

Entre los usos de este modelo figuran:

- Comprender el estado de los procesos de la organización para realizar mejora de procesos
- Determinar la aplicabilidad de los procesos de la organización a un conjunto de requerimientos.
- Determinar la aplicabilidad de los procesos de otra organización ante la contratación de la misma.



Figura 2.3 Diagrama de posibles usos de SPICE

Documentos de SPICE: Proveen un esquema para la evaluación del proceso software.

- **Parte 1 - Guía Introductoria**

Describe cómo seguir las partes y su interacción.

Explica los requerimientos que contiene y su aplicabilidad para conducir evaluaciones, para construir y seleccionar herramientas de soporte y para construir procesos extendidos.

- **Parte 2 - Modelo de referencia para procesos y para capacidad de los procesos**

Explicado a continuación

- **Parte 3 – Como realizar el assessment**

Describe los requerimientos para el método de assessment, sin llegar a definir el método (solo los requerimientos).

- **Parte 4 - Guía para conducir la assessment**

Es aplicable a todas las organizaciones y posibilita usar una variedad de métodos y técnicas soportados por diversas herramientas.

- **Parte 5 – Modelo de assessment y guía de indicadores**

Define un esquema de elementos requeridos para desarrollar un instrumento de ayuda en el assessment de un proceso.

Es una guía para los compradores o diseñadores en la selección y utilidad de varios tipos de instrumentos de assessment.

- **Parte 6 - Calificación y entrenamiento de evaluadores**

Describe la competencia, educación, entrenamiento y experiencia requerida por los evaluadores.

- **Parte 7 - Guía para usar en la mejora del procesos**

Especifica como definir y usar los resultados de un proceso de evaluación para la mejorara de procesos. Incluye ejemplos.

- **Parte 8 - Guía para usar en la determinación de la capacidad de proceso del proveedor**

Especifica como definir y usar los resultados de un proceso de assessment para determinar la capacidad del proceso.

- **Parte 9 - Vocabulario**

Documento Parte 2 - Modelo de referencia para procesos y para la capacidad de los procesos

Define el modelo de referencia, incluyendo los requerimientos para el modelo de assessment, pero no puede ser usado como modelo de assessment.

- El modelo de referencia se compone de dos dimensiones: La dimensión del proceso y la dimensión de la capacidad del proceso. Los procesos (y categorías de procesos) que define la primer dimensión son mapeables a los niveles de capacidad definidos por la segunda dimensión.

- **Dimensión del proceso**

Describe la organización típica de desarrollo de software (ver figura 2.4).

El modelo de referencia agrupa los procesos en la dimensión del proceso de acuerdo a tres grupos de procesos del ciclo de vida que contienen categorías de procesos. Ellos son:

- **Procesos primarios del ciclo de vida**, que consiste de las siguientes categorías de procesos:
 - **Customer-supplier (CUS)**: Procesos que impactan directamente al cliente.
 - **Engineering (ENG)**: Procesos que especifican, implementan o mantienen un sistema y productos de software.
- **Procesos de soporte del ciclo de vida**, que tienen la única categoría:
 - **Support (SUP)**: Procesos que hacen posible y soportan la performance de otros procesos del proyecto.
- **Procesos de organización del ciclo de vida**, que consta de las siguientes categorías:
 - **Management (MAN)**: Procesos que establecen el proyecto, y coordinan y gerencian sus recursos.
 - **Organization (ORG)**: Procesos que establecen los objetivos de negocio de la organización y desarrollan procesos, productos y recursos que ayudan a que la organización alcance sus objetivos.

A su vez, cada categoría de procesos contiene numerosos procesos.

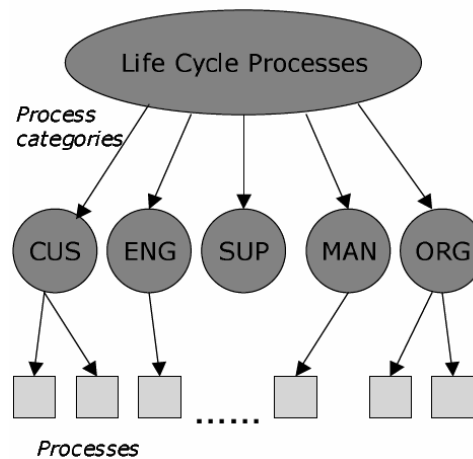


Figura 2.4 Estructura de SPICE

- **Dimensión de la capacidad del proceso**

Define los perfiles de niveles de madurez, donde cada proceso es valido que recaiga en uno de los siguientes niveles de capacidad:

Nivel 0. Incompleto

Hay fallas generales como para cumplir con el propósito del proceso. Hay poca o dificultosa identificación de los workproducts o salida de los procesos

Nivel 1. Ejecutado

El propósito del proceso es generalmente alcanzado. El cumplimiento puede no ser rigurosamente planeado y seguido. Existe identificación de los work products del proceso.

Nivel 2. Administrado

El proceso genera work products de acuerdo a los procedimientos especificados que son planeados y seguidos. Los work products se alinean a los estándares y a los requerimientos.

Nivel 3. Establecido

El proceso es realizado y administrado utilizando un proceso definido que esta basado en los buenos principios de ingeniería de software. Las implementaciones particulares del proceso utilizan procesos aprobados, ajustados a versiones estándares y documentados con el propósito de cumplir con las expectativas del proceso.

Nivel 4. Predecible

El proceso definido es realizado consistentemente dentro de los límites de control definidos para lograr sus objetivos de procesos definidos.

Nivel 5. Optimizado

La performance del proceso es optimizada para alcanzar las necesidades del negocio actuales y futuras, y repetidamente el proceso logra sus objetivos definidos de negocio.

La medición de la capacidad esta basada contra un conjunto de atributos de procesos.

Los atributos del proceso son característicos de un proceso que puede ser evaluado sobre una escala de cumplimiento, proveyendo una medición del nivel de capacidad de los procesos.

Nota: Contenido desarrollado en base a las referencias: 18, 24, 25, 26, 64, 65

2.4 BOOTSTRAP

Bootstrap es una metodología para mejorar la calidad de los procesos de desarrollo de software, que es desarrollada y mantenida por el Bootstrap Institute.

El principio de la aproximación Bootstrap es determinar a través de un assessment cual es el faltante entre el estado actual de los procesos y el estado deseado para un aspecto particular del negocio.

Bootstrap esta basado en el SW-CMM y utiliza las guías de la serie de la ISO 9000. A su vez, Bootstrap cumple totalmente con la ISO/IEC 15504 (SPICE).

Si bien los niveles de madurez de Bootstrap difieren de los de SW-CMM, es posible comparar los resultados de assessment de ambos.

Principales características

- Para conducir el assessment se ofrece un modelo de referencia y un mecanismo para ranquear y dar puntaje.
- Provee perfiles de niveles de madurez y de capacidad, tanto para la organización como para los proyectos.
- La salida del assessment esta expresada en niveles de madurez (por proceso).
- Las áreas de proceso de Bootstrap no están limitadas a un único nivel de madurez (como en las KPA de SW-CMM)
- Bootstrap usa una aproximación no normativa (por ejemplo, durante el assessment una organización puede ignorar algunos procesos que no son importantes para alcanzar los objetivos).
- Permite opcionalmente determinar el grado de cumplimiento de 20 elementos de la ISO 9001.
- Permite opcionalmente determinar el grado de cumplimiento los niveles de madurez de SW-CMM.

Estructura

Bootstrap ofrece dos importantes ayudas para conducir un assessment: un modelo de referencia y un mecanismo para ranquear y dar puntaje (dimensión de la capacidad).

- **Modelo de referencia (o modelo de proceso)**

Es usado como una referencia y guía para identificar buenas practicas de administración e ingeniería de software. Está basado en la ISO 12207 y la ISO 15504 (SPICE).

Los procesos de una organización de software son categorizados según:

- Organización
- Metodología

La categoría de procesos llamada metodología es dividida en:

- Funciones independientes del ciclo de vida
 - Funciones dependientes del ciclo de vida
 - Funciones relacionadas al proceso
- Tecnología.

Estas categorías son mostradas en la figura 2.5.

Las funciones son descompuestas en procesos, y cada proceso es descrito como un conjunto de prácticas bases.

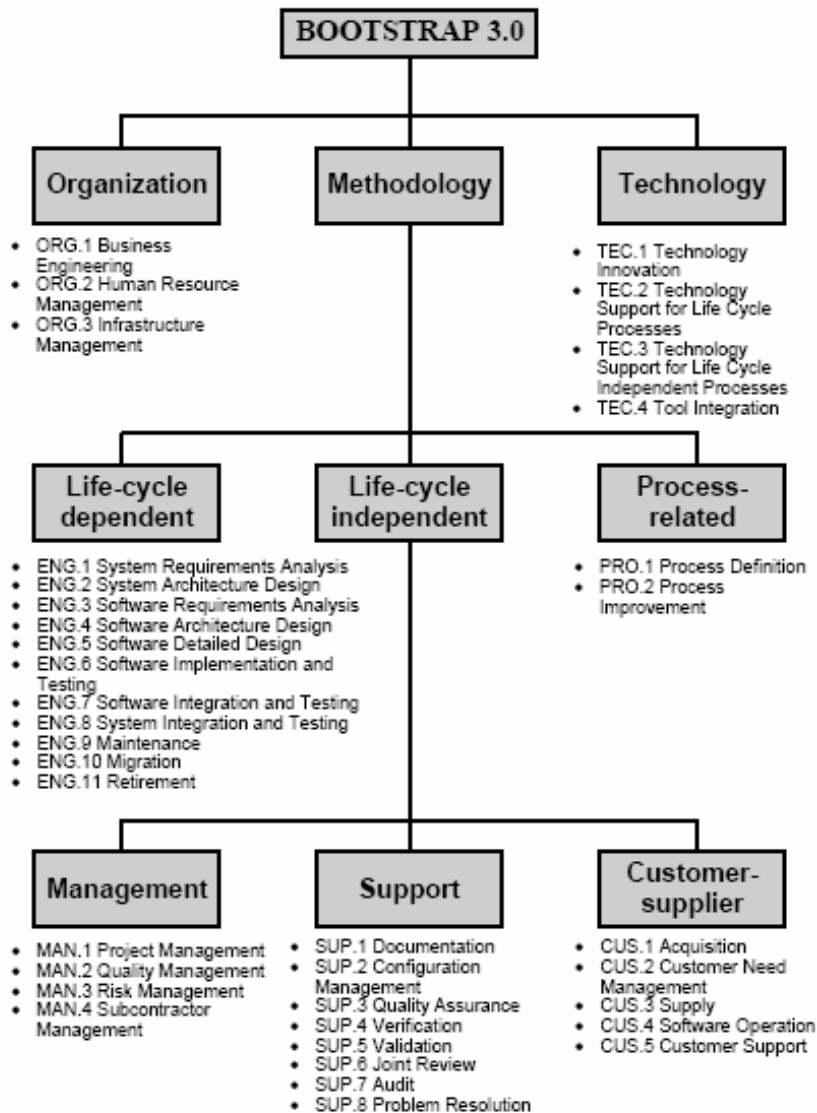


Figura 2.5 Categorías de procesos en Bootstrap

• **Mecanismo para ranquear y dar puntaje (dimensión de la capacidad)**

En el assessment se utilizan un conjunto de cuestionarios para determinar el nivel de madurez por proceso.

La dimensión de capacidad del modelo Bootstrap consiste de seis niveles de capacidad que coinciden con los niveles de la ISO 15504 (SPICE)

- **Nivel 0. Incompleto**
- **Nivel 1. Ejecutado**
- **Nivel 2. Administrado**
- **Nivel 3. Establecido**
- **Nivel 4. Predecible**
- **Nivel 5. Optimizado**

Cada nivel es descrito como un conjunto de una o más prácticas de administración. Mediante el uso de cuestionarios a cada proceso se lo compara contra el modelo de referencia para determinar cuales prácticas de administración están implementadas. Esto es utilizado como el indicador de la capacidad del proceso. Un nivel de madurez es alcanzado implementando las prácticas administrativas adicionales requeridas para ese nivel.

Nota: Contenido desarrollado en base a las referencias: 17, 21

2.5 CMMI-SW staged

El modelo de CMMI involucra los conceptos establecido por el Modelo de Madurez de la Capacidad para Software (SW-CMM) expandiendo el concepto CMM a múltiples disciplinas o cuerpos de conocimiento, como por ejemplo:

- **Ingeniería de Sistemas:** cubre el desarrollo total de sistemas, los cuales pueden o no incluir software.
- **Ingeniería de Software:** cubre el desarrollo de sistemas software.
- **Integración de Proceso de desarrollo y Producto:** es un enfoque sistemático para lograr una colaboración periódica de los participantes relevantes a lo largo de la vida del producto para satisfacer las necesidades del cliente, las expectativas y los requisitos.
- **Fuentes de suministro:** Esta disciplina cubre la adquisición de productos cuando se requiere de un análisis de las fuentes y del monitoreo de las actividades de suministros antes de liberar el producto.

La organización puede usar el modelo CMMI para ayudar a fijar sus objetivos de mejora de proceso, mejorar los procesos y para proveer una guía para asegurar un proceso estable, capaz y maduro. El modelo CMMI seleccionado sirve como guía para la mejora del proceso organizacional.

El modelo CMMI tiene dos tipos de representaciones:

- Staged (por etapas), que se enfoca en la medición del proceso usando niveles de madurez.
- Continua, que enfoca la medición del proceso para cada una de las áreas del proceso, usando niveles de capacidad.

Estructura

La representación staged organiza las áreas de proceso en cinco niveles de madurez para soportar y guiar a la mejora de proceso. Los componentes del modelo están graficados en la figura 2.6.

Se agrupan las áreas del proceso por nivel de madurez, indicando cuales áreas de proceso se deben implementar para alcanzar cada nivel de madurez.

Los niveles de madurez muestran la evolución de la organización dentro del ámbito de la mejora de procesos.

Cada área de proceso involucra objetivos y prácticas específicos que se basan en objetivos y prácticas genéricas.

La representación staged usa cuatro características comunes para organizar las prácticas genéricas:

- Compromisos para desarrollar
- Habilidades a desarrollar
- Orientación de la implementación
- Verificación de la implementación

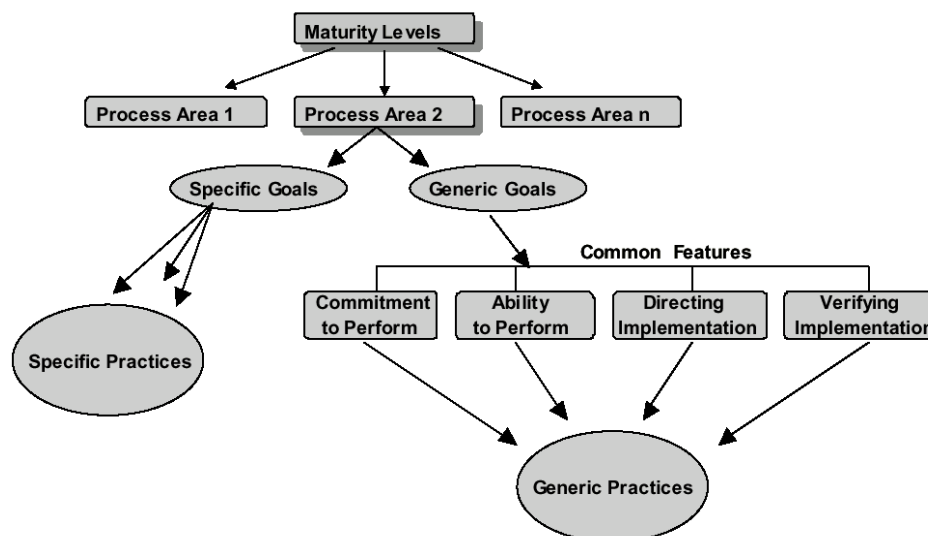


Figura 2.6 Componentes del modelo CMMI-SW

Un nivel de madurez consiste de prácticas genéricas y prácticas específicas relacionadas para un conjunto de áreas de proceso predefinidas para mejorar el desempeño de toda la organización.

El nivel de madurez de organización provee una forma de predecir el desempeño de la organización en una o varias disciplinas.

Cada nivel de madurez establece una parte importante del proceso organizacional, y prepara a la organización para ir hacia el siguiente nivel de madurez.

Los niveles de madurez son medidos por el logro de los objetivos específicos y los objetivos genéricos asociados con cada conjunto de áreas predefinidas.

Niveles de madurez del CMMI-SW staged

Los niveles del CMMI staged tienen las mismas definiciones que el CMM, aunque se hicieron algunos cambios en los nombres de los niveles 2 y 4 para enfatizar con mayor claridad la evolución de la administración de procesos desde un enfoque cualitativo a uno cuantitativo

Nivel 1: Inicial

Nivel 2: Gestionado

Nivel 3: Definido

Nivel 4: Gestionado cuantitativamente

Nivel 5: Optimizado

El CMMI-SW staged contiene 25 áreas de proceso y 411 prácticas para las cuatro disciplinas actualmente cubiertas, mientras que el CMM contiene 18 KPA y 150 prácticas. Las áreas de proceso por nivel se muestran en la tabla 2.2

Nivel de Madurez	Área de Proceso		Categoría
5	Causal Analysis and Resolution	CAR	Support
5	Organizational Innovation and Deployment	OID	Process Mgt
4	Organizational Process Performance	OPP	Process Mgt
4	Quantitative Project Management	QPM	Project Mgt
3	Decision Analysis and Resolution	DAR	Support
3	Integrated Project Management	IPM	Project Mgt
3	Integrated Teaming	IT	Project Mgt
3	Organizational Environment for Integration	OEI	Support
3	Organizational Process Definition	OPD	Process Mgt
3	Organizational Process Focus	OPF	Process Mgt
3	Organizational Training	OT	Process Mgt
3	Product Integration	PI	Engineering
3	Requirements Development	RD	Engineering
3	Risk Management	RSKM	Project Mgt
3	Technical Solution	TS	Engineering
3	Validation	VAL	Engineering
3	Verification	VER	Engineering
2	Configuration Management	CM	Support
2	Measurement and Analysis	MA	Support
2	Process and Product Quality Management	PPQA	Support
2	Project Monitoring and Control	PMC	Project Mgt
2	Project Planning	PP	Project Mgt
2	Requirements Management	REQM	Engineering
2	Supplier Agreement Management	SAM	Project Mgt

Tabla 2.2 Relación Nivel x Área de proceso x Categoría en CMMI Staged

El nivel 2 de CMMI quedó relativamente intacto respecto del SW-CMM:

- Software Subcontracting ha sido renombrado como Supplier Agreement Management y cubre un rango de situaciones de adquisición y contratación.
- Measurement and Análisis es una nueva área de procesos que consolida las prácticas previamente encontradas en el CMM como Measurement and Análisis Common Feature en una sola área de proceso.

El nivel 3 fue el más modificado:

- La práctica de CMM llamada Software Product Engineering ha sido expandida en 5 áreas de procesos en el CMMI:
 - Requirements Development Addresses
 - Technical Solution
 - Product Integration
 - Verification
 - Validation
- Risk Management es una nueva área al igual que Decision Analysis and Resolution.
- Se agregan dos áreas de procesos: Integrated Teaming y Organized Environment for Integration.

En el nivel 4 se modificó:

- Software Quality Management and Quantitative Product Management del CMM ha sido reemplazada por dos nuevas áreas de procesos: Organizational Process Performance y Quantitative Project Management.

El nivel 5 no ha cambiado dramáticamente con la liberación del CMMI.

- Process Change Management y Technology Change Management del CMM han sido combinadas en un área de proceso: Organizational Innovation and Deployment.
- Defect Prevention ha sido renombrada como Causal Análisis and Resolution

Nota: *Contenido desarrollado en base a las referencias: 19, 20, 87*

2.6 Trillium

El modelo está orientado a productos de telecomunicaciones. Fue desarrollado por Bell Canada, Northern Telecom y Bell-Northern Research. Bell Canada lo utiliza para evaluar el desarrollo de productos y soportar la capacidad de los actuales y potenciales proveedores.

Algunas características de Trillium son:

- La arquitectura del modelo está basada en mapas de ruta, en vez de áreas clave de proceso.
- Tiene una orientación al producto, en lugar de hacia el software
- Hace foco en el cliente, en la madurez tecnológica y en la orientación a las telecomunicaciones.

Trillium apunta a mejorar la capacidad de desarrollo de las organizaciones en base a la entrega consistente de productos o en base a la mejora de productos:

- Alcanzando los requerimientos del usuario
- Teniendo mínimos defectos
- Al menor costo de ciclo de vida
- En el menor tiempo

Trillium está basado en el modelo SW-CMM, pero lo complementa con aspectos de la ISO 9001, ISO 9003-3, etc. El alcance del modelo se extendió para incluir tanto software como hardware.

Estructura

Trillium tiene cinco niveles especificados en el modelo que son mostrados en la figura 2.7. Estos niveles son:

- No estructurado
- Repetible y orientado al proyecto
- Definido y orientado al proceso
- Integrado y gerenciado
- Totalmente integrado

A su vez Trillium esta dividido en 8 áreas de capacidad, cada una de las cuales contiene diferentes prácticas, agrupadas en mapas de ruta.

- Organizational Process Quality
- Human Resource Development and Management
- Process
- Management
- Quality System
- Development Practices
- Development Environment
- Customer Support

Los mapas de ruta contienen prácticas de varios niveles de madurez. De esta forma, a medida que una organización avanza al próximo nivel de madurez, se implementan las variantes mas maduras de las prácticas de distintos mapas de ruta.

Trillium contempla explícitamente las actividades de soporte en Customer Support capability area. Esta área de capacidad consta de 6 mapas de ruta que abarcan los problemas de respuesta al sistema, ingeniería de uso, modelado del costo del ciclo de vida, documentación de usuario, ingeniería del cliente y capacitación del usuario.

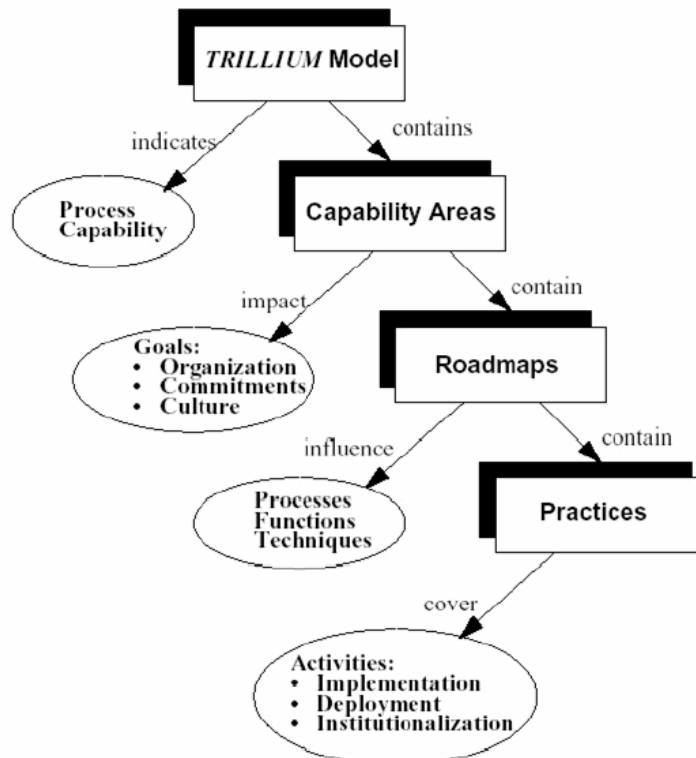


Figura 2.7 Estructura de Trillium

Nota: Trillium no puede ser fácilmente aplicada a una industria de servicios de IT.

Capítulo 3

Tarea 1

Definir objetivos de la organización sobre la implementación y definir criticidad

3.1 Identificación

Tarea 1

3.2 Introducción al capítulo

Los tiempos actuales son muy volátiles desde el punto de vista de los negocios, y para sobrevivir se debe ser capaz de institucionalizar cambios que sean buenos tanto desde el aspecto técnico como desde el aspecto económico, teniendo en cuenta al mismo tiempo los riesgos y las incertidumbres.

Los objetivos organizacionales contemplados por el framework de mejora son:

- Reducir el Time to market
- Incrementar la productividad
- Evitar costos
- Mejorar la calidad

El propósito de este capítulo es seleccionar un subconjunto de estos objetivos, de modo que al realizar la implementación de la mejora de procesos se haga foco en el marco de proceso y en los objetivos elegidos por la organización para la implementación, especialmente a la hora de la elección de las mejores prácticas a implementar.

3.3 Estructura de la tarea

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> • Interés de la organización por implementar 	<p style="text-align: center;">TASK: Tarea 1</p> <p>Definir objetivos de la organización sobre la implementación y definir criticidad</p>	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> • Dato01: Lista de objetivos principales • Dato31: Criticidad del software desarrollado
	<p style="text-align: center;">VERIFY</p> <p>Justificación y aprobación de los altos mandos de la organización.</p>	

3.4 Consideraciones a la hora de plantear los objetivos de la organización

Las razones por las cuales la organización impulsa este cambio, pueden ser:

- Seguir el ritmo a la competencia
- Alcanzar beneficios económicos
- Los cambios pueden ser estimulados por la necesidad de nuevos productos o líneas de productos o una nueva tecnología
- Alcanzar la eficiencia

Las decisiones a tener en cuenta a la hora de seleccionar los objetivos pueden ser:

- ¿La implementación resultará en mejoras que permitirán ahorrar dinero, bajar costos, incrementar productividad, acelerar los desarrollos, y / o mejorar la calidad? Si la respuesta es si, ¿Cuanto costará comparado con los números promedios de la industria? ¿Cuales son las alternativas?
- ¿Cual es el impacto de la innovación en la organización?

- ¿Quiénes son los clientes para el cambio y cuáles son sus expectativas? ¿Es el cambio propuesto compatible con la cultura de la organización?
- ¿La competencia ha seguido alternativas similares? Si la respuesta es sí: ¿Cuánto gastaron y cuál fue el resultado?
- ¿Se cuenta con el talento y otros recursos necesarios para realizar la innovación? Si la respuesta es no, ¿Cómo se adquirirán y a qué costo?
- ¿Porque se debe invertir en una idea como esta en lugar de otras similares?
- ¿La organización es consistente con los objetivos propuestos?
- ¿La organización es compatible con la madurez de los procesos a implementar?

3.5 Consideraciones sobre los posibles objetivos a seleccionar

3.5.1 Reducir el time to market

El time to market se refiere a la cantidad de tiempo que le toma a una organización en desarrollar un producto y tenerlo listo para vender o proporcionarlo. Una organización que pone rápidamente sus productos en el mercado se beneficia con ventajas competitivas sobre sus competidores.

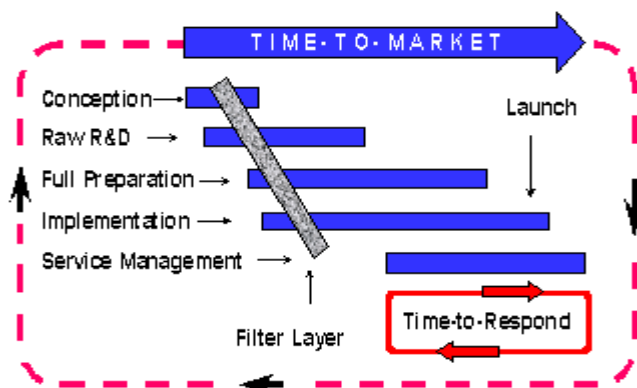


Figura 3.1 Ciclo Time to market

El ciclo del time to market (figura 3.1) comienza con la concepción del producto e incluye aquellos aspectos que están involucrados en el desarrollo del producto, pero también incluye aquellos elementos de negocio que soportan la puesta del producto en el mercado.

El tiempo de respuesta del ciclo es crucial porque esta en función de las necesidades de los clientes y el mercado, a los cuales se les debe responder lo más rápido posible.

El foco del cambio en el time to market no afecta principalmente al producto, pero sí afecta a las dinámicas y arquitectura de la organización. Este cambio afecta a la totalidad de la organización.

El time to market provee una visión a largo plazo de las formas por las cuales el valor es creado como un proceso integrado, corriendo como ejecuciones en paralelo a través de todo el negocio. Analizar el negocio en términos de este proceso otorgará una clara dirección acerca de las prioridades de cambio.

Time to market puede ser visto como un indicador de performance, ya que puede verse como la velocidad que tiene la organización.

Por ejemplo, cuanto más rápido sea el time to market, mejor será la organización para desarrollar, lanzar y soportar sus desarrollos.

Nota: Contenido desarrollado en base a las referencias: 27, 31

3.5.2 Mejorar la calidad

La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario.

“... existe una tendencia marcada en incrementar la calidad de los ambientes de desarrollo, el proveedor que mejor lo logre va ser el que lidere el mercado en los próximos años.”

Los factores que determinan la calidad del software se centran en tres de los aspectos importantes del software:

- Características operativas
 - Corrección (¿Hace lo que se desea?)
 - Fiabilidad (¿Son confiables sus resultados?)
 - Eficiencia
 - Seguridad (Integridad)
 - Facilidad de uso
- Capacidad de soportar cambios
 - Facilidad de mantenimiento
 - Flexibilidad
 - Facilidad de prueba
- Adaptabilidad a nuevos entornos
 - Portabilidad
 - Reusabilidad
 - Interoperabilidad (¿Puede operar con otros sistemas?)

Para mejorar la calidad la mayoría de las organizaciones deben cambiar su entendimiento sobre el tema, ya que tradicionalmente la calidad es vista como el costo puntual por retrabajo y corrección de defectos. La visión que deberían tener es que la calidad es un generador de valor que mejora la satisfacción del cliente, alinea a los sistemas con los objetivos corporativos de la organización, y reduce el costo global de propiedad del software.

Para mejorar la calidad las organizaciones deben enfatizar los siguientes puntos:

- **Liderazgo:** Los mandos altos deben ver a la calidad como una prioridad primordial, respaldándola con acciones claves. Estas acciones pueden ser:
 - Establecer objetivos por factores (prevención de defectos, entrega a tiempo, niveles de servicio)
 - Monitoreo constante del progreso
 - Fomentar la comunicación abierta
 - Establecer los objetivos de calidad como parte de las evaluaciones profesionales
- **Entrenamiento efectivo:** Todos los miembros del staff deben recibir adecuado entrenamiento para ayudarlos a comprender los procesos de calidad.
- **Motivación continua:** Compromiso de todos los niveles de la organización para asegurar que la calidad es más que un simple slogan. Utilizar incentivos y premios podría ser una herramienta útil.
- **Eficiencia en la operación:** Más allá de la eliminación de defectos de software, la calidad se refiere a la performance del sistema, su disponibilidad y sus niveles de servicios. Las organizaciones deben monitorear continuamente los indicadores claves de calidad.

Las organizaciones deberían ver a la calidad como:

- Un profundo proceso cultural que no se hace de un solo paso, sino continuamente.
- Una inversión que recae en beneficios competitivos, no como un costo.

Nota: Contenido desarrollado en base a las referencias: 29, 32, 68, 71

3.5.3 Evitar costos

Es una práctica que la mayoría de las organizaciones adoptan después de ver la facilidad con la que los costos relativos a IT salen de control.

Esta es una medida tomada en el presente y que tiene por objetivo reducir los costos en el futuro.

No es correcto pensar que la mejora de la productividad es equivalente a la reducir / evitar costos, ya que en muchos casos incrementar la productividad resulta en un incremento en los costos.

Es preferible evitar costos que reducirlos, ya que la primera opción trabaja en base al futuro y no con los gastos actuales. La forma de hacerlo es presentando ideas que justifiquen el ahorro de costos trasladándolos a acciones más inmediatas.

Algunas guías a tener en cuenta al evitar costos son:

- Separar las consideraciones de costos respecto de las de productividad.
- Tener conocimiento del sistema monetario (que ocurre cuando sobra dinero del presupuestado y como se podría utilizar)
- Saber como están asociados los costos actuales respecto de los diferentes centros de costos.
- Saber cuales costos se pueden controlar y quienes controlan los otros costos.

Algunos ejemplos de cómo evitar o recortar costos pueden ser:

- Acelerar el Time to market de las iniciativas de negocio basadas en tecnología.
- Basarse en iniciativas a corto plazo.
- Realizar un proceso de outsourcing selectivo.
- Incrementar el staff propio de la organización.
- Racionalizar y simplificar la infraestructura.
- Mejorar las prácticas de adquisición de servicios y productos de IT.

Nota: *Contenido desarrollado en base a las referencias: 33, 82, 83*

3.5.4 Incrementar la productividad

Tradicionalmente la productividad se mide por cantidad de miles de líneas de código hechas por hora. Sin embargo, el proceso clave en la actividad de desarrollo es la transformación de ideas en productos. Entonces, para medir la productividad real del desarrollo de software, se debe ver en que nivel de eficiencia y eficacia se transforman ideas en productos. A partir de este hecho, se obtienen las siguientes definiciones de productividad del desarrollo de software:

- Para las compañías que desarrollan y venden software como un producto, la productividad puede ser definida como la ganancia producida por cada empleado.
- Para organizaciones internas de IT, la productividad puede ser definida como el rédito creciente observado en el negocio versus cada dólar gastado por la organización de IT.

Si la ganancia del trabajo generado por hora se incrementa, crecen nuestros niveles de ingreso. Por el contrario, si la productividad recae, las ganancias también declinan.

En un mercado donde todo es muy parejo, la compañía más productiva disfrutará de mejores beneficios.

De esta forma, las claves para mantener e incrementar las ganancias en la industria del desarrollo de software se deben a mejoras anuales sobre la productividad del desarrollo de software.

Por otro lado, no se puede depender del constante crecimiento del sector tecnológico como vía para darle alas al crecimiento de la productividad. Una solución a este problema es realizar un serio esfuerzo para incrementar la productividad a través del uso más eficiente del trabajo y el darle un valor más efectivo a los clientes. Esta es la forma en que los sectores económicos maduros han incrementado la productividad por décadas.

Existen tres aproximaciones para la mejora de la productividad:

- Reducir los costos del producto eliminando inversiones en características del producto a las cuales el cliente no le da valor.
- Reducir los costos indirectos mediante la agilización de procesos y eliminación de ineficiencias en el desarrollo, entrega y soporte del producto.
- Incrementar ganancias agregando valor a los productos. De esta forma, los clientes pagaran más por ellos.

Capítulo 4

Tarea 2

Definir / identificar marco de proceso utilizado

4.1 Identificación

Tarea 2

4.2 Introducción al capítulo

El objetivo de esta tarea es seleccionar o definir un marco de proceso que utilice la organización como estrategia a la hora de hacer sus desarrollos de software para cumplir con los objetivos organizacionales.

Esta tarea es de fundamental importancia ya que permite establecer los aspectos importantes que caracterizan a los procesos de la organización, en base a:

- Workflow de tareas desempeñadas durante el ciclo de vida de desarrollo de software
- Objetivos de negocio que tiene la organización mostrados implícitamente a través del marco de proceso
- Software de documentación utilizados durante las distintas etapas
- Roles involucrados
- Criterios de entrada y salida de cada una de las etapas

Para ello, es de vital importancia (es el objetivo principal de esta tarea) seleccionar / identificar el framework de proceso que satisfaga los objetivos del negocio.

El resultado del proceso de mapeo entre los objetivos de la organización versus las practicas del modelo de referencia consistirá en tener el conjunto de practicas alineadas con el modelo, permitiendo satisfacer esos objetivos en forma efectiva ya que las practicas siempre apuntaran a los objetivos.

4.3 Estructura de la tarea

<p style="text-align: center;">ENTRY</p> <p>Dato01: Lista de objetivos principales</p>	<p style="text-align: center;">TASK: Tarea 2</p> <p>Identificar o seleccionar un marco de proceso para la organización que sea usado como estrategia de desarrollo y que satisfaga los objetivos de la organización</p>	<p style="text-align: center;">EXIT</p> <p>Dato30: Framework de procesos</p>
	<p style="text-align: center;">VERIFY</p> <p>El marco de proceso en uso o a ser utilizado esta identificado, y existe aceptación entre los lideres y gerentes respecto de la identificación</p>	

4.4 Marcos de procesos a seleccionar

Antes de exponer algunos de los marcos de procesos que son utilizados en el mercado, se debe notar que existen dos corrientes de marcos de procesos:

- Los marcos de proceso pesados (también llamados métodos plan-driven)
- Los marcos de proceso livianos (también conocidos como métodos ágiles)

Nota: En las siguientes secciones se enuncian las características generales de estos marcos de proceso, y puntualmente se dan ejemplos de algunos de ellos. Sin embargo, analizar en profundidad los marcos disponibles en el mercado y como aplicarlos en la organización esta fuera del alcance de este trabajo.

4.4.1 Marcos de proceso pesados

Los marcos de proceso pesados se caracterizan por:

- Hacen foco en predictibilidad y repetitividad
- Procesos de mejora repetidos, estandarizados y de mejora incremental
- Documentación detallada: formularios, procedimientos, guías, etc.
- Una arquitectura de sistema de software bien definida
- Descripciones detalladas de planes, roles, responsabilidades y work products
- Grupos de procesos que contienen recursos específicos: monitoreo, control y capacitación sobre los procesos
- Gerenciamiento de riesgos sobre la marcha
- Hacen foco sobre validación y verificación

Las organizaciones o proyectos donde es mas frecuente aplicar los marcos de proceso pesados se caracterizan por tener:

- Alta criticidad
- Pocos cambios en los requerimientos
- Alto numero de desarrolladores
- Una cultura organizacional que demanda orden

4.4.1.1 Ejemplos de marcos de proceso pesados

4.4.1.1.1 PSP (Personal Software Process)

Descripción

Es un framework estructurado que se compone de formularios, guías y procedimientos desarrollados por el SEI. Este framework guía a un ingeniero en el uso de un proceso definido, medido, planificado y controlado a nivel calidad.

El aprendizaje del PSP sigue una aproximación de mejora evolutiva. Un ingeniero aprendiendo a integrar PSP en sus procesos arranca en un nivel de proceso 0 hasta alcanzar un nivel de madurez en el nivel 3 (ver figura 4.1).

Cada nivel se compone de detallados scripts, cheklists y templates que guían al ingeniero a través de los pasos requeridos.

PSP provee una realimentación basada en mediciones que ayuda al ingeniero a mejorar su proceso de software.

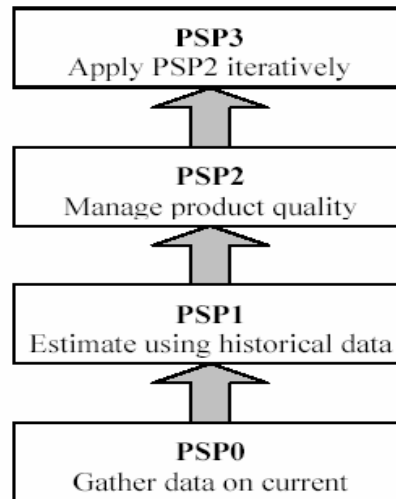


Figura 4.1 Niveles PSP

PSP tiene fuertes principios:

- Cuanto mas tiempo permanece un defecto en el software, mas costoso será detectarlo y corregirlo.
- La prevención de defectos es más eficiente que su corrección.

Documentos y artefactos

Los artefactos que usa PSP son scripts, formas, templates, standards y checklists:

- **Scripts**: Proveen una estructura ordenada de pasos que el ingeniero debería realizar para completar un paso del proceso. Cada script cuenta con un criterio de aceptación o salida. PSP posee seis clases de diferentes scripts:
 - Scripts de proceso: Contiene los pasos para completar un proceso.
 - Planning scripts: Instrucciones para planificar el desarrollo.
 - Scripts de diseño de alto nivel: Provee instrucciones para crear diseños de alto nivel.
 - Scripts de revisiones de diseño de alto nivel: Provee instrucciones para revisar diseños de alto nivel.
 - Scripts de desarrollo: Delinea los pasos para diseñar, codificar, compilar y testear.
 - Scripts de postmortem: Provee las instrucciones para analizar y sumarizar los datos recolectados en forms y templates.
- **Formularios**: Son usados para completar con la información recolectada. Los formularios se usan cuando la cantidad de datos es fija. Hay tres tipos de formularios
 - Almacenamiento de defectos: Recolectar información acerca de los defectos que son introducidos y eliminados del proyecto.
 - Log de tiempos: Se almacena cuanto tiempo es invertido en las actividades de desarrollo.
 - Plan del proyecto: Es un formulario usado para analizar, sumarizar y utilizar de acuerdo a la información recolectada.
- **Templates**: PSP tiene templates para reportes de test, estimación de tamaño, planificación de cronograma, log de seguimiento de ítems, y para la creación de escenarios
- **Checklists**: Ayudan a seguir completamente un procedimiento. La idea es que el desarrollador aprenda de los defectos que genera.

Roles

PSP tiene un único rol, que es el ingeniero de software.

Procesos

PSP esta basado en cuatro niveles de procesos personales. Las habilidades de un nivel deben ser dominadas antes que el ingeniero pase al próximo nivel.

- **Nivel 0** (Mediciones personales): Las entradas al PSP son los requerimientos. Uno de sus scripts tiene tres etapas estilo cascada: planeamiento, desarrollo (incluyendo diseño, código, compilación y test) y postmortem. Se comienza estableciendo un baseline personal de sus procesos de desarrollo mediante mediciones básicas, tales como tiempo gastado en un programa, defectos insertados y corregidos, etc. Este nivel es luego mejorado agregando standards de codificación, mediciones de tamaño y el desarrollo de un plan personal de mejora de procesos.
- **Nivel 1** (Planeamiento personal): Basado en las mediciones del nivel anterior, el ingeniero estima que tan grande será un nuevo programa. Cada nuevo proyecto se registra con su tiempo gastado. Esta información es usada para planificación y estimación de tareas y cronogramas.
- **Nivel 2** (Calidad Personal): El foco se pone en la prevención de defectos y su eliminación. Se utilizan checklists para las revisiones de código y diseño. Se introducen especificaciones de diseño y técnicas de análisis.
- **Nivel 3** (Ascendiendo): El programador emplea un modelo incremental de desarrollo para proyectos dividiendo el problema en partes pequeñas, y luego iterativamente aplica los principios de PSP en cada parte.

Ventajas

- Mejora en la estimación de tiempos y tamaño
- Mejora de productividad
- Reducción de tiempo de testeo
- Mejora de la calidad

Desventajas

- Algunas personas no lo pueden aplicar por no ser receptivas en la recolección de información detallada.
- La longevidad de PSP requiere disciplina

Aclaración

PSP puede ser usado como un programa de entrenamiento, mejorando la calidad de aquellas personas que componen el staff de desarrollo, debido a que es muy personal.

Nota: *Contenido desarrollado en base a las referencias: 44, 45*

4.4.1.1.2 RUP (Rational Unified Process)

Descripción

RUP es un framework de procesos genérico que puede ser especializado para una gran variedad de sistemas de software, diferentes áreas de aplicación, diferentes tipos de organización, diferentes niveles de competencia, y diferentes tamaños de organizaciones.

La idea alrededor de la creación de RUP fue la de capturar muchas de las mejores prácticas del desarrollo moderno de software y que puedan ser fácilmente compuestas en un proceso adaptable a un amplio rango de proyectos y organizaciones.

RUP trabaja con técnicas orientadas a objetos y utiliza UML como notación principal.

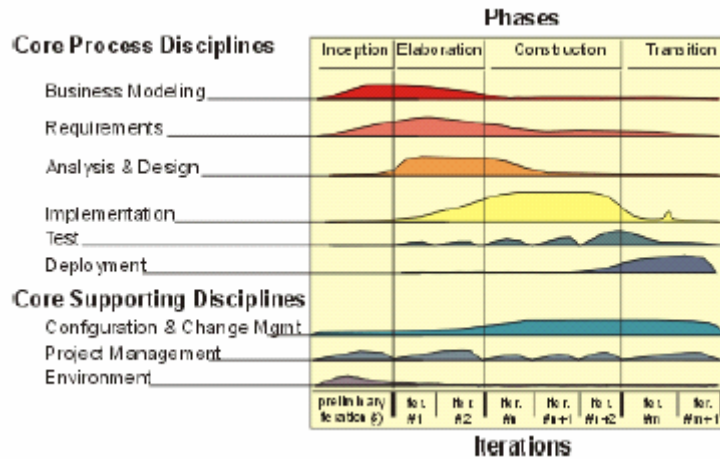


Figura 4.2 Estructura de procesos del RUP

En la figura 4.2 el eje horizontal representa la línea del tiempo para un ciclo de vida de un proyecto, mientras que en el eje vertical se muestran las disciplinas principales del proyecto que agrupa actividades de ingeniería de software que se ejecutan simultáneamente.

La altura de un área indica cuanto trabajo es gastado en una disciplina en un punto del tiempo.

Cada una de las disciplinas mostradas en la estructura de procesos del RUP abarca un conjunto de actividades.

El RUP contiene actividades, artefactos, guías y ejemplos necesarios para modificar y configurarlo para la organización que lo quiere adoptar.

Documentos y artefactos

Una actividad es una pieza de trabajo que puede ser ejecutada por un rol. La granularidad de una actividad es generalmente de unas pocas horas. La actividad es expresada en términos de crear o actualizar artefactos.

Un artefacto es una pieza de información o entidad física que es producida. Los artefactos son usados como entradas por los trabajadores para realizar actividades y a su vez son el resultado de esas actividades.

Cada disciplina tiene su propio conjunto de artefactos. RUP define cerca de 30 artefactos de alto nivel.

Cada rol es responsable por cada uno de los artefactos de sus disciplinas.

En general, los artefactos en RUP no son documentos en papel. En su lugar, los artefactos son contenidos dentro de la herramienta que es usada para crearlos, mantenerlos y ayudar al desarrollador a progresar con el producto.

Los artefactos generalmente tienen asociados guías y checkpoints, los cuales representan información sobre como desarrollar, evaluar y usar los artefactos. También los templates acompañan a los artefactos.

Roles

Se definen los roles necesarios para completar las disciplinas. Los roles tienen la responsabilidad por determinadas actividades y sus artefactos asociados. Un rol en RUP puede ser realizado en forma individual o grupal. De la misma forma, un individuo o grupo puede desempeñarse en varios roles.

Ejemplos de roles recomendados pueden ser:

- Analista
- Diseñador
- Implementador
- Reviewer
- Test designer
- Tester
- Integrador
- Project manager
- Technical writer
- Arquitecto de software
- Diseñador de interfaces de usuario

Procesos

RUP esta basado en las siguientes mejores practicas:

- Desarrollar software iterativamente
- Administrar requerimientos
- Usar arquitecturas basadas en componentes
- Visualización del modelo de software
- Verificación continua del software
- Control de cambios de software

En RUP el trabajo es dividido en 9 workflows (ver figura 4.3), y se corresponden con los mostrados en el eje vertical de la figura antes mostrada:

- Business modeling: Describe los procesos del negocio de la compañía para los cuales el software es desarrollado.
- Administración de requerimientos: Elicitación, organización y documentación.
- Análisis y diseño: Creación de la arquitectura y diseño del sistema de software.
- Implementación
- Test
- Desarrollo: Packaging del software, creación de scripts de instalación, escribir documentación para el usuario final, y otras tareas necesarias para entregar el software a los clientes.
- Project Management
- Management de la configuración y los cambios
- Environment

Cada ciclo del producto es dividido en cuatro fases; cada fase puede correr en una o mas iteraciones (en el grafico, son Elab#1, Elab#2,). Cada iteración se construye sobre los resultados de las iteraciones previas.

Las fases son:

- **Concepción:** Define los objetivos del proyecto, incluyendo el business case.
- **Elaboración:** Creación y validación de la arquitectura del sistema de software, captura de los requerimientos más importantes, y planeamiento y estimación del resto del proyecto. Los casos de uso desarrollados en la concepción son explotados aquí.
- **Construcción:** Implementación del sistema basado en la arquitectura creada en la fase anterior.
- **Transición:** Testeo beta del sistema con algunos clientes y preparación del release.



Figura 4.3 Flujos de trabajo de RUP

Nota: Contenido desarrollado en base a las referencias: 36, 38, 39

4.4.1.2 Otros marcos de proceso pesados

- OPF (OPEN Process Framework)

4.4.2 Marcos de proceso livianos

Las organizaciones o proyectos donde es más frecuente aplicar los marcos de proceso livianos se caracterizan por tener:

- Baja criticidad
- Alto cambio en los requerimientos
- Reducido número de desarrolladores
- Una cultura organizacional que es propensa al caos

4.4.2.1 Ejemplos de marcos de proceso livianos

4.4.2.1.1 XP (Extreme Programming)

Descripción

Es una metodología centrada en potenciar las relaciones interpersonales como clave para el éxito en el desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en:

- realimentación continua entre el cliente y el equipo de desarrollo
- comunicación fluida entre todos los participantes
- simplicidad en las soluciones implementadas
- coraje para enfrentar los cambios

XP se define como especialmente adecuada para proyectos con requerimientos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico.

Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre.

Documentos y artefactos

Con XP las historias de usuarios y otras prácticas reemplazan la necesidad de artefactos. Por esa razón XP parece ser inmune a la criticidad de ser muy pesado en artefactos.

La mayoría de la bibliografía que trata XP no menciona los términos artefactos y workproducts. Sin embargo, muchas de las referencias son artefactos. Se pueden contar más de 30, entre los que figuran:

- Historias
- Restricciones
- El diseño (CRC, UML)
- Estándares de codificación
- Releases
- Plan de iteración
- Defectos (e información asociada)
- Testeos
- Documentación de soporte
- Etc.

Roles

- **Programador.** El programador escribe las pruebas unitarias y produce el código del sistema.
- **Cliente.** Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
- **Encargado de pruebas (Tester).** Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- **Encargado de seguimiento (Tracker).** Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- **Entrenador (Coach).** Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- **Consultor.** Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto en el que puedan surgir problemas.
- **Gestor (Big boss).** Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación.

Procesos

A grandes rasgos el ciclo de desarrollo consiste en los siguientes pasos:

1. El cliente define el valor de negocio a implementar.
2. El programador estima el esfuerzo necesario para su implementación.
3. El cliente selecciona qué construir de acuerdo con sus prioridades y las restricciones de tiempo.
4. El programador construye ese valor de negocio.
5. Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. No se debe presionar al programador a realizar más trabajo que el estimado, ya que se perderá calidad en el software o no se cumplirán los plazos. De la misma forma el cliente tiene la obligación de manejar el ámbito de entrega del producto, para asegurarse que el sistema tenga el mayor valor de negocio posible con cada iteración.

El ciclo de vida ideal de XP consiste de seis fases:

- Exploración
- Planificación de la Entrega (*Release*)
- Iteraciones
- Producción
- Mantenimiento
- Muerte del Proyecto.

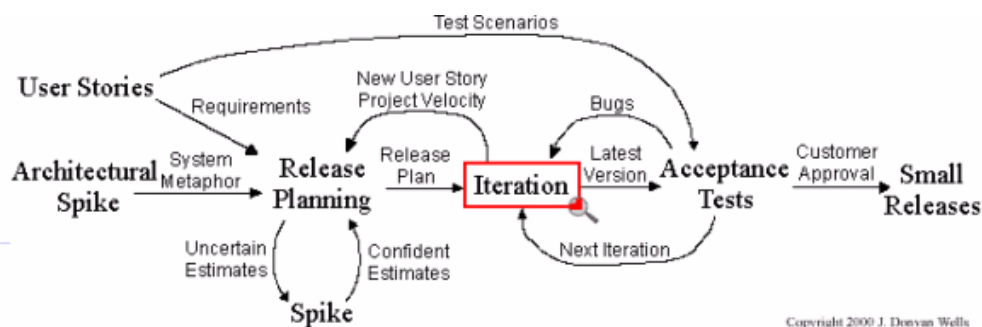


Figura 4.4 Formato de un proyecto XP

Prácticas

La principal hipótesis que se realiza en XP es la posibilidad de disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto, lo suficiente para que el diseño evolutivo funcione. Esto se consigue gracias a las tecnologías disponibles para ayudar en el desarrollo de software y a la aplicación disciplinada de las siguientes prácticas:

- **El juego de la planificación:** Hay una comunicación frecuente entre el cliente y los programadores. El equipo técnico realiza una estimación del esfuerzo requerido para la implementación de las historias de usuario y los clientes deciden sobre el ámbito y tiempo de las entregas y de cada iteración.
- **Entregas pequeñas:** Producir rápidamente versiones del sistema que sean operativas, aunque no cuenten con toda la funcionalidad del sistema. Esta versión ya constituye un resultado de valor para el negocio. Una entrega no debería tardar más 3 meses.
- **Metáfora:** El sistema es definido mediante una metáfora o un conjunto de metáforas compartidas por el cliente y el equipo de desarrollo. Una metáfora es una historia compartida que describe cómo debería funcionar el sistema.
- **Diseño simple:** Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- **Pruebas:** La producción de código está dirigida por las pruebas unitarias. Éstas son establecidas por el cliente antes de escribirse el código y son ejecutadas constantemente ante cada modificación del sistema.
- **Refactorización (Refactoring):** Es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios. Se mejora la estructura interna del código sin alterar su comportamiento externo.
- **Programación en parejas:** Toda la producción de código debe realizarse con trabajo en parejas de programadores. Esto conlleva ventajas implícitas (menor tasa de errores, mejor diseño, mayor satisfacción de los programadores, ...).
- **Propiedad colectiva del código:** Cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- **Integración continua:** Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.
- **40 horas por semana:** Se debe trabajar un máximo de 40 horas por semana. No se trabajan horas extras en dos semanas seguidas. Si esto ocurre, probablemente está ocurriendo un problema que debe corregirse. El trabajo extra desmotiva al equipo.
- **Cliente in-situ:** El cliente tiene que estar presente y disponible todo el tiempo para el equipo. Este es uno de los principales factores de éxito del proyecto XP. El cliente conduce constantemente el trabajo hacia lo que aportará mayor valor de negocio y los programadores pueden resolver de manera inmediata cualquier duda asociada. La comunicación oral es más efectiva que la escrita.
- **Estándares de programación:** XP enfatiza que la comunicación de los programadores es a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación para mantener el código legible.

Nota: Contenido desarrollado en base a las referencias:37, 40, 41

4.4.2.1.2 FDD (Feature Driven Development)

Descripción

FDD no hace énfasis en la obtención de los requerimientos sino en como se realizan las fases de diseño y construcción. Sin embargo, fue diseñado para trabajar con otras actividades de desarrollo de software y no requiere la utilización de ningún modelo de proceso específico. Además, hace énfasis en aspectos de calidad durante todo el proceso e incluye un monitoreo permanente del avance del proyecto. Comparada con las demás metodologías, FDD afirma ser ideal para el desarrollo de sistemas críticos.

FDD esta pensado para proyectos con tiempo de desarrollo relativamente cortos (menos de un año). Se basa en un proceso iterativo con iteraciones cortas (aproximadamente de dos semanas) que producen un software funcional que el cliente y la dirección de la empresa pueden ver y monitorear. A su vez, esto implica que los desarrolladores tendrán nuevas actividades que realizar en dicho período de tiempo. Esto hace que la motivación del equipo se mantenga durante todo el proyecto debido a que se ven los resultados periódicamente.

Documentos y artefactos

- Diagramas de clases (a distintos niveles de abstracción)
- Lista de características de alto nivel
- Documentación de entendimiento del dominio
- Lista de características priorizadas
- Plan inicial
- Modelos detallado de diseño
- Diagramas de secuencias
- Código

Roles

FDD clasifica sus roles en las siguientes tres categorías:

- **Key roles**
 - **Project Manager:** Es el líder administrativo y financiero del proyecto. Una de sus tareas principales es proteger al equipo de distracciones externas y permitir que el equipo de pueda trabajar en las condiciones apropiadas. En el FDD el Project Manager tiene la última palabra sobre temas referidos al alcance, tiempo y personal.
 - **Chief Architect:** Es el responsable por el diseño global del sistema y de la ejecución de todas las etapas del diseño. También tiene la última palabra sobre las decisiones del diseño de todo el sistema. Si es necesario, este rol puede ser dividido en dos roles como ser Domain Architect y Technical Architect.
 - **Development Manager:** Lleva diariamente las actividades de desarrollo y resuelve cualquier conflicto que pueda ocurrir con el equipo. Además, este rol incluye la responsabilidad de resolver problemas referentes a los recursos. Las tareas de este rol pueden ser combinadas con las del Chief Architect o el Project Manager.
 - **Chief Programmer:** Es un desarrollador con experiencia, el cual participa en el análisis de los requerimientos y el diseño del proyecto. El Chief Programmer es el responsable de guiar a pequeños equipos en el análisis, diseño y desarrollo de nuevas funcionalidades. Además, selecciona las funcionalidades a desarrollar de la lista de funcionalidades de la próxima iteración en la última fase del FDD, identifica las clases y el Class Owner que se necesita en el equipo para la iteración. Con la ayuda de otros Chief Programmers resuelve problemas técnicos y de recursos, y reporta el progreso del equipo durante la semana.
 - **Class Owner:** Trabaja bajo la guía del Chief Programmer en las tareas de diseño, codificación, testing y documentación. Es responsable del desarrollo de las clases que se le asignaron como propias. Para cada iteración los Class Owner participan en la decisión de que clase será incluida en la lista de funcionalidades de la próxima iteración.
 - **Expertos del dominio:** Los expertos del dominio pueden ser un usuario, un cliente, un esponsor, un analista del negocio o una mezcla de estos. Su tarea es poseer el conocimiento de los diferentes requerimientos del sistema. El experto del dominio pasa el conocimiento a los desarrolladores de manera tal que asegure que estos entreguen un sistema completo.

- **Domain Manager:** Lidera al grupo de expertos del dominio y resuelve sus diferencias de opinión concernientes a los requerimientos del sistema.
- **Supporting roles**
 - **Release Manager:** Controla el avance del proceso mediante la revisión de los reportes del Chief Programmer y realiza pequeñas reuniones con él. Además, reporta el progreso del proyecto al gerente.
 - **Language Lawyer/Language Guru:** Es un miembro del equipo responsable de poseer un vasto conocimiento, por ejemplo, un lenguaje específico de programación o tecnología. Este rol es particularmente importante cuando el equipo trabaja con una nueva tecnología.
 - **Build Engineer:** Es la persona responsable de preparar, mantener y correr el proceso de construcción, incluidas las tareas de mantenimiento de las versiones y la publicación de la documentación.
 - **Toolsmith:** Es un rol para la construcción de herramientas específicas para el desarrollo, conversión de datos y testeo. Además, puede trabajar en la preparación y mantenimiento tanto de bases de datos o sitios web destinados al proyecto.
 - **System Administrador:** La tarea del administrador del sistema es configurar, administrar y reparar los servidores, estaciones de trabajo y equipos de desarrollo y testeo utilizados por el equipo.
- **Additional roles**
 - **Tester:** Verifica que el sistema recién creado cumpla con los requerimientos del cliente. Puede llegar a ser una persona independiente del equipo del proyecto.
 - **Deployer:** Es el encargado de convertir la información existente requerida por el nuevo sistema y participa en el lanzamiento de los nuevos productos.
 - **Technical Writer:** Se encarga de preparar la documentación para los usuarios, quienes pueden formar parte o no del equipo del proyecto.

Procesos

El proceso consiste de cinco fases secuenciales durante las cuales el diseño y la construcción del sistema se llevan a cabo (ver figura 4.5):

- **Develop an Overall Model:** Cuando esta fase comienza, los expertos del dominio ya tienen una idea del contexto y los requerimientos del sistema. Es probable que el documento de especificación de requerimientos ya exista. Sin embargo, FDD no hace énfasis en la recolección y administración de los requerimientos. Los expertos del dominio presentan un informe llamado walkthrough en el cual los miembros del equipo y el Chief Architect son informados a través de una descripción de alto nivel del sistema.
El dominio global es dividido en diferentes áreas y se realiza un walkthrough detallado para cada una de dichas áreas por parte de los expertos del dominio. Luego de cada walkthrough, un grupo de desarrolladores realizan un modelo de objetos para el área del dominio. Además, el equipo de desarrollo discute y decide cual es el modelo de objetos más apropiado para cada área del dominio. Simultáneamente, el modelo global del sistema es construido.
- **Build a Features List:** Los walkthroughs, el modelo de objetos y los requerimientos existentes ofrecen una buena base para construir una features list que resuma la funcionalidad del sistema a ser desarrollado. En dicha lista, el equipo de desarrolladores presenta cada una de las funcionalidades evaluadas por el cliente. Las funcionalidades son presentadas por cada área del dominio y éstas forman un Major List Sets. Dicha lista es dividida en subconjuntos en base a la funcionalidad. Estas representan diferentes actividades con un área específica del dominio. La features list es revisada por los usuarios y esponsors del sistema para su validación y aprobación.
- **Plan by Feature:** En esta etapa se incluye la creación de un plan de alto nivel, en el cual la features list es ordenada en base a la prioridad y a la dependencia entre cada feature. Además, las clases identificadas en la primera etapa son asignadas a cada programador.
- **Design and Build by Feature:** Un conjunto de features son seleccionadas de la features list. El diseño y construcción de la funcionalidad es un proceso iterativo durante el cual las features seleccionadas son producidas. Una iteración puede llevar desde unos pocos días a un máximo de dos semanas. Este proceso iterativo incluye tareas como inspección del diseño, codificación, testing unitario, integración e inspección del código. Luego que la iteración llega a su fin se realiza un main build de la funcionalidad en el cual se integra la funcionalidad. Dicho main build se realiza mientras la siguiente iteración comienza.

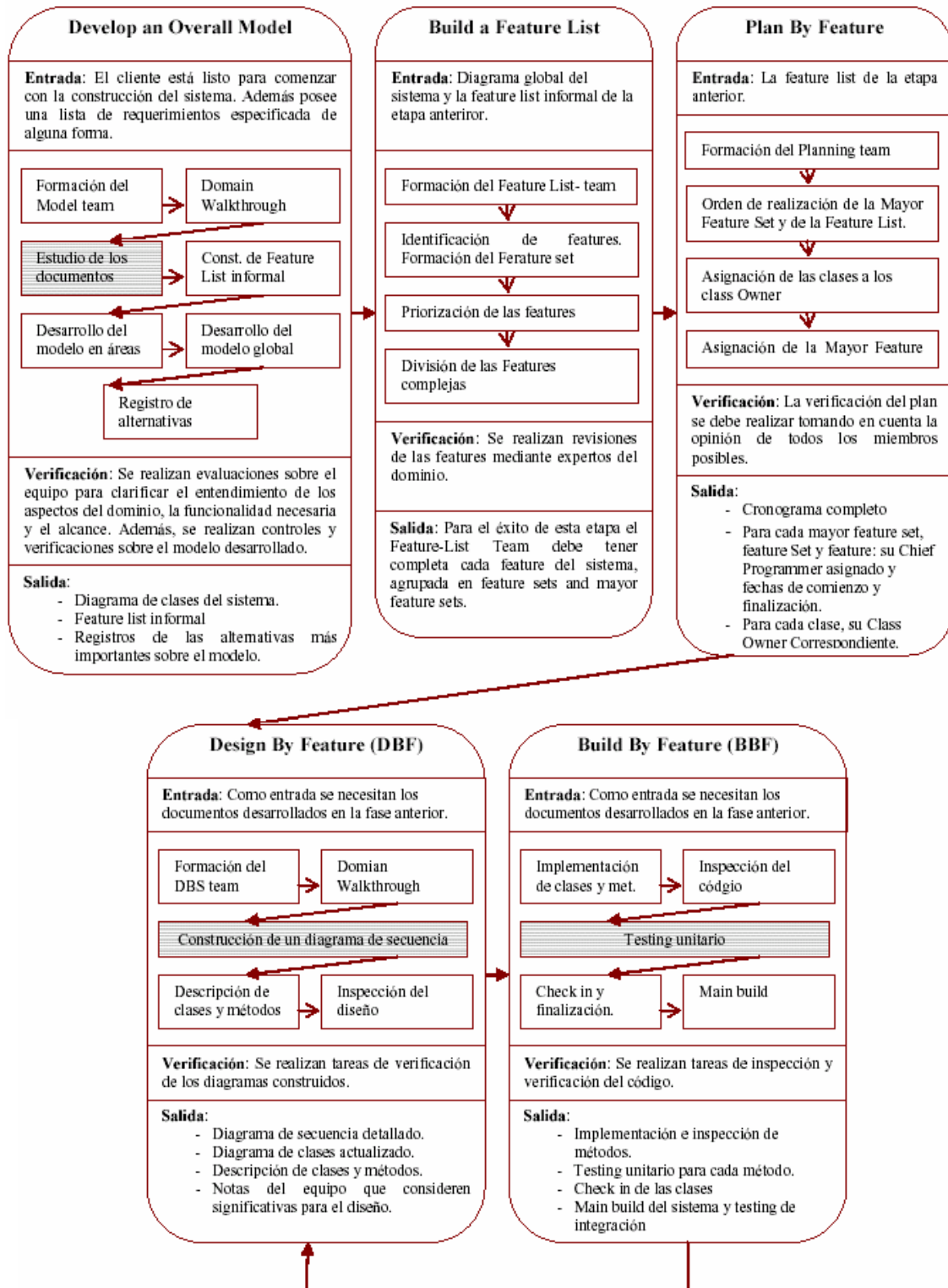


Figura 4.5 Procesos del FDD

Nota: Contenido desarrollado en base a las referencias: 42, 43, 94

4.4.2.2 **Otros marcos de proceso livianos**

- AM - Agile Modeling
- ASD - Adaptive Software Development
- Crystal Clear
- DSDM - Dynamic Systems Development Method
- Lean Software Development
- Scrum
- TDD - Test-Driven Design
- XBreed
- Microsoft Solutions Framework (MSF)

4.4.3 **Software Product Line Framework**

Descripción

Es un conjunto de sistemas de software que comparten un conjunto de características comunes y gerenciales que satisfacen las necesidades de un segmento de mercado o misión, y que son desarrollados a partir de un conjunto de activos claves en forma preestablecida. Esto último tiene por consecuencia economizar la producción.

Los activos principales son artefactos reusables que constituyen la base del software product line.

La forma de lograr esta economía se debe a que cada producto esta formado por componentes que son considerados activos comunes. Estos activos, pueden ser customizados (parametrización, herencia). Estos componentes se ensamblan junto con otros similares de acuerdo a una amplia arquitectura de línea de producto. Entonces, construir un nuevo producto es mas un problema de arquitectura (integración) mas que de creación (programación).

Una vez que se establece el repositorio de activos principales, cada vez que se construye un nuevo producto se logran ahorros en base a:

- Requerimientos
- Arquitectura
- Componentes
- Modelo y análisis
- Testing
- Planning
- Procesos
- Personas

Procesos

La esencia de un product line involucra el desarrollo de activos principales y el desarrollo de productos utilizando estos activos.

Los nuevos productos pueden ser construidos a partir de activos principales, o los activos principales pueden ser extraídos de los productos. Esto se realiza en el marco de las 3 actividades principales (ver figura 4.6):

- Desarrollo de activos principales
- Desarrollo de productos
- Management

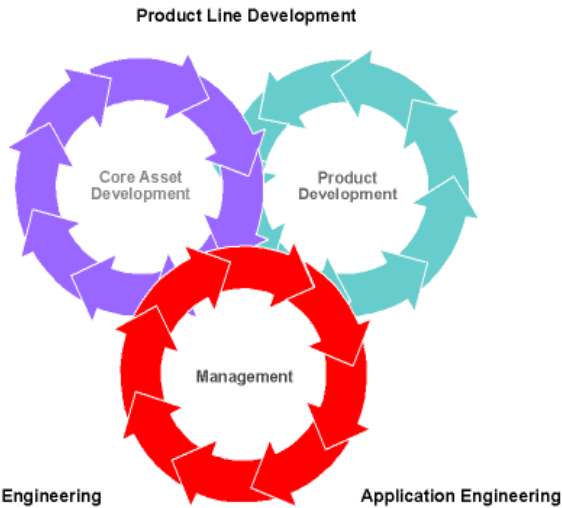


Figura 4.6 Las 3 actividades elementales del SPL

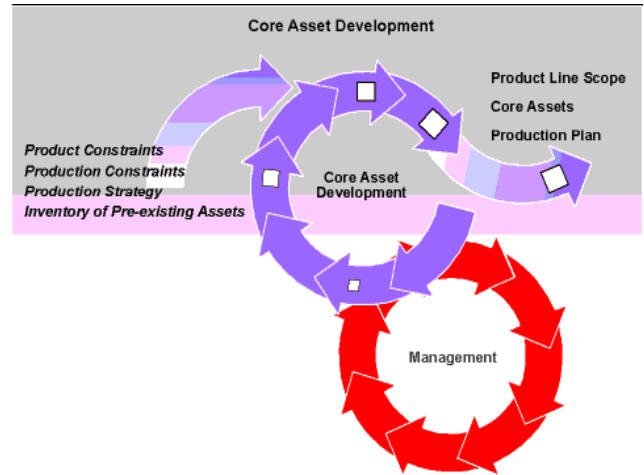


Figura 4.7 Desarrollo de activos base para SPL

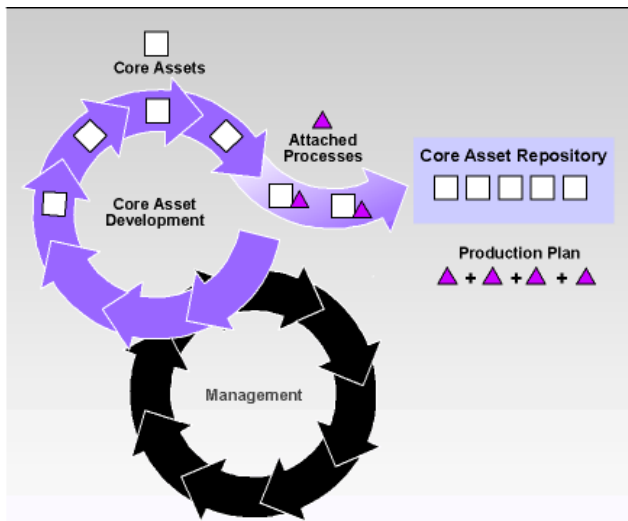


Figura 4.8 Procesos adjuntos del SPL

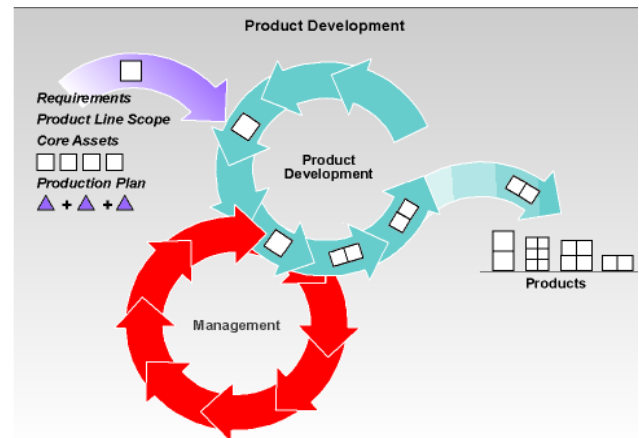


Figura 4.9 Desarrollo de productos en SPL

Cada rotación del círculo representa una de las actividades principales.

Las flechas rotando indican que no solo los activos principales son usados para desarrollar productos; también se usan revisiones de activos principales existentes, o quizás los activos podrían evolucionar a partir del desarrollo de un producto.

Hay un fuerte ciclo de realimentación entre los activos principales y los productos (figuras 4.7, 4.8 y 4.9), ya que los activos principales son actualizados cuando los nuevos productos son desarrollados.

El uso de los activos es seguido, y los resultados son realimentados a la actividad de desarrollo. Como resultado, el activo principal se vuelve más genérico por considerarlo potencial para nuevos productos.

El gerenciamiento debe complementar el cambio cultural para poder ver a los nuevos productos en base a los activos principales disponibles: o bien los nuevos productos se deben alinear a los activos principales existentes, o bien los activos principales deben ser actualizados para reflejar los nuevos productos que están siendo comercializados.

Para poder realizar esas actividades, las mismas se organizan a través de áreas prácticas. Un área práctica es el cuerpo del trabajo o la colección de actividades que una organización debe dominar (realizarlo en forma repetida y exitosa) para llevar a cabo exitosamente el trabajo esencial de la product line.

Las áreas prácticas se organizan según las siguientes categorías:

- **Ingeniería de software:** Agrupa las áreas prácticas que son necesarias para aplicar la tecnología apropiada para crear o evolucionar tanto productos como activos principales.

Sus áreas prácticas son:

- Definición de la arquitectura
- Evaluación de la arquitectura
- Desarrollo de componentes
- Utilización de COTS
- Extracción de activos existentes
- Ingeniería de requerimientos
- Integración de sistemas de software
- Testing
- Comprensión de dominios relevantes

- **Gerenciamiento técnico:** Agrupa las áreas prácticas de management necesarias para diseñar la creación y evolución de activos principales y productos.

Sus áreas prácticas son:

- Configuration Management
- Recolección de datos, métricas y seguimientos
- Análisis de hacer, comprar, extraer o delegar
- Definición de procesos
- Scoping
- Planeamiento técnico
- Gerenciamiento de riesgos técnicos
- Soporte a la herramienta

- **Gerenciamiento organizacional:** Agrupa las áreas prácticas necesarias para la instrumentación de la totalidad del esfuerzo de la product line.

Sus áreas prácticas son:

- Construcción del Business case
- Gerenciamiento de la interfase de usuario
- Desarrollo de una estrategia de adquisición
- Financiación
- Lanzamiento e institucionalización
- Análisis de mercado
- Operaciones
- Planeamiento operacional
- Gerenciamiento del riesgo organizacional
- Estructurar la organización
- Predicción tecnológica

Beneficios

- Obtener productividad a gran escala
- Reducir el time to market
- Incrementar la calidad del producto
- Incrementar la satisfacción del cliente
- Uso eficiente de los recursos humanos
- Capacidad para mantener crecimientos sin precedente
- Capacidad para mantener presencia en el mercado.

Nota: *Contenido desarrollado en base a la referencia: 95*

4.5 **Notas y conclusiones**

No es la intención de esta tarea identificar o imponer un marco de proceso en particular; pero sí es objetivo del proceso de implementación (y particularmente de esta tarea) notificar la necesidad de la identificación / selección del marco de proceso como un paso previo.

Capítulo 5

Tarea 3

Assessment

5.1 Identificación

Tarea 1

5.2 Introducción al capítulo

El objetivo que presenta esta tarea es el de realizar un assessment del estado de los procesos vinculados con el desarrollo de software en algún punto de la implementación para una organización.

Siguiendo con la aproximación del proceso de mejoras IDEAL, esta tarea está directamente afectada a la etapa de diagnóstico; esta inspirada en el método CBA IPI versión 1.2, que es una herramienta de diagnóstico desarrollada por el SEI que se utiliza para generar conclusiones y clasificaciones relativas a las KPA de SW-CMM respecto del entorno de assessment. Las conclusiones son generadas de acuerdo a los datos recolectados a partir de cuestionarios, revisión de documentos, presentaciones y entrevistas con Srs. Managers, líderes de proyectos y profesionales de software.

El valor agregado y diferencias principales que hay entre esta tarea y método CBA IPI, son:

- CBA IPI está pensado para ser utilizado en grandes organizaciones, sin embargo, esta tarea puede ser aplicada a cualquier tipo de organización.
- La tarea es para realizar el assessment sin limitarse únicamente al SW-CMM como modelo de referencia. Como complemento, se consideran issues no contemplados por el modelo de referencia que dan un valor agregado a la hora de confeccionar el plan de acción y posterior confección del business case.
- Se brinda un amplio conjunto de templates y ejemplos que funcionan como herramientas complementarias.

Los objetivos principales de esta tarea son:

- Soportar, permitir y fomentar el compromiso de la organización para la mejora de los procesos de software
- Proveer una detallada foto de las debilidades y fortalezas del actual proceso de software de la organización, identificando también las áreas de proceso a ser mejoradas.

Los objetivos antes nombrados se logran en base a que el equipo de assessment debe:

- Proveer a la organización con la información necesaria para construir baselines de la capacidad de software de la organización.
- Identificar los principales issues que no contemplados por el modelo de referencia que tengan impacto en los esfuerzos para mejorar el proceso de software.
- Identificar fortalezas y debilidades usando el modelo de referencia.

El método se basa en los siguientes principios:

- Utilizar el modelo de referencia como modelo de procesos.
- Basarse en un proceso de assessment formal
- Involucrar a los seniors managers como sponsors del assessment.
- Basar el assessment en las necesidades, objetivos, tipo de software y marco de proceso.

La salida de este proceso, puede ser utilizada (entre otras cosas) para establecer planes de acción sobre el proceso de software.

Un requisito importante vinculado con esta tarea es la confidencialidad. La información relevada, procesada y desarrollada en esta tarea debe ser mantenida bajo estricta confidencialidad, ya sea fuera de la organización, como así también entre proyectos.

Esta tarea se ejecuta una primera vez como requisito previo a la tarea de elaboración del plan de acción, proporcionando las entradas necesarias para la confección del plan.

En este punto, el assessment se caracteriza generalmente por la poca o nula utilización de métricas, ya que la organización está probablemente en los niveles más bajos careciendo de cualquier tipo de métricas o haciéndolo en forma inconsistente.

Luego de ejecutar la tarea de assessment por primera vez se obtendría en forma incremental un grado de agilidad

que beneficiaría a las posteriores tareas de assessment dentro de la implementación ya que se establecen las bases para que los posteriores assessments lo utilicen como referencia.

Nota: El contenido del presente capítulo fue desarrollado en base investigaciones hechas en las referencias: 46, 47, 48, 49, 50, 51, 52, 58, 60

5.3 Estructura de la tarea

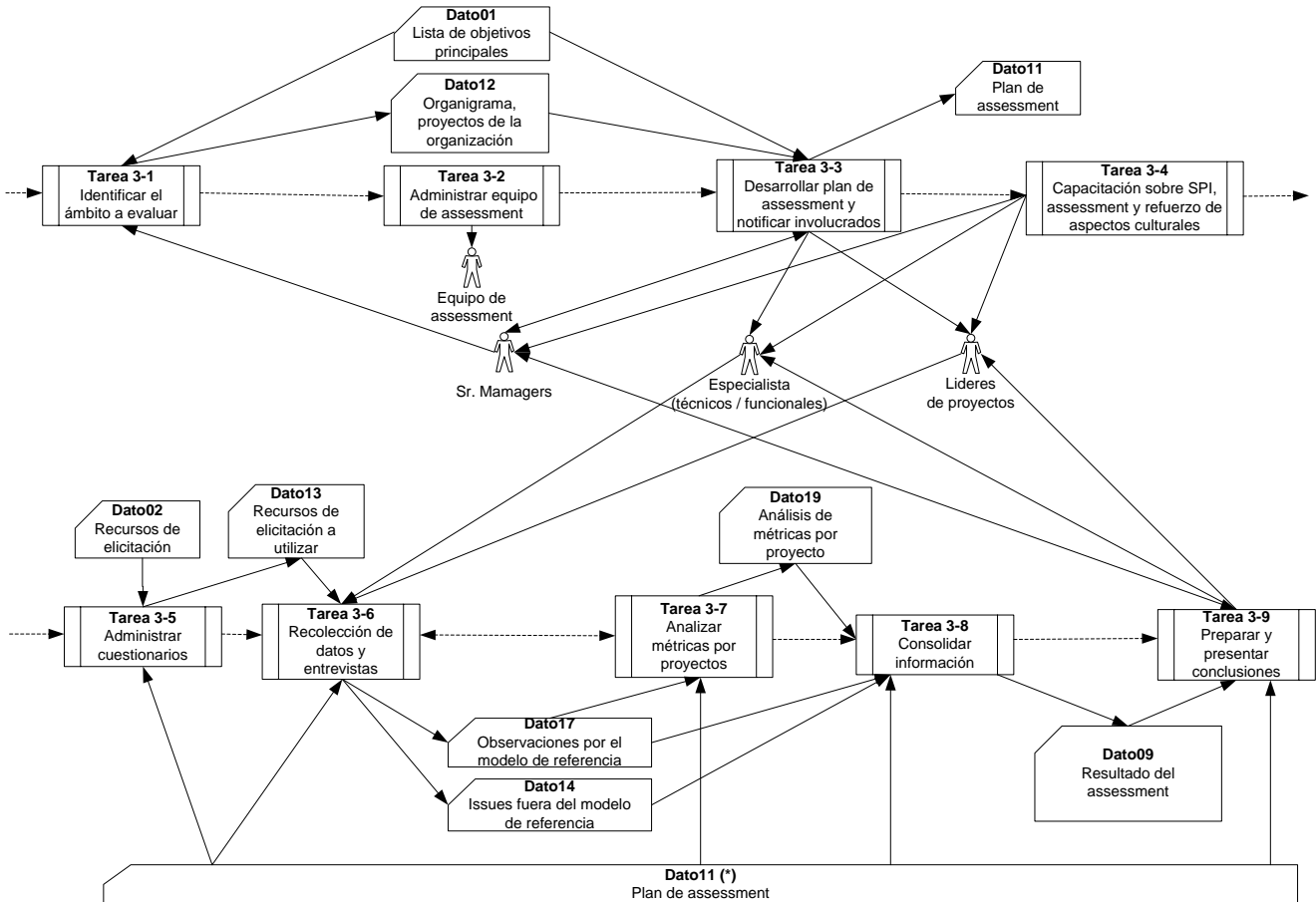


Figura 5.1 Estructura tarea 3

5.3.1 Tarea 3-1: Identificar el ámbito a evaluar

Propósito

Desarrollar un entendimiento común entre el líder del assessment y el Sr. Manager (el esponsor) respecto de los objetivos a evaluar (tarea 1), alcance, restricciones, roles, responsabilidades y salidas a producir. La idea detrás de esta tarea es ganar compromiso para realizar el assessment.

Con la colaboración del Sr. Manager se confecciona el organigrama descompuesto en proyectos de la organización.

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato01: Lista de objetivos principales • Entrevistas con Sr. Manager 	<p>TASK: Tarea 3-1</p> <p>Identificar el ámbito a evaluar</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato12: Organigrama, proyectos de la organización • Compromiso de la alta gerencia con el proceso de assessment
	<p>VERIFY</p> <p>Validar Organigrama y proyectos de la organización relevados con Sr. Manager</p>	

5.3.2 Tarea 3-2: Administrar equipo de assessment

Propósito

El equipo de assessment deberá estar compuesto por un número variable de integrantes que varía en función del tipo de organización y proyectos a evaluar. Sin embargo se recomienda que su número no varíe entre 4 y 10 participantes.

<p>ENTRY</p> <ul style="list-style-type: none"> Tarea 3-1 cumplida 	<p>TASK: Tarea 3-2</p> <p>Administrar equipo de assessment</p>	<p>EXIT</p> <ul style="list-style-type: none"> Equipo de assessment constituido
	<p>VERIFY</p> <p>Disponibilidad de equipo de assessment capacitado.</p>	

Composición del equipo de assessment

- Líder de assessment:** Persona que lidera la evaluación y que posiblemente sea personal del SEPG o personal externo a la organización. Debe tener amplia experiencia en el desarrollo de software, experiencia y conocimiento en el modelo de referencia, y entrenamiento y experiencia en assessment.
- Integrantes de la organización a realizarle el assessment:** Se participan a integrantes de la organización que aporten conocimiento del dominio de aplicación de la organización. Deben cumplir con los requisitos del equipo de assessment.

Requisitos para componer el equipo de assessment

- Ser dirigido por un líder de assessment. El papel del líder es de vital importancia.
- Todos los integrantes deben poseer conocimientos profundos del modelo de referencia. Este conocimiento incluye la habilidad de explicar las áreas de proceso y sus propósitos, y proveer ejemplos relevantes de la entidad evaluada.
- Conocimientos del proceso de assessment.
- Conocimiento de conceptos de mejora de procesos de software.
- Experiencia en ingeniería de software.
- Experiencia en gerenciamiento.
- Experiencia en las fases del ciclo de vida y actividades funcionales del dominio de aplicación.

Atribuciones / obligaciones del líder de assessment:

- Tiene la responsabilidad total por el assessment.
- Asignar roles a los miembros del equipo.
- Asegurar que el plan y normas establecidas sean cumplidas.
- Monitorear la performance del equipo.
- Resolver conflictos.

Atribuciones / obligaciones de los miembros del equipo de assessment:

- Asumir uno o más roles, responsabilidades por documentación o seguimiento respecto de una o más áreas de proceso en las que se especializa.

Puntos a tener en cuenta para esta tarea

- Delinear las reglas de procedimiento a utilizar por el equipo durante el assessment
- Acordar los detalles del assessment
- Notificar a los participantes del equipo de assessment de su participación

5.3.3 Tarea 3-3: Desarrollar plan de assessment y notificar

Propósito

Producir y conseguir la aprobación del plan de assessment. Esto implica relevar cierta información con el fin de realizar las estimaciones de tiempo, costo, recursos y disponibilidad de las personas involucradas en las entrevistas.

<p>ENTRY</p> <ul style="list-style-type: none"> • Tarea 3-2 cumplida (equipo de assessment constituido) • Dato12: Organigrama, proyectos de la organización • Dato01: Lista de objetivos principales 	<p>TASK: Tarea 3-3</p> <p>Desarrollar plan de assessment y notificar involucrados</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato11: Plan de assessment
	<p>VERIFY</p> <p>Aprobación del plan de assessment por Sr. Managers (esponsores)</p>	

Actividades vinculadas a relevar información sobre estimaciones y disponibilidad

- 1- Seleccionar cuales son los proyectos sobre los cuales se va a realizar el assessment.
Los proyectos deben ser elegidos de forma que representen a la organización que esta siendo evaluada.
Los factores que deben ser considerados en la selección de proyectos para que representen a la organización son:
 - Dominio de aplicación
 - Tecnología empleada
 - Alcance, en función de tamaño del producto, tamaño del equipo, fases del ciclo de vida, y duración
 - Impacto del proyecto en el negocio respecto de ganancias, costos y valor estratégico.
- 2- Selección de personal de la organización que participara del assessment, confirmación de participación y disponibilidad del mismo.
Se deberá seleccionar a personal que sea clave en la parte técnica y a funcionales altamente calificados con alta participación en los proyectos elegidos. Dentro de esta categoría están:
 - Srs. Managers
 - Líderes de proyectos
 - Especialistas técnicos y funcionales

La selección de los mismos respecto de su participación debe ser balanceada entre el valor que pueda aportar en las entrevistas versus la interrupción de sus tareas cotidianas en la organización.
Particularmente para la selección de los representantes de áreas funcionales se recomienda que los mismos sean especialistas, no gerentes ni staff.
- 3- Confirmar los detalles de logística en la organización.
- 4- Confirmar la disponibilidad de documentación a relevar en tiempo y forma.
- 5- Estimación de tiempos, recursos y costos vinculados a entrevistas, revisión, análisis de estándares de productividad y calidad.
- 6- Identificación de riesgos y restricciones que pueden afectar al plan de assessment.
- 7- Obtener aprobación del esponsor.

Los puntos 3, 4, 5 y 6 deben ser considerados puntualmente para la organización a evaluar, y el resultado de los mismos deberá estar reflejado en el plan de assessment, detallado en el anexo Dato12.

5.3.4 Tarea 3-4: Capacitación SPI, assessment y refuerzo de aspectos culturales

Propósito

Capacitar a los participantes del assessment sobre:

- Conceptos relativos a mejora de procesos.
- Propósito y objetivos de la implementación del framework para la mejora de procesos.
- Tarea de assessment:
 - Alcance
 - Participación
 - Responsabilidades
 - Aporte individual
 - Aporte grupal
 - Aporte a la organización
 - Objetivos
 - Pasos a seguir

<p>ENTRY</p> <ul style="list-style-type: none"> • Participantes del assessment de todos los niveles de la organización notificados 	<p>TASK: Tarea 3-4</p> <p>Capacitación SPI, assessment y refuerzo de aspectos culturales</p>	<p>EXIT</p> <p>Conocimientos impartidos a integrantes sobre:</p> <ul style="list-style-type: none"> • SPI • Assessment • Aspectos culturales
	<p>VERIFY</p> <p>Mediante evaluaciones de los conocimientos impartidos a los alumnos, se valida el éxito de la tarea.</p>	

5.3.5 Tarea 3-5: Administrar cuestionarios

Propósito

Determinar cuales serán los cuestionarios a utilizar en las entrevistas basándose en un pool de recursos de elicitación, el cual es mantenido y enriquecido a través de las distintas implementaciones.

Los criterios a utilizar para la selección o customización de los tipos de cuestionarios, se basan en los siguientes criterios:

- 1) Objetivos que persigue la organización
- 2) Personal sobre el cual se realizara las entrevistas
- 3) Disponibilidad del personal
- 4) Otros criterios

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato02: Recursos de elicitación • Dato11: Plan de assessment 	<p>TASK: Tarea 3-5</p> <p>Administrar cuestionarios</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato13: Recursos de elicitación a utilizar
	<p>VERIFY</p> <p>Verificación de la documentación seleccionada por parte del líder de assessment</p>	

5.3.6 Tarea 3-6: Recolección de datos y entrevistas

Propósito

Revelar toda información vinculada al assessment de la organización tanto para las áreas de proceso del modelo de referencias como así también para información no incluida en el modelo que permita el desarrollo posterior del plan de trabajo poniendo énfasis en los objetivos perseguidos por la organización.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> • Dato13: Recursos de elicitación a utilizar • Dato11: Plan de assessment 	<p>TASK: Tarea 3-6</p> Recolección de datos y entrevistas	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> • Dato14: Issues fuera del modelo de referencia • Dato17: Observaciones por el modelo de referencia
	<p>VERIFY</p> Validación de información relevada	

Etapas de la tarea

Etap 1: Recolección de datos: La información recogida en esta etapa se clasifica en presentaciones, formularios y documentos que poseen los proyectos.

Se pretende identificar documentación y otros artefactos que demuestren el uso de prácticas institucionalizadas por la organización dentro del alcance definido por el assessment.

La idea es relevar información previa a la etapa de entrevistas, para minimizar el tiempo de participación de los entrevistados.

Cada documento analizado es localizado e identificado para su posterior utilización (si corresponde).

Ejemplos de la utilidad de la información relevada en esta etapa son:

- a. Poder formular en las entrevistas posteriores preguntas específicas respecto de cómo es realizado el trabajo.
- b. Confirmar la institucionalización de una práctica.
- c. Etc.

La recolección de datos es realizada por integrantes del equipo de assessment, preferentemente por individuos especializados en la KPA de la que sea desea relevar información.

Etap 2: Encuestas y entrevistas: Los Sr. Managers, Líderes de proyectos y especialistas participantes de los proyectos elegidos a evaluar son encuestados en base a las encuestas seleccionadas (por ejemplo, cuestionario de madurez).

Modo de conducir las entrevistas

Características especiales que hay que tener en cuenta:

- Un entrevistado puede cumplir varios roles
- De acuerdo al punto anterior, una misma persona será entrevistada varias veces

Por lo tanto es muy importante poner énfasis en que el entrevistado haga foco al proyecto y al rol para el cual se lo entrevista abstrayéndose de sus otros roles y/o participación en otros proyectos.

Entrevistadores

Las entrevistas son realizada por integrantes del equipo de assessment, preferentemente por individuos del equipo de assessment que son especialistas en las áreas de proceso de las que sea desea relevar información.

Entrevistados

Son aquellos integrantes de los proyectos que fueran elegidos oportunamente en la tarea 3.3.

Estrategia

Al comenzar las entrevistas se parte de la base que toda la información esta no confirmada, y a su vez, que no hay nada que sea sabido. La información ira pasando de un conjunto al otro a medida que se vaya entrevistando, probando y validando la información.

Si bien es dependiente del tipo de estructura de la organización, una posible estrategia de entrevistas puede ser una aproximación top-down, donde el orden de ir entrevistando es:

- Srs. Managers.
- Lideres de proyectos
- Individuos de los proyectos

La mayor parte del tiempo consumido en las entrevistas debería ser empleado con los especialistas técnicos y funcionales del proyecto ya que están en contacto permanente con los procesos implementados en la organización.

Cantidad de entrevistas

La cantidad de entrevistas a realizar variará dependiendo de la información que se va relevando, pero básicamente tenemos:

- **Primera entrevista:** Se refiere a temas generales a nivel organización o que apuntan a prácticas de áreas de proceso en forma general. Un conjunto de preguntas de ejemplo sobre esta clase de información a relevar son mostradas en el Anexo Dato02 - Complemento Preguntas entrevistas (ver ejemplo en figura 4.2).

Ejemplo de Cuestionario a utilizar en entrevistas								
Datos Entrevista								
Fecha:								
Id Entrevista:								
Entrevistador:								
Datos del encuestado								
Nombre:								
Legajo:								
Rol encuestado:								
Proyecto por el que se encuesta:								
Ejemplos de preguntas a nivel General								
Id	Audiencia			Pregunta	Aplicabilidad			Comentarios Soporte organizacional (políticas / procesos y procedimientos)
	Sr. Managers	Lideres Proy.	Staff		No aplica	Parcialmente	Totalmente	
G 01	X	X		Indicar si se tiene identificado algún nivel de madurez objetivo a alcanzar				
G 02	X	X	X	Describir gráficamente la organización a la que pertenece el encuestado, incluyendo entidades (sectores especiales) tales como SQA, CM, SEPG.				
G 03	X	X	X	Para cada una de las entidades incluidas en la respuesta a G02, indicar quienes son los integrantes de las mismas.				
G 04	X	X		Indicar cuales son los productos y proyectos afectados a desarrollo o mantenimiento				

Figura 4.2 Porción de cuestionario a utilizar en entrevistas

Las preguntas relacionadas al ámbito de las áreas de proceso del modelo de referencia, están dirigidas a un determinado tipo de entrevistado (Sr. Managers, Lideres de proyectos, o especialistas).

Las preguntas están agrupadas por área de proceso, y cada pregunta busca obtener la mayor cantidad de información.

Las preguntas contenidas en esta entrevista son preguntas exploratorias de final abierto que se caracterizan por:

- Brindar conocimiento sobre como las áreas de proceso están implementadas
- Determinar el modo en el que un proceso ha sido institucionalizado
- Identificar documentos a nivel implementación

- **Entrevistas sucesivas:** Son una serie de entrevistas que apuntan a buscar información faltante, contradicciones, validaciones, precisiones, etc. El foco de las entrevistas irá variando a medida que avancen las entrevistas. Las preguntas contenidas en este tipo de entrevistas contienen preguntas directas y se caracterizan por:
 - Validar la información relevada
 - Hacer foco en necesidades de información específica
 - Volver a entrevistar a individuos entrevistados anteriormente

Si quedan puntos con dudas o por verificar o corroborar, se procede a una nueva entrevista con quien corresponda. La nueva entrevista se arma en base a los resultados del análisis de las entrevistas previas.

Si no quedan más dudas o puntos por verificar o corroborar, se da por terminada esta serie de entrevistas.

Cardinalidad de entrevistados y entrevistadores

Si bien es más productivo entrevistar a grupos que a individuos puntuales, la cardinalidad variará en base al rol del entrevistado y la posibilidad que múltiples roles entrevistados al mismo tiempo generen distorsiones en las respuestas (por motivos de cadena de mando, liderazgo en el grupo, asignaciones de trabajo, etc.):

- Para la primera entrevista:
 - Los Srs. Managers y los líderes de proyecto son entrevistados individualmente por un entrevistador
 - Los especialistas (técnicos y funcionales) del proyecto son entrevistados en grupos por un entrevistador.
- Para las siguientes entrevistas:
 - Los Srs. Managers y los líderes de proyecto son entrevistados individualmente por varios entrevistadores
 - Los especialistas (técnicos y funcionales) del proyecto son entrevistados en grupos por varios entrevistadores.

Información que se desea relevar con las entrevistas

- 1) **Issues del modelo de referencia.** Estas observaciones están basadas en el modelo de referencia y son validadas y corroboradas.

Se pretende conocer como el trabajo es realizado en base a:

- 1-1) Conocer cuales prácticas aplican a nivel organización
- 1-2) Conocer cuales prácticas aplican a nivel proyecto
- 1-3) Procedimientos asociados (a nivel organización y a nivel proyecto)
- 1-4) Procesos asociados (a nivel organización y a nivel proyecto)
- 1-5) Políticas asociadas (a nivel organización y a nivel proyecto)
- 1-6) Baselines utilizados (a nivel organización y a nivel proyecto)

- 2) **Issues complementarios al modelo de referencia**

- 2-1) Costos x recurso. *Ver template en Anexo Dato 14*
- 2-2) Infraestructura tecnológica de la organización. *Ver template sobre información relevada en Anexo Dato 14*
- 2-3) Herramientas Case utilizadas en proyectos. Se releva los distintos tipos de herramientas usadas en los proyectos
 - Gestión de contenidos
 - Administración de requerimientos
 - Administración de configuración
 - Herramientas de planeamiento de proyectos
 - Herramientas de garantía de calidad
 - Seguimiento de problemas / errores

- Herramientas de diseño

En el anexo Complemento - Ejemplos de herramientas Case (ver ejemplo en figura 5.3), se brinda un listado de herramientas Case utilizadas en la industria agrupadas por actividad afectada.

Actividad relacionada	Herramienta	Fabricante
Bug / defect / problem tracking	+1CR	+1 Software Engineering
	Bugzilla	
	ClearDDTS	Rational Software Corporation
	ClearQuest	Rational
	Defect WorkFlow	SoftQuest Systems
	Digit	
	McCabe TRUEtrack	McCabe & Associates
	SilkRadar	Segue Software, Inc.
	TeamTrack	TeamShare, Inc.
	Test Management System	Rex Black Consulting Services, Inc.
Vantive Quality	Vantive Corporation	
Case Tools	Erwin	
	Poseidon for UML	Gentleware
	Rational Rose	
Comunicación	Software de emails / extranet / intranet / Messenger	
	Software para Teleconferencias	
	+1CM	+1 Software Engineering
	AccuRev/CM	AccuRev Inc.
	AllChange	Intasoft Ltd.
	Change and Configuration Control (CCC/Harvest)	Computer Associates
	ClearCase	Rational
	CM Synergy	Telelogic AB
	CMVision and Configuration Management Facility (CMF)	Expertware
	CMWin	Expertware, Inc.
	Configuration Management Version Control (CMVC)	IBM Corp.
	CONTROL-CM	

Figura 5.3 Porción de planilla de herramientas case

Nota: Ver template Anexo Dato 14 para registrar la información relevada.

- 2-4) Información del ámbito de RRHH. *Por su importancia, se dedica una sección especial a este tipo de información a relevar.*
- 2-5) Información del ámbito administrativo. *Ver template información relevada en Anexo Dato 14*
- 2-6) Aspectos culturales de la organización. *Por su importancia, se dedica una sección especial a este tipo de información a relevar.*

Información a relevar del ámbito de RRHH

Puede que el modelo de referencia no considere aspectos de recursos humanos, pero es de suma importancia incluirlos dentro del plan de mejoras de los procesos.

Esto se debe a que el plan de mejora servirá para mejorar los procesos, pero son las personas quienes utilizan los procesos para generar los productos. Si no se tiene una buena gestión de los RRHH, se tendrá una falla muy importante a nivel calidad dentro de la mejora continua a la que apunta la implementación.

Se reconoce y se pone especial énfasis en que no existe mejora de procesos que no sea acompañada de una mejora en los RRHH. Es por eso que se tiene en cuenta el relevamiento de aspectos de la gestión de RRHH.

Algunas de esas conclusiones serán derivadas al sector de RRHH para que planifique y ponga en práctica la implementación de prácticas que conduzcan a las conclusiones presentadas a través de un programa de mejoras que corra en paralelo con el de mejora de procesos.

Las demás serán incluidas en el plan de mejoras.

Principales problemas de la gestión de RRHH

- Existe en el mercado una alta rotación de profesionales
- Generalmente los ambientes de trabajo son poco amigables
- Existen problemas de comunicación
- No se aplican o se aplican ineficientemente las políticas de motivación

- Se excluye al sector de RRHH en de proyectos de mejora organizacional
- Se limita su funcionalidad netamente al control de personal
- Otras.

Principales responsabilidades que son del ámbito de RRHH

- Selección de personal
- Aspectos vinculados con entrenamiento / capacitación
- Aspectos de modelo de desempeño (plan de carrera y evaluación)
- Estrategias de compensación
- Ambiente de trabajo (Comodidad, privacidad, iluminación, mobiliario, salud y seguridad)
- Resolución de problemas del ámbito laboral
- Gestión por competencias: Conjunto de comportamientos que describen el nivel optimo de performance para un determinado rol, equipo, función u otro contexto de trabajo.
- Planificación del uso de personal
- Motivación del personal a través de sus necesidades
 - Social: Proporcionar facilidades para la comunidad, permitir comunicaciones informales, etc.
 - Estimación: Logro de reconocimientos, recompensas
 - Realización personal: Aprendizaje, asignación de responsabilidad

Para poder determinar cuales son los aspectos vinculados a fortalezas y debilidades dentro del ámbito de RRHH se debe realizar un relevamiento de esos aspectos a través de encuestas.

En el Anexo Dato02 - Complemento Preguntas RRHH se presenta un ejemplo de cuestionario de RRHH (ver ejemplo en figura 5.4)

Ejemplo de Cuestionario a utilizar en entrevistas de Recursos Humanos										
Datos Entrevista										
Fecha:										
ID Entrevista										
Entrevistador:										
Datos del encuestado										
Nombre:										
Legajo:										
Rol encuestado:										
Proyecto por el que se encuesta:										
Planes de carrera para los empleados										
Ejemplos de preguntas del ámbito de diseño y desarrollo organizacional										
ID	Audiencia			Pregunta	Aplicabilidad					Comentarios
	Sr. Managers	Líderes Proy.	Staff		No aplica	Inaceptable	Pendiente	Satisfactoria	Muy efectiva	
A 01				Tiene definido la organización los conceptos de misión, visión y valores?						
A 02				Existe un plan de Recursos Humanos que esté en línea con los objetivos de la organización?						
A 03				Existe un políticas, procedimientos y normas que guíen la operatoria de Recursos Humanos?						
A 04				Existe un sistema o procesos que vinculen planes de carrera y conjunto de habilidades de los empleados con los objetivos de la organización?						
A 05				Se involucra a los empleados en los planes de la organización y en la toma de decisiones?						

Figura 5.4 Porción de cuestionario de RRHH

Información a relevar sobre aspectos culturales de la organización

Importancia

Los factores que contribuyen al éxito de la organización son las personas, los procesos y los productos. La mejora de los procesos debe estar alineada con la mejora de las personas, ya que esto ayuda a los individuos a mejorar, y de esta forma mejorar los productos y los procesos usados en la organización.

De esta forma la mejora de procesos implica cambios que para que surtan efecto deben estar alineados con los aspectos culturales, es decir se debe reconocer e identificar los aspectos culturales existentes en la organización, en los proyectos, y a veces a nivel personal, con el propósito de conocer cuales factores promoverán o impedirán el éxito de la implementación.

Otros factores que destacan la importancia de la cultura organizacional son:

- La cultura organizacional sirve de marco de referencia a los miembros de la organización y da las pautas acerca de cómo las personas deben conducirse en ésta.
- La cultura determina la forma como funciona una empresa, ésta se refleja en las estrategias, estructuras y sistemas.
- Permite un alto grado de cohesión entre sus miembros, siempre y cuando sea compartida por la mayoría.
- La cultura organizacional es el conjunto de creencias y prácticas ampliamente compartidas en la organización. Tiene una influencia directa sobre el proceso de decisión y sobre el comportamiento de la organización.
- La cultura organizacional está integrada a la satisfacción y bienestar individual y colectivo.

Una de las posibles formas de caracterizar la cultura de una organización es a través de un modelo de 3 capas (figura 5.5):

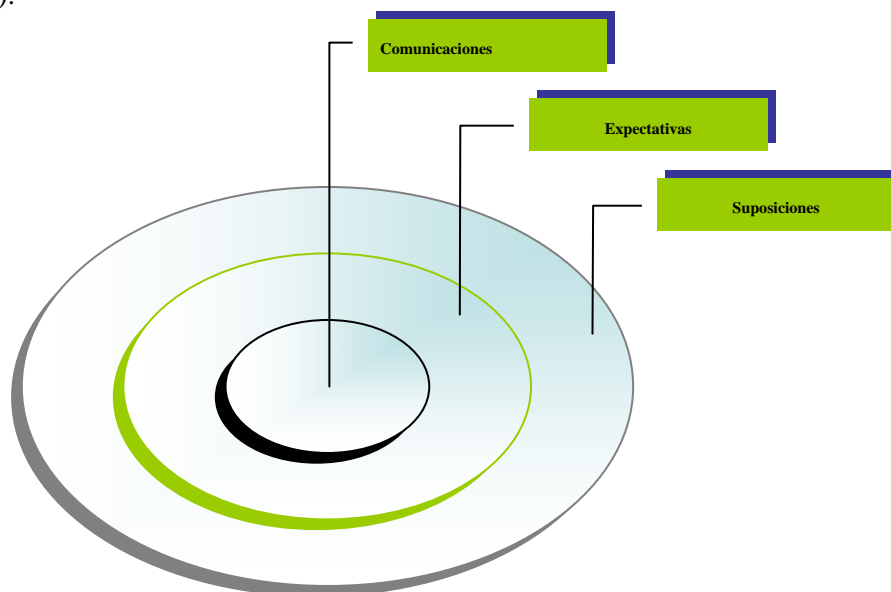


Figura 5.5 Modelo cultural de 3 capas para organizaciones

Capa de comunicaciones: Incluye a los artefactos visibles (por ejemplo documentos, datos, registros de calidad, etc.) y las actividades para conseguirlos. Son aspectos visibles que la gente puede ver, escuchar y hablar acerca de ellos.

Capa de expectativas: Incluye los objetivos de la organización, métricas, sus valores, y todo lo que se considera como comportamiento deseable o resultado deseable. Se basan en ciertas suposiciones que son centrales a la cultura de la organización.

Capa de suposiciones: Representa aspectos invisibles de la cultura que son ideas subconscientes que no se discuten. No se discuten porque todos saben y están de acuerdo de que son así. Estas suposiciones afectan las expectativas de los individuos, que a su vez afectan a las actividades y los artefactos producidos en la organización.

Por lo tanto, podemos definir a la cultura organizacional como un patrón de suposiciones básicas que los integrantes de una organización han aprendido a medida que realizan su trabajo y trabajan en conjunto.

Para ello es necesario realizar un relevamiento de los aspectos culturales de la organización a través de un cuestionario.

El objetivo del cuestionario es determinar como los individuos ven su entorno de trabajo (a nivel organización / proyecto) desde una perspectiva personal.

En el Anexo Dato02 - Complemento - Preguntas aspectos culturales se brindan un ejemplo de cuestionario para aspectos culturales (ver ejemplo en figura 5.6).

Ejemplo de Cuestionario sobre aspectos culturales					
Datos Entrevista					
Fecha:					
Id Entrevista:					
Entrevistador:					
Datos del encuestado					
Nombre:					
Legajo:					
Rol encuestado:					
Proyecto por el que se encuesta:					
ID	Categoría de suposición			Pregunta	Requiere ejemplos
	Colaborativa	Individual	Unificado		
01	X			Que piensa Ud sobre la habilidad de sus superiores (Lideres de proyecto o Sr. Managers) para colaborar y utilizar como entradas a los miembros del equipo?	
02	X			Cree que existe una clima de equipo dentro de su proyecto / organizacion? Haria algun cambio?	
03	X			Puede Ud colaborar en la toma de decisiones que la organizacion / proyecto realiza y que lo afectan? Justifique.	
04	X			Percive Ud que dispone de suficiente informacion para tomar decisiones colaborativas?	
05	X			Se siente Ud comprometido con la organizacion y con el uso de colaboracion? Justifique.	
06	X			Se enorgullece alcanzando logros laborales? Alienta a otros individuos a conseguir logros? En que forma los alienta / ayuda?	
07		X		Siente Ud. que sus necesidades de desarrollo personal son satisfechas? Justifique	
08	X	X		A que se asemeja su ambiente de trabajo organizacional? Por que lo considera de esa forma?	

Figura 5.6 Porción de cuestionario de aspectos culturales

Cada pregunta apunta a una determinada categoría de suposición: Colaborativa, Individual o Unificada. Luego que las encuestas son completadas se procede al procesamiento de la información agrupando las suposiciones por categoría e identificando las respuestas como positivas o negativas.

Las combinaciones a las que se puede llegar luego de la compilación de los resultados son:

- **Ambiente de organización colaborativo y unificado:** Se tiene que la colaboración es una creencia compartida que funciona para producir una mejora general a través de una fuerza colectiva
- **Ambiente unificado e individual:** Las creencias son compartidas pero apuntan a obtener ganancias individuales.
- **Ambiente colaborativo e individual:** Las creencias son compartidas y apuntan al beneficio colectivo e individual, mejorando los logros para ambas entidades

Con estos resultados se pretende:

- Mantener intactas aquellas influencias positivas identificadas.
- Desarrollar nuevas influencias que promuevan el éxito organizacional. Esto lleva implícito que los individuos adapten sus normas y valores a normas y valores deseados a nivel equipo y organización
- Complementar al plan de acción con los aspectos culturales que se mantienen y los que se promueven
- Evitar distintos tipos de reacciones negativas, entre las que encontramos:
 - Disconformidad personal
 - Dejar de formar parte de la organización
 - Oponerse a la implementación

Nota: El contenido de las secciones de aspectos culturales y de RRHH están desarrollados en base a las referencias: 54, 55, 59, 66, 67, 98

Etapas 3: Validación de la información relevada:

Los resultados recavados en las entrevistas permitirían tener una visión completa y detallada del proyecto, ciclo de vida de desarrollo de software y los componentes del modelo de referencia que participan del assessment.

Toda la información recolectada debe ser consolidada siguiendo reglas de corroboración, que exigen al menos:

- Que las observaciones provengan de al menos dos fuentes independientes
- Que hayan sido brindadas en diferentes sesiones
- Que puedan ser confirmadas
- Que efectivamente se llevan a cabo en la práctica mediante pruebas concretas.
- Que no sean redundantes con otras observaciones.
- Que sean significativas en el sentido que puedan expresar una debilidad, fortaleza o aceptación de una práctica alternativa.

La validación y consolidación de la información relevada es realizada por especialistas del área de proceso enfocada, bajo la aprobación del líder del equipo de assessment.

El resultado de la consolidación de la información es un conjunto de observaciones categorizadas respecto de las áreas de proceso del modelo de referencia que intervienen en el assessment.

Si fuera necesario se iteraría entre la recolección de datos, encuestas y entrevistas hasta conseguir la completitud, detalle deseado y cumpliéndose las reglas de corroboración.

5.3.7 Tarea 3-7: Analizar métricas por proyectos

Propósito

Relevar si a nivel proyectos y organización se utilizan mediciones de calidad y productividad de los proyectos, basándose en algún criterio de comparación.

Las mediciones a relevar están vinculadas a:

- Productividad
- Calidad
- Rework
- Efectividad del testing
- Otras.

<p>ENTRY</p> <ul style="list-style-type: none"> Dato17: Observaciones por el modelo de referencia Dato11: Plan de assessment 	<p>TASK: Tarea 3-7</p> <p>Analizar métricas por proyectos</p>	<p>EXIT</p> <p>Dato19: Análisis de métricas por proyecto</p>
	<p>VERIFY</p> <p>Verificación de resultados por parte del líder de assessment</p>	

En el Anexo19 se presenta un ejemplo de planilla que incluye un conjunto de posibles métricas a considerar en la evaluación. Ver ejemplo en figura 5.7.

Métrica / Proyecto		Proyecto 1						
		Registra mediciones?	Baseline mediciones	Patron comparación			Historial Desvios / Mejoras	m
				Mediciones anteriores	Standard industria	Ultima - 2		
Area Afectada	Métrica							
PP & PT	Desvíos de tamaño							
	Desvíos de esfuerzo							
	Desvíos de costo							
	Desvíos en el camino crítico							
	Desvíos en las actividades de SCM planificadas							
	Desvíos en las actividades de SQA planificadas							
	Horas reales insumidas en cada tarea vs horas estimadas							
	Porcentaje de desvío de esfuerzo registrado mensualmente							
	Porcentaje de desvío de cronograma registrado mensualmente							
	Porcentaje de desvío de costo registrado mensualmente							
	Capacitaciones realizadas en el mes							
	Magnitud final del proyecto							
	Tasa de errores del proyecto							
	Cantidad final de requerimientos de cambio							
Cantidad final de issues del proyecto								
RM	Top five mensual							
	Nivel de Exposición de los riesgos							
QA	Cantidad / tipo de errores de SQA							
	Esfuerzo insumido en actividades de							

Figura 5.7 Porción posibles métricas usadas en proyectos

Algunos ejemplos de métricas son:

- **Métricas de Planeamiento y Seguimiento de Proyectos**
 - Desvíos de tamaño.
 - Desvíos de esfuerzo.
 - Desvíos de costo.
 - Desvíos en el camino crítico.
 - Desvíos en las actividades de SCM planificadas.
 - Desvíos en las actividades de SQA planificadas
 - Horas reales insumidas en cada tarea vs. horas estimadas.
 - Porcentaje de desvío de esfuerzo registrado mensualmente.
 - Porcentaje de desvío de cronograma registrado mensualmente.
 - Porcentaje de desvío de costo registrado mensualmente.
 - Capacitaciones realizadas en el mes.
 - Magnitud final del proyecto.
 - Tasa de errores del proyecto.
 - Cantidad final de requerimientos de cambio.
 - Cantidad final de issues del proyecto

- **Métricas de Administración de Riesgos**
 - Top five mensual.
 - Nivel de Exposición de riesgos.

- **Métricas de SQA**
 - Cantidad / tipo de errores de SQA.
 - Esfuerzo insumido en actividades de SQA

- **Métricas de SCM**
 - Cantidad / tipo de errores de SCM
 - Esfuerzo insumido en actividades de SCM
 - Cantidad de cambios por estado

- **Métricas de Revisión por Pares**
 - Cantidad de errores registrados por severidad.
 - Esfuerzo insumido en revisiones.
 - Esfuerzo insumido en correcciones surgidas de las revisiones

- **Métricas de Testeo**
 - Cantidad de errores/severidad surgidos por ciclo de testing.
 - Esfuerzo insumido en actividades de testing.

- **Métricas de tamaño del software**
 - Puntos de función.

Hay que destacar que para cada métrica que se evalúe por proyecto, se puede caer en las siguientes categorías dependiendo de la forma en que se registren o comparen las mediciones:

- No se registran mediciones a utilizar para cierta métrica.
- Si se registran mediciones para una determinada métrica, lo cual implica la existencia de un baseline de mediciones. Al registrarse mediciones se pueden establecer comparaciones de mejora / desvío respecto del patrón de comparación. Los patrones de comparación pueden caer en alguna de las siguientes categorías:
 - Las mediciones se comparan contra mediciones anteriores.
 - Las mediciones registradas se comparan contra estándares de calidad y productividad de la industria, lo cual implica el acceso a dichos estándares.

Hay que destacar, que a medida que se avance en las etapas de implementación, se irán incrementando el número de métricas que compongan el baseline de la organización.

5.3.8 Tarea 3-8: Consolidar información

Propósito

Compilar a nivel organización toda la información relevada para los proyectos involucrados en el assessment.

<ul style="list-style-type: none"> • Dato11: Plan de assessment • Dato19: Análisis de métricas por proyecto • Dato17: Observaciones por el modelo de referencia • Dato14: Issues fuera del modelo de referencia 	TASK: Tarea 3-8 Consolidar información	EXIT Dato09: Resultado del assessment
	VERIFY Verificación de resultados por parte del líder de assessment	

La salida de esta tarea, esta compuesta de tres partes:

○ **Findings**

Características de la organización, indicando en que son buenos y en que no lo son, justificando la característica.

○ **Información relativa a la consolidación de observaciones vinculadas al modelo de referencia**

El propósito es compilar y consolidar a nivel organización toda la información relevada en las observaciones de los proyectos involucrados en el assessment.

Las observaciones desarrolladas están basadas en referencia a las áreas de proceso del modelo de referencia.

El objetivo es brindar una forma de expresar estructuradamente la salida del assessment y que dicha salida sea utilizada como template para los siguientes assessments. De esta forma la organización puede construir un baseline del estado actual de implementación.

En el Anexo09 (ver ejemplo en figura 5.8) se proporciona una planilla para registrar la información consolidada de los activos analizados.

Características de la planilla:

- Las actividades están agrupadas por objetivos
- Los objetivos están agrupados por área de proceso.
- Las áreas de proceso están agrupadas por nivel de madurez.
- Al pie de cada nivel se sumaliza la información, y una sumarización global se presenta a nivel planilla
- El valor que presenta esta herramienta es que consolida y da una rápida visión del estado actual de implementación. Luego, las instancias de esta planilla irían formando un baseline del estado de avance en la implementación.

Objetivo / Practica clave	ID	•?	Description - Peer Reviews KPA	Cumplimiento [0(no aplica) ,1 (aplica parcialmente) ,2(aplica totalmente)]
Objetivo 1	PRD01		Se planean las actividades de revisión por pares.	
Objetivo 2	PRD02		Los defectos en los productos de trabajo del software se identifican y se eliminan.	
Compromiso 1	PRC01	•	El proyecto sigue una política organizacional escrita para realizar revisiones por pares. Esta política especifica típicamente que:	
Habilidad 1	PRH01	•	Se proporcionan recursos y financiamiento adecuados para realizar revisiones por pares en cada producto del trabajo de software que se revisara.	
Habilidad 2	PRH02		Los líderes de la revisión por pares reciben el entrenamiento requerido acerca de como conducir revisiones por pares.	
Habilidad 3	PRH03		Los revisores que participan en revisiones por pares reciben el entrenamiento requerido sobre los objetivos, los principios, y los metodos de revisiones por pares.	
Actividad 1	PRA01	•	Se planean las revisiones por pares, y se documentan los planes. Estos planes:	
Actividad 2	PRA02	•	Las revisiones por pares se realizan segun un procedimiento documentado. Este procedimiento especifica típicamente que:	
Actividad 3	PRA03		Se registran los datos sobre la conduccion y los resultados de las revisiones por pares.	
Medición 1	PRM01		Las mediciones se toman y se utilizan para determinar el estado de las actividades de la revisión por pares.	
Verificación 1	PRV01	•	El grupo de la garantia de oalidad del software revisa y/o audita las actividades y los productos de trabajo para las revisiones por pares e informar los resultados. Como minimo, las revisiones y/o las auditorias verifican que:	
Políticas	PRPOL01		The project follows a written organizational policy for performing peer reviews. This policy typically specifies that: The organization identifies a standard set of software work products that will undergo peer review. Each project identifies the software work products that will undergo peer review. Peer reviews are led by trained peer review leaders.	

Figura 5.8 Porción de planilla para consolidar assessment SW-CMM

Los criterios para completar esta planilla son:

- Se confecciona en base al consenso del equipo completo de assessment.
 - Para cada fila que representa un objetivo, área clave o artefacto se debe considerar si se cumple con el issue en cuestión. Los posibles valores a poner son:
 - **Satisfecho Totalmente (2)**
 - **Satisfecho parcialmente (1):** Se considera parcialmente satisfecha (1) cuando una práctica no esta cubierta del todo, o no es bien interpretada o en las entrevistas surgió algún nivel de incertidumbre con relación al uso
 - **No satisfecho (0)**
 - Si todos los integrantes de la evaluación consideran que la práctica esta totalmente satisfecha (contemplando la existencia de opiniones en el grupo que la practica esta parcialmente satisfecha), entonces la tarea se considera satisfecha.
 - Pero si alguno de los integrantes considera que la tarea no esta satisfecha, entonces dicha tarea automáticamente califica como no satisfecha.
 - Luego un objetivo es considerado satisfecho si todas sus actividades son consideradas satisfechas.

- Posteriormente un área de proceso es considerada satisfecha si todos sus objetivos fueron calificados como satisfechos.
- Y por último, un nivel se considera alcanzado cuando todas sus áreas de proceso han sido satisfechas.
- **Columna instancia template organizacional:** Si el proyecto en evaluación tiene un documento (procedimiento por ejemplo) realizado a partir del template organizacional. Si existe, debe contener su nombre (y ubicación). Puede que no aplique esta entrada para una tarea en particular
- **Columna existe baseline:** Si la tarea en cuestión hace uso de un baseline, hay que poner el nombre en esta entrada. (puede que no aplique esta entrada para una tarea en particular)
- **Información no vinculada al modelo de referencia.** Le da un valor agregado al assessment, ya que considera issues no perteneciente al modelo y que permitirán desarrollar un plan de acción con alternativas para la organización de acuerdo a los objetivos que persiga en base a beneficios cuantitativos y cualitativos que no estén vinculados directamente al modelo.

Los aspectos vinculados a recursos tecnológicos (hardware, software, etc.) con que cuenta la organización antes de comenzar el proceso de implementación tiene una gran importancia desde el punto de vista económico, ya que los costos asociados son altos y el tipo de organización puede ser pequeña o mediana.

Dentro del Anexo09 se muestran una serie de planillas a completar con información relativa a estos issues, donde se encuentran la siguiente información

- Herramientas Case
- Software de apoyo a actividades

5.3.9 Tarea 3-9: Preparar y presentar conclusiones

La etapa de entrega de resultados consiste en:

- Presentación de los resultados a todos los participantes
- Una sesión final con la gerencia para clarificar detalles
- Efectuar actividades de cierre de la evaluación.
- Aprobación del assessment.

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato11: Plan de assessment • Dato09: Resultado del assessment 	<p>TASK: Tarea 3-9</p> <p>Preparar y presentar conclusiones</p>	<p>EXIT</p> <p>Sr. Managers, Líderes de proyectos, Funcionales y especialistas son notificados sobre resultados. Aprobación de resultados de assessment por Sr. Managers.</p>
	<p>VERIFY</p> <p>Verificación de resultados por parte del líder de assessment</p>	

Capítulo 6

Tarea 4

Plan de acción

6.1 Identificación

Tarea 4

6.2 Introducción al capítulo

El plan de acción es una respuesta formal y escrita al assessment, y constituye el mapa de la mejora de procesos. Se puede considerar al plan de acción como el corazón de la mejora de procesos.

La respuesta al assessment pretende realizar mejoras a nivel:

- Procesos de desarrollo de software
- Aspectos Culturales
- Aspectos vinculados a RRHH

La estrategia para realizar estas mejoras es realizando un plan de acción que mejore los procesos de desarrollo, y que incluya issues vinculados a aspectos culturales y de RRHH. Aquellos aspectos culturales no tenidos en cuenta en el plan de acción son derivados al sector de RRHH bajo una lista de acciones de RRHH y culturales para que este departamento los implemente.

El plan de acción se compone de:

- **Plan táctico de mejoras (Global):** Son las acciones que se deben aplicar para realizar mejoras a nivel organizacional con el propósito de lograr los objetivos propuestos por la organización (objetivos estratégicos). Puede ser visto como un plan maestro (o template para los planes operacionales) que contempla las particularidades de los proyectos involucrados en la organización. Este plan es desarrollado en la presente tarea.
- **Conjunto de planes operacionales (específicos para cada proyecto):** Estos planes son instancias creadas a partir del plan táctico de mejoras (que contempla los resultados del assessment para cada proyecto) para que reflejen características particulares de cada proyecto sobre como implementar el plan táctico en un proyecto en particular. De esta forma cada el plan operacional de cada proyecto satisface en forma particular el punto de vista táctico con visión a lo estratégico.

Estos planes son desarrollados con la colaboración de individuos de los proyectos (líderes y especialistas) para obtener un plan realista y al mismo tiempo que motive a los futuros usuarios de los procesos.

Nota: *El contenido del presente capítulo fue desarrollado en base a investigaciones hechas en las referencias: 14, 15, 22, 53, 62, 57, 63, 70, 72, 81, 85, 86, 89, 90*

6.3 Estructura de la tarea

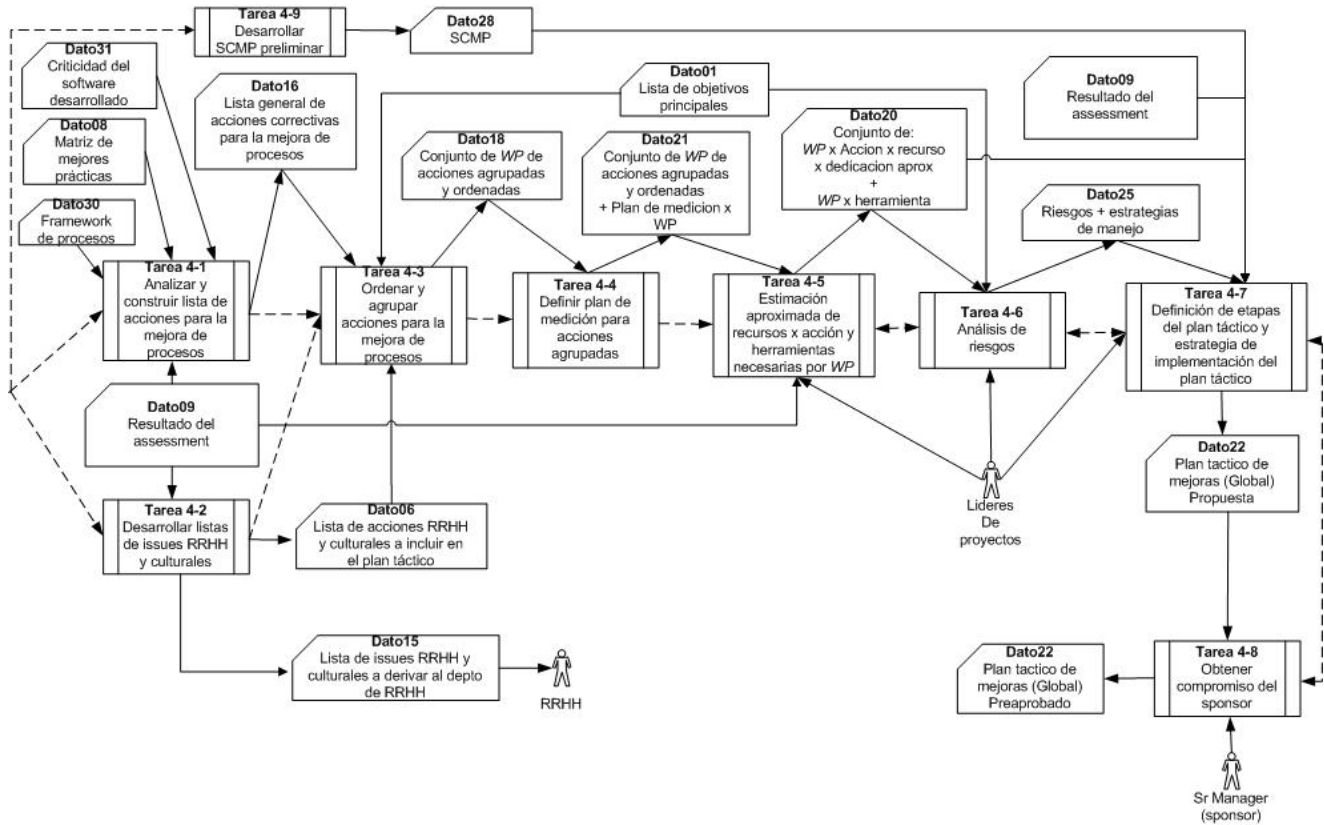


Figura 6.1 Estructura tarea 4

6.3.1 Tarea 4-1: Analizar y construir lista de acciones para la mejora de procesos

Propósito

Construir una lista de acciones relativas a la mejora de procesos que guíen hacia:

- La lista de recomendaciones (producto final de la tarea de assessment)
- Acciones básicas necesarias para la implementación de la mejora de procesos

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato08: Matriz de mejores prácticas • Dato09: Resultado del assessment • Dato31: Críticidad del software desarrollado • Dato30: Framework de procesos 	<p>TASK: Tarea 4-1</p> <p>Analizar y construir lista de acciones para la mejora de procesos</p> <hr/> <p>VERIFY</p> <p>Tener desarrollada la lista de acciones, y validarla cruzándola contra la lista de objetivos principales y mejores practicas.</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato16: Lista general de acciones correctivas para la mejora de procesos
--	--	--

La justificación de esta tarea se debe principalmente a:

- Se necesita de un conjunto de acciones que sean los cimientos para brindar un soporte integral a la mejora de procesos
- La lista de recomendaciones del assessment generalmente no es lo suficientemente específica
- Se necesita identificar acciones específicas para realizar las implementaciones vinculadas a esas recomendaciones

Esta lista de acciones específica que es lo que se debe hacer y como se debe implementar para poder traducir las recomendaciones en acciones que guíen hacia la construcción posterior de los planes.

Procedimiento para desarrollar la lista de acciones

Encontramos dos conjuntos de acciones a desarrollar en esta tarea:

- **Acciones que dan soporte a la mejora de procesos**

Estas acciones se ejecutan por única vez en la organización como paso previo a la implementación en sí (ver figura 6.2).

Entre sus acciones se destacan:

- Establecimiento y capacitación del grupo de SEPG.
- Definición, implementación y soporte para el uso de un repositorio de procesos de software de la organización

Se establece para recolectar y dejar disponible información acerca de procesos de software y a los productos derivados de dichos procesos.

- Definición, implementación y soporte para el uso de una librería de documentación relativa a los procesos de software

Su propósito es:

- Almacenar documentos de procesos que son potencialmente útiles para los proyectos actuales y futuros.
- Permitir compartir su contenido a toda la organización.
- Definición, implementación y soporte para el uso de un repositorio de los framework de desarrollo de software aprobados para ser usados en la organización (*tópico analizado en el capítulo 1*).
- Definición, implementación y soporte para el uso de un repositorio del marco de proceso de desarrollo estándar y sus derivados.
- Definición, implementación y soporte para el uso de una herramienta para la carga de horas del staff respecto de las tareas realizadas para monitorear el esfuerzo realizado.
- Definición, implementación y soporte de herramienta de administración de proyectos.
- Definición, implementación y soporte de herramienta para registrar defectos encontrados con el propósito de hacer seguimiento de los mismos.

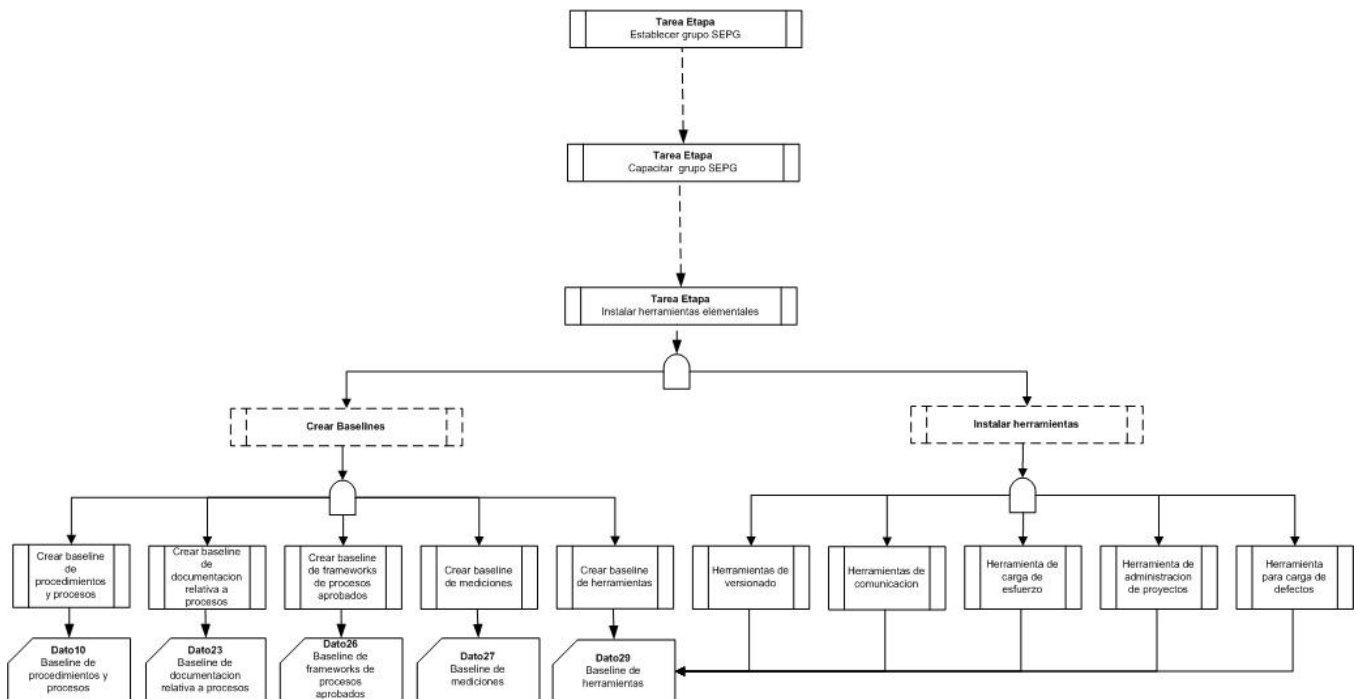


Figura 6.2 Estructura tarea para soporte a mejora de procesos

• **Acciones relativas a las recomendaciones del assessment**

- En función de la lista de recomendaciones, identificar que objetivo persigue cada recomendación y cuales acciones se pueden tomar para alcanzar ese objetivo

En base a lo anterior se pueden dar los siguientes escenarios:

- Basándose en las fortalezas detectadas en el assessment
 - Si hay buenas instancias de los procesos implementados en algunos proyectos pueden proveer buenas prácticas a ser institucionalizadas en los demás proyectos.
- Basándose en las debilidades detectadas en el assessment
 - Identificar acciones a tomar para implementar procesos para aquellas recomendaciones del assessment identificadas como debilidades.
 - También puede darse el caso que una instancia de proceso sea conveniente excluirla.

Cada acción incluida en la lista debe ser acompañada (ver figura 6.3) por:

- Objetivos
- Lista de recomendaciones atacadas por la acción
- Detalle de pasos a seguir
- Deliverables esperados.
- Análisis de las consecuencias de implementar esa acción.

Lista de acciones para la mejora de procesos

Fecha creación: <Fecha>
 Organización: <Nombre organización>
 Versión: <Nro de versión>

Id Acción	Objetivo perseguido	Id Recomendación asociada	Detalle Acción	Id mejor practica asociada	Deliverable esperado

Responsable armado de lista: <Responsable>

Figura 6.3 Porción template de acciones correctivas

6.3.2 Tarea 4-2: Desarrollar listas de issues RRHH y culturales

Propósito

Aquellos issues no contemplados por el modelo de referencia que no hayan sido tenidos en cuenta en la tarea 4-1 respecto a aspectos culturales y de RRHH deben ser analizados y agrupados en dos categorías:

- Lista de issues RRHH y culturales a derivar al Dpto. de RRHH (ver ejemplos en figura 6.4).
- Lista de issues RRHH y culturales a incluir en el plan táctico (ver ejemplos en figura 6.5).

El criterio por el que se incluyen en una lista u otra esta relacionado con el grado de especialización requerido para la implementación de prácticas relativas a estos issues, donde se clasifica:

- En base a la complejidad se derivan algunos issues culturales y de RRHH al Dpto. de RRHH para que los considere, planifique e implemente de forma óptima.
- Los issues culturales y de RRHH restantes pasan a ser tenidos en cuenta para que formen parte de las actividades del plan táctico de implementación.

<p>ENTRY</p> <ul style="list-style-type: none"> Dato09: Resultado del assessment 	<p>TASK: Tarea 4-2</p> <p>Desarrollar lista de issues RRHH y culturales</p>	<p>EXIT</p> <ul style="list-style-type: none"> Dato15: Lista de issues RRHH y culturales a ser tenidos en cuenta por Dpto. de RRHH Dato06: Lista de acciones RRHH y culturales a incluir en el plan táctico
	<p>VERIFY</p> <p>Tener desarrollada la lista de issues RRHH y culturales, y validarla cruzándola contra la lista de objetivos principales. Deberán participar de la validación los integrantes del departamento de RRHH.</p>	

Algunos posibles issues a derivar al departamento de recursos humanos

- Definición de Valores a nivel organizacional**

Los valores representan la base de evaluación que los miembros de una organización emplean para juzgar situaciones, actos, objetos y personas.

Los valores son los cimientos de cualquier cultura organizacional; definen el éxito en términos concretos para los empleados y establecen normas para la organización.

La importancia del valor radica en que se convierte en un elemento motivador de las acciones y del comportamiento humano; define el carácter fundamental y definitivo de la organización; crea un sentido de identidad del personal con la organización.

- Definición de Misión a nivel organizacional**

La misión puede verse como el nivel de percepción que tenga un individuo acerca de las respuestas satisfactorias que se le dé acerca del objetivo principal, ético y trascendente de la existencia de la organización.

Proporciona sentido y propósito, definiendo una función social y metas externas para una institución y definiendo funciones individuales con respecto a la función organizacional.

Una visión exitosa generalmente está constituida por grupos de individuos comprometidos con la organización y que están dispuestos a proporcionar todo su potencial para el logro de los objetivos propuestos.

- Definición de Visión a nivel organizacional**

Todo proceso de transformación organizacional requiere que se genere una visión compartida de la alta gerencia acerca de lo que se quiere alcanzar; es decir, toda la actividad organizacional se transforma en parte de un propósito mayor encarnado en los bienes y/o productos de dicha organización. No hay organización inteligente sin visión compartida.

- Desarrollo de la satisfacción del individuo y mitigar el desgaste**

La gente necesita estar satisfecha con su trabajo. Deben satisfacerse sus necesidades físicas y emocionales. Entre las necesidades psicológicas que se deben satisfacer en un clima corporativo están:

- Las necesidades de realizar y ser reconocido por esa realización.
- Identificarse con un grupo social y tener un sentido de pertenencia.

La satisfacción de éstas necesidades ayuda a proporcionar la motivación necesaria para que los empleados permitan mejorar lo que está a su alrededor, aunque no se les haya pedido expresamente hacerlo.

- Establecer un sistema de reconocimiento apropiado y sistema de premios**

El sistema de recompensa debe estar basado en la efectividad del proceso donde se comparte el éxito y el riesgo.

- Capacitar liderazgo para demostrar y comunicar valores culturales**

Capacitar continuamente para dar apoyo a crear y fortalecer el sentido de compromiso del personal, cambiar actitudes y construir un lenguaje común que facilite la comunicación, comprensión e integración de las personas.

Definitivamente lo que más ayuda a la organización a alcanzar el éxito en sus procesos, es contar con gente

altamente preparada e identificada con la organización.

• **Mejorar las habilidades de comunicación**

Se debe estimular la participación de los trabajadores tanto en el diseño de sus puestos como en la búsqueda e implantación de mejoras y lograr una relación de cooperación.

• **Promover el trabajo en equipo**

Desarrollar la capacidad de un grupo para trabajar conjuntamente en armonía, a fin de alcanzar objetivos comunes. Esto significa que la gente tenga mutua confianza y se sienta cómoda en los equipos, pero antes deben sentirse cómodos en sus funciones individuales. Donde hay trabajo en equipo el grupo participa en la toma de decisiones y al hacerlo, los miembros del grupo se solidarizan con las necesidades y con los sentimientos que se expresan libremente. El clima estimula continuamente la franqueza. Se comparte el crédito por las realizaciones y los estilos predominantes de influencia son aquéllos que se basan en una visión de futuro (común para todos) y en una continua participación y creación de confianza.

• **Promover la responsabilidad**

Los integrantes del grupo deben estar dispuestos a asumir responsabilidad para hacer cambios. La responsabilidad significa estar dispuesto a querer realizar las tareas desde la primera vez y procurar el mejoramiento continuo por siempre. El nivel de responsabilidad está íntimamente relacionado con el hecho de que el sistema gerencial esté fomentando responsabilidad y confiabilidad.

Fecha creación: <Fecha>

Organización: <Nombre organización>

Versión: <Nro de versión>

Id issue RRHH derivado	Objetivo perseguido	Detalle del objetivo
01	Desarrollo de modelos mentales	Definición a nivel organizacional de: <ul style="list-style-type: none"> • Valores • Misión • Estrategia • Visión • “Como se hacen las cosas en la organización”
02	Motivación	<ul style="list-style-type: none"> • Desarrollo de la satisfacción del individuo y mitigar el desgaste • Establecer un sistema de reconocimiento apropiado y sistema de premios. • Programa de refuerzo de valores
03	Capacitación	<ul style="list-style-type: none"> • Capacitar liderazgo para demostrar y comunicar valores culturales • Promover el alineamiento de los equipos con la organización. • Habilidades de comunicación. • Promover el trabajo en equipo • Sentido del trabajo • Responsabilidad • Inclinación participativa • Lealtad • <u>Involucración</u> afectiva

Figura 6.4 Ejemplo de lista de issues RRHH y culturales a derivar

Para muchas organizaciones, un cambio organizacional puede significar pasar de una cultura tradicional (en la cual prevalecen estilos burocráticos, motivacionales y valores por el poder y la afiliación, y un clima de conformidad) a una cultura del desempeño, donde es posible aportar nuevas ideas; la gente puede asumir riesgos calculados y es incentivada a establecerse metas retadoras, mediante el reconocimiento del mérito y resultados excelentes.

Un posible modelo para realizar los cambios por parte del Dpto. de RRHH podría ser el planeado de Lippitt, Watson y Westley que consta de 5 fases:

- Desarrollo de la necesidad de cambio (descongelación de Lewin).
- Establecimiento de una relación de cambio.

- Trabajo para lograr el cambio (avance).
- Generalización y estabilización del cambio (recongelación)
- Logro de una relación Terminal.

A continuación se muestra un ejemplo de issues RRHH y culturales a incluir en el plan táctico

Lista de issues culturales y RRHH a considerar en la mejora de procesos

Fecha creación: <Fecha>
Organización: <Nombre organización>
Versión: <Nro de versión>

Id issue RRHH	Objetivo perseguido	Detalle del objetivo
01	Motivación	<ul style="list-style-type: none"> ● Poner énfasis en que el proceso de mejora es parte del trabajo cotidiano de todos ● Motivar al individuo respecto de cómo el éxito de sus tareas se vinculan con el trabajo del grupo y la organización.
02	Capacitación previa a cada etapa	<ul style="list-style-type: none"> ● Explicar cuales son los objetivos de la etapa a implementar ● Capacitar a cada uno con las tareas que debe hacer ● Dar a conocer de que forma se medirá el éxito del cumplimiento de una tarea / etapa ● Comunicar de que forma se ponen a disposición los datos e información del proyecto para cada uno de los individuos.

Responsable armado de lista: <Responsable>

Figura 6.5 Ejemplo de lista de issues RRHH y culturales a tener en cuenta en la mejora de procesos

6.3.3 Tarea 4-3: Ordenar y agrupar acciones para la mejora de procesos

Propósito

Desarrollar una serie de work packages de acciones correctivas utilizando un criterio de agrupación.

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato16: Lista general de acciones correctivas para la mejora de procesos • Dato01: Lista de objetivos principales 	<p>TASK: Tarea 4-3</p> <p>Ordenar y agrupar acciones para la mejora de procesos</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato18: Conjunto de Work packages de acciones agrupadas y ordenadas
	<p>VERIFY</p> <p>Validar que de acuerdo al criterio de agrupación elegido, los workpackages sean consistentes, complementarios y no conflictivos entre si.</p>	

La justificación para esta tarea es que las acciones identificadas pueden ser muy diversas, y agrupándolas por un criterio se pueden coordinar esfuerzos de modo que la implementación de un conjunto de acciones apunte a objetivos comunes.

El criterio de agrupación esta fuertemente ligado al contexto de la organización.

En la figura 6.6. se muestra un ejemplo de cómo agrupar las acciones correctivas.

Posibles criterios para la agrupación de acciones correctivas

- Las acciones correctivas pueden ser agrupadas en base al nivel de la organización al que impactan. Algunos ejemplos son:
 - A nivel proyecto
 - A nivel unidad de negocio
 - A nivel gerencia
 - Otros

Se aconseja que las mejoras a diferentes niveles de la organización sean implementadas por distintos grupos desarrolladores de procesos.

- Agrupar por mejor práctica común a un conjunto de acciones.
- Si varias acciones involucran a un tópico en particular, es preferible agruparlas por ese tópico.
- Varias acciones pueden ser agrupadas si afectan a un determinado producto o segmento de productos dentro del proyecto u organización.
- Si varios procesos en la organización fueron identificados con síntomas similares de un problema en particular, dicho problema podría ser mejor atacarlo a nivel organización.
- Agrupar teniendo en cuenta los objetivos perseguidos por la organización.

Cada work package resultado de la agrupación deberá contar con la siguiente información

- Id de work package
- Lista de recomendaciones atacadas por el work package
- Dependencias entre las acciones que lo componen.
- Para cada acción especificar:
 - Lista con las mejores prácticas a utilizar
 - Lista de deliverables esperados
 - Indicadores a usar para medir el progreso de la implementación del work package

Nota acerca de las acciones de soporte a la mejora de procesos: Se recomienda agrupar todas aquellas tareas de soporte integral a la mejora de procesos bajo un único work package.

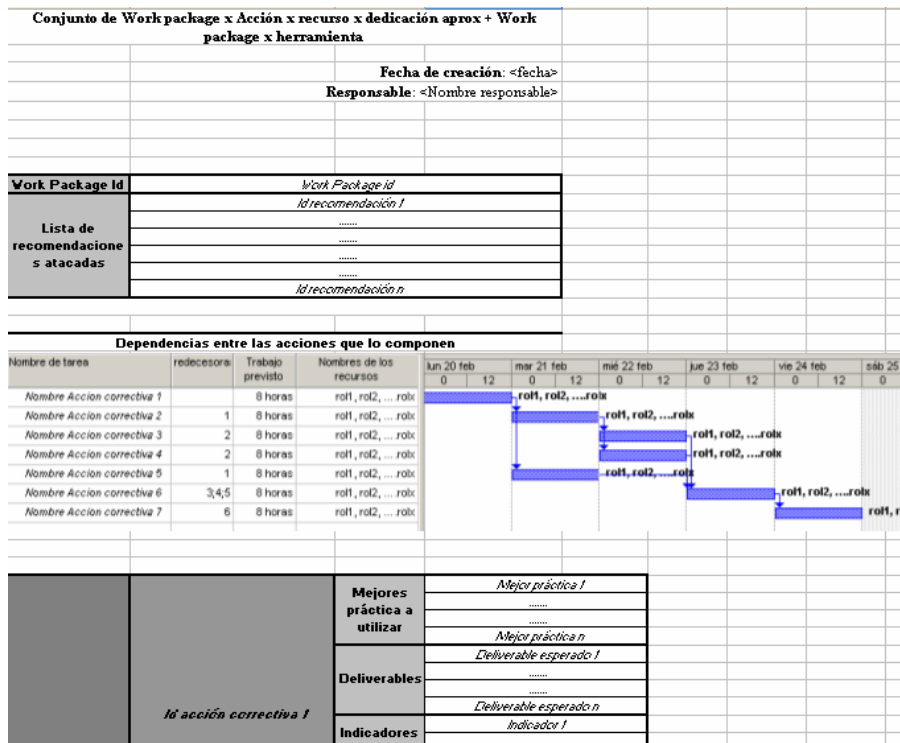


Figura 6.6 Ejemplo / template de agrupación de acciones correctivas

6.3.4 Tarea 4-4: Definir plan de mediciones para acciones agrupadas

Propósito

Para cada work package se debe desarrollar un plan de medición a utilizar durante la tarea de implementación con el fin de monitorear y controlar el avance de la implementación.

<p>ENTRY</p> <ul style="list-style-type: none"> Dato18: Conjunto de WP de acciones agrupadas y ordenadas 	<p>TASK: Tarea 4-4: Definir plan de mediciones para acciones agrupadas</p>	<p>EXIT</p> <ul style="list-style-type: none"> Dato21: Acciones agrupadas y ordenadas por etapa + plan de medición por WP
	<p>VERIFY</p> <p>Cada workpackage tiene definido un conjunto de mediciones para monitorear y controlar el avance de las tareas del workpackage.</p>	

Por definición un plan de métricas describe que es lo que se debe medir, porque medir, y como se analizara y responderá al análisis.

De acuerdo a los objetivos abarcados por el work package, se especifica un plan de medición que sirva de soporte durante la implementación del work package (etapa), tanto para la tarea de instalación como para la de apalancamiento (ambas corresponden a la tarea de implementación).

Las métricas a utilizar en la implementación pueden estar en uno de los siguientes grupos dentro de la implementación del work package:

6.3.4.1 Métricas que se utilizan en la tarea de instalación de la etapa (work package) (perteneciente a la tarea de implementación).

Estas métricas son útiles para controlar y monitorear el avance la instalación de la etapa respecto a la parte del plan que rige la implementación de la etapa.

Su análisis permitirá corregir los planes para una mejor implementación de esta etapa en el proyecto

actual y de la posterior implementación de la etapa en los demás proyectos. De esta forma se logra realiza prevención de defectos.

Ejemplos de estas métricas:

6.3.4.1.1 Métricas de esfuerzo para la instalación / desarrollo de activos de proceso de la etapa:

Es una medición clave para validar que lo que se esta ejecutando esta dentro del presupuesto. Para monitorear el esfuerzo utilizar una aplicación donde el staff involucrado cargue a periodos regulares la cantidad de tiempo insumido para realizar sus actividades. La aplicación deberá codificar cada entrada de la misma forma que se codifico la actividad en el plan mas algún posible digito que represente el retrabajo sobre la misma.

6.3.4.1.2 Métricas de defectos producidos al instalar o desarrollar activos de proceso: Debido a su impacto directo sobre la calidad, estas mediciones son generalmente consideradas muy importantes. Los defectos deben ser registrados en un log de defectos, registrándose información desde su detección hasta su clausura.

El uso de estas mediciones permite hacer el seguimiento de defectos y particularmente modificar los planes realizando detección y eliminación de errores para aquellos proyectos que utilicen el plan modificado al implementar la misma etapa. De esta forma se implementa también prevención de defectos.

6.3.4.1.3 Métricas de recursos utilizados respecto del plan (dinero, tiempo, staff, recursos críticos, otros).

6.3.4.1.4 Métricas de progreso en la instalación respecto del estimado a realizar

6.3.4.2 Métricas que se utilizan en la tarea apalancamiento de la etapa (perteneciente a la tarea de implementación)

Estas métricas son útiles para controlar y monitorear la forma en que se utilizan en la organización los activos de proceso desarrollados en la tarea de instalación, perteneciente a la tarea de implementación.

Estas métricas permitirán apalancar los activos, institucionalizándolos, haciendo correcciones en tiempo de uso, etc.

Ejemplos de estas métricas:

6.3.4.2.1 Proporción de proyectos que adhieren a los procedimientos sobre el total de proyectos

6.3.4.2.2 Medición de efectividad del proceso: Medida en que las personas que usan el proceso obtienen resultados satisfactorios, por ejemplo cantidad de no conformidades detectadas en las revisiones de calidad.

6.3.4.2.3 Mejora proporcional del tiempo de entrega al cliente desde la implementación de los nuevos procesos

6.3.4.2.4 Variación en quejas por defectos

6.3.4.2.5 Variación en proporción de horas extra trabajadas por el staff

6.3.4.2.6 Métricas financieras: Costos y Valor ganado (Earned Value)

6.3.4.2.7 Métricas de esfuerzo en el uso de los activos de proceso de la etapa (mismas características que 5.3.4.1.1)

6.3.4.2.8 Métricas de defectos por uso de los work products (mismas características que 6.3.4.1.2)

6.3.4.2.9 Métricas de volatilidad de procesos y políticas ya definidas.

Contenido del plan de métricas

- Cuales serán las métricas a ser colectadas y como son definidas.
- Cuales son los objetivos perseguidos por cada métrica.
- Mecanismos y técnicas para recolectar información: La recolección debe ser mayormente automática y no intrusiva.
 - Con que frecuencia se tomaran mediciones
- Mecanismos y técnicas para analizar la información
 - Como serán utilizadas las métricas del workproduct para identificar y medir la performance
 - Niveles de rango de aceptación de las mediciones
 - Como se usaran las métricas para determinar que tan bien se esta llevando a cabo el proceso respecto de milestones y objetivos de calidad del proceso.
- Reportes a generar
- Responsabilidades
 - Quien relevara la información
 - Quien analizará la información

En el anexo dato21 se presenta un template para volcar el plan de medición.

6.3.5 Tarea 4-5: Estimación aproximada de recursos x acción y herramientas necesarias por Work Package

Propósito

Complementar los work packages con estimaciones de uso de recursos y herramientas necesarias. Para esto es necesario realizar un análisis de las herramientas existentes (ver figura 6.7) versus alternativas existentes en el mercado.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> • Dato21: Acciones agrupadas y ordenadas por etapa + plan de medición por WP • Dato09: Resultado del assessment 	<p style="text-align: center;">TASK: Tarea 4-5</p> <p>Estimación aproximada de recursos x acción y herramientas necesarias por Work package</p>	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> • Dato20: Conjunto de WP x Acción x recurso x dedicación aprox. + WP x herramienta
	<p style="text-align: center;">VERIFY</p> <p>Para cada workpackage tener definida una estimación de los recursos y dedicación necesarios, y esa estimación debería ser aceptada por el/los líderes de proyectos involucrados.</p>	

Script para realizar esta tarea

1. Realizar una reunión con los líderes de los proyectos de la organización con el propósito de:
 - 1.1 Para cada work package, realizar
 - 1.1.1 Para cada acción correctiva dentro del work package, realizar
 - 1.1.1.1 Definir cuales recursos son necesarios
 - 1.1.1.2 Definir cual será la dedicación necesaria para cada tipo de recurso
 - 1.1.1.3 En base a
 - Sección “Herramientas Case usadas en el proyecto” dentro del documento que se desarrolló en el assessment.
 - Conocimiento de herramientas disponibles en el mercado (Ver tabla con ejemplos de herramientas disponibles en el mercado)
 Definir cuales herramientas son necesarias para la acción correctiva
 - 1.1.1.4 Iterar con 1.1.1.1 hasta obtener consenso entre los líderes sobre hallazgos para la acción correctiva
 - 1.1.2 Iterar con 1.1.1 hasta obtener consenso entre los líderes sobre hallazgos para el work package y posibles incompatibilidades entre el uso de herramientas con otros work packages
- 2 Documentar los hallazgos (ver figura 6.8)

Actividad relacionada	Herramienta	Fabricante	Actividad relacionada	Herramienta	Fabricante
Bug / defect / problem tracking	+1 CP	+1 Software Engineering	Human resources and financial system	Personnel System	
	Bugzilla			ActiveWatch	Mercury Interactive
	ClearDDTS	Rational Software Corporation		Mc Cabe QA	Mc Cabe & Associates
	ClearQuest	Rational		Resource Standard Metrics	M Squared Technologies
	Detect WorkFlow	SoftQuest Systems		Tau Logiscope	Telelogic AB
	Digite			Together ControlCenter	Borland, Inc.
	Mc Cabe TRUTrack	Mc Cabe & Associates		xSuds	Bellcore
	SilkRadar	Segue Software, Inc.		AutoPlan Enterprise	Tufan, Inc.
	TeamTrack	TeamShare, Inc.		Catalyze	SteelTrace
	Test Management System	Rex Black Consulting Services, Inc.		Cradle Version 5.2	Structured Software Systems Ltd
Case Tools	Vantive Quality	Vantive Corporation	in-Step	microTOOL GmbH	
	Erwin		Juggler	Catalyst Software Ltd	
Comunicación	Poseidon for UML	Genteware	Microsoft Project	Microsoft	
	Rational Rose		MS Business Scorecard Accelerator	Microsoft	
Comunicación	Software de emails / extranet / intranet / Messenger		Prinopia/PRINCE	British Aerospace Ltd. Software Tool	
	Software para Teleconferencias		SoftTrak	SoftQuest Systems	
Configuration Management	+1 CM	+1 Software Engineering	Requirements Management	Analyst Pro Version 5.0	Goda Software, Inc.
	AccuRev/CM	AccuRev Inc.		Awom Sys Version 6.0	Structured Technology Group, Inc
	AllChange	Intasoft Ltd.		CaliberRM 2005	Borland
	Change and Configuration Control (CCCM/ Harvest)	Computer Associates		CARE3.2 (Computer Aided Requirements Engineering)	SOPHIST GROUP
	ClearCase	Rational		Catalyze	SteelTrace
	CM Synergy	Telelogic AB		ClearSpecs Composer	LineSpecs Software
	CMVision and Configuration Management Facility (CMF)	Expertware		EasyRM Version 1.06	Cybernetic Intelligence GmbH
	CMWin	Expertware, Inc.		Gatherspace	Gatherspace
	Configuration Management Version Control (CM/VC)	IBM Corp.		GMARC (Generic Model Approach to Requirements Capture)	Computer System Architects Ltd.
	CONTROL-C/S	Thompson Automation Software		IBM Rational RequisitePro	IBM Rational Software
	CVS (Concurrent Versions System)	Freeware		IRQA (Integral Requisite Analyze)	TCP Sistemas e Ingeniería
	DRTS	ILSI		RDT (Requirements Design & Traceability)	Igatech
	Endeavor Workstation	Computer Associates		Requirements Traceability Management (RTM)	Serena Software, Inc.
	ExoConf	Exocoft AB		RequisitePro	Rational
	FpVC	Gregory Nisewich		Verotrace	Verotrace
	GNU C/SSC	GNU Freeware		Autotester	Autotester Inc.
	Human-Oriented Programming Environment (HOPE)	Aladdin Knowledge Systems, Inc.		CC-OPAIRA-TSM	CC GmbH
	NeumaCM+	Neuma Technology Inc.		Cyano Test	Cyano SA
	Perforce	Perforce Software		Debuggers en gral de los lenguajes	
	PrimeCode	Data Design Systems, Inc.		e-TEST Suite	Emplix
	Product Configuration Management (PCM)	Tesseract Technologies		ILZU	Imbus AG
	PVCS	MERANT		ITOM	Imbus AG
	QEF	QEF Advanced Software Inc.		QARun	Compuware
	Razor	Visible Systems		QuickTest Professional	Mercury
	Revision Control Engine (RCE)	DuraSoft		Rational Robot (Rational Suite TestStudio)	IBM Rational Software GmbH
	SCCS Source Code Control System	Freeware		Rational team test	
	Software Manager	Vertical Sky		Rational Visual Test	IBM Rational Software GmbH
	Source Code Manager	Unipress Software Inc		Rational XDE Tester	IBM Rational Software GmbH
	Star Team	StarBase Corporation		SilkTest	Segue Software
	TeamConnection	IBM Corp.		Test Partner	Compuware
Visual SourceSafe	Microsoft	TESTSCOPE	SCOPE GmbH		
DocEXPRESS	Telelogic AB	Unified TestPro (UTP)	Software Development Technologies		
Document publishing / FileCM	loatech Systems Ptv Ltd	Vermont HighTest Plus	Vermont Creative Software		

Figura 6.7 Porción de ejemplo de herramientas disponibles en el mercado

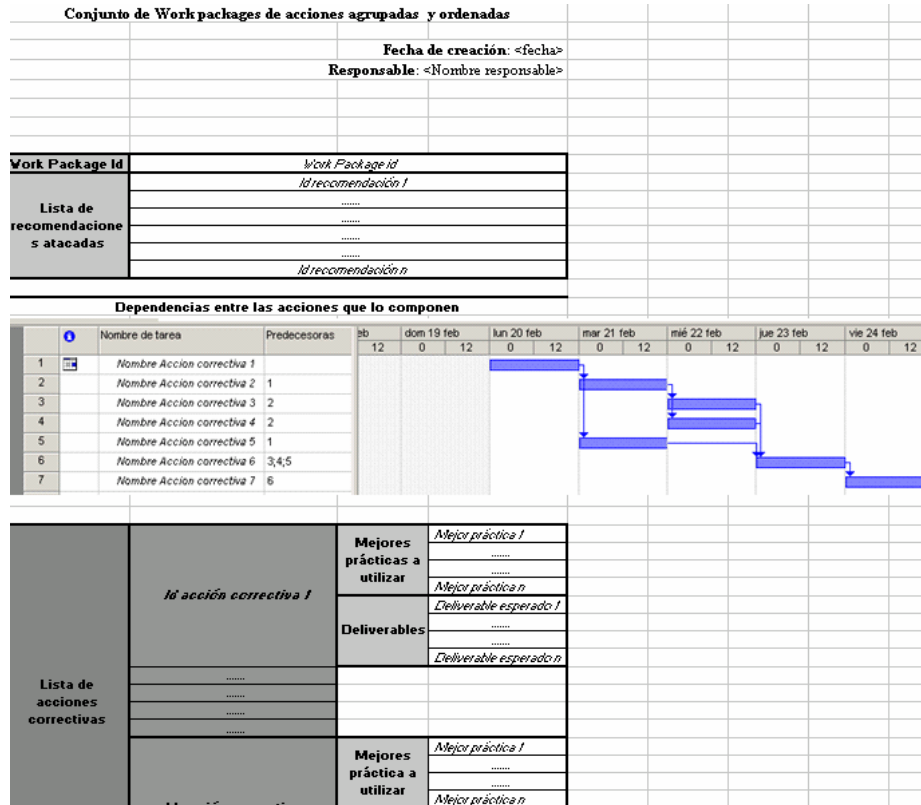


Figura 6.8 Porción de ejemplo / template de resultado tarea 4-5

6.3.6 Tarea 4-6: Análisis de riesgos

Propósito

Complementar con información vinculada a riesgos para cada work package.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> Dato20: Conjunto de: WP x Acción x recurso x dedicación aprox. + WP x herramienta 	<p style="text-align: center;">TASK: Tarea 4-6: Análisis de riesgos</p> <hr/> <p style="text-align: center;">VERIFY</p> <p>Tener complementados los workpackages con riesgos y acciones a tomar ante su ocurrencia. Esta información debe ser validada con los líderes de proyectos involucrados.</p>	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> Dato25: Riesgos + estrategias de manejo
--	---	---

El objetivo de esta tarea es identificar los riesgos potenciales que pueden aparecer durante el transcurso de la implementación del plan en cada uno de los proyectos considerando las posibles futuras consecuencias de decisiones hechas en el presente.

Ejemplos de los posibles riesgos que se pueden presentar

- Financieros
 - Costos y presupuesto
 - Monto del desembolso general
- Deliverables
 - Contenido
 - Mantenimiento
- Estabilidad organizacional
 - Riesgos sobre los productos y la organización si las acciones son tomadas (o no) y las implicancias de realizar cambios oportunos
- Usuarios de los procesos
 - Barreras psicológicas o culturales
 - Experiencia y capacidad para operar bajo nuevos procesos
- Entorno de desarrollo de la mejora de procesos
- Administración del proyecto de mejora de procesos
 - Experiencia del equipo de trabajo
 - Complejidad del proyecto
 - Ventana de tiempo para la implementación
 - Restricciones de planificación
- Staff vinculado a la mejora de procesos
 - Disponibilidad de recursos
- Tecnología o infraestructura
 - Dependencia de otros sistemas
 - Conocimiento de entorno tecnológico
 - Capacidad de mantener y actualizar las tecnologías claves

Con el fin de poder realizar esta tarea se necesita de la participación de los líderes de los proyectos afectados a proyectos involucrados en la implementación de modo que aporten su conocimiento, visión y experiencia en el actual uso del framework de desarrollo.

Esta tarea se compone de un grupo de subtareas, entre las cuales se puede iterar repetidas veces hasta completar la lista de riesgos y estrategias bajo un consenso general. El resultado de esta tarea puede ser desarrollado como en la figura 6.9.

Sub-etapas de la tarea

- **Identificar los riesgos**

Nombrar y describir los riesgos específicos del proyecto / organización para su posterior análisis y decidir bajo que aproximación tratarlos.

Cada riesgo identificado debe ser:

- Catalogado y evaluado por su potencialidad: alta, media o baja.
- Condiciones bajo las cuales se dispara.
- Consecuencias (impacto)

Salida: lista de riesgos a ser analizados

- **Analizar riesgos identificados**

Los riesgos son analizados para ver de que forma cada proyecto se expone a cada riesgo y cual es la importancia de cada uno, estimando el impacto de los mismos.

Para cada riesgo se debe:

- Determinar probabilidad de ocurrencia
- Identificar pedidas si el riesgo ocurre
- Calcular la exposición de cada proyecto a cada riesgo

La exposición de un riesgo se define como el producto de la probabilidad que el riesgo ocurra y la magnitud da las consecuencias por su ocurrencia.

Posibles valores para probabilidad de ocurrencia

- Frecuencia casi nula
- Frecuencia ligera
- Frecuencia moderada
- Frecuencia alta

Posibles valores para magnitud de ocurrencia

- Severidad reducida
- Severidad normal
- Severidad grande
- Severidad catastrófica

Una vez que se tiene los riesgos analizados se deben rankear en base a la probabilidad y magnitud de la ocurrencia.

Este análisis se hace antes de comenzar el proyecto, sin embargo, puede identificarse nuevos riesgos, lo cual implica la actualización de la lista de riesgos.

Salida: conjunto de riesgos claves acordados por los líderes de proyecto para que sean manejados.

- **Definir las acciones para manejar los riesgos**

Establecer un conjunto de acciones realizables para minimizar la exposición a los riesgos identificados y de esa forma asegurar el éxito de los proyectos.

Para cada riesgo clave, realizar:

- Identificar una estrategia de respuesta: aproximación para manejar el riesgo o planes de contingencia
- Estimar el esfuerzo o costo requerido para la acción de manejo del riesgo

La estrategia de respuesta debe:

- Apropiaada a la severidad del riesgo
- Exitosa respecto del factor tiempo
- Realista dentro del contexto del proyecto

Ejemplos de estrategias de respuesta para manejar riesgos

- Aceptar el riesgo: Es apropiado cuando el costo para mitigarlo excede la exposición, y la exposición es aceptable.
- Transferir el riesgo a algún otro (ya sea cliente o partner), o en su defecto compartir el riesgo.
- Financiar y tomar recursos (capacitar o contratar) para reducir la probabilidad que el riesgo se vuelva un problema.
- Financiar y tomar recursos (capacitar o contratar) para reducir la pérdida asociada si el riesgo se vuelve un problema.

Salida: conjunto de riesgos claves correspondidos con acciones o planes de contingencia que estén justificados respecto de los beneficios del proyecto.

• **Controlar los riesgos**

Definir indicadores, disparadores de alertas para la ocurrencia del riesgo, y asegurar mediante acciones que el riesgo este bajo control.

En base a los disparadores y al monitoreo de indicadores, se invoca a las acciones para el manejo de riesgos o al plan de contingencia.

Salida: Asegurarse a futuro que las exposiciones del riesgo están a niveles aceptables.

Risk Id	Descripción del riesgo	Lista de proyectos donde impacta el riesgo	Probabilidad de ocurrencia	Magnitud de ocurrencia	Estrategia de respuesta Id

Estrategia de respuesta Id	Tipo de estrategia	Descripción de estrategia	Lista de responsables

Figura 6.9 Porción de ejemplo / template de análisis de riesgo

6.3.7 Tarea 4-7: Definición de etapas del plan táctico y estrategia de implementación del plan táctico

Propósito

Priorizar los work packages definidos y con toda la información desarrollada armar una el plan táctico de mejoras (Global).

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato09: Resultado del assessment • Dato20: Conjunto de: WP x Acción x recurso x dedicación aprox. + WP x herramienta • Dato25: Riesgos + estrategias de manejo 	<p>TASK: Tarea 4-7 Definición de etapas del plan</p> <hr/> <p>VERIFY Tener desarrollado el plan táctico de mejoras. El contenido del mismo debe ser validado con los líderes de los proyectos involucrados.</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato22: Plan táctico de mejoras (Global) Propuesta
---	---	--

Descomposición de la tarea

- **Priorización de los work packages en base a un criterio para establecer precedencias de implementación.**

El objetivo es consolidar todas las acciones para lograr un único plan de implementación basándose en los work packages definidos.

Mapeo de objetivos organizacionales vs. Work packages

- Cada etapa se corresponderá con un work package en particular.
- Cada work package comprende a un conjunto de acciones correctivas
- Cada acción correctiva esta vinculada a un conjunto de recomendaciones del assessment.
- Cada recomendación del assessment esta ligado a un conjunto de objetivos de la organización / conjunto de practicas.

Por lo tanto:

Una etapa es un conjunto de acciones correctivas expresadas como tareas que guían a un conjunto de prácticas que promueven alcanzar un subconjunto de objetivos de la organización.

Es necesaria y de vital importancia la participación de los líderes para tener una validación de las precedencias de la implementación en base a algún criterio de priorización

Ejemplos de criterios para priorizar las etapas

- En base al tiempo de implementación: Se podrían clasificar en base a si son implementaciones a corto, mediano o largo plazo. Conseguir objetivos a corto plazo produce rápidos resultados para la organización.
- En base a las áreas de proceso que atacan cada uno de los work packages
- En base a la disponibilidad de recursos para ejecutar cada etapa
- En base a la dependencia explícita de deliverables entre work packages

Salida: Una secuencia ordenada de work packages que compondrán la implementación.

- **Definición de la estrategia de implementación del plan táctico**

Esta parte de la tarea apunta a definir la forma en que se va a realizar la implementación sobre pilotos, y la manera en que se van a institucionalizar las etapas. Para ello es necesario definir los siguientes criterios:

- **Número de proyectos pilotos a utilizar:** La cantidad de pilotos a utilizar varia de acuerdo a:
 - Pueden ser uno o n pilotos dependiendo de la complejidad de los proyectos de la organización, por ejemplo diferentes tecnologías, diferentes dominios de aplicación, etc., de modo que todos los pilotos son representativos del pool de proyectos de la organización.
 - Tener varios pilotos para probar la implementación de cada una de las etapas, definiendo el

resultado de los pilotos que es lo que posteriormente se va a institucionalizar (ideal para organizaciones grandes.)

- Utilizar directamente proyectos reales como pilotos para aquellas áreas donde las practicas de las etapas se pueden aplicar.
- **Estrategia de despliegue:** Se refiere a la forma en que se institucionalizará una etapa a través de los proyectos de la organización. Existen dos puntos a considerar en la estrategia:
 - Sobre cuales proyectos de la organización aplicar la mejora de procesos
 - Todos los proyectos nuevos y en curso adhieren a la mejora de procesos.
 - Solo los nuevos proyectos van a adherirse a la mejora de procesos, mientras que los proyectos en curso adhieren según conveniencia (por ejemplo, si el proyecto en curso esta en la etapa de diseño, y el plan tiene una buena estrategia de testeo, entonces ese proyecto implementará las practicas de testeo).
 - Solo los nuevos proyectos van a adherirse a la mejora de procesos, mientras que los proyectos en curso siguen con sus procesos actuales.
 - Forma de implementar la mejora entre los proyectos elegidos
 - Big Bang sobre los proyectos siendo SEPG quien desarrolle el plan operacional de cada proyecto para que se implementen las practicas de las etapas.
 - Big Bang sobre los proyectos, pero cada proyecto se hace responsable de desarrollar su propio plan operacional. Luego QA monitorea que cada uno de los planes operacionales estén alineados al plan táctico.
 - Institucionalizar las etapas del plan en forma secuencial sobre cada uno de los proyectos.

Salida: Estrategia de implementación definida.

- **Integración de implementaciones entre proyectos**

Puede darse el caso que entre implementaciones de proyectos existan secciones coincidentes. Ejemplos de secciones coincidentes a la hora de implementar pueden ser:

- Sección de instalación desarrollo de activos del proceso para las etapas entre proyectos
- Mecanismos de medición de instalación o apalancamientos entre etapas de proyectos.
- Guías de capacitación para involucrados en etapas de proyectos.

Para tales casos se procede a realizar una integración para cada solapamiento entre proyectos, definiendo una única:

- Instalación o desarrollo de activos del proceso
- Instalación de mecanismo de medición
- Desarrollo de guía de capacitación

y luego para los demás proyectos coincidentes se realiza una instanciacion parametrizando para ajustar la sección en común a un determinado proyecto.

Salida: Work packages integrados (factorizados).

- **Desarrollo del plan táctico**

Se reúne toda la información que se fue construyendo gradualmente a lo largo de esta tarea bajo un formato estructurado de plan táctico, como se muestra en la tabla 6.1

Secciones que debe contener el Plan Táctico	Deliverables asociados de apoyo
• Prefacio	A desarrollar como parte de esta subtarea
• Registro de cambios del plan	A desarrollar como parte de esta subtarea
• Propósito	A desarrollar como parte de esta subtarea
• Alcance	A desarrollar como parte de esta subtarea
• Objetivos	Dato01: Lista de objetivos principales
• Documentos relacionados	Dato09: Lista de objetivos principales x rating KPA Organización + Issues no CMM
• Estrategia de implementación	A desarrollar como parte de esta subtarea
• Organización de la mejora de procesos	A desarrollar como parte de esta subtarea
• Etapas que componen el plan	
○ Dependencias entre etapas del plan	A desarrollar como parte de esta subtarea
○ Para cada etapa del plan	
• Objetivos globales de la etapa del plan	Dato20: Conjunto de: WP x Acción x recurso x dedicación aprox. + WP x herramienta
• Herramientas necesarias para esta etapa	
• Deliverables de la etapa del plan	
• Plan de medición de la etapa	
• Dependencias entre tareas de la etapa	
• Para cada tarea de la etapa	
• Mejores prácticas a utilizar	Dato20: Conjunto de: WP x Acción x recurso x dedicación aprox. + WP x herramienta
• Deliverables esperados	
• Factor de éxito	
• Recursos necesarios y dedicación	
• Riesgos asociados al plan y acciones para manejarlos	Dato25: Riesgos + estrategias de manejo
• Suposiciones y restricciones	A desarrollar como parte de esta subtarea

Tabla 6.1 Composición de plan táctico

Salida: En el Anexo 21 se presenta un template con indicaciones para especificar el plan táctico

6.3.8 Tarea 4-8: Obtener compromiso del sponsor

Propósito

Obtener el visto bueno del sponsor (Sr. Manager) respecto de la propuesta presentada como plan táctico.

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato22: Plan táctico de mejoras (Global) Propuesta 	<p>TASK: Tarea 4-8: Obtener compromiso del sponsor</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato22: Plan táctico de mejoras (Global) Preaprobado
	<p>VERIFY</p> <p>Validar alcance y contenido del plan con el sponsor.</p>	

Como objetivo se persigue contar con un plan táctico que sea respaldado por el Sr. Manager que funciona como sponsor para tener su apoyo y visto bueno sobre el plan elaborado.

Si el sponsor hiciera recomendaciones sobre posibles modificaciones se iteraría hacia las primeras sub tareas de elaboración del plan táctico para hacer las modificaciones necesarias hasta conseguir el apoyo del sponsor.

Con el apoyo del sponsor se pretende:

- Obtener conformidad sobre el plan táctico desarrollado.
- Obtener compromiso sobre las acciones y recursos necesarios para el plan
 - Disponibilidad de recursos internos y externos
 - Disponibilidad de herramientas.
 - Etc.
- Si bien en esta tarea del framework de implementación no se abordan cuestiones monetarias, el sponsor será capaz de calcular groseramente cual será la inversión necesaria para implementar el plan en los proyectos involucrados.

Nota: Los temas vinculados con gastos, costos y beneficios serán tratados en la tarea posterior de elaboración del Business Case.

La salida de esta tarea es el plan táctico en sí, pero con la conformidad parcial del sponsor.

Un ejemplo de la salida para esta tarea es presentada en el Anexo 22.

6.3.9 Tarea 4-9: Desarrollar SCMP preliminar

Propósito

Desarrollar una versión mínima del plan de configuración de software (SCMP) para dar soporte a aquellos ítems de configuración necesarios desde el inicio de la implementación.

<p>ENTRY</p> <ul style="list-style-type: none"> • Comienzo de la tarea de planificación 	<p>TASK: Tarea 4-9 Desarrollar SCMP preliminar</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato28: SCMP
	<p>VERIFY</p> <p>Validar completitud del plan, tanto por contenido como por los ítems de configuración a contemplar.</p>	

En esta tarea se produce una versión preliminar porque tiene por objetivo completar la sección de SCI (software configuration identification), particularmente en lo relativo al establecimiento de los baselines y a los procesos de cambio de esos baselines.

Los baselines a incluir son los mínimos necesarios para la puesta en marcha al momento de implementar. Dentro de estos baselines al menos se encuentran:

- Baseline de procedimientos y procesos
- Baseline de documentación relativa a procesos
- Baseline de frameworks de procesos aprobados
- Baseline de mediciones
- Baseline de herramientas

En el anexo 28 se muestra un template para especificar el SCMP, y en la figura 6.10 se muestra la estructura general del plan.

- 1- Introducción**
 - 1-1 Propósito**
 - 1-2 Alcance**
 - 1-3 Términos mas usados**
- 2 Administración del SCM**
 - 2-1 Organización**
 - 2-1.1 SCCB**
 - 2-1.2 SCRIB**
 - 2-1.3 SCM staff**
 - 2-1.4 Administrador SCM**
 - 2-2 Responsabilidades** (dentro del área de SCM)
- 3 Actividades del SCM**
 - 3-1 SCI**
 - 3-1.1 Selección de los CSCIS**
 - 3-1.2 Establecimiento formal de los baselines**
 - 3-1.3 Métodos de identificación**
 - 3-1.4 Procedimientos**
 - 3-2 SCC**
 - 3-2.1 Proceso de cambios de los baselines**
 - 3-2.1.1 Clasificación de cambios**
 - 3-2.1.2 Formularios de solicitud de cambios**
 - 3-2.2 Procedimientos**
 - 3-3 SCSA**
 - 3-3.1 Registros**
 - 3-3.2 Reportes**
 - 3-3.3 Procedimientos**
 - 3-4 SCA**
 - 3-4.1 FCA**
 - 3-4.2 PCA**
 - 3-4.3 Procedimientos**

Figura 6.10 Estructura del SCMP

Capítulo 7

Tarea 5

Desarrollo del Business Case

7.1 Identificación

Tarea 5

7.2 Introducción al capítulo

Las organizaciones hacen inversiones en ingeniería de software para incrementar su negocio, satisfacer los requerimientos de cliente, o mejorar el faltante de productividad y calidad.

Estas inversiones generalmente vienen en forma de adopción de prácticas, herramientas, tecnología, métodos y procesos, educación y capacitación, o adopción de estándares.

Desde el punto de vista de los inversores:

- Desean saber por que deben gastar sus limitados recursos en la propuesta a presentar y no en otra
- Quieren profundizar en sus costos y beneficios
- Quieren estar seguros que la “historia” presentada es creíble y alcanzable

Un BC (Business Case) es una propuesta detallada de inversión que provee un análisis de todos los costos, beneficios y riesgos asociados con la inversión de la implementación del plan táctico de mejoras (global) preaprobado, pero evaluado sobre cada proyecto donde se desea implementar.

Desarrollar el BC es una etapa clave dentro de los procesos del ciclo de vida del proyecto.

EL BC es importante porque provee a quienes toman decisiones con información financiera, competitiva y otras formas de justificación que demuestran que la idea de negocio bajo consideración es conveniente para la organización y que los números que lo rodean tienen un sentido financiero para invertir en talentos, tiempo, dinero y otros recursos requeridos en la propuesta.

Por lo expuesto, a través del BC se debe convencer a los Sr. Managers que la recompensa potencial justifica el costo.

La propuesta será aprobada si se han analizado las opciones adecuadamente y si están asociadas a un plan factible y de bajo riesgo que conduce a lo que se promete.

El BC no es un documento estático. El BC también se utiliza en el proceso de decisión continúa a través del resto de las tareas que componen la implementación, ya que es revisado y revalidado luego que cada etapa del plan táctico se considera cumplida para verificar si los objetivos claves y los beneficios han sido alcanzados.

Por tal motivo el BC se debe mantener consistente con el plan táctico, en el sentido que un cambio en el plan debe impactarse en el documento del BC.

Del mismo modo que se procede con las actualizaciones en los planes, las distintas instancias del BC que se desarrollen deben ser registradas en un baseline creado para tales efectos.

Nota: *El contenido del presente capítulo fue desarrollado en base a investigaciones hechas en las referencias: 74, 75, 76, 77, 78, 79, 80, 83, 84, 99, 100, 101*

7.3 Perfiles necesarios para desarrollar el Business Case

Se deben tener los siguientes conocimientos o habilidades:

- Finanzas
- Administración económica
- Habilidades técnicas
- Habilidades de adquisición
- Haber participado en el desarrollo del plan táctico de mejoras (global)

7.4 Principios para el desarrollo de un Business Case

- Siempre que fuese posible se debe usar la unidad \$ como denominador común. El valor actual o el peso de esos fondos es determinado a lo largo del tiempo usando la base de valor actual.
- Los costos deprimidos (sunk cost) son irrelevantes: todos los costos previos a la toma de decisión de implementar no tienen incidencia en la toma de la decisión en si.
- La decisión de inversión debería reconocer el valor del dinero en el tiempo. El dinero tiene un valor que se incrementa en el tiempo por la presencia de la inflación. Si la inversión no proporciona una tasa de retorno superior a la que ofrece un banco, la propuesta no sería atractiva.
- Las decisiones separables deben ser consideradas separadamente.
- La decisión debe considerar tanto los factores cuantitativos como cualitativos.
- Los riesgos asociados con la decisión deben ser cuantificados si fuese posible. Todas las decisiones están basadas en proyecciones a futuro. Existe incertidumbre o riesgo asociados con las predicciones.
- Contemplar un análisis de riesgos y estrategias para mitigarlos.
- El tiempo asociado con la toma de decisión es crítico.
- El proyecto, su alcance y objetivos se alinean a las políticas y estrategias de la organización
- El alcance del proyecto satisface los objetivos de la organización y entrega los beneficios y salidas delineadas en el BC. La propuesta ofrece una opción apropiada de implementación con costos razonables para su realización.
- Los análisis financieros y de costos son exactos, comprensivos y basados en suposiciones transparentes y robustas; y hay una razón objetiva para creer que el proyecto puede ser entregado según lo planificado.
- Los stakeholders claves son consultados y/o considerados dentro de las posibilidades al momento del desarrollo de la propuesta.

7.5 Estructura de la tarea

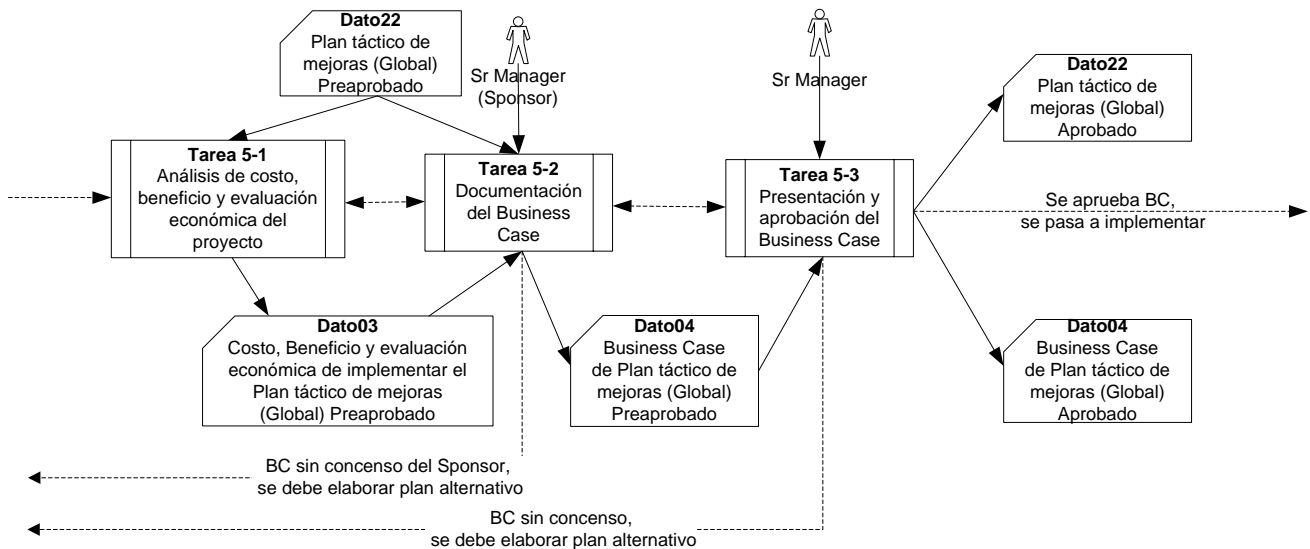


Figura 7.1 Estructura tarea 5

Aclaración: Si bien la tarea del desarrollo del BC figura en la metodología como una tarea específica, en la práctica lo relativo a análisis de costos, análisis de beneficios y evaluación económica del proyecto se hace en forma solapada con la tarea de elaboración del plan táctico de mejoras.

7.5.1 Tarea 5-1: Análisis de costo, beneficio y evaluación económica del proyecto

Propósito

Desarrollar la evaluación económica del proyecto a partir de los costos y beneficios identificados para el Plan táctico de mejoras (Global) Preaprobado.

<p>ENTRY</p> <ul style="list-style-type: none"> Dato22: Plan táctico de mejoras (Global) Preaprobado 	<p>TASK: Tarea 5-1</p> <p>Análisis de costo, beneficio y evaluación económica del proyecto</p>	<p>EXIT</p> <ul style="list-style-type: none"> Dato03: Costo, Beneficio y evaluación económica de implementar el Plan táctico de mejoras (Global) Preaprobado
	<p>VERIFY</p> <p>Validar exhaustivamente la inclusión de todos los costos y beneficios de toda índole. Este análisis debe ser cruzado contra el plan táctico de mejoras una vez finalizado.</p>	

Descomposición de la tarea

- Subtarea Análisis de Costos
- Subtarea Análisis de Beneficios
- Subtarea Evaluación económica del proyecto

Las subtareas a y c deben evaluarse para cada uno de los proyectos donde se desea implementar.

La subtarea c se desarrolla en base a los resultados obtenidos por las subtareas a y b aplicadas a cada proyecto.

Subtarea análisis de Costos

El costo es la consecuencia económica resultante de crear un nuevo y mejorado proceso de software.

$$Costos = \sum_{i=1}^n Costo_i$$

Se deben incluir todos costos asociados con la inversión, desde el inicio hasta el desarrollo e implementación, independientemente de quien vaya a pagarlos.

Tipos de costos a identificar

- Costos fijos directos:** Se corresponden con los gastos de desarrollo e implementación. Estos costos están generalmente bien documentados y son fáciles de determinar, salvo para proyectos que involucren nueva tecnología o nuevo software de aplicación.

Las principales categorías de este tipo de costos son:

- Hardware y periféricos
- Software enlatado o customizado
- Relevamiento inicial de información o conversión de información histórica
- Equipo de telecomunicaciones
- Actualizaciones de software
- Especificaciones de usuario
- Diseño y programación
- Espacio físico de las instalaciones, mobiliario e ítems relacionados
- Capacitación inicial a usuarios
- Costos de transición, tales como correr sistemas en paralelo o conversión de sistemas
- Aseguramiento de calidad y revisiones post implementación.

- **Costos directos:** Son aquellos que suceden a lo largo del ciclo de vida de la implementación. Se deben identificar tanto los costos para operar los activos del proceso, como los de desarrollar o implementar una determinada alternativa.

Las principales categorías de este tipo de costos son:

- Los salarios de staff de IT
- Las actualizaciones y los mantenimientos de software
- Abastecimiento de computadoras
- Soporte a usuarios
- Capacitación permanente
- Revisiones y auditorías

- **Costos indirectos o Costos ocultos:** Por definición es el conjunto de erogaciones relacionado con la producción, salvo las materias primas y la mano de obra directa.

Las principales causas de los costos ocultos son:

- Pérdida inicial de la productividad:
 - Al principio la productividad puede caer mientras los miembros del staff aprenden a trabajar con los nuevos procesos o herramientas.
 - Caída de la productividad del personal altamente productivo por ayudar a los nuevos empleados a resolver sus problemas.
 - Cuando las organizaciones pagan en forma indirecta a profesionales que no son de sistemas para realizar soporte a funciones de sistemas (peer group support).
- Soporte IT a nivel corporación.
- Gastos generales corporativos: La forma de contabilizarlos es en base al porcentaje que representa el staff de cada departamento dentro de la compañía para luego asignar en forma proporcional a cada departamento los gastos generales corporativos.
- Recursos humanos: Se deben identificar todos aquellos costos vinculados a la incorporación y/o movimiento interno de personal en la compañía.

- **Costos del cliente:** Pueden ser directos o indirectos. Un ejemplo de este costo se da en el caso que la compañía realiza un cambio de tecnología para uno de sus productos consumidos por clientes. En el caso que el cliente no cuente con tecnología adecuada para consumir el producto expresado en la nueva tecnología, existiría un costo para el cliente.

- **Costos intangibles:** Algunos ejemplos son:
 - Impacto del incumplimiento de la legislación.
 - Deterioro del conocimiento gerencial.
 - Potenciales costos por litigio debido a la inadecuada administración de la información.

Nota: En el desarrollo de esta tarea se dan ejemplos de costos relativos al uso de SW-CMM como modelo de referencia

SW-CMM incurre en costos para el desarrollo de los procesos que resultan en una mayor productividad y menor mantenimiento.

Principales costos de implementar SW-CMM

- **Costos de procesos para SW-CMM:** Costo de desarrollar políticas y procedimientos para el SW-CMM.
- **Costo de los productos de SW-CMM:** Costo por el desarrollo de documentos, autorizaciones de trabajo, registros, reportes y minutas de reuniones para el SW-CMM.
- **Costos de assessment del SW-CMM:** Costos de planificación, preparación y reuniones vinculadas al assessment.
- **Costos de software del SW-CMM:** Costos del análisis, diseño e implementación del software.
- **Costos de las reuniones para SW-CMM:** Costo de realizar el proceso de inspecciones de software
- **Costos de testeo del SW-CMM:** Costo del testeo de software basados en los defectos vinculados al proceso de inspección de software.
- **Costos de mantenimiento del SW-CMM**

Subtarea análisis de Beneficios

Los beneficios son el valor económico resultante de un nuevo y mejorado proceso de software.

$$\text{Beneficios} = \sum_{i=1}^n \text{Beneficio}_i$$

Beneficios comunes de una implementación de SPI

- Mayor calidad (muchos menos defectos)
- Menor mantenimiento (menos retrabajo)
- Mayor productividad (menores costos de desarrollo)
- Tiempos de ciclos mas rápidos (time to market mas rápidos)
- Mayor valor (mas características en los productos)
- Mayor variedad (mas variaciones en los productos)
- Mayor satisfacción en los clientes (mas contratos)

Tipos de beneficios a analizar

- **Beneficios tangibles (cuantificables):** Son servicios cuantificables o ganancias financieras para la organización. Representan aquellos beneficios que son medidos e idealmente cuantificados bajo una moneda.
- **Beneficios intangibles (no cuantificables):** Son mejoras no cuantificables para la mejora de la organización. Estas son herramientas valiosas para convencer a los Sr. Managers para emprender la propuesta presentada.
- **Beneficios condicionales:** Algunos beneficios pueden depender de eventos fuera del alcance de las opciones definidas. Obtener estos beneficios condicionales generalmente depende de la forma que se reestructura la organización. Los beneficios condicionales necesitan particular atención:
 - Para la evaluación de su certeza
 - Para proveer información a los gerentes sobre la mejor forma de administrar los riesgos asociados con la inversión
 - Para proveer un indicador de su ventana de tiempo para el aumento de los beneficios (asumiendo que los beneficios actuales van a aumentar)

Métodos de análisis de Beneficios

Los métodos para el análisis de beneficios son técnicas para identificar, medir y cuantificar los beneficios del SPI. Dentro de los métodos para medir beneficios encontramos los siguientes ejemplos:

- **Productividad**

Es la medición de que tan rápido y que tanta cantidad de software se produce. También puede ser visto como la cantidad de productos o servicios creados.

La productividad es la proporción, velocidad o capacidad de un proceso de software.

Entonces, la productividad es una medición de que tan rápido y cuantos productos o servicios de software son producidos en un lapso de tiempo (horas, días, meses, duración del proyecto, etc.).

Sin embargo, un problema de software puede tener muchas soluciones diferentes, derivando en casos como que una solución puede ser más eficaz que otra, o más legible que otra, o más mantenible, etc. De esto se desprende que las soluciones se pueden medir con atributos diferentes, y la consecuente insignificancia de comparación entre soluciones.

La productividad esta relacionada con el costo y con el tiempo de ciclo, en el sentido que si se incrementa la productividad se reducen los costos y el tiempo de ciclo. Si se reducen los costos y el tiempo de ciclo se termina reduciendo los costos del ciclo de vida.

Por lo general, las estimaciones de productividad se basan en medir algunos atributos del software y dividir el resultado entre el esfuerzo total requerido por el desarrollo. Existen dos tipos de medidas:

- **Medidas relacionadas con el tamaño:** Se relacionan con el tamaño de alguna salida de una actividad

Ejemplos:

- A) Líneas de código fuente entregadas (la medida más común utilizada)
- B) Numero de instrucciones de código objeto entregado
- C) Numero de paginas de la documentación de un sistema

“A” es una métrica ampliamente usada en la medida de la productividad que se calcula contando el número total de líneas del código fuente que se entrega dividido por el tiempo total de programadores mes requeridos para completar el proyecto. Este tiempo incluye los tiempos de análisis, diseño, codificación, pruebas y documentación.

El problema que posee es que puede haber más de una instrucción por línea, y por lo tanto no habría relación simple entre las instrucciones del programa y las líneas de un listado.

Un ejemplo de cómo medir la productividad es a través del siguiente factor:

$$\text{Productividad} = \frac{\text{Líneas de código}}{\text{Hora}}$$

Ejemplo: Una productividad de 100 líneas de código por hora es una proporción del número total de líneas de código desarrolladas para el número total de horas usadas en el proyecto. Esta productividad es aplicada a cada uno de los roles involucrados en el proyecto (analistas, arquitectos, programadores, testers). Representan la productividad de todos, no solo de los programadores.

- **Medidas relacionadas con la función:** Se relacionan con la funcionalidad total del software entregado. Se expresa en términos de la cantidad de funcionalidad útil producida en un tiempo dado, independizándose del lenguaje de programación.

Ejemplos:

- A) Puntos de función: Se expresa como el número de puntos de función que son implementados por persona – mes. El número total de puntos de función en un programa se calcula estimando las siguientes características del programa:
 - Entradas y salidas
 - Interacciones con el usuario
 - Interfaces externas
 - Archivos usados por el sistema

Cada una de estas características se evalúa de forma individual acorde a su complejidad y se la pondera con un factor de peso.

La suma ponderada de esas cantidades da lugar a los puntos de función no ajustados

$$UFC = \sum (\text{elemento de algún tipo dado}) * (\text{peso})$$

Una vez calculado el UFC, se lo multiplica por un factor de complejidad del proyecto en base que contempla características como: procesamiento distribuido, reutilización, facilidad de instalación, etc.

- B) Puntos de objeto: Son una alternativa a usar con lenguajes 4GL. Los puntos de objeto no son clases de objetos del paradigma de objetos, sino que es una estimación de:
- Numero independiente de pantallas que se despliegan (valoradas por complejidad de 1 a 3)
 - Numero de informes que se generaron (valorados por complejidad de 2 a 8)
 - Numero de módulos en lenguaje 3GL que complementan el código 4GL (vale 10 puntos cada modulo).

- **Densidad de defectos**

Es el número estimado de errores o fallas que todavía residen en el software después que el proyecto de software se ha terminado.

La densidad de defectos es una proporción de los defectos residuales respecto del tamaño del software o líneas de código.

$$\text{Densidad de defectos} = \frac{\text{Defectos}}{\text{Líneas de código}}$$

Ejemplo: Una densidad de 10 defectos por cada 1000 líneas de código, significa que existe una proporción equivalente de errores remanentes en el producto terminado. Los errores pueden estar en los requerimientos, diseño, código, testeos, otros.

La calidad de un producto de software es mayor si el radio es menor.

La densidad de defectos también es una medición de la satisfacción del cliente aplicado a sus necesidades, requerimientos y expectativas, ya que puede interpretarse como que la medición de defectos es la medición del número de requerimientos del usuario que no están satisfechos en el producto de software.

La densidad de defectos no es una medida de la calidad del código, ya que no se limita al código fuente del software, sino que aplica a los requerimientos, diseños, código y testeos.

- **Calidad**

Es una medición de que tan conformes están los clientes en función del número de requerimientos de cliente que un producto o servicio ha alcanzado y satisfecho.

De esta forma, una calidad pobre representa una proporción baja del número de requerimientos de cliente que se han cumplido. Esto se debe a que los defectos son otra forma de decir que los requerimientos no han sido cumplidos.

Los defectos están referidos a toda clase de defectos: requerimientos, diseño, código y testeos.

- **Eficiencia en la eliminación de defectos**

La eficiencia en la eliminación de defectos es la proporción de defectos eliminados respecto de los defectos que se vienen evadiendo y los defectos insertados.

Este es un indicador de que tan bueno o efectivo ha sido el proceso de revisión.

Este indicador también puede ser usado para administrar la calidad a través del ciclo de vida del software, ya que se puede usar como medición para setear objetivos de mejoras en los procesos y productos.

$$EED = \frac{\text{Defectos eliminados}}{\text{Defectos evadidos} + \text{Defectos insertados}} * 100 \%$$

Donde

Defectos eliminados son el número de defectos de software detectados por una actividad de evaluación (Ej. revisiones individuales, procesos de inspección de software, y testing).

Defectos evadidos son el número de defectos que existían antes del cálculo.

Defectos insertados son el número de defectos de software asociados a la actividad actual

Generalmente este indicador es calculado por periodos: entre fases, o entre actividades. Esto se hace para evaluar la efectividad de una fase o de un proceso de software puntual.

- **Esfuerzo de software**

El esfuerzo de software es el número de horas requeridas para el desarrollo de software, generalmente medido en horas-hombre, o meses-hombre.

El esfuerzo de software es el tiempo requerido para analizar, diseñar, desarrollar y testear productos de software.

El esfuerzo de software generalmente no contempla el esfuerzo de mantenimiento, el cual debe ser estimado separadamente.

Existen algunos modelos como los expuestos a continuación, pero por su antigüedad probablemente fueron diseñados para usarse con proyectos de software de millones de líneas de código. Para utilizarlos, quizás deberían ser ajustados.

Fuente	Modelo
Boehm	Meses = 3 * KSLOC ^{12.1}
Walston/Felix	Meses = 2.5 * KSLOC ^{91.0}
Bailey/Basili	Meses = 5 .5 + 73.0 * KSLOC ^{15.1}
Doty	Meses = 288 .5 * KSLOC ^{047.1}

- **Costo total del ciclo de vida**

Es un método para determinar tanto el costo del desarrollo de software como el costo de mantenimiento.

Se considera que el costo de mantenimiento de un proyecto de software es 20 veces mayor al costo de su desarrollo.

La porción del ciclo de vida del software dedicada a mantenimiento es donde se encuentran los mayores beneficios de aplicar los métodos de SPI.

Este costo se construye sobre los principios de densidad de defectos, calidad, eficiencia en la eliminación de defectos y los modelos de eliminación de defectos.

$$TLC = \text{Tamaño} * 10.51 - \text{Horas de inspección} * 99 - \text{Horas de prueba} * 9$$

Donde

Tamaño se refiere al tamaño estimado del software en líneas de código.

La constante 10.51 es una simplificación de la proporción de los costos de mantenimiento y la inserción de defectos.

Las horas de inspección son el número de horas que se planea gastar para realizar el proceso de inspección de software.

Las horas de prueba son el número de horas planeadas para realizar el testeado del software.

Subtarea Evaluación económica del proyecto

Una vez que los costos y los beneficios han sido identificados, se debe realizar una evaluación económica del proyecto.

La evaluación económica del proyecto es la magnitud económica de usar el modelo de referencia para crear nuevos y mejorados procesos de software.

Si bien algunos beneficios pueden no ser cuantificables, también deben ser incluidos en el análisis.

Este análisis se debe hacer desde dos perspectivas:

- Desde el punto de vista de la organización como proveedor de productos o servicios
- Desde el punto de vista del cliente como consumidor del producto servicio desarrollado

Para el caso puntual de SW-CMM la proporción de beneficios respecto de los costos es alta debido al incremento en la productividad y calidad, y a la reducción de los costos de desarrollo y mantenimiento. La mayoría de sus beneficios se deben a la institucionalización de las revisiones por pares o al proceso de inspección de software.

Tips para realizar la evaluación económica del proyecto

- Los principales ahorros de costos provienen de los recursos dedicados. Cuando se desee estimar la salvedad de estos costos, mantenerlos separados al expresar la salvedad en si.
- Evaluar de que forma la organización realizara ahorros en base a los recursos asignados, y estimar cuanto tiempo llevara realizarlos.
- Utilizar técnicas analíticas y modelos financieros
 - Relación Beneficio / Costo (BC/R)
 - Retorno de inversión (ROI)
 - Valor Presente Neto (NPV)
 - Tasa Interna de Retorno (TIR)
- El BC puede describir solo costos totales.
- Asignar un período de análisis: Debido a que la tecnología afecta al ciclo de vida de la inversión, asegurarse que los costos y beneficios de los proyectos se calculan sobre un determinado período de tiempo.
- Utilizar valores expresados en \$ ajustados al valor presente cuando fuese posible

Técnicas analíticas y modelos financieros

Relación beneficio/costo (B/CR)

Es simplemente la relación de los beneficios respecto de los costos.

B/CR mide la magnitud económica de usar un nuevo y mejorado proceso de software.

El valor resultante de la fórmula es cuanto se recibirá por cada peso invertido en el proceso por el uso del método de SPI.

$$BC/R = \frac{Beneficios}{Costos}$$

Retorno de inversión (ROI)

ROI es la proporción de la cantidad de dinero que se ganará al realizar una inversión.

Por ejemplo un ROI del 10% implica que por cada 10\$ invertidos, se ganará 1\$.

ROI es una medición numérica del dinero o beneficio.

$$ROI = \frac{Beneficios - Costo}{Costo} * 100 \%$$

Criterio de conveniencia: Si el ROI es mayor que el ROI mínimo requerido por la organización se acepta, si es menor, la propuesta no sería conveniente.

Desventajas: Se basa en ingresos contables y no en flujos de caja, no tiene en cuenta el momento en el cual se producen los ingresos y los egresos, esto es, le da igual importancia al flujo de caja del primer año que la flujo de caja del último año, en otras palabras, ignora el valor del dinero en el tiempo.

Valor presente neto (NPV)

El NPV representa el Valor Presente de los flujos salientes de caja menos la cantidad de la inversión inicial. El Valor Presente del flujo de caja futuro es calculado utilizando el costo del capital como un factor de descuento.

Por lo tanto NPV es el valor económico del dinero actual evaluado en el futuro teniendo en cuenta la inflación.

$$NPV = VP_{Beneficios} - VP_{Inversiones}$$

Criterio de conveniencia: Si NPV > 0

Ventajas

- Considera el valor del dinero en el tiempo
- Considera la vida útil

Desventajas

- La determinación de la tasa de descuento
- La variación del costo de capital

Tasa interna de retorno (TIR)

La Tasa Interna de Retorno es la tasa de interés que hace la ecuación de la Inversión Inicial (I) con el Valor Presente (PV) de los futuros flujos de caja entrantes. Esto es, a la Tasa Interna de Retorno, I = PV o NPV = 0.

Cuando se calcula el TIR, el NPV se fija en cero y se resuelve para un interés (i). En este caso, el factor de descuento es (1 + i) ya que se desconoce el interés verdadero, solamente se conoce el interés deseado.

$$VP_{Beneficios} = VP_{Inversiones}$$

Criterio de conveniencia: Si TIR > tasa mínima de rendimiento requerida por la organización.

Ventajas

- Considera el valor del dinero en el tiempo
- Considera la vida útil
- Proporciona una tasa en vez de valores monetarios
- No es necesaria la tasa de descuento para su calculo

Desventajas

- Considera que todos los ingresos se deben reinvertir a la misma tasa de TIR
- Difícil de calcular manualmente

Desarrollo de una salida común para esta tarea

En el Anexo03 (ver figura 7.2) se presenta un template para documentar los análisis y evaluaciones realizadas en esta tarea. A continuación se muestra una porción del template:

Costos Indirectos / ocultos		Total Beneficios xxx	
CluO	xxx		
CluOn	xxx		
Total xxx			
Costos del Cliente			
CC	xxx		
CC n	xxx		
Total xxx			
Total Costos xxx			
Costos Intangibles		Beneficios Intangibles	
Costo Intangible 1		Beneficio Intangible 1	
...		...	
Costo Intangible n		Beneficio Intangible n	
Suposiciones para Análisis de Costos o Análisis de Beneficios			
ID	Tipo de Análisis	Descripción suposición	
Resultados de Evaluación económica			
Método de evaluación 1	«Nombre método de evaluación 1»		
Parámetro	Valor	Descripción / comentario	
Resultado método de evaluación	«Valor resultado»	Aplica conveniencia?	Comentarios acerca del resultado

Figura 7.2 Porción de template para análisis y evaluación económica

7.5.2 Tarea 5-2: Documentación del Business Case

Propósito

Construir el BC en base al plan táctico de mejoras (global) preaprobado, a sus riesgos y planes de contingencia, y a la salida de la tarea análisis de costo, beneficio y evaluación económica de implementar el Plan táctico de mejoras (Global) Preaprobado.

En esta etapa se debe preparar toda la documentación para demostrar la relevancia y viabilidad del plan de implementación. Esta documentación es para ser presentada ante un determinado público llamado audiencia.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> • Dato22: Plan táctico de mejoras (Global) Preaprobado • Dato03: Costo, Beneficio y evaluación económica de implementar el plan táctico de mejoras (global) preaprobado 	<p style="text-align: center;">TASK: Tarea 5-2</p> <p style="text-align: center;">Documentación del Business Case</p>	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> • Dato04: Business Case de plan táctico de mejoras (global) preaprobado
	<p style="text-align: center;">VERIFY</p> <p>El sponsor evalúa el Business Case, tanto por lo desarrollado a nivel contenido como por contenido impacto económico en el Plan de Táctico de mejoras.</p>	

Audiencia del BC

Se debe identificar quienes son las personas que toman las decisiones. Puede darse el caso que haya varias audiencias a distintos niveles o a diferentes partes del proceso.

El documento debe ser ajustado a las necesidades, intereses, expectativas y nivel de entendimiento de la audiencia.

Participación del Sponsor en la tarea

El contenido del BC una vez terminado es valorado por el Sr.Manager (Sponsor). Como respuesta a esta valoración, pueden darse las situaciones mostradas en la tabla 7.1:

Resultado evaluación del Sponsor	Acción a tomar
El Sponsor sugiere profundizar el nivel de detalle con otros modelos económicos o técnicas analíticas.	Mejorar el BC sobre el mismo plan táctico y volver a presentarlo
El Sponsor aprueba la propuesta	Se continúa con la secuencia de tareas realizando la presentación del BC al resto de los Srs. Managers (o quien tome las decisiones).
El Sponsor objeta la propuesta presentada	Se deberá retroceder y desarrollar un Plan Táctico opcional utilizando mejores prácticas alternativas para satisfacer las correcciones indicadas.

Tabla 7.1 Acciones a tomar según evaluación del BC por parte del sponsor

El anexo Dato04 provee un posible template de BC a utilizar. A continuación se detalla la información que debe contener cada una de las secciones del BC.

Detalle de información a incluir en el BC

1) Introducción

1-1) Identificación del documento

Se detalla la siguiente información:

- Id del BC
- Versión
- Título
- Organización
- Autor
- Fecha creación
- Id plan táctico de mejoras (global) preaprobado asociado

1-2) Tema

El tema del BC debería ser definido en base a los objetivos propuestos.

Necesita describir en forma total y precisa tanto la acción principal como las acciones relacionadas.

También debe contener en forma puntual el alcance y los objetivos que hay detrás de cada acción.

Además de los objetivos organizacionales relevados para la implementación se pueden plantear los siguientes objetivos:

- Objetivos de negocio
- Objetivos financieros
- Objetivos funcionales
- Objetivos operacionales

1-3) Propósito

Si bien el título del documento puede sugerir una aproximación general, o apuntar a una acción central, puede darse el caso que el mismo no mencione el propósito del BC.

El propósito debe indicar:

- Para que será usado el BC (por ejemplo para la toma de decisiones)
- Como será utilizado el BC

1-4) Resumen ejecutivo

Debe ser conciso, cuidadosamente preparado y formateado debido a que parte de la audiencia leerá solo esta sección.

El éxito en los emprendimientos de software es una función proporcional de contenidos y empaquetamiento.

Contenidos a considerar en esta sección:

- Una descripción de la propuesta y del concepto de negocio
- La recomendación, junto con cualquier calificación importante
- Un claro vínculo hacia los objetivos del negocio y hacia la sección involucrada dentro del plan táctico de mejoras (global) preaprobado.
- Mencionar cualquier información relacionada que sea muy apropiada a la recomendación, tales como riesgos y restricciones. El detalle de cada una de estas será desarrollado en las siguientes secciones
- Una explicación de por que se considera apropiado implementar la opción presentada

2) Métodos y supuestos

La credibilidad y efectividad del BC dependen en gran parte de la habilidad del autor para explicar el origen de los datos y resultados derivados.

2-1) Métodos financieros

Se debe notificar a la audiencia en forma anticipada que los análisis posteriores se harán utilizando técnicas analíticas y modelos financieros.

Por tal motivo, es conveniente enumerarlos en esta sección.

2-2) Suposiciones

Sean de la clase que sean las suposiciones (predicción, simplificación, clasificación) sin importar que tan necesarias, obvias o apropiadas parezcan, necesitan ser explícitamente registradas en el BC sin importar que la audiencia interprete lo mismo que el autor del BC para cada suposición.

Ejemplos de suposiciones críticas:

- Drivers de ganancia
- Costos operativos y de capital
- Factores ambientales y sociales
- Restricciones financieras
- Disponibilidad y expertise de recursos
- Tópicos políticos, marcos regulatorios, legislativos y otros aspectos importantes que puedan entorpecer la propuesta

2-3) Alcance

El alcance es el rango que contempla el BC entre distintas dimensiones.

La definición del alcance le indica a la audiencia de quienes serán los costos y beneficios, y de donde proviene el impacto financiero.

Algunas de las dimensiones posibles son:

- Tiempo
 - ¿Cuándo comienza y cuándo termina el periodo de análisis?
 - ¿El análisis está sincronizado con el año fiscal o con el año calendario?
- Ubicación geográfica
- Organización o función

- ¿El análisis cubre a un grupo, departamento o división específica?
- ¿El análisis se aplica a una determinada función?
- ¿El análisis se aplica a determinado personal y a otros no?

- Tecnología

2-4) Costos y beneficios contemplados

De acuerdo a lo desarrollado en la tarea Análisis de costo, beneficio y evaluación económica del proyecto, se vuelcan en esta sección lo referente al análisis de costos y análisis de beneficios, sin tener en cuenta para ambos casos los que son intangibles.

2-5) Métodos y fuentes de datos

Describir las fuentes de datos y métodos utilizados para asignar valores a los costos y beneficios.

Tips para el desarrollo:

- Incluir descripciones cuando no se pueda asegurar que la audiencia conozca de forma no ambigua como se desarrollaron los datos.
- Si no son obvios o bien conocidos, describir los métodos para asignar valores a costos y beneficios.

3) Impacto en el negocio

3-1) Resultados financieros

La pieza central del BC es la aplicación de los modelos financieros y las técnicas analíticas.

Esta sección se completa en base al resultado de la tarea Análisis de costo, beneficio y evaluación económica del proyecto, particularmente con lo referente a evaluación económica del proyecto.

Para cada modelo financiero o técnicas analíticas se debe detallar:

- Los parámetros usados con comentarios para cada uno que demuestren el valor asignado
- El valor obtenido del método, complementado con comentarios
- Descripción de conveniencia de la propuesta respecto del valor obtenido
- Cuando fuese posible, complementar el resultado con un gráfico

3-2) Resultados no financieros

Estos son impactos que no pueden ser cuantificados en forma monetaria.

Esta sección se completa con lo desarrollado en la tarea Análisis de costo, beneficio y evaluación económica del proyecto, particularmente para el análisis de costos y beneficios que contemplan los casos intangibles.

Estos resultados al no ser expresados en términos económicos, sería ideal determinar formas de observarlos y verificarlos.

4) Riesgos

Esta sección se completa con lo contemplado en la sección de riesgos del plan táctico de mejoras (global) preaprobado.

En esa sección se enumeran los distintos riesgos, probabilidades y acciones a tomar.

5) Conclusiones

Se organiza en base a los objetivos detallados en la sección “tema del BC”, ya que las conclusiones deben hacer foco en la contribución que realiza la propuesta para cada uno de estos objetivos en base a los análisis realizados.

Nota: Si los proyectos considerados trabajan con distintas tecnologías o múltiples ambientes, existen dos posibles estrategias para desarrollarles el BC:

- Unificar métricas entre proyectos (por ejemplo puntos de función)
- Hacer un BC para cada uno de los ambientes

7.5.3 Tarea 5-3: Presentación y aprobación del Business Case

Propósito

Presentar ante la audiencia el BC desarrollado buscando la aprobación de la propuesta.

ENTRY	TASK: Tarea 5-3	EXIT
<ul style="list-style-type: none"> Dato04: Business Case de Plan táctico de mejoras (Global) Preaprobado 	Presentación y aprobación del Business Case	<ul style="list-style-type: none"> Dato22: Plan táctico de mejoras (Global) Aprobado Dato04: Business Case de Plan táctico de mejoras (Global) Aprobado
	<p>VERIFY</p> <p>Validar que se cuenta con un conjunto de elementos que posibilitan la presentación del caso en su conjunto con material que fundamente cada análisis contemplado.</p>	

La propuesta de inversión competirá por obtener la atención de quienes toman decisiones y la organización en su conjunto. Obtener la atención de esta audiencia es crucial para que la propuesta pueda ser promovida, expuesta y vendida.

Muchas de las propuestas son rechazadas como consecuencia del no entendimiento de la misma por parte de la audiencia.

La audiencia puede ver a la propuesta como algo que “solo afecta a los centros de costos”. De esta forma no se consigue comunicar los beneficios potenciales del BC.

Tips a considerar para presentar el BC

- Es más efectivo presentar el BC en términos a largo plazo.
- Para no sobrecargar de información a la audiencia, hacer la presentación bajo el formato de una historia debido a los siguientes beneficios:
 - Las historias se componen de imágenes sensoriales que cautivan activamente a la audiencia.
 - Las historias promueven la participación de la audiencia dentro del marco de referencia.
 - Las historias permiten captar inmediatamente la idea.
 - Las historias son más fáciles de recordar que los hechos o definiciones abstractas.
- Si existe más de una clase de audiencia, identificarlas y preparar estrategias de comunicación y mensajes para cada una.
- Utilizar el mismo lenguaje y argumentos que utiliza la audiencia, para demostrarle con sus propios términos que la propuesta es beneficiosa.
- Se debe tener claridad en los mensajes: Se debe pensar, hablar y trabajar hasta conseguir claridad a nivel estratégico y operacional.
- Ayudar a la audiencia que toma decisiones educándolos (si fuese necesario) para que comprendan que están tomando una decisión de inversión.
- Es vital promover la confianza en la audiencia respecto que la implementación se realizara según lo que se presenta en el BC, sin tener importantes sobrepuestos.

Resultado de la tarea

Pueden sucederse tres resultados distintos luego de la presentación. Los resultados y las acciones a tomar se presentan en la tabla 7.2:

Resultado presentación	Acción a tomar
La audiencia sugiere profundizar con otros modelos económicos o técnicas analíticas.	Mejorar el BC sobre el mismo Plan Táctico y volver a presentarlo
La audiencia aprueba la propuesta	EL BC y el Plan Táctico cambian a un estado de aprobado y el paso siguiente es comenzar el proceso de implementación en si.
La audiencia objeta la propuesta presentada	Se deberá retroceder y desarrollar un Plan Táctico opcional utilizando mejores prácticas alternativas para satisfacer las correcciones indicadas.

Tabla 7.2 Acciones a tomar según evaluación del BC a nivel organización

Capítulo 8

Tarea 6

Implementación

8.1 Identificación

Tarea 6

8.2 Introducción al capítulo

El objetivo de esta tarea es realizar las implementaciones de las etapas en los proyectos bajando el nivel de abstracción definido en el plan táctico hasta el nivel de los proyectos siguiendo la estrategia de implementación definida (guías de customización, número de pilotos, forma de despliegue y otros).

Para eso cada proyecto desarrolla una versión customizada del plan táctico utilizando guías para ajustarse dicho plan.

Consideraciones sobre la implementación:

- El proyecto esta dividido en etapas de implementación y cada etapa se compone de actividades.
- Cada actividad se compone de:
 - Objetivos perseguidos
 - Recomendaciones del assessment involucradas
 - El cuerpo de la acción en sí
 - Recursos
 - Deliverables
 - Otros
- Por lo tanto la etapa termina conteniendo la reunión de cada clase de los componentes de sus actividades (objetivos, indicadores, otros).

Introducción a los pasos para la implementación

- Cada etapa se implementa en al menos un piloto de acuerdo a la estrategia de implementación.
- Una vez instalada una etapa en el piloto, se procede a:
 - Apalancar el uso de dicha etapa en el piloto.
 - Instalar la siguiente etapa-piloto que corresponda.
- Cuando una etapa en el piloto esta implementada (finaliza el apalancamiento de la etapa en los pilotos) se pasa a institucionalizar esa etapa en el resto de los proyectos de la organización.
- Los pasos para institucionalizar etapas que ya han sido implementadas en los pilotos son los mismos que para implementar la etapa en los pilotos.
 - Instalar la etapa en los proyectos participantes de la institucionalización.
 - Apalancar el uso de dicha etapa en los proyecto.
 - Volver al paso de instalar la etapa en los proyectos participantes de la institucionalización.
- De esta forma se van probando etapas en los pilotos para que luego cada etapa sea institucionalizada.
- A medida que se implementa se va refinando (si hiciera falta) el plan táctico de mejoras. Esto puede derivar en modificaciones sobre:
 - El Business Case
 - La estrategia de implementación

La idea de utilizar pilotos antes de la institucionalización de una etapa se debe a que son útiles para testear la solución tanto a nivel proceso de implementación (viabilidad del plan táctico) como a nivel de la solución propuesta antes de llevar a cabo la institucionalización de la misma

A medida que se avanza en el plan de implementación van a haber prácticas de etapas muy institucionalizadas y las últimas prácticas de las etapas instaladas van a estar en vías de ser institucionalizadas.

Nota: El contenido del presente capítulo fue desarrollado en base a investigaciones hechas en las referencias:12, 14, 15, 28, 34, 61, 73, 88, 91, 92

8.3 Estructura de la tarea

Debido a que en la estructura existen dos tareas que se encargan de realizar la implementación de etapas, ya sea en los proyectos pilotos como en aquellos proyectos participantes de la institucionalización, se considera que las dos tareas son una especialización de la tarea “implementar etapa”, siendo la diferencia entre ambas el tipo de proyectos sobre el que se implementa.

Para una mejor lectura del diseño se plantea la estructura en dos niveles de abstracción según se presenta en las figuras 8.1 y 8.2.

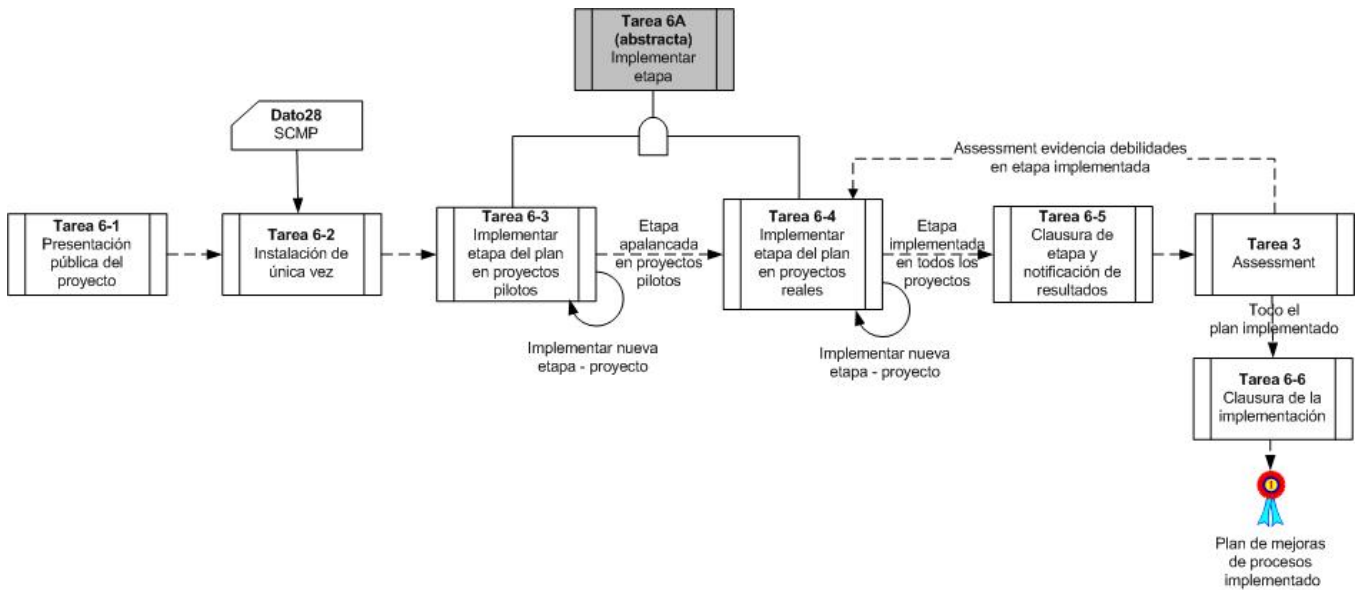


Figura 8.1 Estructura de la tarea implementación a gran nivel de abstracción

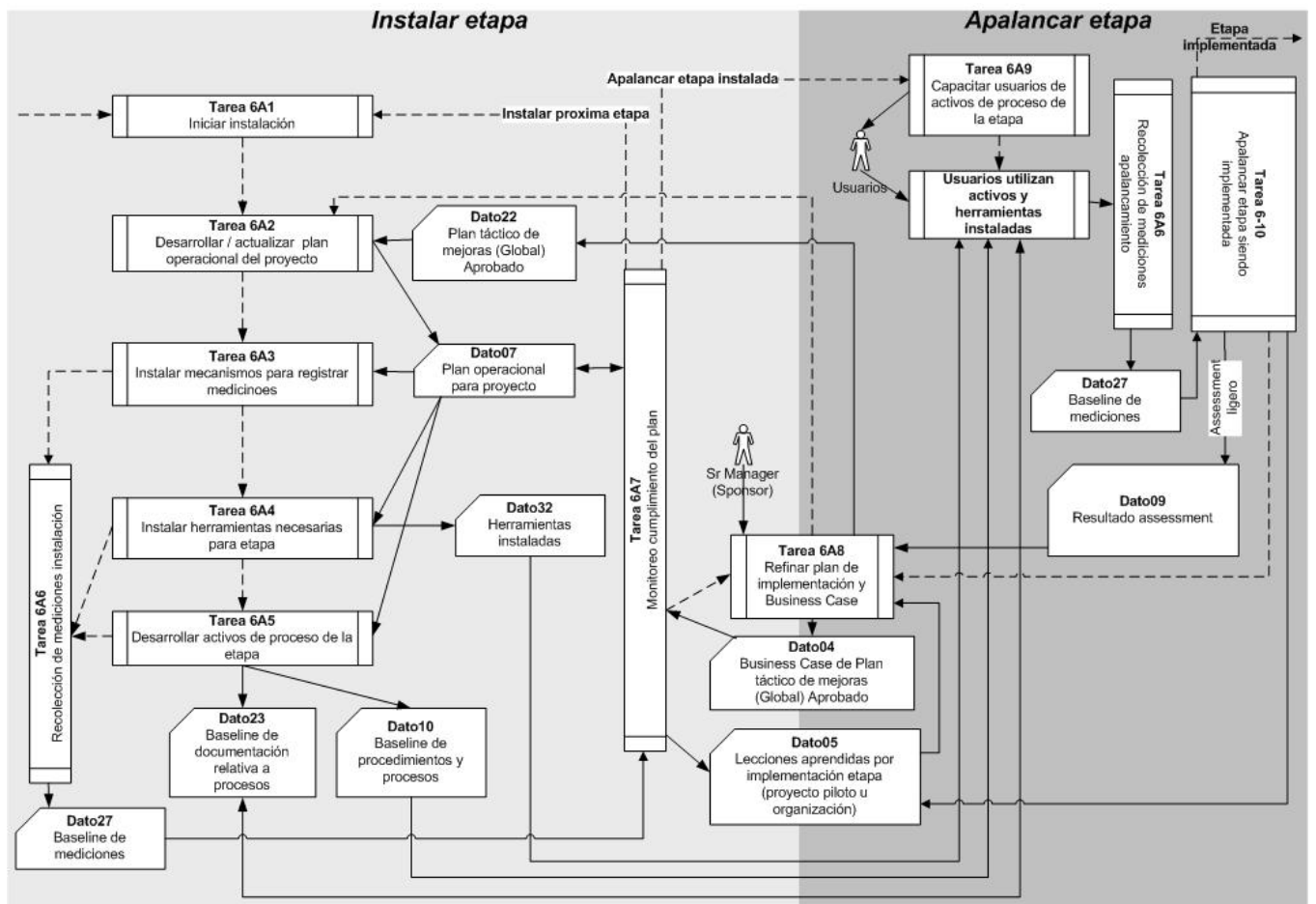


Figura 8.2 Estructura de explosión tarea implementar etapa

8.3.1 Tarea 6.1: Presentación pública del proyecto

Propósito

El propósito de esta tarea es generar una plataforma de entendimiento y cooperación en todo lo concerniente a la ejecución del resto del proyecto.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> Dato22: Plan táctico de mejoras (Global) Aprobado 	<p style="text-align: center;">TASK: Tarea 6-1</p> <p style="text-align: center;">Presentación pública del proyecto</p>	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> Notificación del plan a involucrados en el proyecto de implementación
	<p style="text-align: center;">VERIFY</p> <p>Las características claves de la implementación (contexto, visión, estrategia, otros) son incluidas y dejadas en claro a todos los participantes de la presentación.</p>	

Se presenta públicamente el emprendimiento a realizar que se contrasta con el plan táctico aprobado por la gerencia.

Para llevar a cabo la presentación es necesario identificar a todos los involucrados de los distintos niveles organizacionales.

Los participantes de la presentación deben estar “maduros” y debe existir un clima que permita mantener presión tanto en los participantes de la presentación como en toda la estructura de la organización del proyecto.

La presentación no debe ser realizada demasiado tarde, porque se perderían los efectos beneficiosos de esta tarea.

Con los participantes identificados se deberá desarrollar una agenda de contenidos que debe contener al menos los siguientes puntos:

Definir el contexto

- Establecer el contexto significa ser muy claro respecto del lugar que ocupa el cambio dentro de la estrategia de negocio de la organización.

Visión del proyecto:

- Mostrar cual es la dirección a tomar por la organización y cual es el punto al que se va a llegar.
- Contar cuales serán los futuros desarrollos y cambios esperados en la organización.

Motivación para el cambio

- Describir los beneficios esperados para la organización.
- Describir los beneficios esperados para el personal involucrado.

Estrategia

- Mostrar la estrategia de implementación para alcanzar los cambios a través de los objetivos organizaciones, el plan táctico y los planes operacionales a desarrollar.

Participación

- Informar cuales serán los proyectos involucrados.
- Informar principales asignaciones de roles.

Pasos próximos a seguir....

Los resultados que produce la tarea de arranque son:

- El enfoque sistemático del arranque hace que todos los participantes sean concientes de los resultados y de la naturaleza del proceso.
- Se crea un compromiso entre todas las partes interesadas para conseguir los objetivos del proyecto.
- Se clarifica el proyecto al estudiar y analizar su planificación en todos los aspectos.

8.3.2 Tarea 6-2: Instalación de única vez

ENTRY	TASK: Tarea 6-2	EXIT
<ul style="list-style-type: none"> • Dato28: SCMP • Dato22: Plan táctico de mejoras (Global) Aprobado 	<p>Instalación de única vez</p> <p>VERIFY</p> <p>Todos los baselines y herramientas involucrados fueron instalados, configurados y están listos para ser utilizados.</p>	<ul style="list-style-type: none"> • SEPG funcionando • Herramientas elementales instaladas • Baselines elementales instalados

Esta etapa se ejecuta una única vez como primera tarea dentro del plan de implementación y su contenido se despliega en la figura 8.3. Se implementa a nivel organización, lo que implica que su implementación es común y única a todos los proyectos sobre los cuales se va a implementar.

Las entradas a esta tarea son:

- El pan táctico de mejoras para:
 - Establecer el SEPG
 - Instalar las herramientas elementales
- SCMP
 - Instalación de los baselines elementales

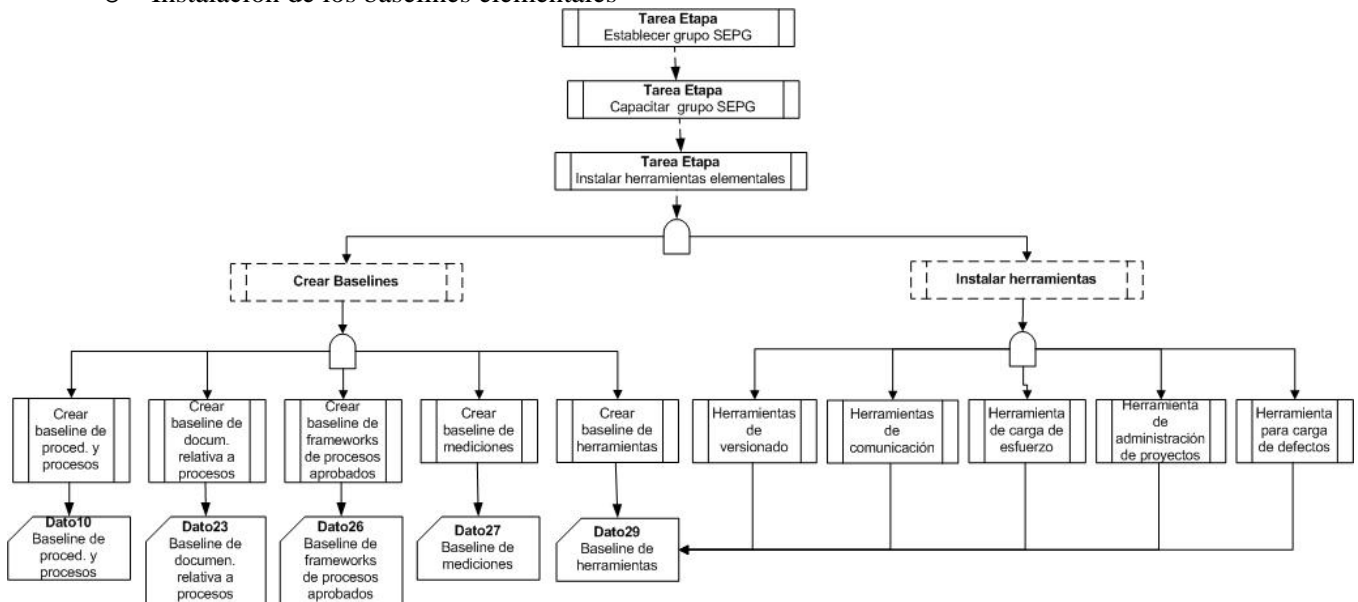


Figura 8.3 Estructura tarea para soporte a mejora de procesos

Instalar herramientas elementales: Entre las indispensables se encuentran:

- Herramientas de versionado.
- Herramientas o mecanismo de comunicación.
- Herramienta para la carga de esfuerzo por persona versus actividad
- Herramienta de administración de proyectos.
- Herramienta para registrar defectos.

Crear baseline de procedimientos y procesos

El objetivo de este baseline es recolectar y hacer disponible información relativa a los procesos y procedimientos a utilizar en la organización.

Crear baseline de documentación relativa a procesos

Esta baseline contiene ejemplos de documentos y porciones de documentos que son esperados para que se utilicen en proyectos futuros. Los ejemplos pueden abarcar activos de proceso tales como procesos de software definidos para proyectos, estándares, procedimientos, planes de desarrollo de software, planes de medición, material de capacitación, otros.

Crear baseline de frameworks de procesos aprobados

Debido a que la organización puede producir software para una variedad de clientes o usuarios, trabajar con un único framework puede no ser apropiado para todas las situaciones. Como consecuencia, es necesario disponer de un baseline de frameworks aprobados para utilizar en los proyectos de la organización.

Crear baseline de mediciones

El objetivo de este baseline es recolectar y hacer disponible información relativa a las mediciones que se van a recolectar durante la tarea de implementación, tanto para la instalación de la etapa como para el apalancamiento. Puede darse el caso que los baselines estén implícitamente definidos y mantenidos en las herramientas que proporcionan las mediciones.

8.3.3 Tarea 6A (abstracta): Implementar etapa

El propósito de esta tarea es realizar la implementación de una etapa en un proyecto.

Las tareas donde se realizan implementaciones de etapas son:

- Tarea 6-3: Implementar etapa del plan en proyectos pilotos
- Tarea 6-4: Implementar e institucionalizar etapa del plan en proyectos restantes

Debido a que ambas tareas son idénticas excepto por el entorno sobre el cual corren (las primeras corren sobre proyectos pilotos, mientras que la segunda es para implementar sobre proyectos participantes de la institucionalización).

La estructura de esta tarea es mostrada en la sección 7.3.

La tarea de implementación por si misma puede ser dividida en dos secciones:

- La que trata la instalación de la etapa
- El complemento de la anterior que es el apalancamiento de la etapa

Ambas secciones están compuestas de múltiples tareas.

En los siguientes párrafos se tratan los objetivos de cada una de estas dos secciones, como así también los deliverables producidos y las tareas que las componen.

8.3.3.1 Tarea 6A - sección de instalación de la etapa

El propósito de esta sección es:

- Desarrollar el plan operacional de implementación para el proyecto basándose en el plan táctico
- En base al plan operacional se desarrollan las siguientes actividades:
 - Capacitación a integrantes de la instalación
 - Instalación de mecanismos de captura de mediciones
 - Instalación de herramientas necesarias para la etapa a implementar
 - Desarrollo de activos de proceso relativos a la etapa a implementar

Con el propósito de realizar una correcta instalación, se monitorea la ejecución de la instalación teniendo en cuenta el plan operacional versus las mediciones registradas a través de los mecanismos de captura de mediciones. Si se observa que existe un desvío (en forma favorable o negativa) respecto del plan operacional, se procede a la modificación del plan táctico para que el plan contemple los casos observados en la instalación. De esta forma la organización se beneficia en términos de prevención de defectos y eficiencia para eliminarlos.

Al finalizar la instalación de una etapa se tiene una etapa instalada en un determinado proyecto.

Los pasos a seguir al finalizar la instalación de una etapa en un proyecto son:

- Próxima instalación de una etapa dentro del mismo entorno (proyectos pilotos o proyectos participantes de la institucionalización). Pueden darse dos casos:
 - Instalar la misma etapa en los demás proyectos dentro del mismo entorno
 - Instalar la próxima etapa dentro del plan operacional dentro del mismo entorno
- Se procede a desarrollar actividades de apalancamiento sobre la etapa del proyecto recién instalado.

Nota: Para aquellos proyectos que compartan secciones de instalación se debe proceder de acuerdo a la sección “Integración de implementaciones entre proyectos” dentro de la tarea de desarrollo del plan de acción, donde la instalación común se realiza una única vez, y las demás son instancias de la anterior.

8.3.3.1.1 Tarea 6A1: Iniciar instalación

<p>ENTRY</p> <ul style="list-style-type: none"> • Inicio de instalación de etapa en proyecto 	<p>TASK: Tarea 6A1 Iniciar instalación</p> <hr/> <p>VERIFY</p> <ul style="list-style-type: none"> • Documento que constate la verificación de las precondiciones. • Los involucrados en la etapa son notificados sobre las características de la implementación de la etapa. 	<p>EXIT</p> <ul style="list-style-type: none"> • Requisitos validados para la ejecución de la instalación y presentación de etapa a involucrados
---	--	---

Se descompone en dos subtareas

- **Verificación de precondiciones para comenzar etapa**
 Debe validarse la disponibilidad de los requisitos necesarios para la realización de la etapa. Entre los ítems a validar figuran:
 - Fondos económicos suficientes para el desarrollo de la etapa.
 - Recursos necesarios comprometidos (tanto staff como bienes materiales: hardware, software, mobiliarios, otros)

Nota: Ante la ausencia de alguna de las precondiciones necesarias para comenzar la implementación de la etapa, se debe proceder de acuerdo la respuesta para este tipo de riesgo desarrollada en la sección plan de contingencia del plan táctico.

- **Presentación de etapa a involucrados**
 Deben ser participados a esta presentación todos los afectados por el cambio, todos los miembros involucrados en la etapa actual del proyecto afectado.
 Los puntos a tratar en la presentación son:
 - Objetivos a lograr por la etapa
 - Beneficios esperados por la etapa a nivel proyecto y a nivel organizacional
 - Descripción de deliverables esperados
 - Mostrar la estrategia de implementación para alcanzar los cambios a través de los objetivos organizaciones, el plan táctico y los planes operacionales a desarrollar
 - Asignación de roles al personal
 - Describir el plan de capacitación
 - Publicación de los principales riesgos de la etapa

Dentro de los beneficios esperados para esta subtaska figuran:

- El enfoque sistemático del arranque hace que todos los participantes sean concientes de los resultados y de la naturaleza del proceso.
- Se crea un compromiso entre todas las partes interesadas para conseguir los objetivos del proyecto.
- Se establece una matriz de tareas versus responsabilidades
- Se establece consenso respecto de la metodología y el sistema de información a utilizar en la implementación del la etapa.

8.3.3.1.2 Tarea 6A2: Desarrollar / actualizar plan operacional del proyecto

<p>ENTRY</p> <ul style="list-style-type: none"> • Requisitos validados para la ejecución de la instalación y presentación de etapa a involucrados • Dato22: Plan táctico de mejoras (Global) Aprobado 	<p>TASK: Tarea 6A2</p> <p>Desarrollar / actualizar plan operacional del proyecto</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato07: Plan operacional para proyecto
	<p>VERIFY</p> <p>Validar que el plan operacional:</p> <ul style="list-style-type: none"> • Se alinea con el plan táctico de mejoras • Consistencia según los líderes de proyectos involucrados. 	

Es la tarea que registrará cuales acciones deben ser seguidas durante la implementación de la etapa.

Los planes operacionales son instancias particulares para un proyecto creadas a partir del plan táctico de mejoras. El plan operacional contempla los resultados del assessment para el proyecto sobre el que se esta implementando. De esta forma el plan operacional refleja las características particulares del proyecto sobre como implementar el plan táctico en un proyecto en particular.

De esta forma el plan operacional del proyecto satisface el punto de vista táctico con visión a lo estratégico.

La forma de desarrollar los planes operacionales puede variar entre las siguientes opciones:

- El desarrollo de los planes operacionales pueden ser responsabilidad de cada uno de los proyectos respecto del plan táctico.
- Los planes operacionales para todos los proyectos pueden ser desarrollados por el SEPG.

8.3.3.1.3 Tarea 6A3: Instalar mecanismos para registrar mediciones

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato07: Plan operacional para proyecto 	<p>TASK: Tarea 6A3</p> <p>Instalar mecanismos para registrar mediciones</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Mecanismos para la recolección de mediciones de instalación instalados y funcionando • Mecanismos para la recolección de mediciones de apalancamiento instalados
	<p>VERIFY</p> <ul style="list-style-type: none"> • Mecanismos para la recolección de mediciones de instalación instalados y funcionando • Mecanismos para la recolección de mediciones de apalancamiento instalados 	

De acuerdo a lo establecido en la sección plan de medición para la etapa actual dentro del plan operacional, se instalan los mecanismos para registrar mediciones.

Los mecanismos a instalar deben contemplar dos conjunto de métricas:

- **Métricas que se utilizan en la tarea de instalación de la etapa** (ver 6.3.4.1)

Algunos ejemplos de las mediciones a tomar para la instalación son:

- Esfuerzo para la instalación / desarrollo de activos de proceso de la etapa.
- Defectos producidos al instalar o desarrollar activos de proceso.
- Recursos utilizados en el plan (dinero, tiempo, staff, recursos críticos, otros).
- Progreso en la instalación respecto del estimado.

Los mecanismos de recolección de mediciones de instalación son configurados y se dejan funcionando al finalizar la presente tarea.

- **Métricas que se utilizan en la tarea apalancamiento de la etapa** (ver 6.3.4.2)

- Proporción de proyectos que adhieren a los procedimientos sobre el total de proyectos
- Medición de efectividad del proceso.

- Mejora proporcional del tiempo de entrega al cliente desde la implementación de los nuevos procesos
- Variación en quejas por defectos.
- Variación en proporción de horas extra trabajadas por el staff.
- Financieras: Costos y Valor ganado (Earned Value).
- Esfuerzo de utilizar los activos de proceso de la etapa.
- Defectos por uso de los activos del proceso.
- Volatilidad de procesos y políticas ya definidas.

Los mecanismos de recolección de mediciones de apalancamiento sólo son instalados para que posteriormente en la implementación sean puestos en funcionamiento.

Nota: Debido a que los mecanismos de medición entre etapas de proyectos puede ser coincidentes, se debe proceder de acuerdo a la sección “Integración de implementaciones entre proyectos” dentro de la tarea de desarrollo del plan de acción, instalando una única vez el mecanismo para un proyecto e instanciando ese mecanismo para los demás proyectos coincidentes.

8.3.3.1.4 Tarea 6A4: Instalar herramientas necesarias para etapa

<p align="center">ENTRY</p> <ul style="list-style-type: none"> • Dato07: Plan operacional para proyecto 	<p align="center">TASK: Tarea 6A4</p> <p align="center">Instalar herramientas necesarias para etapa</p>	<p align="center">EXIT</p> <ul style="list-style-type: none"> • Dato32: Herramientas instaladas
	<p align="center">VERIFY</p> <p align="center">Las herramientas están instaladas, configuradas y listas para ser utilizadas</p>	

Esta tarea persigue la instalación (o desarrollo) de todas aquellas herramientas que son imprescindibles para la implementación de la etapa.

Cada instalación de una herramienta comprende las siguientes actividades:

- Desarrollo o instalación de la herramienta en si misma, acompañado del test de la misma.
- CM debe incluir la herramienta dentro del baseline de productos.
- Desarrollar un instructivo para dar capacitación sobre el uso de la herramienta.
- Poner las herramientas a disposición para su utilización.

8.3.3.1.5 Tarea 6A5: Desarrollar activos de proceso de la etapa

<p>ENTRY</p> <ul style="list-style-type: none"> • Dato07: Plan operacional para proyecto 	<p>TASK: Tarea 6A5</p> <p>Desarrollar activos de proceso de la etapa</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato23: Alimentación del Baseline de documentación relativa a procesos • Dato10: Alimentación del Baseline de procedimientos y procesos
	<p>VERIFY</p> <ul style="list-style-type: none"> • Los procesos son desarrollados de acuerdo al plan operacional del proyecto • Se genera documentación para cada activo desarrollado de acuerdo a guías de desarrollo. • Los activos son agregados a los baselines correspondientes 	

Los grupos de desarrollo establecidos y capacitados desarrollan los activos del proceso especificados en el plan operacional.

Dependiendo de la etapa y del proyecto, los activos del proceso a desarrollar son:

- Desarrollo de políticas organizacionales
- Desarrollo de procedimientos
- Desarrollo de planes
- Desarrollo de guías y criterios

Todos los activos de proceso que se desarrollen deberán ser testeados de acuerdo a lo descrito en el plan operacional; para luego incorporarlos a su baseline correspondiente.

Nota común para las tareas 6A3, 6A4 y 6A5 : Durante el desarrollo de estas tareas de acuerdo al plan operacional y bajo la supervisión del SEPG, pueden surgir diferencias o acciones a tomar no contempladas en el plan. Para esos casos, las tareas van poblando el repositorio de lecciones aprendidas por implementación.

8.3.3.1.6 Tarea 6A6: Recolección de mediciones instalación

<p>ENTRY</p> <ul style="list-style-type: none"> • Tarea 6A4: Instalar herramientas necesarias para etapa • Tarea 6A5: Desarrollar activos de proceso de la etapa 	<p>TASK: Tarea 6A6</p> <p>Recolección de mediciones instalación</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato27: Alimentación de Baseline de mediciones
	<p>VERIFY</p> <p>Los mecanismos de recolección de medidas están funcionando y registran resultados reales.</p>	

A medida que se van desarrollando las tareas de instalación y desarrollo de activos de proceso, se van tomando automáticamente las mediciones correspondientes de acuerdo a lo establecido en el plan y a lo instalado en la tarea 6A3.

Las mediciones registradas serán la entrada para poder realizar el monitoreo del cumplimiento del plan operacional.

8.3.3.1.7 Tarea 6A7: Monitoreo cumplimiento del plan

ENTRY	TASK: Tarea 6A7 Monitoreo cumplimiento del plan	EXIT
<ul style="list-style-type: none"> • Dato07: Plan operacional para proyecto • Dato27: Baseline de mediciones • Dato04: Business Case de Plan táctico de mejoras (Global) Aprobado 	<p style="text-align: center;">VERIFY</p> <ul style="list-style-type: none"> • Se utiliza el Plan operacional del proyecto como fuente de validación para cruzar contra las mediciones registradas. • Todas las lecciones aprendidas efectivamente son registradas. 	<ul style="list-style-type: none"> • Dato05: Lecciones aprendidas por implementación etapa (proyecto piloto u organización)

El desarrollo de esta tarea se alinea con las actividades del área de proceso de seguimiento del proyecto.

Su objetivo es:

- Verificar que el proyecto sigue el plan operacional
- Asegurar que los objetivos de mejoras están siendo alcanzados
- A medida que surjan actualizaciones sobre el plan táctico se deberá alinear el plan operacional a la nueva versión.

Se revisa el progreso de las actividades monitoreando y tomando acciones sobre las mediciones registradas por la tarea recolección de mediciones instalación. Algunos ejemplos de esas mediciones son:

- Esfuerzo para la instalación / desarrollo de activos de proceso de la etapa (ver 6.3.4.1.1)
- Defectos producidos al instalar o desarrollar activos de proceso (ver 6.3.4.1.2)
- Recursos utilizados respecto del plan (dinero, tiempo, staff, recursos críticos, otros) (ver 6.3.4.1.3)
- Progreso en la instalación respecto del estimado a realizar (ver 6.3.4.1.4)

Problemas que pueden surgir del monitoreo

- **Problemas de performance**
 - Problemas técnicos inesperados
 - Insuficientes recursos técnicos
 - Se presentan dificultades técnicas insuperables
 - Problemas de rendimiento o calidad
 - Adelantos tecnológicos afectan al proyecto
- **Problemas de costo**
 - Dificultades técnicas requieren mas recursos
 - Estimaciones iniciales fueron muy bajas
 - Presupuesto inadecuado
- **Problemas de tiempo**
 - Las dificultades técnicas llevan mas tiempo de lo planeado para resolverse
 - Las estimaciones de tiempo iniciales fueron muy optimistas
 - Las entradas de materiales, staff o equipos no estuvieron disponibles a tiempo
- **Problemas de planificación**
 - Se presentan pequeños trabajos o ajustes necesarios durante la ejecución del proyecto. Estos problemas se llaman issues y pueden ser numerosos y en consecuencia pueden retrasar el proyecto. Los issues deben ser seguidos en el proyecto donde se detectan hasta ser resueltos para luego ser incorporados como lección aprendida y una eventual posterior incorporación al plan táctico para que los demás proyectos consideren ese issue.

En la figura 8.4 se muestra un template donde volcar los issues encontrados

Issue log id	Issue				Creación		Estado			Severidad			Solucion			Reasignación		
	Proyecto	Tipo proyecto	Ubicación / identificación	Descripción	Autor	Fecha	Abierto	En curso	Reasignado	Cerrado	Muy importante	Importante	Cosmética	Responsable	Fecha cierre	Comentarios	Fecha	Nuevo responsable

Figura 8.4 Tabla para seguimiento de issues de implementación

Durante el transcurso de la ejecución de esta tarea se van registrando aquellas lecciones aprendidas de la instalación (sean problemas o no) en el repositorio de lecciones aprendidas.

La ejecución del monitoreo del cumplimiento del plan puede derivar en alguno de los siguientes resultados:

Acción 1: Se observan desviaciones de las mediciones registradas respecto de las contempladas en el plan para la sección plan de mediciones. En este caso, se invoca directamente a la tarea “Refinar plan de implementación y Business Case” para que se hagan las correcciones pertinentes de acuerdo a la ejecución de la instalación de la etapa en el proyecto.

- Si la tarea “Refinar plan de implementación y Business Case” identifica que es un problema local al proyecto, entonces localmente se modifica el plan operacional.
- Si la tarea Refinar plan de implementación y Business Case identifica que no es un problema local del proyecto, entonces esta hace las modificaciones sobre el plan táctico y el Business Case

Acción 2: La instalación de la etapa en el proyecto termina satisfactoriamente de acuerdo al plan operacional. Luego, se procede a:

Acción 2-1: Se detienen los mecanismos para la recolección de mediciones de instalación para la etapa recién instalada del proyecto involucrado.

Acción 2-2: Se ponen en funcionamiento los mecanismos para la recolección de mediciones de apalancamiento para la etapa recién instalada para el proyecto involucrado. Acto seguido se procede al desarrollo de las actividades correspondientes a la sección de apalancamiento.

Acción 2-3: Si el proyecto donde se terminó la instalación no es el último proyecto dentro del mismo entorno (proyectos pilotos o proyectos participantes de la institucionalización) en el que se instala la presente etapa, se pasa a instalar la misma etapa en otro proyecto dentro del mismo entorno (proyectos pilotos o proyectos participantes de la institucionalización) según corresponda en el plan táctico.

Acción 2-4: Si el proyecto donde se terminó la instalación es el último proyecto dentro del mismo entorno (proyectos pilotos o proyectos participantes de la institucionalización) en el que se instala la presente etapa, se pasa a instalar la próxima etapa en otro proyecto dentro del mismo entorno (proyectos pilotos o proyectos participantes de la institucionalización) según corresponda en el plan táctico.

8.3.3.2 Tarea 6A8 – Refinar plan de implementación y Business Case

Propósito

Debido a que las soluciones nunca trabajan totalmente como se planean es necesario ajustar los planes y el Business Case a medida que se van implementando las etapas en los proyectos de la organización.

<p style="text-align: center;">ENTRY</p> <ul style="list-style-type: none"> • Dato05: Lecciones aprendidas por implementación etapa (proyecto piloto u organización) • Dato09: Resultado del assessment 	<p style="text-align: center;">TASK: Tarea 6A8</p> <p style="text-align: center;">Refinar plan de implementación y Business Case</p> <hr/> <p style="text-align: center;">VERIFY</p> <p>Para el plan Táctico de mejoras y el Business Case:</p> <ul style="list-style-type: none"> • Son refinados ante el suceso de eventos de cambios • Se mantienen consistentes entre si. • Los cambios son aceptados por la alta gerencia, minimamente con el sponsor del proyecto. 	<p style="text-align: center;">EXIT</p> <ul style="list-style-type: none"> • Dato04: Business Case de plan táctico de mejoras (global) aprobado • Dato22: Plan táctico de mejoras (global) aprobado
--	---	--

Esta tarea no pertenece exclusivamente a la sección de instalación ni a la de apalancamiento, sino que es útil para ambas secciones.

El disparador de esta tarea es que las mediciones registradas por la instalación o el apalancamiento difieren de lo esperado por el plan.

La entrada para desarrollar el refinamiento del plan táctico son las lecciones aprendidas por implementar etapa en proyecto, o el resultado del assessment para el caso en que al realizar el assessment alguno de los proyectos demuestran tener debilidades respecto de la etapa implementada.

Independientemente de cuales sean los cambios introducidos, las modificaciones hechas al plan deben ser acompañadas de modificación apropiadas al Business Case.

Los tipos de modificaciones a realizar pueden ser:

- Modificar la estrategias de implementación del plan táctico
 - Número de pilotos
 - Forma de desplegar las etapas
- Mantener la estrategia de implementación y modificar aspectos de las etapas del plan táctico
 - Planificación de la etapa
 - Recursos necesarios
 - Deliverables a producir
 - Indicadores a utilizar
 - Mejores prácticas
 - Descripción de actividades
 - Riesgos y estrategias de manejo
- Una combinación de ambas.

Las acciones a llevar a cabo son dependientes de lo observado en las lecciones aprendidas. Sin embargo, sea cual sea la modificación a realizar, el impacto del cambio se manifiesta dentro de un conjunto de escenarios. Los posibles escenarios de cambios y estado de implementación a conciliar son mostrados en la figura 8.5.

	Proyectos pilotos						Proyectos participantes de la institucionalización					
	Instalación			Apalancamiento			Instalación			Apalancamiento		
	Realizada	Realizándose	Por realizar	Realizada	Realizándose	Por realizar	Realizada	Realizándose	Por realizar	Realizada	Realizándose	Por realizar
Cambio en estrategia de implementación												
Cambio en etapas del plan táctico												

Figura 8.5 Escenarios de cambios según estado de de implementación

Cualquier modificación que se haga sobre el plan táctico o sobre el Business Case debe ser incorporado a su baseline correspondiente.

8.3.3.3 Tarea 6A - Sección de apalancamiento de la etapa

El objetivo de esta sección es realizar ajustes sobre la etapa que se ha instalado y que esta implementándose. Para cumplir este objetivo, los usuarios utilizan los activos del proceso, y en base a ese uso se recolectan mediciones para ver en que medida la etapa que se esta implementado cumple con sus objetivos. Tanto las mediciones como la forma de evaluar el cumplimiento de los objetivos son desarrolladas dentro de las tareas que componen esta sección.

8.3.3.3.1 Tarea 6A9: Capacitar usuarios de activos del proceso de la etapa

<p>ENTRY</p> <ul style="list-style-type: none"> Etapa instalada íntegramente para un proyecto 	<p>TASK: Tarea 6A9</p> <p>Capacitar usuarios de activos de proceso</p>	<p>EXIT</p> <ul style="list-style-type: none"> Los integrantes del proyecto que participan de la etapa instalada están capacitados para utilizar los activos de proceso de la etapa Desarrollo de guías de capacitación de la etapa
	<p>VERIFY</p> <p>Mediante las guías desarrolladas, se capacitan y evalúan a los participantes para verificar la incorporación de los temas impartidos.</p>	

El propósito de esta tarea es capacitar a los usuarios acerca de los activos de proceso instalados o desarrollados durante la sección de instalación.

La falta de capacitación (que puede llevar a lo sumo algunas horas) puede convertirse en una barrera importante para conseguir la exitosa implementación.

Es conveniente que no pasen más de 2 semanas entre la finalización de la capacitación y la puesta en práctica de los conocimientos impartidos.

Para llevar a cabo la capacitación se deben utilizar guías de capacitación, y registrar información sobre la capacitación brindada:

- Audiencia (alumnos) de la capacitación
- Preparación necesaria para participar
- Objetivos del entrenamiento
- Dedicación necesaria para la capacitación
- Criterios para determinar la completitud satisfactoria de los alumnos
- Registro del desempeño de los participantes

Las guías de capacitación deben ser incorporadas a un repositorio para que puedan ser usadas por otros proyectos al apalancar una etapa con idénticas características. Esto surge en el marco de lo contemplado dentro del plan táctico: pueden coincidir las guías de capacitación de la misma etapa para distintos proyectos, y ese aspecto está definido en la sección “Integración de implementaciones entre proyectos” dentro de la tarea de desarrollo del plan de acción.

8.3.3.3.2 Tarea 6A10: Apalancar etapa siendo implementada

<p>ENTRY</p> <ul style="list-style-type: none"> Dato27: Baseline de mediciones 	<p>TASK: Tarea 6A10</p> <p>Apalancar etapa siendo implementada</p>	<p>EXIT</p> <ul style="list-style-type: none"> Dato05: Lecciones aprendidas por implementación etapa (proyecto piloto u organización) Dato09: Lista de objetivos principales x rating KPA Organización + Issues no CMM
	<p>VERIFY</p> <ul style="list-style-type: none"> Impartir el soporte necesario basándose en el análisis de las mediciones registradas versus los objetivos perseguidos por el plan. Registrar oportunamente las lecciones que se van aprendiendo. 	

Apalancar y aprender de las experiencias de los procesos, junto con la administración cuantitativa de los procesos son considerados factores muy importantes dentro de los modelos de referencia.

Esta tarea provee la oportunidad formal para comparar la solución que se está implementando respecto de los objetivos esperados.

Como resultado macro de esta actividad, los procesos y las estructuras organizacionales pueden ser modificados para reflejar las lecciones aprendidas y tener la ventaja de realizar optimizaciones potenciales.

Apalancar una etapa de un proyecto implica que para una etapa de un proyecto ya instalada se debe realizar un ciclo de operaciones (*) durante un periodo de tiempo (**) hasta que se demuestre que las prácticas / objetivos atacados por la etapa bajo apalancamiento son alcanzados (***)

El párrafo anterior hace una serie de llamadas que son los aspectos fundamentales del apalancamiento:

- Ciclo de operaciones del apalancamiento (*)
- Período de tiempo bajo el que se lleva a cabo el apalancamiento (**)
- Detección que las prácticas / objetivos de la etapa bajo apalancamiento son alcanzados (***)

En las siguientes secciones se tratan cada uno de ellos

Ciclo de operaciones del apalancamiento

El ciclo de operaciones comprende los pasos mostrados en la figura 8.6:

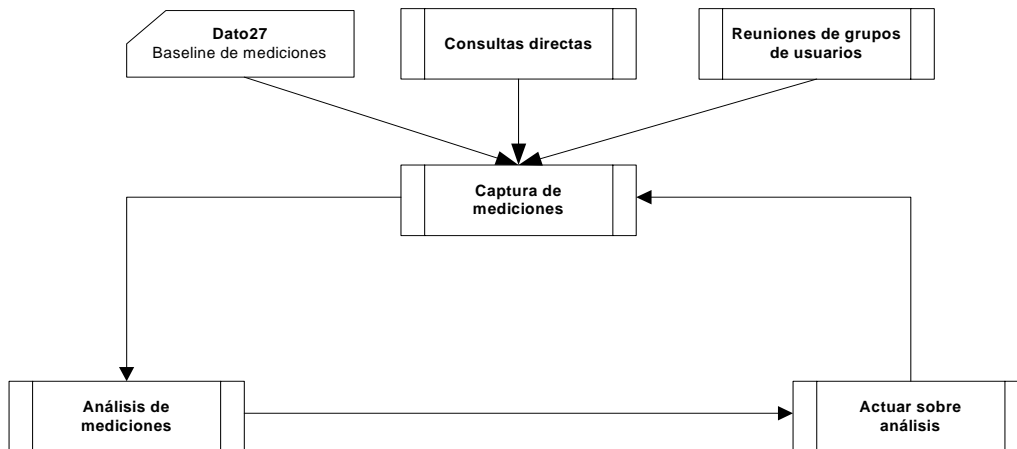


Figura 8.6 Ciclo de operaciones de apalancamiento

- **Operación 1: Captura de mediciones para realizar el apalancamiento:** Comprende la captura de las mediciones tomadas por los mecanismos instalados durante la etapa de instalación, y puestos en funcionamiento al finalizar dicha sección.

Si bien el aprendizaje y la mejora de procesos es siempre posible de lograr, la identificación de éxitos y fallas y la determinación de sus causas resulta más fácil y menos subjetivo si se dispone de información cuantitativa de la performance de los procesos del proyecto.

Entre las mediciones a recolectar sistemáticamente se encuentran (ver 5.3.4.2)

- Proporción de proyectos que adhieren a los procedimientos sobre el total de proyectos
- Medición de efectividad del proceso
- Mejora proporcional del tiempo de entrega al cliente desde la implementación de los nuevos procesos
- Variación en quejas por defectos
- Variación en proporción de horas extra trabajadas por el staff
- Financieras: Costos y Valor ganado (Earned Value)
- Esfuerzo de utilizar los activos de proceso de la etapa
- Defectos por uso de los activos de proceso
- Volatilidad de procesos y políticas definidas

Otras fuentes de información de mediciones:

- Consultas directas
- Reuniones de grupos de usuarios

- **Operación 2: Análisis de mediciones:** Se realiza un análisis de las mediciones recolectadas con el fin de comprender la situación real del proyecto utilizando las prácticas de la etapa.

La capacidad de la organización para monitorear los procesos del proyecto y aplicar controles apropiados para asegurar el éxito está en función de la posibilidad que los objetivos del proyecto estén expresados en términos cuantitativos, y que la información cualitativa del progreso del proyecto esté disponible.

El análisis de las mediciones debe ser ejecutado a determinada frecuencia debido a que su objetivo principal es brindar información para la toma de acciones correctivas y preventivas a tiempo.

Existen dos posturas acerca de la frecuencia en que se realiza el análisis de las mediciones:

- Al alcanzar hitos dentro de la planificación
- Realizar un análisis en base a eventos: Los eventos pueden ser completar determinada tarea. El análisis evalúa cuando la tarea se realizó adecuadamente y determina si se deben tomar acciones correctivas para la tarea. Este tipo de análisis provee una mejor granularidad de control.

Dentro de los análisis a desarrollar figuran:

- Efectividad del proceso
- Tiempos de entrega al cliente
- Financieros
- De que forma se cumplen o no los propósitos deseados.
- Que cosas trabajaron bien y cuales no
- Reducción de las no conformidades
- Reducción de errores
- Seguimiento del proceso
- Funcionamiento de las prácticas
- Utilidad de las prácticas para el proyecto
- Obtención de objetivos perseguidos por las mejoras contempladas en la etapa
- Verificar cumplimiento del circuito de calidad
- Tiempo promedio para corregir defectos

Nota: Cada resultado de análisis puede ser expresado mediante un reporte

Esquema de análisis de las mediciones

Posibles razones	Acciones a considerar
Las mediciones actuales son menores a las estimadas utilizando los límites permitidos	
Razón 1	Conjunto de posibles acciones para razón 1
...	...
Razón n	Conjunto de posibles acciones para razón n
Las mediciones actuales son mayores a las estimadas utilizando los límites permitidos	
Razón 1	Conjunto de posibles acciones para razón 1
...	...
Razón k	Conjunto de posibles acciones para razón k

Tabla 8.1 Tabla de análisis de mediciones registradas

- **Operación 3: Actuar sobre resultados de análisis:** Algunas de las decisiones a tomar pueden ser:

- Extender la planificación para mantener la calidad
- Agregar recursos para mantener la planificación
- Cambiar los procesos para mejorar la performance
- Reubicar recursos para soportar las actividades claves
- Invocar planes de contingencia
- Identificar aspectos que se pueden hacer más efectivos o eficientemente
- Mantener o cambiar la estrategia de institucionalizar las etapas

Nota: Todas las acciones enumeradas anteriormente se manifiestan directamente como lecciones aprendidas, que a su vez derivan en la invocación de la tarea “Refinar plan de acción y Business Case”.

Período de tiempo bajo el que se lleva a cabo el apalancamiento

Debe pasar un determinado lapso de tiempo hasta que se considera institucionalizada. Ese periodo es necesario para afianzar, para ajustar y para capturar la evidencia. Hay que tener siempre en cuenta la magnitud del cambio que se está realizando en la organización: se están haciendo cambios culturales, de forma de trabajar, etc. y eso lleva tiempo para internalizarlo.

El tiempo de apalancamiento es muy dependiente del estado de la organización, de las prácticas actuales que tengan, etc.

Realizar una estimación del tiempo necesario para realizar el apalancamiento debe estar basado en al menos las siguientes variables:

- Resultado del assessment para el proyecto donde se esta implementando
 - Formalidad de los procesos actuales
 - Uso de herramientas de automatización
 - Formación de recursos humanos
 - Otros.
- Áreas clave que ataca la etapa siendo implementada
- Tiempo necesario para pasar al nivel que comprende las áreas claves atacadas por la etapa

Para el caso de SW-CMM las mediciones reportas por el SEI muestran una estadística sobre los valores actuales de la industria para avanzar de nivel, según se muestra en la tabla 8.2.

	Nro. de empresas que componen el universo	Número de meses para avanzar de nivel		
		Percentil 25	Mediana	Percentil 75
Avanzar del nivel 1 al 2	175	13	19	40,75
Avanzar del nivel 2 al 3	262	12,5	19	28,5

Tabla 8.2 Cantidad de meses para avanzar de nivel en SW-CMM

Detección que las prácticas / objetivos de la etapa bajo apalancamiento son alcanzados

La forma de determinarlo es a través de un assessment ligero (cuestionario o encuesta que no debe ser extenso) para evaluar la capacidad del proyecto al implementar la etapa.

Los objetivos y prácticas de la etapa se consideran institucionalizados cuando:

- Poseen la infraestructura y mecanismos de realimentación necesarios para soportar la persistencia y la consistencia de los procesos y procedimientos institucionalizados por la etapa.
- Facilidad para poder cambiar los procesos y procedimientos cuando es apropiado.
- Existe para los procesos un alto grado de repetitibilidad, estandarización, efectividad, durabilidad y alto grado de control.
- Existen prácticas que permitan comunicar los procesos (que los procesos estén definidos, documentados y comprendidos).

Todo lo anterior se consigue si las para las características comunes (compromisos para el desarrollo, habilidades, medición y análisis y verificación de la implementación) de cada una de las áreas de proceso incluidas en la etapa son parte de la cultura organizacional.

Los pasos a seguir una vez detectados que los objetivos y prácticas de la etapa fueron alcanzados, son:

- Detener los mecanismos de recolección de mediciones de apalancamiento para el conjunto etapa y proyecto apalancados.
- Validar que los demás proyectos hayan alcanzado los objetivos y las prácticas de la etapa para poder pasar a la tarea de clausura de etapa y notificación de resultados.

8.3.4 Tarea 6-5: Clausura de etapa y notificación de resultados

Propósito

Esta tarea manifiesta la entrada en operación normal de todo lo implementado para una etapa en particular.

<p>ENTRY</p> <ul style="list-style-type: none"> • Todos los proyectos participantes de la institucionalización tienen implementada una nueva etapa 	<p>TASK: Tarea 6-5</p> <p>Clausura de etapa y notificación de resultados</p>	<p>EXIT</p> <ul style="list-style-type: none"> • Dato05: Lecciones aprendidas por implementación etapa (proyecto piloto u organización)
	<p>VERIFY</p> <ul style="list-style-type: none"> • La clausura de la etapa se desarrollo totalmente de acuerdo a lo contemplado en el plan. • Se registraron las lecciones aprendidas por la implementación de la etapa en todos los proyectos participantes. 	

La clausura de la etapa abarca las siguientes actividades:

- Quitar de inversión o transferencia de recursos a otros sistemas
- Preparar evaluaciones personales de desempeño
- Auditar los cambios finales y costos
- Notificación oficial a los Sr. Managers que la etapa ha sido completada
- Terminar las operaciones on-site
- Redistribuir personal, material, equipo y otros recursos hacia proyectos o ubicaciones apropiadas

8.3.5 Tarea 3: Assessment

Propósito

Realizar un assessment integral como el descrito en la tarea 3 para ver las debilidades y las fortalezas de la organización una vez finalizada la institucionalización de las etapas del plan en los proyectos involucrados.

El detalle de esta tarea puede ser consultado en el capítulo 3.

Como resultado de la ejecución del assessment, se van a tener las siguientes opciones:

- Se comprueba que los proyectos tienen todavía ciertas debilidades importantes respecto de las prácticas y objetivos esperados. Puede haber 2 posibles escenarios respecto de los pasos a seguir:
 - Si los problemas son puntualmente sobre proyectos, la acción a tomar es volver a la tarea “Institucionalizar etapa del plan” haciendo ajustes sobre el plan operacional de los proyectos detectados con debilidades.
 - Si los problemas son comunes entre los proyectos, la acción a tomar es volver a la tarea “Institucionalizar etapa del plan” haciendo ajustes sobre el plan táctico para que todos los proyectos implementen las modificaciones ajustando cada plan operacional.
- No se detectan debilidades y el assessment demuestra haber alcanzado los objetivos y estar utilizando correctamente las prácticas. En este caso, el paso a seguir es ejecutar la tarea “Clausura de la implementación”.

8.3.6 Tarea 6-6: Clausura de la implementación

Propósito

La clausura de la implementación no apunta a la liberación de recursos como ocurre en la clausura de la etapa, sino a hacer un análisis sobre el proyecto terminado.

<p>ENTRY</p> <p>Assessment demostró haber alcanzado los objetivos del plan</p>	<p>TASK: Tarea 6-6</p> <p>Clausura de la implementación</p>	<p>EXIT</p> <ul style="list-style-type: none"> Dato05: Lecciones aprendidas por implementación etapa (proyecto piloto u organización)
	<p>VERIFY</p> <p>Generar como lección aprendida el análisis integral de la implementación de todas las etapas en todos los proyectos.</p>	

El objetivo de esta etapa es examinar el proyecto, identificando que cosas anduvieron bien y cuales no dentro del ámbito de la planificación.

El resultado de esta tarea es un conjunto de resultados expresados como una exhaustiva lista de recomendaciones de mejoras que son registradas en el repositorio de seguimiento de implementación de proyecto x etapa.

El mecanismo para llevarlo a cabo es a través de un método estructurado de postmortem del proyecto. Las principales características del método son:

- Proveer una estructura a los involucrados en la implementación para que provean sus recomendaciones acerca de su participación en el proyecto.
- Realizar dos reuniones de una o dos horas con un intermedio para analizar las recomendaciones formuladas.
- La salida es casi una captura directa de las notas del encuentro, minimizando el esfuerzo necesario.
- El contenido de la agenda es:
 - Establecer un gráfico de línea de tiempo de los principales eventos o actividades, poniendo fecha al inicio y al final del proyecto. Los participantes discuten la ocurrencia de los principales eventos y actividades para poder agregarlos a la línea de tiempo. Este provee un marco de referencia común para la discusión.
 - Cada participante identifica las áreas más importantes para discutir.
 - Considerar de a un área de discusión y utilizar la técnica de discusión brainstorming. Cada participante identifica que cosas fueron positivas y cuales no. Esto se repite hasta que todas las ideas hayan sido expresadas.
 - Refinar la lista generada comparando e identificando las principales fortalezas y debilidades. Para cada entrada se generan recomendaciones.

Conclusiones

Se pudo desarrollar un framework de mejoras de procesos que:

- Es independiente del modelo de referencia para la mejora de procesos de software.
- Proveer de un framework detallado al nivel de tareas que deben ser llevadas a cabo para conducir la mejora.
- Contemplar los aspectos económicos como parte del procesos de aceptación de la propuesta de mejora
- Utilizar variables que afectan directamente a la implementación:
 - Objetivos perseguidos por la organización para la mejora
 - Tipo de software desarrollado por la organización
 - Marcos de proceso
- Incluir en el marco de la mejora de procesos de software aspectos de la organización vinculados a:
 - Recursos Humanos
 - Aspectos culturales.

Luego de desarrollar el trabajo, se pueden enunciar las siguientes “lecciones aprendidas”

- No existe una implementación adecuada de un modelo de madurez si no se centra y conduce por los drivers del negocio (estrategias, problemas a resolver, reclamos de mercados, etc.)
- Debe existir una relación entre la visión del negocio y la implementación.
- Independientemente de la implementación, los objetivos del negocio dirigen mejoras en aquellas áreas que ellos mejoran.
- Una implementación debe considerar no solo los aspectos técnicos, sino también los aspectos culturales y de recursos humanos.
- El uso del framework no es prescriptivo, en los siguientes aspectos:
 - No debe ser considerado una bala de plata.
 - Depende del buen sentido común
 - Depende de la aplicación profesional.
- No necesariamente tener un proceso formal es condición suficiente para asegurar la calidad, pero si contribuye a mejorar las variables económicas y los objetivos del proyecto.
- Tomando una organización mediana de las características del caso práctico y extrapolando a que muchas de las organizaciones similares en Argentina se encuentran en igual situación, se podría llegar a deducir que las organizaciones medianas de la industria nacional deberían invertir mas en optimizar sus procesos para lograr un crecimiento sostenido.

Futuros trabajos

Entre las posibles extensiones al presente trabajo, se pueden citar:

- Realizar un caso práctico llevando a cabo la implementación en si hasta alcanzar la mejora de procesos según los objetivos planteados.
- Desarrollar un sistema de aplicaciones que permitan automatizar el seguimiento y control de todas las tareas que componen el framework.

Vocabulario

Baseline: Versión formalmente aprobada de un ítem de la configuración, formalmente determinada y fijada en un momento específico durante el ciclo de vida del ítem de configuración, sin tener en cuenta el medio.

Best Practice: Procedimiento bien definido que es conocido por producir resultados cercanos a lo óptimo.

Business Driver: Requerimiento especificado por las necesidades del mercado u organización que influencia la definición de los objetivos del negocio o la estrategia de la organización.

Capacidad: Habilidad de mantener los atributos medibles de los procesos en intervalos de tolerancia

Framework: Diagrama estructural que relaciona las partes componentes de una entidad conceptual con las demás

Madurez: Es el conocimiento y mejora de la capacidad alcanzada con la practica con el paso del tiempo

Modelo de referencia: Modelo que comprende definiciones de procesos descriptas en términos de propósitos y salidas de procesos, junto a una arquitectura que describe las relaciones entre los procesos

Modelos de implementación de ciclos de vida:

- IDEAL (Initiating, Define, Establishing, Acting, Leveraging)
- PDCA (Plan, Do, Check, Act)
- DUME (Define, Use, Measure, Evaluate)
- OODA (Define, Orient, Decide, Act)
- TQM
 - Identify customer / supplier relationships
 - Determine customer needs / expectations
 - Define work process
 - Identify candidate change
 - Implement change in a controlled environment
 - Evaluate effects of change
 - Change process permanently

Política: Principio conductor, generalmente establecido por los Srs. Managers, el cual es adoptado por una organización o proyecto para influenciar o determinar decisiones

Práctica: Actividad o tarea que conduce a la implementación o institucionalización de un atributo específico del proceso

Procedimiento: Descripción escrita del curso de acción a ser tomado para realizar una determinada tarea

Proceso: Conjunto de actividades, métodos, prácticas y transformaciones que la gente usa para desarrollar y mantener software y sus productos asociados (planes, diseños, código, casos de prueba, manuales de usuario, otros)

Process framework: Template que especifica y coordina los distintos procesos necesarios para desarrollar software. Su objetivo es especificar que procesos son necesarios los cuales deben ser customizados por la organización.

Referencias

- [1] CMM in Practice - Process for Executing Software Projects at Infosys. P.Jalote ~ SEI Series in Software Engineering - ISBN 0201616262
- [2] Capability Maturity Model for Software (Version 1.1). B.Curtis, M.B.Chrissis, C. Weber ~ SEI {Marzo 1993}
- [3] Capability Maturity Modeling at the SEI. M.Konrad, M.B.Chrissis, J.Ferguson, S.Garcia, B.Hefley, D.Kitson ~ SEI
- [4] El sector de software y servicios informáticos (SSI) en la Argentina: Situación actual y perspectivas de desarrollo. D.Chudnovsky, A.López, S.Jelitsko
- [5] IDEAL: A User's Guide for Software Process Improvement. B.McFeeley ~SEI
- [6] Software Process improvement Guidebook. ~ Software Engineering Laboratory Series, NASA
- [7] A Software Engineering Body of Knowledge (Version 1.0). T.B.Hilburn, I.Hirmanpour ~ SEI
- [8] Facts and Fallacies of Software Engineering. R.L. Glass ~ Addison Wesley - ISBN: 0321117425
- [9] SWEBOK, Guide to the Software Engineering Body of Knowledge (Versión 2004). ~ IEEE
- [10] Schaum's Outline of Theory and Problems of Software Engineering (Primera edición). D.Gustafson ~ McGraw-Hill - ISBN: 0071377948
- [11] Benefits of CMM-Based Software Process Improvement: Initial Results. J.Herbsleb, A.Carleton ~ SEI
- [12] Process Improvement Approaches – Their Implementation. I.Seward ~ Oxford Software Engineering
- [13] The Cascading Benefits of Software Process Improvement. B.Curtis ~ TeraQuest {2000}
- [14] Achieving Process Maturity in Small Organizations or Organizations with Small Projects. D.L.Johnson ~ Logos {Marzo 1999}
- [15] Applying the CMM to Small Organizations and Small Projects. D.L.Johnson ~ Logos {Abril 1998}
- [16] Model for Telecom Product Development & Support Process Capability (Versión 3.0). ~ Bell Canada {1994}
- [17] Software Process Assessment and Improvement - 5 Years of Experience with BOOTSTRAP. Stienen, Engelmann, Gierszal ~ Synspace {1997}
- [18] ISO/IEC TR 15504-1 - 15504-9, Information technology – Software process assessment. ~ IEEE
- [19] Capability Maturity Model Integration for Software Engineering Staged Representation (CMMI-SW, V1.1). ~ SEI
- [20] Capability Maturity Model Integration for Software Engineering Continuous Representation (CMMI-SW, V1.1). ~ SEI
- [21] TPI, BOOTSTRAP and testing. D.Robben ~ Sogeti Nederland B.V.
- [22] Software process engineering systems: models and industry cases. ~ University of Oulu - ISBN 9514265084 {2001}
- [23] Towards Mature IT Services. F.Niessink, H.van Vliet ~ Faculty of Mathematics and Computer Science, Vrije Universiteit Amsterdam
- [24] SPI frameworks - TQM, CMM, SPICE, ISO 9001, QIP experiences and trends. R.Conradi ~ Norwegian University of Science and Technology
- [25] SPiCE Benchmarking. C.Steinmann, H.Stejdir ~ ARC Innovative Software Services
- [26] Employee Motivation and Information using SPICE. The Road to Software Process Improvement. C.Steinmann ~ Synspace
- [27] Mastering Time-to-Market: a Handbook. Hall & Woodman - *website: <http://www.t2m.mcmail.com/why.htm>*
- [28] Application Development Tools. IdeaByte IT Trends {2003}
- [29] Factors in Software Quality. J.McCall, P.Richards, G.Walters ~ NTIS {Noviembre 1977}
- [30] UML Modeling of Five Process, Maturity Models (Version 1). S.Alexandre, N.Habra ~ CETIC - FUNDP {2003}
- [31] Ten Techniques for Trimming Time to Market. D.P.Olivier ~ Medical Device & Diagnostic Industry Magazine
- [32] Software Quality: The Elusive Target. B.Kitchenham, S.L.Pfleeger ~ IEEE
- [33] Small Cap and Growth Investing. A.Damodaran
- [34] Balancing Agility and Discipline: A Guide for the Perplexed. B.Boehm, R.Turner ~ Addison Wesley {2003}
- [35] Introduction to the Team Software Process. Reading, Humphrey ~ Addison Wesley {2000}
- [36] The Rational Unified Process Made Easy: A Practitioner's Guide to the RUP. P.Kroll, P.Kruchten ~ Addison Wesley {2003}
- [37] Extreme Programming Explained: Embrace Change. K.Beck ~ Addison-Wesley - ISBN 201616416 {2000}
- [38] The Rational Unified Process and the Capability Maturity Model - Integrated Systems/Software Engineering. B.Gallagher, L.Brownsword ~ SEI
- [39] IBM RUP and Beyond - Delivering project success through organisational culture. J.Holmes ~ Capgemini UK
- [40] Extreme Programming Explored (1st edition). W.C.Wake ~ Addison-Wesley - ISBN: 0201733978
- [41] Extreme Programming from a CMM Perspective. M.C.Paulk ~ SEI
- [42] Metodología FDD. L.Calabria ~ Universidad ORT Uruguay {2003}
- [43] Procesos de desarrollo: RUP, XP y FDD. A.Molpeceres - *website: www.javahispano.org* {2002}
- [44] Applying the PSP in Industry. M.Morisio ~ IEEE
- [45] Using a Personal Software Process to Improve Performance. W.Hayes ~ Carnegie Mellon University
- [46] CMM-Based Appraisal for Internal Process Improvement (CBA IPI) Method Description (Version 1.2). D.K.Dunaway, S.Masters ~ SEI {Noviembre 2001}
- [47] CMM Appraisal Framework (Version 1.0). S.Masters, C.Bothwell ~ SEI {Febrero 1995}
- [48] The Federal Aviation Administration Integrated Capability Maturity Model Appraisal Method (Version 1.0). L.Ibrahim, L.LaBruyere

- ~ Federal Aviation Administration {Abril 1999}
- [49] CBA - IPI Methodology. ~ Satyam Computer Services Ltd.
- [50] Software process assessment and improvement. ~ European Space Agency
- [51] Software Process Assessments: Issues and Lessons Learned. M.C.Paulk, W.S.Humphrey ~ SEI
- [52] Comprehensive Appraisal Method (Version 1.0). ~ Integrated System Diagnostics, Inc
- [53] Improvement and Process Assessment. H. van Loon ~ Synspace
- [54] The cultures of work organizations. H.M.Trice, J.M.Beyer ~ Prentice Hall {1992}
- [55] People Capability Maturity Model (Version 2.0). ~ SEI
- [56] Ingeniería de Software (5ta Edición). Ian Sommerville
- [57] After the Appraisal: A Systematic Survey of Process Improvement, its Benefits, and Factors that Influence Success. D.R.Goldenson, J.D.Herbsleb ~ SEI
- [58] Why Do Organizations Have Assessments? Do They Pay Off?. D.K.Dunaway, R.Berggren ~ SEI
- [59] Using an organizational culture analysis to design interventions for change. L.L.Forsythe ~ AORN Journal {Junio 2005}
- [60] Interim Profile: Development and Trial of a Method to Rapidly Measure Software Engineering Maturity Status. R.Whitney, E.Nawrocki ~ SEI
- [61] Software Engineering Process Group Guide. P.Fowler, S.Rifkin ~ SEI
- [62] Space product assurance, Software process assessment and improvement. ~ European Cooperation for Space Standardization
- [63] Defense Industrial Financial Management System - Project Management Plan. ~ USA Air Force Material Command
- [64] Analyzing the Conceptual Relationship Between ISO/IEC 15504 (Software Process Assessment) and the CMM. M.C.Paulk ~ SEI
- [65] Spice, Software process assessment, Guide for use in process improvement (Version 1.0). ~ SPICE Project Organization
- [66] The Era of Culture in Quality Improvement. D.M.Boan ~ Delmarva Foundation for Medical Care
- [67] People Capability Maturity Model. B.Curtis, W.E.Hefley, S.Miller ~ SEI
- [68] IEEE Guide for Software Quality Assurance Planning. ~ IEEE {1995}
- [69] A Guide to the Project Management Body of Knowledge. ~ PMI - ISBN 1880410230
- [70] Manager's Handbook for Software Development. ~ National Aeronautics and Space Administration
- [71] Procedimientos para la garantía de la calidad para una entidad productora de software. S.A.Cardenas ~ CEIS, ISPJAE. Cuba
- [72] Software Risk Management. R.P.Higuera, Y.Y.Haimes ~ SEI
- [73] Software management for executives guidebook. ~ Systems Engineering Process Office (SEPO), San Diego
- [74] Creating and Using a Business Case for IT Projects. ~ Project Management Office, Chief Information Officer Branch, Treasury Board Secretariat, Canada {Febrero 1998}
- [75] Business Case Development Guidelines. ~ Gateway Initiative {Abril 2005}
- [76] Return on Investment of Software Process Improvement – Metrics and Models for Software Engineering. D.F.Rico
- [77] Making the software Business Case, Improvement by the numbers. D.J.Reifer ~ Addison Wesley - ISBN 0201728877
- [78] Selling the IT Business Case to Senior Management. ~ Iain Massey & Associates {2005}
- [79] OMB Exhibit 300 Preparation Guide. ~ U.S. Department of energy {Julio 2002}
- [80] Economic Justification - A Step-by-Step Guide to Optimizing IT Investments that Forge Alliances Between IT and Business ~ Microsoft Corporation
- [81] Risk Management Processes for Software Engineering Models. M.Myerson ~ Artech House Computer Science Library - ISBN: 0890066353
- [82] Mejora de Procesos de Software y Coste de la Calidad: El "qué" y el "porqué" para aumentar los beneficios de las organizaciones de software. ~ European Software Institute
- [83] Costs and Benefits of Software Process Improvement. K.El Eman, L.C. Briand ~ IESE
- [84] Software Process Improvement: Modeling Return on Investment. D.F.Rico
- [85] Dirección Integrada de Proyecto – Project Management (3ra edición). ~ Universidad Politécnica de Madrid
- [86] Project Management – A managerial approach (4ta edición). J.R.Meredith, S.J.Mantel
- [87] CMMI V1.1 and Appraisal Tutorial. M.Phillips ~ SEI
- [88] Process Maturity Profile, Software CMM - 2005 End-Year Update. ~ SEI {Marzo 2006}
- [89] Modeling Project Management. R.M.Wideman ~ AEW Services {2003}
- [90] Practical Project Management: Tips, Tactics, and Tools. H.A.Levine ~ AEW Services {2003}
- [91] Total Project Management of Complex Projects - Improving Performance with Modern Techniques. R.M.Wideman ~ Acres International Limited
- [92] A Review of Managing High-Technology Programs & Projects. R.M.Wideman ~ AEW Services {2003}
- [94] Feature Driven Development. - *website: <http://www.featuredrivendevelopment.com>*
- [95] A Framework for Software Product Line Practice (Version 4). ~ SEI - *website: <http://www.sei.cmu.edu/productlines/framework.html>*
- [96] Open Process Framework. - *website: <http://www.opfro.org/>*
- [97] Agile Alliance. - *website: www.agilealliance.org*
- [98] Human Resources Capacity/Capability Assessment. ~ Solutions 2000 & Hiebert Group - *website: <http://www.solutions2000.ca/hrassessment.htm>*

- [99] Justifying Information Technology investments in tight times. ~ BearingPoint - *website: <http://www.bearingpoint.com>*
- [100] Return on Investment from Software Process Improvement. ~ U.S.DoD - *website: <http://www.softwaretechnews.com>*
- [101] Business Case Analysis - Tools, Training, and Resources for Building the Financial Business Case. - *website: <http://www.solutionmatrix.com/>*
- [102] An Investigation into Software Process Improvement in the Small and its Application in a Scandinavian Picture Agency Group. R.T.Johannesen ~ University of Sunderland {Octubre 2004}
- [103] A Software Process Improvement Framework for Small Organizations. D.Vasiljevic, S.Skoog ~ Blekinge Institute of Technology {Agosto 2003}
- [104] Software Process Improvement With CMM. J.Raynus ~ Artech House - ISBN 0890066442
- [105] CMM Implementation Guide - Choreographing Software Process Improvement. K.Caputo ~ Addison Wesley - ISBN 0201379384
- [107] Tesis de magister en ingeniería del software “Certificación de procesos de desarrollo de software” Basado en el Estándar ISO 9001:2000. C.G. Rivero Bianchi ~ ITBA {2004}