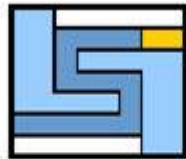


El proceso de Ingeniería de Requisitos en el ciclo global del software

Rosa María Torres de Paz, 6249286-W
(rosamaria.torresdepaz@gmail.com)

Supervisado por los Profs. Drs. Beatriz Bernárdez y Amador Durán



Departamento de
Lenguajes y Sistemas Informáticos
Universidad de Sevilla

Trabajo de investigación presentado en el Departamento de Lenguajes y Sistemas Informáticos de la Universidad de Sevilla como parte de los requisitos a cumplir para obtener el título de Doctora Ingeniera en Informática (Periodo de Investigación)

Índice

1	Introducción.....	3
2	Hipótesis y objetivos.....	5
3	Trabajo Relacionado.....	9
3.1	Ingeniería de Requisitos.....	9
3.2	Métrica v3.....	15
3.2.1	Conceptos Generales en Métrica v3.....	15
3.2.2	Elementos principales en Métrica v3.....	16
3.2.3	Requisitos en Métrica v3.....	17
3.2.3.1	Plan de Sistemas de Información.....	17
3.2.3.2	Estudio de Viabilidad del Sistema.....	18
3.2.3.3	Análisis del Sistema de Información.....	19
3.2.3.4	Diseño del Sistema de Información.....	22
3.2.3.5	Otros procesos.....	23
3.2.4	Conclusiones.....	25
3.3	CMMI.....	27
3.3.1	Conceptos Generales de CMMI.....	27
3.3.2	Elementos Principales de CMMI.....	27
3.3.3	Requisitos en CMMI.....	34
3.3.3.1	Desarrollo de Requisitos (Requirement Development RD).....	34
3.3.3.2	Gestión de Requisitos (Requirement Management REQM).....	36
3.3.4	Conclusiones.....	37
3.4	ITIL.....	38
3.4.1	Conceptos Generales de ITIL.....	38
3.4.2	Elementos Principales de ITIL.....	40
3.4.3	Requisitos en ITIL.....	43
3.5	CMMI-ITIL.....	49
3.6	COBIT.....	52
3.7	SOMA.....	55
4	Conclusiones y trabajo futuro.....	59
5	Bibliografía.....	61
6	Índice de Figuras.....	64
7	Índice de Tablas.....	65

1 Introducción

El contexto del presente trabajo está enmarcado dentro de la Ingeniería del Software, y en particular en la Ingeniería de Requisitos como pieza clave y fundamental a la hora de, por un lado proporcionar productos de calidad y en plazo; y por otro de ser el primer eslabón en la cadena estructurada y rigurosa de producción de software necesaria para alcanzar esa calidad, cumpliendo plazos y costes.

En esta introducción se pretende presentar el documento de la forma más lógica posible. Intentando, que se vayan induciendo los motivos por los que se han escogido los distintos elementos que se estudian en los capítulos posteriores.

Conociendo los resultados publicados de los proyectos de desarrollo de Software y trabajando diariamente en este campo, quería ahondar en el problema y analizar las posibles carencias de la situación actual analizando sus causas. El objetivo principal es identificar cuáles pueden ser las contribuciones a realizar para mejorar los resultados de nuestros días en el campo de la Ingeniería de Requisitos en su interacción con la Ingeniería del Software y la gestión de proyectos.

En primer lugar, quise averiguar el estado de la primera actividad a realizar en el desarrollo o mantenimiento de los sistemas de información: determinar qué tiene el sistema que realizar. En el campo de ingeniería de requisitos quería saber si había normativa, si había herramientas que lo soportaran, si había metodologías, etc.

Parecía lógico el hecho de continuar por los primeros esfuerzos realizados en el área de calidad de los sistemas de información: la utilización de una metodología de desarrollo. Es decir, eliminar el concepto de la informática como arte de programación, para aproximarnos a una metodología de trabajo. Como metodología de estudio elegí Métrica v3. De esta forma, los defectos y carencias de la metodología nos irían marcando el camino a seguir de los siguientes trabajos.

El grado de flexibilidad de la metodología, junto a la falta de indicadores (métricas que midan el proceso) que nos mostrasen cómo y dónde se puede mejorar, dejaban patente que se necesita un componente más. En esta línea, CMMI responde a cuestiones como cuál es el grado de organización o de madurez, cuál es el grado de independencia ante las rotaciones del personal al basarnos en procesos definidos, cuáles son las relaciones entre los procesos, cómo ser competitivo ante un entorno tan cambiante, etc.

Métrica v3, con todos sus beneficios, nos iba a mostrar una debilidad importante, que radica en la necesidad que tiene la propia metodología en reforzar el proceso de mantenimiento de un producto software.

La dependencia tecnológica actual de las organizaciones requiere que los servicios que ofrecen y utiliza dicha organización en su día a día, se integren no sólo en las líneas estratégicas sino como parte crucial del mantenimiento y evolutivo de los productos utilizados. Es aquí dónde las buenas prácticas de ITIL tienen su lugar.

ITIL incide en que los servicios ofrecidos y consumidos por la organización, hay que gestionarlos y mejorarlos, para que la operativa no sólo no se vea interrumpida, sino mejorada.

La dependencia de la tecnología en todos los sectores de la sociedad, provoca que los usuarios sean más exigentes con dichos servicios y que la capacidad productiva de dichos servicios deba ser mejorada.

En este punto del trabajo, había otra cuestión que estaba sin cubrir y a la que me enfrento diariamente: la arquitectura orientada a servicios (SOA), sobre todo desde la perspectiva de como la arquitectura de sistemas de información es tratada por las metodologías de desarrollo, por los procesos que gobiernan la organización y por los servicios que se exponen y que son utilizados internamente. En este campo aparece SOMA (Service-Oriented Modeling and Architecture).

Por tanto, el objetivo marcado en este trabajo era analizar el estado del arte en la ingeniería de requisitos y su importancia dentro de la gestión de proyectos en estas tres direcciones: metodológico, procesos y servicios y su relación en la mejora de los productos TI.

Si la evolución en el campo de las TI pasa por una mejor estructuración y enlazado de las distintas piezas que forman la cadena de elaboración de los productos, un objetivo tan difícil, como atrayente es averiguar cómo sistematizar y enlazar un componente psicológico y de comunicación: los requisitos.

Así el presente trabajo se estructura de la siguiente forma:

- Capítulo 2: Hipótesis y objetivos de partida de la memoria.
- Capítulo 3: Trabajo relacionado con los objetivos. Este apartado cubre los siguientes temas:
 - Revisión del estado actual de los requisitos
 - Tratamiento de los requisitos dentro de la metodología Métrica v3
 - Tratamiento de los requisitos dentro de CMMI
 - Tratamiento de los requisitos dentro de ITIL
 - La unión entre CMMI e ITIL
 - Los indicadores COBIT
 - La metodología SOMA
- Capítulo 4: Las conclusiones obtenidas y los trabajos futuros

2 Hipótesis y objetivos

Desde hace tiempo se conocen los datos del alto coste que implica reparar un error o defecto una vez realizada la entrega del producto. En comparación con lo que hubiera implicado detectarlo en fases más tempranas del desarrollo, varios estudios nos muestran una oscilación entre un 60% y un 100 % de sobrecoste [1], [2]. Situación que ratifica Boehm en [3].

También se conoce desde hace tiempo que el 56% del total de errores encontrados en los productos software, son generados en la fase de Requisitos [4]. El tanto por ciento del esfuerzo total del proyecto que se destina en la fase de Requisitos no llega al 20% independientemente del modelo de desarrollo (iterativo, en cascada, etc) o del tipo de proyecto (nuevo, de mantenimiento) [5].

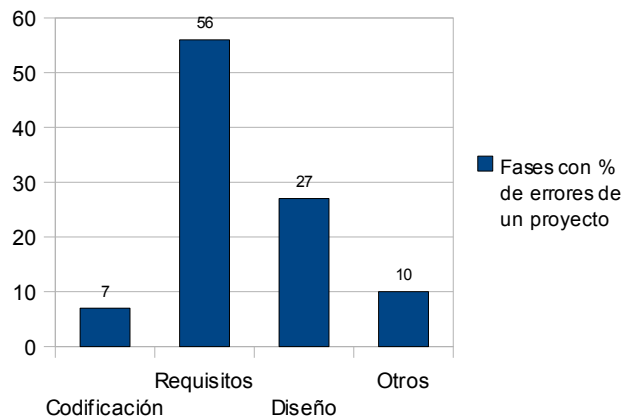


Figura 1: Proporción de Errores en cada una de las fases de desarrollo [4]

Si se tiene además en cuenta el gasto invertido en el mantenimiento del software, tal y como se observa en Figura 2: Evolución temporal del coste de Mantenimiento [6], es indudable que el problema persiste o que la situación debe mejorarse.

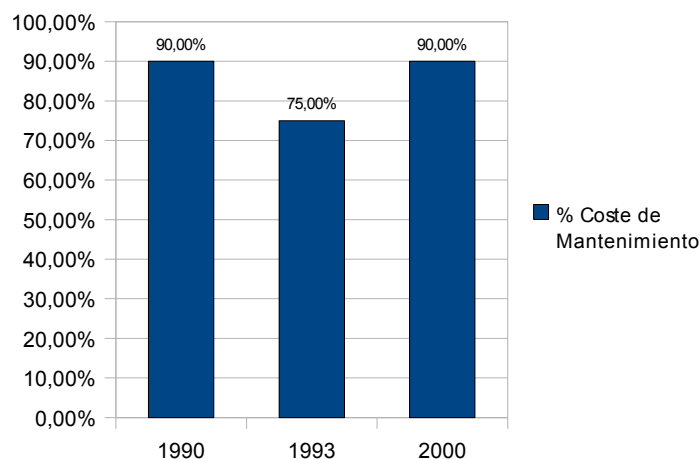


Figura 2: Evolución temporal del coste de Mantenimiento [6]

Acciones para solucionar esta situación se han realizado desde muchas perspectivas: desde nuevos lenguajes, metodologías en gestión de proyectos, nuevos paradigmas tan influyentes hoy en día como es la Orientación a Servicios, etc. Estas actuaciones avalan los avances producidos en la disciplina de Ingeniería del Software.

De hecho, en la figura 3 se observan las curvas de tendencias de los resultados de los proyectos de los datos ofrecidos por los informes CHAOS [7]

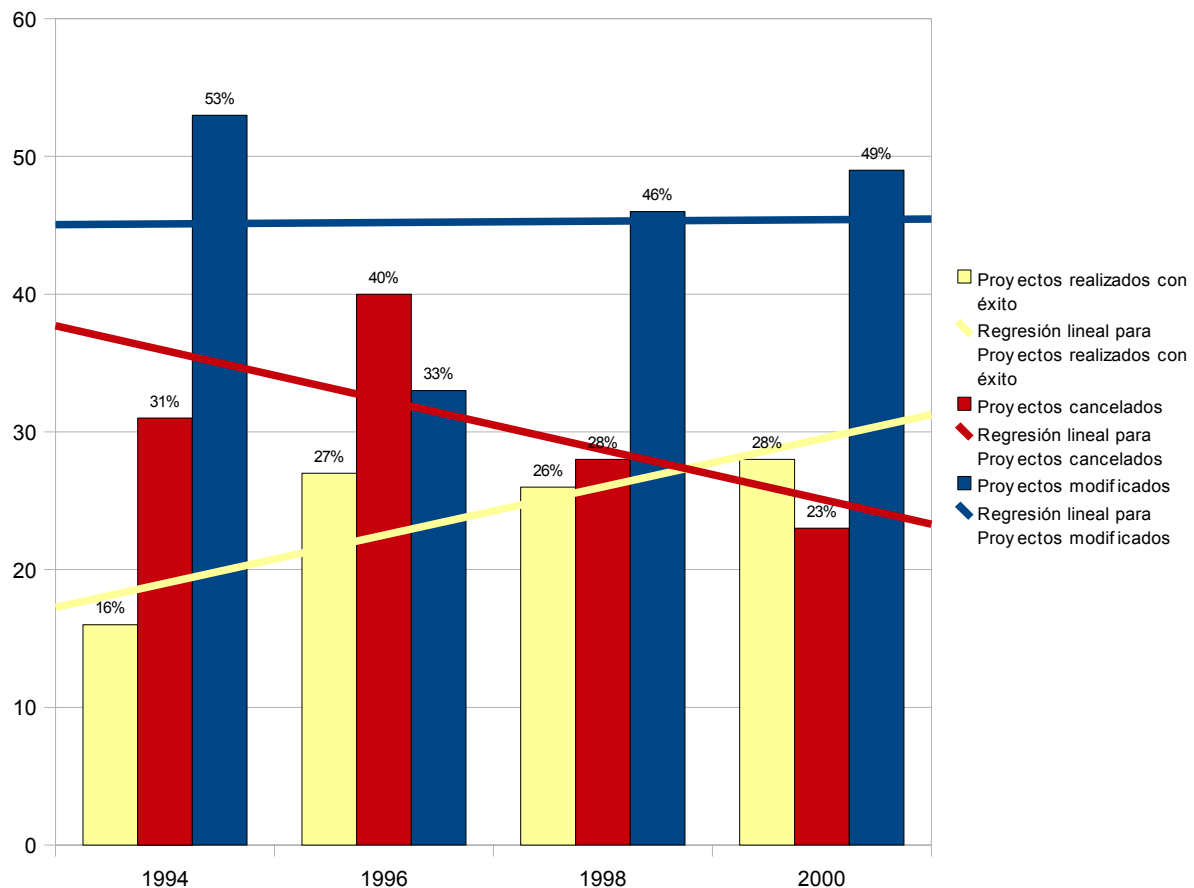


Figura 3: Resultados de los Informe CHAOS [7]

Se define como un *proyecto realizado con éxito*, aquellos que cumplen con el alcance definido, son terminados en tiempo y con el coste, de los datos con los que fueron planificados. Se definen como *proyectos cancelados*, aquellos que no llegaron a terminarse. *Proyectos modificados* son aquellos que fueron modificados en alguno de estos parámetros: alcance inicial, tiempo estimado y/o coste planificado.

Se observa en Figura 3: Resultados de los Informe CHAOS [7] cómo la tendencia de los proyectos realizados con éxito es ascendente, mientras que los proyectos fracasados es inversa. No obstante en el año 2000, el porcentaje de los proyectos terminados con éxito alcanza el 28% frente a los modificados que son 49%.

Hay un avance del informe correspondiente al año 2006, encontrada en [8], que indica que los

proyectos terminados con éxito representan un 35 %, frente al 46% de proyectos modificados y un 19% de proyectos cancelados. El informe no ha sido publicado oficialmente a fecha de la escritura del trabajo.

Según el informe del 2000, los factores que hacen que un proyecto llegue a realizarse con éxito son:

- Apoyo de la dirección (18 % de factor de importancia)
- Usuarios involucrados (16 %)
- Experiencia en la dirección del proyecto (14 %)
- Objetivos de negocio claros (12%)
- Alcance realista (10 %)
- Infraestructuras de software estándar (8 %)
- Requisitos acordados (6%)
- Metodologías formales (6%)
- Estimaciones fiables (5%)
- Otros (5 %)

La mayoría de esos factores atienden a cuestiones organizativas, sociales y en definitiva de mentalidad o de filosofía de gestión de proyectos y de gestión de la organización. Estas causas contrastan con los de los años anteriores dónde se evidenciaban más las carencias en la ingeniería de requisitos.

En los últimos años se ha desarrollado un intenso trabajo en esta línea, consiguiendo transformar la disciplina informática en ingeniería. Como cualquier otra ingeniería, con la automatización del proceso en todos los componentes y roles que intervienen en este área, se espera obtener los beneficios de una sistematización en el procedimiento. De este modo, el coste, el tiempo y la calidad sean aspectos controlables en un producto software.

La IEEE Computer Society en colaboración con la ACM, en “Guide to the Software Engineering Body of Knowledge” [9] ha realizado una recopilación de lo que la ingeniería del software debe suponer. Algunos de sus objetivos son:

- promover una visión consistente de la ingeniería del software
- establecer los límites de la ingeniería informática en relación a otras áreas del conocimiento
- definir los objetivos de la disciplina

Las áreas definidas que componen a la ingeniería informática son:

- Software Requirements
- Software Design
- Software Construction
- Software Testing
- Software Maintenance
- Software Configuration Management

- Software Engineering Management
- Software Engineering Process
- Software Engineering Tools and Methods
- Software Quality.

Conociendo la situación actual de los proyectos, y los trabajos realizados para trasladar esta área del conocimiento en ingeniería, los objetivos del trabajo son:

- Analizar las iniciativas más relevantes en la sistematización y control del proceso de desarrollo del software.
- Partiendo de que el sector de las tecnologías de la información (en adelante TI) pertenece al sector económico de Servicios, proporcionando servicios al propio sector; se analizará la característica diferenciadora frente a otras ingenierías: la relevancia de la ingeniería de requisitos en el producto final y por tanto de la percepción de la calidad del mismo.
- Adaptación de los dos factores anteriores a la demanda actual de diseño de sistemas: arquitecturas orientadas al servicio.
- Profundizar en la capacidad de ofrecer servicios con una calidad establecida.

Las dimensiones del trabajo se pueden observar en Figura 4: Dimensiones del Trabajo, es decir, los tres ejes que van a regir el trabajo: ingeniería de requisitos, metodologías y recomendaciones en los procesos de desarrollo y la interoperabilidad de sistemas.

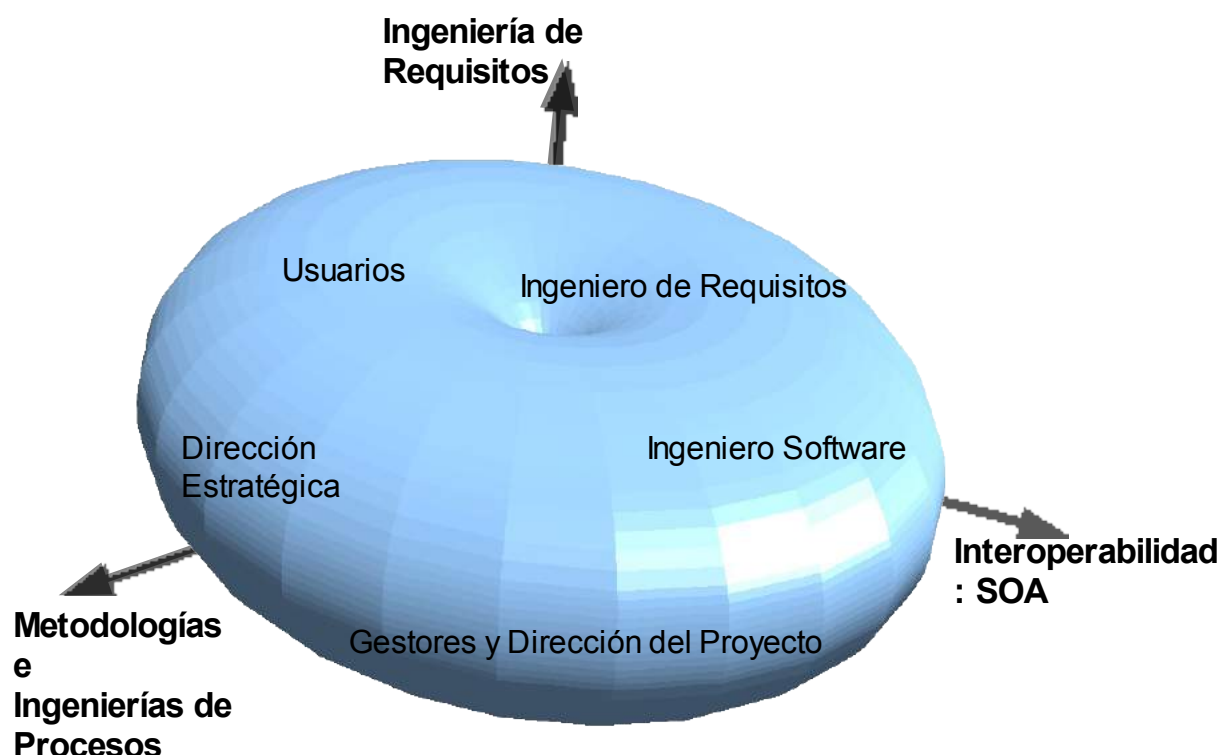


Figura 4: Dimensiones del Trabajo

3 Trabajo Relacionado

3.1 Ingeniería de Requisitos

En este apartado se realizará una revisión de la ingeniería de requisitos y las normas que cubren la definición de los mismos.

Tal y como se he visto en el apartado de Hipótesis y Objetivos y como se ratifica en [10], la necesidad de una ingeniería de requisitos dentro de la ingeniería del software es imprescindible para obtener productos de calidad. Ya se indica en [11] [12] que la calidad no puede implementarse en la etapas finales de los procesos de desarrollo sino que es una característica intrínseca al propio producto, iniciándose en las especificaciones de los productos. Este inicio de las actividades de calidad en la ingeniería de requisitos son la base para el aseguramiento de la calidad del proyecto [13].

Diversas definiciones sobre los requisitos y su tipología han sido aportadas en diversos trabajos. Posibles definiciones del concepto de requisito las encontramos en [14] y a su vez referenciada en [15], donde se define el término como una especificación de lo que debe ser implementado. Son descripciones de cómo el sistema se debe comportar, o las propiedades o atributos que el sistema debe contener. Los requisitos deben ser una restricción en el proceso de desarrollo del sistema.

Dentro de los movimientos más relevantes que en la ingeniería del software han ido surgiendo en los últimos tiempos podemos citar a la necesidad del **modelado de los procesos de negocio** [16] y la **interoperabilidad entre sistemas**. La ingeniería de requisitos debe responder a estas necesidades como parte del ciclo de desarrollo de los sistemas.

Con este enfoque, el giro a la definición de los requisitos aportada en [17], como los servicios que el sistema debe proporcionar junto a las restricciones en la operación del sistema; se establece más acorde a estos nuevos enfoques.

Bajo estas premisas y como puede verse en la figura 5, la ingeniería de requisitos consta de las siguientes actividades principales:

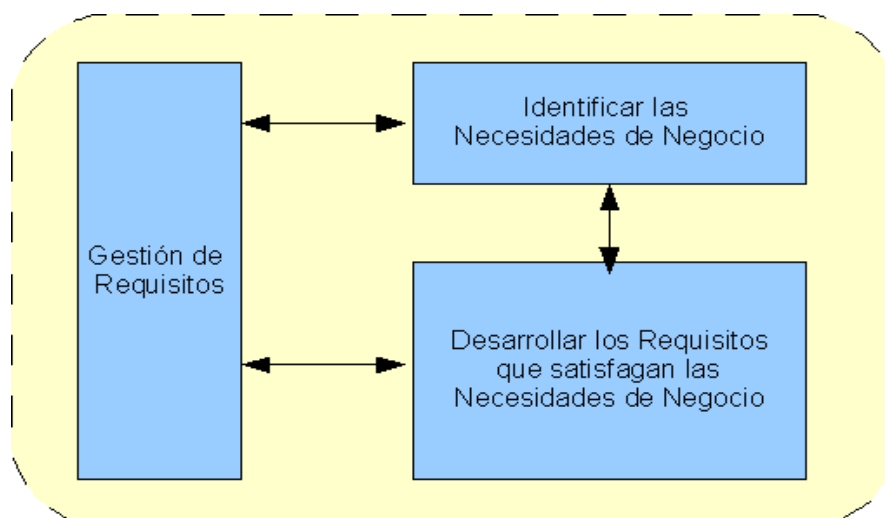


Figura 5: Actividades en la Ingeniería de Requisitos

Estas actividades se desarrollan a lo largo de todo el ciclo de vida del producto, ya que las necesidades del negocio van cambiando conforme el producto se explota en producción. La actividad de gestión de requisitos conlleva una serie de actividades dentro de la gestión de proyectos a nivel global, como son la gestión del cambio, la creación de líneas base, incluyendo aspectos financieros del propio proyecto. Es decir, una ampliación de funcionalidad o error encontrado en el producto debe ser analizado en relación a cuáles han sido las causas, el impacto de la eliminación del error o ampliación de funcionalidad que tiene en el proyecto, tiempo de resolución, coste y presupuesto disponibles, comprobar que el servicio o parte del proceso no es ofrecido por la organización, comprobar que el cambio está en concordancia con las líneas estratégicas y las necesidades del negocio de la organización, etc. La ingeniería de requisitos no debe ser considerada meramente como una elicitación de requisitos aislada en las fases iniciales del proyecto, sino que exige una retroalimentación continua de la organización .

Por todos estos motivos, la ingeniería de requisitos debe reforzar los dos aspectos anteriormente citados: el modelado de procesos y la interoperabilidad. Parte de estos conceptos y las nuevas líneas en la investigación de requisitos pueden leerse en [18].

Con este enfoque, dentro de las actividades generales en la gestión de requisitos, podemos desglosar las actividades principales en:

- **Identificar las necesidades del negocio:**
 - **Estudiar el dominio del problema.** Su objetivo principal es obtener un conocimiento suficiente del dominio del problema como para poder comunicarse eficazmente con clientes y usuarios, comprender su negocio, entender sus necesidades y poder proponer una solución adecuada.
 - **Identificar aspectos positivos y negativos de la situación actual.** Su objetivo principal es hacer explícitos qué aspectos funcionan bien y cuales no antes de la implantación del nuevo sistema. Los aspectos negativos identificados ofrecen la posibilidad de evitar que ocurran en el sistema a desarrollar y los positivos para conservarlos en la medida de lo posible.
 - **Estudiar el modelo de negocio del cliente.** Su objetivo principal es comprender el negocio actual, especialmente sus procesos de negocio. Independientemente de que los procesos de negocio actuales sean manuales o estén automatizados, esta tarea es fundamental para entender el contexto en el que se usará el sistema a desarrollar y promover posibles mejoras.
 - **Estudiar el entorno tecnológico del cliente.** Su objetivo es conocer el entorno tecnológico actual, tanto el *hardware* y el *software*, redes, etc., como la *arquitectura de servicios* que pueda estar implantada.
 - **Obtener y documentar las necesidades de clientes y usuarios.** Su objetivo principal es obtener las *necesidades de clientes y usuarios* y documentarlas como *objetivos de negocio*, incluyendo los *modelos de procesos del negocio a implantar* si difieren significativamente del modelo de negocio actual, o si no se consideró oportuno realizar el estudio del negocio actual previamente en el proyecto.
- **Desarrollar los requisitos de un sistema software que satisfaga las necesidades del negocio**
 - **Desarrollar la visión general del sistema.** Su objetivo principal es comenzar a definir

el sistema software (la *solución*) que satisfaga las necesidades de negocio de clientes y usuarios (el *problema*). Para ello, se definen los *requisitos generales* y se comienzan a especificar los *casos de uso del sistema* en su versión inicial. Conforme avance el proceso de ingeniería de requisitos, los requisitos generales se irán detallando en requisitos más específicos y los casos de uso que se considere oportuno irán evolucionando hacia su forma detallada.

- **Documentar los requisitos del sistema.** Su objetivo principal es continuar con la definición del sistema software a desarrollar, tomando como punto de partida los *requisitos generales* y los *casos de usos en su versión inicial*, y considerando los *objetivos de negocio* y el *modelo de negocio a implantar*. Los casos de uso que se considere necesario, se van completando con más información y los requisitos generales se van detallando en *requisitos funcionales, no funcionales, de integración* y en *restricciones técnicas*.
- **Definir las posibilidades de integración del sistema.** Su objetivo principal es identificar sistemáticamente los servicios en funcionamiento o que estén en desarrollo en el entorno de la organización y que pueden tener impacto en el sistema a desarrollar.
- **Analizar los requisitos del sistema.** Su objetivo principal es analizar los requisitos previamente elicitados y documentados para detectar posibles conflictos.
- **Verificar la calidad de los requisitos del sistema.** Su principal objetivo es la comprobación de si los requisitos se están elaborando correctamente.
- **Validar los requisitos del sistema.** Su objetivo principal es la comprobación de la concordancia entre los requisitos elicitados corresponden a las necesidades de usuarios y clientes.
- **Gestionar los requisitos del sistema software a desarrollar**
 - **Definir el ciclo de vida de los requisitos del sistema.** Su objetivo principal es definir los estados por los que los requisitos van a ir pasando. Es necesario estos estados para controlar las actividades pertenecientes al desarrollo de requisitos.
 - **Gestionar las líneas base de los requisitos del sistema.** Su objetivo fundamental es la generación de versiones del proyecto. Es decir, controlar la consistencia del producto en un instante dado.
 - **Gestionar los conflictos de los requisitos.** Su objetivo principal es la eliminación de posibles conflictos en el desarrollo de los requisitos.
 - **Gestionar las peticiones de cambio.** Su objetivo principal es gestionar todas las peticiones de cambio en los requisitos una vez generada la primera *línea base* de la *Especificación de Requisitos del Sistema*, asegurándose que se sigue el procedimiento de control de cambios establecido en el proyecto y que por lo tanto, puede ser necesaria una modificación en los costes, recursos y fecha de entrega inicialmente acordados.
 - **Gestionar la trazabilidad de los requisitos del sistema.** Su objetivo fundamental es comprobar la implementación de los requisitos y el grado de impacto de los cambios.

En la documentación de los requisitos, se deben tener las siguientes recomendaciones para la buena definición de requisitos en una organización:

- Estructurar los requisitos. Problema de los requisitos duplicados u omitidos.

- Gestionar los requisitos: base para los contratos
- Gestionar las restricciones: rendimiento, seguridad, etc
- Realizar pruebas de los requisitos
- Gestionar el control de cambios de los requisitos
- Obtener métricas de la evolución de los mismos
- Proporcionar ejemplos
- Reutilizar requisitos

Las normas internacionales encontradas que tratan la ingeniería de requisitos en su totalidad o aspectos de ella son las siguientes:

- ISO 20.000. La cobertura de los requisitos en esta norma se plasma en:
 - ISO/IEC 20.000-1:2005.Part 1 Specification. Define los requisitos para la Gestión de Servicios. Es decir, define los requisitos para que los servicios sean ofrecidos con calidad siendo gestionados.
 - ISO/IEC 20.000-2:2005.Part 2 Code of Practice. Proporciona una guía y recomendaciones para la Parte 1.
- IEEE 610.12. Es la norma que especifica la especificación de requisitos. Se complementa con la IEEE 830-1998 para la estructura del documento de la especificación de requisitos. Según la norma 610.12 la definición de los requisitos deben ser:
 - No ambigua
 - Completa
 - Verificable
 - Consistente
 - Fácil de Modificar
 - Rastreadable o trazable
 - Usable en operación y en mantenimiento
- ISO/IEC 12207. Este estándar internacional, en su última revisión del 2008 cubre el ciclo de vida de un proyecto software al completo. En la figura 6 se muestran los procesos cubiertos por esta norma:

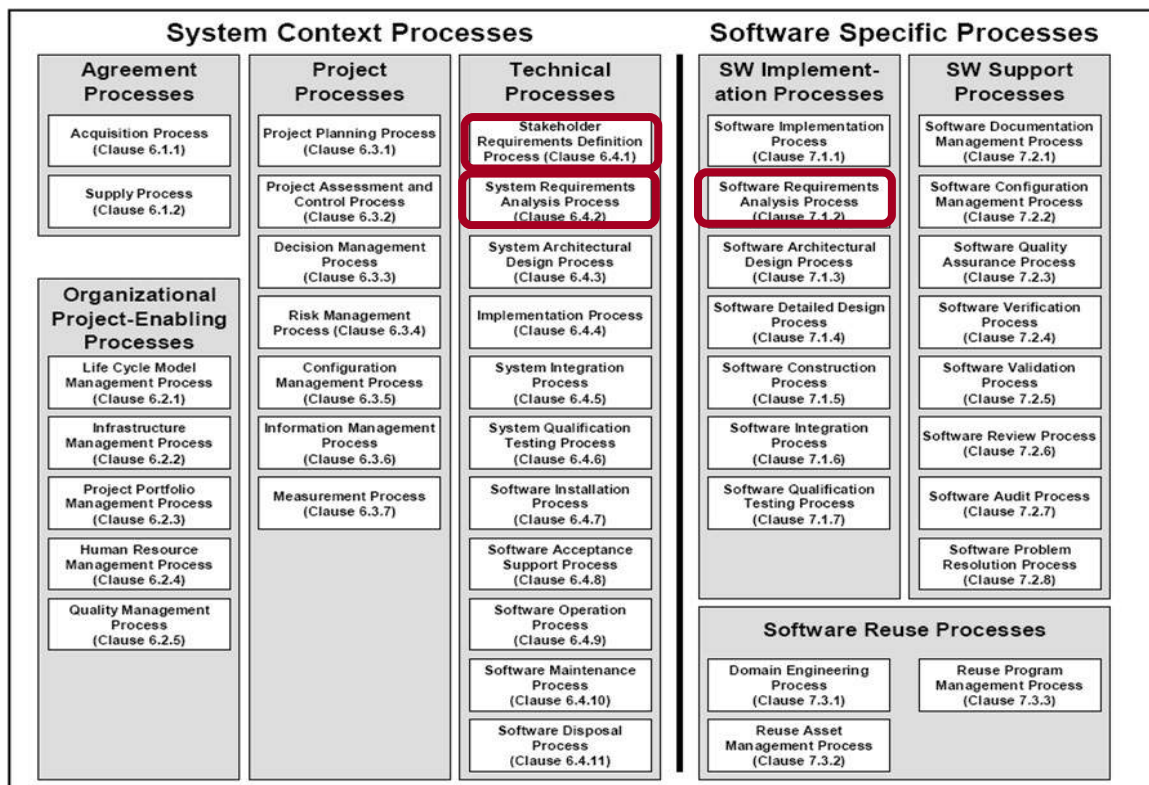


Figura 6: Los requisitos en los procesos de la ISO/IEC 12.207

Para terminar este apartado indicar según [19] las herramientas mas relevantes en el campo:

- Para la Gestión de Requisitos
 - Borland-Caliber Analyst (RM Module)
 - Compuware-OptimalTrace
 - IBM-RequisitePro (Rational), Doors (Telelogic)
 - Jama Software-Contour
 - MKS-Integrity
 - Serena-DimensionsRM
 - TCP Sistemas e Ingeniería- IrqA
 - Siemens PLM Software- TeamCenter
- Para la Definición de Requisitos
 - Atlantic Systems Guild-Volere
 - Axure Software Solutions- RP Pro
 - Borland- Calibre Analyst (DefineIT module)
 - Gorilla Logic- GXE

- iRise- Application Simulator
- Ravenflow- Raven
- Serena- Composer (Mashup Composer; Prototype Composer)

3.2 Métrica v3

Se ha elegido Métrica v3 para analizarla en este estudio por la importancia que tiene en el sector TI el desarrollo de sistemas de información para las Administraciones Públicas. Según [20] 928 Millones de Euros en el año 2008 se han invertido en la AGE (Administración General del Estado)

3.2.1 Conceptos Generales en Métrica v3

Métrica v3 [21] es la metodología realizada por el MAP (Ministerio de Administraciones Públicas) para dar respuesta a las necesidades metodológicas, en el ámbito de las administraciones públicas, al desarrollo de sistemas de información.

Métrica v3, como evolución de la versión anterior (v2), incorpora el concepto de Procesos en lugar de fases y módulos y aportando a través de sus Interfaces los procesos organizativos necesarios en el desarrollo del producto (Gestión de Proyectos, Gestión de la Configuración, Seguridad y Aseguramiento de la Calidad).

También se ha ampliado con la cobertura de la planificación a mayor nivel, la inclusión del mantenimiento, el plan de pruebas, el desarrollo Orientado a Objetos, como características más relevantes.

Para el desarrollo de esta metodología se han basado en los siguientes estándares:

- ISO 12.207 "Information technology –Software life cycle processes", estandar que le proporciona la estructura de los procesos principales:Planificación Desarrollo y Mantenimiento y las interfaces antes mencionadas.
- ISO/IEC TR 15.504/SPICE “Software Process Improvement and Assurance Standars Capability Determination”
- UNE-EN-ISO 9001:2000 Sistemas de Gestión de la Calidad
- UNE-EN-ISO 9000:2000 Sistemas de Gestión de la Calidad, Fundamentos y Vocabulario
- IEEE 610.12-1.990 “Standard Glosary of Software Engineering Terminology”
- Otras metodologías como SSADM, Merise, MAGERIT, etc.

Como cualquier metodología de desarrollo, ofrece una sistematización en las actividades que dan soporte al ciclo de vida de los productos software. Los objetivos que se persiguen, según Métrica v3 son:

- Proporcionar o definir Sistemas de Información que ayuden a conseguir los fines marcados en la Organización mediante la definición de un marco estratégico.
- Dotar a la Organización de productos software que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de Requisitos.
- Mejorar la productividad de los departamentos de TI, permitiendo una mayor capacidad de adaptación al cambio.
- Facilitar la comunicación y el entendimiento entre los distintos participantes.
- Facilitar la operación, mantenimiento y uso de los productos obtenidos.

Como podemos observar la gestión de los requisitos y resaltar el rol jugado por el usuario guían el desarrollo del proceso.

3.2.2 Elementos principales en Métrica v3

El objetivo de este apartado es revisar los elementos principales dentro de Métrica, para poder analizar el tratamiento de los Requisitos dentro de esta metodología.

Los procesos principales en Métrica v3 son los siguientes:

- **Planificación de Sistemas de Información.** En la planificación de Sistemas de Información se debe obtener una definición a nivel estratégico del marco de referencia de los Sistemas de Información. Según Métrica v3 “el resultado del Plan de Sistemas de Información debe orientar las actuaciones en materia de desarrollo de Sistemas de Información, con el objetivo básico de apoyar la estrategia corporativa, elaborando una arquitectura de información y un plan de proyectos informáticos para dar el apoyo a los objetivos estratégicos”.
- **Desarrollo de Sistemas de Información.** Este proceso cubre las actividades propias del desarrollo de un sistema de información.
- **Mantenimiento de Sistemas de Información.** Se entiende por Mantenimiento la generación de una nueva versión del producto, atendiendo a las demandas de los usuarios o por un error detectado en el mismo.

A su vez el **Desarrollo de Sistemas de Información** se divide en los siguientes procesos:

- **Estudio de la Viabilidad del Sistema.(EVS).** El objetivo de este proceso es realizar el análisis de viabilidad del sistema de información prioritario marcado por el Plan de Sistemas, desde un punto de vista táctico, operacional, económico y legal. Puede implicar un desarrollo a medida y/o bien una solución a medida. Dicho estudio puede derivar en varios sistemas de información.
- **Análisis del Sistema de Información.(ASI).** En el análisis obtendremos la especificación detallada, a través de una de sus piezas claves: el Catálogo de Requisitos. De igual manera obtendremos una serie de modelos que sirvan de puente entre las necesidades planteadas por el usuario y el resto de procesos en el desarrollo. También se comienza a elaborar el plan de pruebas.
- **Diseño del Sistema de Información (DSI).** En el desarrollo obtendremos la especificación detallada del entorno tecnológico, los componentes de información y la arquitectura del sistema.
- **Construcción del Sistema de Información (CSI).** A partir de los modelos generados por el proceso anterior, obtendremos la construcción y prueba de los distintos componentes del sistema. Se desarrollan los procedimientos de seguridad y operación, así como los diferentes manuales de usuario.
- **Implantación y Aceptación del Sistema (IAS).** Se realizará la aceptación del sistema por parte del cliente una vez entregado y es llevado a producción.

Los procesos se dividen a su vez en actividades, y éstas en tareas, bajando progresivamente en el nivel de detalle y de definición.

Por último, como elementos principales se disponen de las interfaces. Como hemos comentado las

interfaces están para dar soporte a la procesos anteriores en la parte organizativa y de seguridad. Así Métrica v3 propone:

- Gestión de Proyectos. Realiza la planificación, el seguimiento y el control de las actividades, gestionando los recursos humanos necesarios y materiales.
- Seguridad. Incorpora la seguridad tanto en las actividades del proceso de desarrollo, como la seguridad del propio producto.
- Gestión de la configuración. Identifica y define los procedimientos necesarios en el control de las versiones del producto, así como la configuración del sistema.
- Aseguramiento de la Calidad. Define planes específicos de aseguramiento de calidad aplicables a los proyectos específicos, como velar por la calidad del producto.

3.2.3 Requisitos en Métrica v3

Se va a analizar el tratamiento de los Requisitos a través de todos los procesos anteriormente citados. Realizaremos el recorrido en el mismo orden en el que hemos ido presentando los procesos en el apartado anterior.

3.2.3.1 Plan de Sistemas de Información

El objetivo de realizar un PSI es estudiar la organización para de identificar y analizar las áreas en las que sería necesario abordar un posible desarrollo de Software.

El tratamiento que realiza Métrica v3 de los requisitos en este proceso, donde se estudian las necesidades de información de los procesos de la organización, es la definición de los requisitos generales para la obtención de los modelos conceptuales de información.

La figura 7 muestra las actividades principales del Plan de Sistemas:

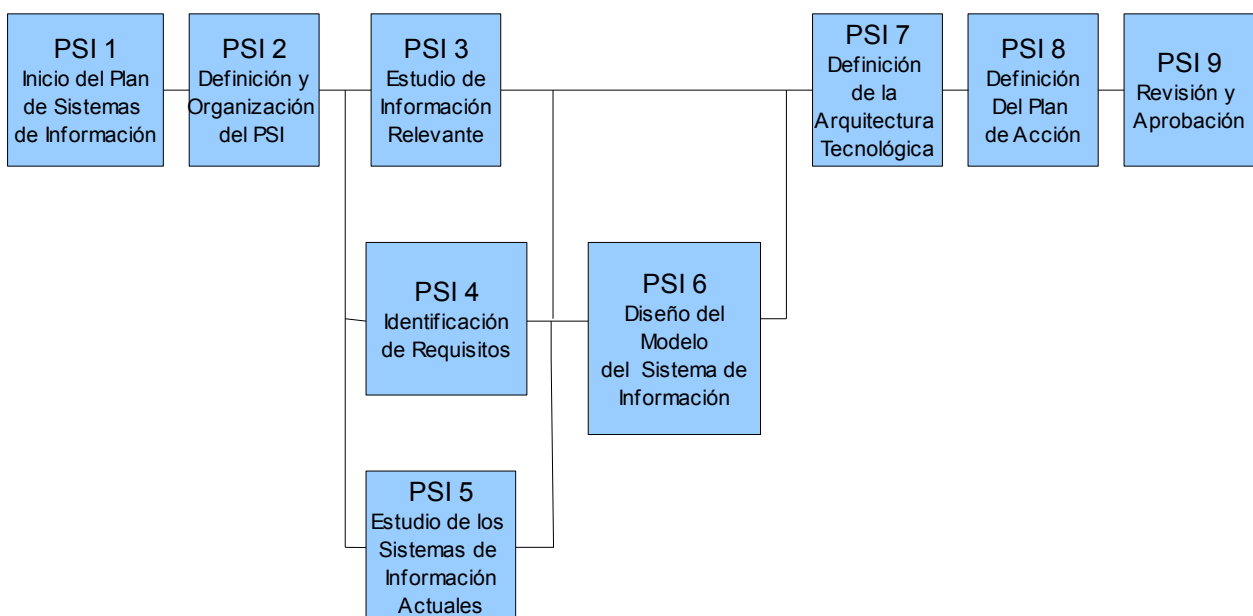


Figura 7: Plan de Sistemas de Información en Métrica v3

- En la actividad PSI 3 “Estudio de información relevante”, se estudian todos los antecedentes necesarios que pueden influir en la realización del Plan. El resultado de este estudio será una de las entradas para la siguiente actividad. Se comienza a construir el Catálogo de Requisitos.
- La actividad PSI 4 “Identificación de Requisitos”, tiene como principal objetivo la especificación detallada de los requisitos generales de información y la clasificación de los mismos en función de su prioridad. Dicha información se plasma en el Catálogo de Requisitos. Se realiza de la siguiente manera:
 - Tarea PSI 4.1 “Estudio de los procesos del PSI”. Se analiza cada proceso afectado por el Plan de Sistemas. Se identifican los requisitos de una manera más precisa.
 - Tarea PSI 4.2 “Análisis de las Necesidades de Información”. Con esta tarea se refuerza la identificación comenzada en la tarea anterior.
 - Tarea PSI 4.3 “Catalogación de Requisitos”. En esta tarea se plasman e incorporan los requisitos al Catálogo de Requisitos.
- En la actividad PSI 6 “Diseño del Sistema de Información”, se elabora un modelo de Sistemas de la Información, teniendo en cuenta la cobertura que los sistemas actuales dan a los requisitos identificados.
- En la actividad PSI 7 “Definición de la Arquitectura Tecnológica”, se define la arquitectura tecnológica que da soporte al resultado de la actividad anterior, teniendo en cuenta sobre todo los requisitos tecnológicos.
- En la actividad PSI 9 “Revisión y Aprobación” se aprueba el Catálogo de Requisitos del PSI por el comité de dirección.

3.2.3.2 Estudio de Viabilidad del Sistema

El objetivo de realizar el estudio de Viabilidad del Sistema es indicar la forma de abordar los proyectos que satisfagan los requisitos planteados, en función de las alternativas de solución. Es importante tener en cuenta que el Catálogo de Requisitos del PSI no es el mismo que el obtenido en este proceso, ya que el anterior engloba a todo el Plan de Sistemas de la Organización.

De forma general, el Estudio de Viabilidad del Sistema consta de las siguientes actividades que se muestran en la figura 8.

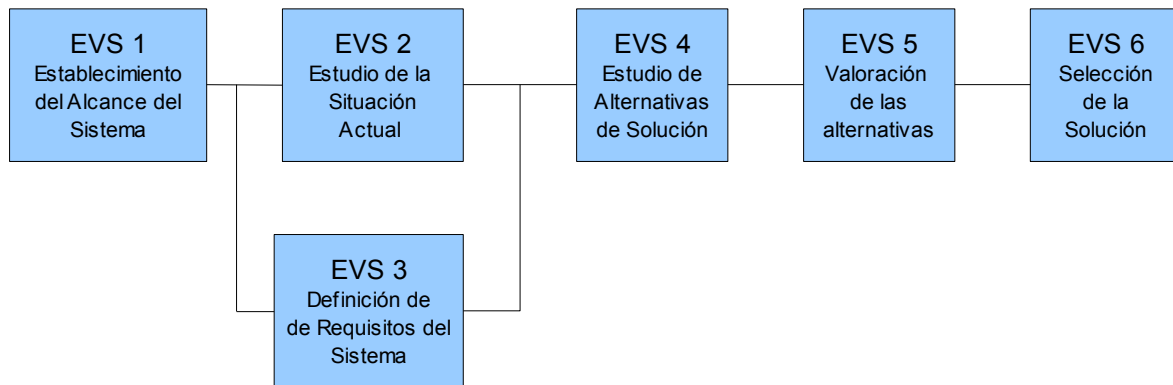


Figura 8: Estudio de viabilidad del sistema en Métrica v3

- En la actividad EVS 1 “Establecimiento del Alcance del Sistema”, se estudian los requisitos del sistema a abordar para determinar el alcance del proyecto.
 - Tarea EVS 1.1 “Estudio de la solicitud”. Se genera el Catálogo de Requisitos específico del proyecto.
 - Tarea EVS 1.2 “Identificación del Alcance del Sistema”. Dentro de la identificación del alcance del sistema, también se plasma en el Catálogo de Requisitos las posibles interrelaciones o dependencias con otros sistemas de la Organización.
- Actividad EVS 3 “Definición de Requisitos del Sistema”. Aunque en la actividad anterior se ha empezado a generar el catálogo de requisitos, es esta actividad la que tiene como misión principal la generación del mismo.
 - Tarea EVS 3.2 “Identificación de Requisitos”. Sesiones de trabajo con los participantes adecuados para realizar una revisión de la información actual, se debe considerar el resultado de la actividad EVS 2 “Estudio de la situación Actual”.
 - Tarea EVS 3.3 “Catalogación de Requisitos”. Se clasifican los requisitos junto a las prioridades. Con esta información, se podrán estudiar las alternativas de solución realizado por la actividad EVS 4 “Estudio de Alternativas de Solución” viendo la cobertura de los requisitos.
- En la actividad EVS 6 “Selección de la Solución”, más en concreto en la tarea EVS 6.2 “Evaluación de las Alternativas y Selección” se debate la o las alternativas de solución, por lo que el Catálogo de Requisitos puede ser modificado. En la tarea 6.3 “Aprobación de la Solución” es aprobado finalmente el Catálogo.

3.2.3.3 Análisis del Sistema de Información

El objetivo de realizar este proceso es obtener el análisis del sistema de información. El proceso PSI genera el Catálogo de Requisitos del modelo estratégico de la organización, y el proceso EVS el Catálogo de Requisitos con los objetivos del sistema, atendiendo a la cobertura que pueda haber con otros sistemas y restricciones técnicas y/o legales. En este subproceso del proceso de Desarrollo del Sistema de Información, es donde se comienzan a desarrollar requisitos.

Según indica Métrica v3 “El objetivo de esta actividad es elaborar un catálogo de requisitos detallado, que permita describir con precisión el sistema de información y que además sirva de base para comprobar que es completa la especificación de los modelos obtenidos” en el resto de las actividades del ASI. Todas estas actividades podrán modificar, ampliar y refinar el Catálogo.

De forma global tendríamos la estructura del ASI como muestra la figura 9 (el diseño estructurado no se tendrá en cuenta en este estudio por estar hoy en día en desuso):

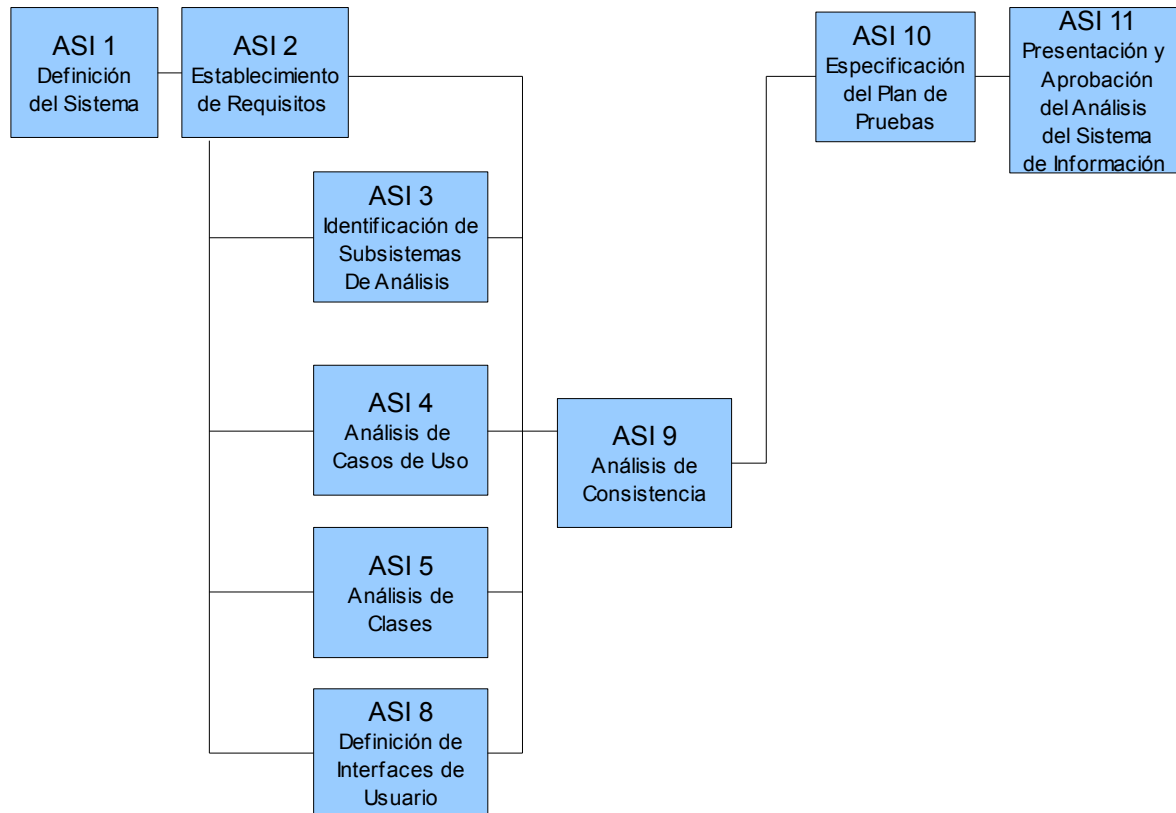


Figura 9: Análisis del Sistema de Información en Métrica v3

- En la actividad ASI 1 “Definición del Sistema”, se toman los productos resultantes del proceso anterior y se adecuan como punto de partida del análisis a elaborar.
 - En la tarea ASI 1.1 “Determinación del Alcance del Sistema” se comienza con la definición del Catálogo de Requisitos del Sistema.
 - En la tarea ASI 1.2 “Identificación del Entorno Tecnológico” se revisan los requisitos del Catálogo añadiendo las posibles condiciones o restricciones del entorno tecnológico.
- La actividad ASI 2 “Establecimiento de Requisitos” realiza la definición, análisis y validación de los requisitos, completando el Catálogo definido en la actividad anterior para obtener un catálogo detallado. Se descompone de las siguientes tareas con una retroalimentación continua entre ellas:
 - Tarea ASI 2.1 “Obtención de Requisitos”. El objetivo de esta tarea es la obtención detallada de lo requisitos. Se deben introducir las posibles restricciones del entorno, tanto a nivel software como hardware. Teniendo ahora en cuenta el punto de vista de los

usuarios expertos, las prioridades se incluirán también en el Catálogo. Propone un ejemplo de tipología de requisitos consistente en:

- Funcionales.
 - Rendimiento.
 - Seguridad.
 - Implantación.
 - Disponibilidad del Sistema.
- Tarea ASI 2.2 “Especificación de Casos de Uso”. Para los requisitos funcionales deben realizarse los Casos de Uso asociados. Para cada caso de uso se deberá realizar una descripción del escenario, precondiciones y postcondiciones, interfaces de usuario y condiciones de error.
 - Tarea ASI 2.3 “Análisis de Requisitos”, tiene por objetivo detectar cualquier inconsistencia en el Catálogo realizado. Se revisan también las prioridades. Es muy importante también este análisis, ya que nos permite estructurar la información obtenida, acercándola a los procesos siguientes gracias a la generalización, relaciones, etc.
 - Tarea 2.4 “Validación de Requisitos”. Se asegura que el Catálogo es válido, y que los requisitos son consistentes y completos.
 - La identificación de subsistemas en la actividad ASI 3, se basa en la afinidad de requisitos (entre otros) para la identificación de subsistemas. El catálogo se actualiza con esta división realizada.
 - En la actividad ASI 8 “Definición de Interfaces de Usuario”, se tendrá que verificar que la interfaz o prototipo realizados satisfacen los requisitos definidos.
 - Actividad ASI 9 “Análisis de Consistencia y Especificación de Requisitos”, verifica la calidad técnica de cada modelo y su coherencia, asegurando que usuarios y analistas tienen el mismo concepto. Tal y como Métrica v3 expone: “Se elabora la **Especificación de Requisitos Software** (ERS), como producto para la aprobación formal, por parte del usuario, de las especificaciones del sistema. Se convierte el ERS en la línea base para los procesos posteriores del desarrollo del software, de modo que cualquier petición de cambio en los requisitos que pueda surgir posteriormente, debe ser evaluada y aprobada”.
- Una vez que se realizan las tareas de Análisis y Validación de los modelos, es en la tarea ASI 9.4 “Elaboración de la Especificación de Requisitos Software (ERS)” donde se incorpora toda la información necesaria para la aprobación del documento.
- En la definición del Plan de Pruebas (ASI 10) es crucial el papel que desempeña el documento ERS para que el sistema sea validado por el cliente y sea aceptado. También se definirán los Requisitos del Entorno de Pruebas: hardware, software base, configuración del entorno, etc.
 - En la actividad ASI 11 “Aprobación del Análisis del Sistema de Información” se procede a la aprobación de la Especificación de Requisitos Software y del Plan de Pruebas, por el comité de dirección.

3.2.3.4 Diseño del Sistema de Información

El objetivo de este proceso es realizar el diseño del sistema de información. En este proceso, referente al ERS, se definen los requisitos de implantación, operación, administración, seguridad, control de acceso, etc. toda una serie de requisitos técnicos. También se explicitan una serie de requisitos necesarios para la implantación del sistema, como son los relacionados con la infraestructura y las necesidades de formación.

Como ya se ha indicado, los requisitos se van ampliando y refinando durante todo el ciclo de vida del producto. Los productos que se han ido obteniendo en procesos anteriores sirven de base a los siguientes. Por ejemplo, en el Diseño de Casos de Uso Reales, se matizará tanto los Casos de Uso realizados en la actividad ASI, como los requisitos hasta entonces plasmados.

Las principales actividades que componen este proceso, lo podemos ver en la figura 10:

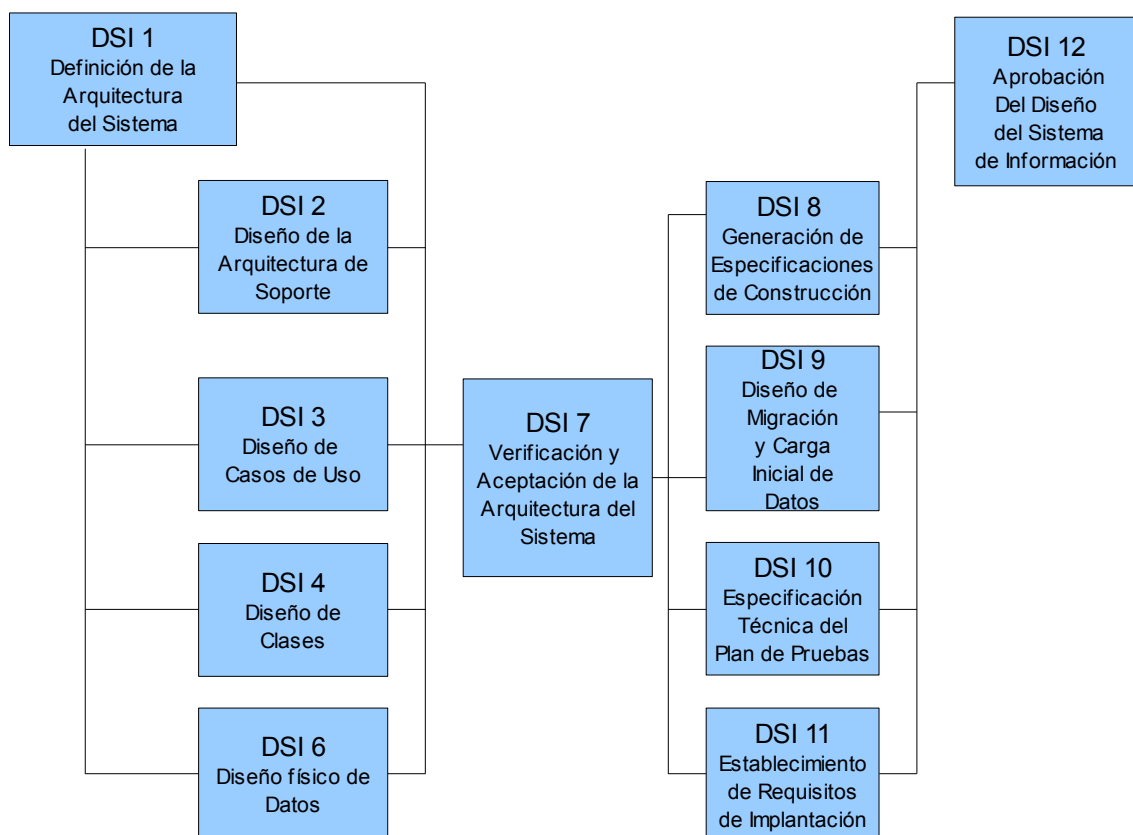


Figura 10: Diseño del Sistema de Información en Métrica v3

- En la actividad DSI 1 “Definición de la Arquitectura del Sistema”, se deciden los requisitos, normas y estándares que se derivan de la elección de la arquitectura del sistema y que determinan no sólo este proceso sino en el siguiente: Construcción del sistema de Información (CSI). A su vez, se adoptan los requisitos del sistema, que van a dar como resultado los procedimientos de operación, administración del sistema, etc. Se refleja en las siguientes tareas:
 - En la tarea DSI 1.2 “Identificación de Requisitos de Diseño y Construcción”.

- En la tarea DSI 1.7 “Especificación de Requisitos de Operación y Seguridad”.

En la mayoría de las restantes tareas, se siguen obteniendo requisitos relativos al diseño, como por ejemplo en la actividad DSI 6 “Diseño Físico de Datos”, donde se establecen requisitos relativos al rendimiento; en la actividad DSI 8 “Generación de especificaciones de Construcción” donde se fijan cuestiones operacionales; en la actividad DSI “Diseño de la Migración y Carga Inicial de Datos”, donde se determinan los requisitos necesarios para la migración de los datos.

- En la actividad DSI 10 “Especificación técnica del Plan de Pruebas”, el catálogo de requisitos, el catálogo de excepciones y el diseño detallado, serán las fuentes que nos permitan especificar las pruebas de tal manera que se pueda garantizar que el sistema realiza lo que el cliente/ usuario necesita y se ha definido.
- En la actividad DSI 11 “Establecimiento de requisitos de Implantación”, va a permitir que la Implantación y Aceptación del Sistema (IAS) se pueda llevar a cabo, gracias a las siguientes tareas:
 - En la tarea DSI 11.1 “Especificación de Requisitos de Documentación de Usuario”.
 - En la tarea DSI 11.2 “Especificación de Requisitos de Implantación”.

3.2.3.5 Otros procesos

Existen otros procesos que son importantes desde el punto de vista de este trabajo. Las interfaces de Métrica tratan aspectos horizontales a todo el proceso de desarrollo, como son la seguridad, el control de la configuración, la gestión de proyectos y la calidad.

1. **Interfaz de Gestión de proyectos.** Las actividades de Gestión de Proyectos se dividen en: actividades de inicio del proyecto (GPI), las actividades de seguimiento y control (GPS), y las de finalización del proyecto (GPF) . Las de GPS se han dividido en Gestión de Incidencias y Gestión de Cambios en los Requisitos.
 - Gestión de Cambios en los requisitos: Según preconiza la metodología, no debería darse por la inclusión de los usuarios en la aprobación de los requisitos. Los cambios en los mismos deben ser gestionados por el Comité de Seguimiento. Es necesario el establecimiento de un mecanismo formal para estos cambios que nos aseguren que el impacto, en cuanto a tiempo y esfuerzo están acordados. Los cambios en los requisitos se registrarán en un documento: Registro de Cambios. Para cada cambio se registrará:
 - Formulario de Petición del Cambio (GPS 5: Petición de Cambios de Requisitos.)
 - Catálogo de necesidades (GPS 6: Análisis de la Petición de Cambio en los Requisitos)
 - Análisis Funcional del Cambio (GPS 6: Análisis de la Petición de Cambio en los Requisitos). En caso de aprobarse se realizará GPS 7: Aprobación de la solución. Se completa la aprobación con:
 - Estimación de Esfuerzo y sus variaciones en coste y tiempo (GPS 8 Estimación del esfuerzo y planificación de la solución)
 - GPS 9 Registro del cambio de Requisitos

Tienen una parte común que son:

- GPS 10: Finalización de la tarea
- GPS 11: Actualización de la planificación
- GPS 12: Reuniones de Seguimiento
- GPS 13: Aceptación.

2. Mantenimiento del Sistema de Información. Proceso del desarrollo de Métrica que tiene por objeto la generación de una nueva versión del sistema, por cualquier correctivo realizado. El mantenimiento que contempla es el correctivo y el evolutivo. El mantenimiento correctivo es el que comprende los cambios para eliminar errores del software. El mantenimiento evolutivo son los cambios para incorporar, eliminar y modificar el producto software para cubrir la expansión o cambio en las necesidades del usuario.

- Registro de la Petición MSI 1, en caso de ser una mejora hay que remitir una especificación de requisitos.
- Análisis de la Petición MSI 2, es un estudio de la Viabilidad del Mantenimiento

3. Interfaz de calidad. Según la norma ISO 9000-2000, se define la calidad como el grado en que un conjunto de características inherentes cumplen con unos requisitos. Es por tanto, que la calidad debe garantizar que el producto satisface todos los requisitos del sistema de información. La interfaz de calidad define sus actividades relacionadas con el proceso de Métrica de desarrollo al cual está auditando.

- Para el Análisis nos encontramos con ASI-CAL 3: Revisión del Análisis de Consistencia. Y en particular la tarea ASI-CAL 3.1 Revisión del Catálogo de Requisitos, valida que los requisitos del sistema son consistentes y que serán satisfechos. La siguiente tarea, ASI-CAL 3.2 Revisión de la Consistencia entre productos, comprueba que los productos obtenidos se ajustan a los estándares y normas y a los requisitos especificados. Revisa la trazabilidad entre los requisitos.
- Para el Diseño se tienen “las revisiones que se centran en confirmar que los requisitos especificados en el Análisis se han traducido en una arquitectura conforme al entorno tecnológico seleccionado”. Se revisan los requisitos para las distintas pruebas a realizar, así como los requisitos no funcionales relacionados con la documentación de usuario e implantación.
 - DSI-CAL 1.1 Revisión de la consistencia entre productos del diseño, comprueba que el sistema realiza los requisitos especificados.
 - DSI-CAL 2.2 Revisión del Plan de Pruebas, las pruebas de implantación deben contemplar los requisitos no funcionales, así como que las pruebas de aceptación comprueben los criterios de aceptación por parte del usuario.
 - DSI-CAL 3 Revisión de los requisitos de implantación. Incluye dos tareas: DSI-CAL 3.1 Revisión de los Requisitos de Documentación de usuario y DSI-CAL 3.2 Revisión de los Requisitos de Implantación (que son los relacionados con la instalación, formación e infraestructura).
- Para la Implantación del Sistema tenemos IAS-CAL 1 Revisión del Plan de Implantación del Sistema.

- Para el Mantenimiento se tiene MSI-CAL 1.1 Revisión del mantenimiento del Sistema de Información. Se revisa periódicamente el Catálogo de Requisitos y que las peticiones de los usuarios son tratadas y tramitadas.
4. **Interfaz de seguridad.** Al igual que con la interfaz anterior, las actividades relativas a la seguridad se vinculan a las actividades del proceso de desarrollo. La seguridad se ocupa de tanto del propio proceso en sí de desarrollo, como de la seguridad que debe contemplar el sistema intrínsecamente. En el EVS se comienzan a elaborar las recomendaciones de seguridad. Métrica v3 indica que se deben incorporar al Catálogo de Requisitos. De forma especial, se tiene DSI-SEG 2 Especificación de Requisitos de Seguridad del Entorno Tecnológico y DSI-SEG 3 Requisitos de Seguridad del Entorno de Construcción.

3.2.4 Conclusiones

Las principales conclusiones alcanzadas tras el estudio de la metodología Métrica v3 son:

- Mantenimiento insuficiente. Tanto en el proceso en sí mismo como al no incluir por ejemplo el mantenimiento adaptativo (modificaciones en los entornos en los que el sistema opera) y perfectivo (mejorar la calidad interna de los sistemas como reestructuraciones del código, etc)
- Enlazar el mantenimiento con la gestión de proyectos desde la propia gestión de proyectos.
- Demasiada flexibilidad de la metodología al no definir el qué y el cómo. Dificultad de adopción, sobre todo a medianas y pequeñas empresas.
- Los cambios en los requisitos se aprueban, y después se valora el esfuerzo y coste.
- No hay una taxonomía clara para los requisitos ni ninguna plantillas o definición de atributos, que nos indiquen qué capturar de los requisitos.
- Dispersión de la definición de los requisitos, tanto en el desarrollo propio del sistema, como de la tipología (por ejemplo, los requisitos de seguridad son establecidos en la interfaz de Seguridad, sin embargo la tarea DSI 1.7 indica que se establecen los requisitos de operación y seguridad).
- La verificación de requisitos y la trazabilidad enunciadas pero sin rigurosidad y nivel de detalle.

Tarea	Producto de salida	Técnicas o Prácticas	Objetivo
PSI 4 Identificación de Requisitos: - PSI 4.1 Estudio de los Procesos del PSI - PSI 4.2 Análisis de las Necesidades de la Información - PSI 4.3 Catalogación de Requisitos	- Modelo de procesos de la organización (PSI 4.1) - Necesidades de información (PSI 4.2) - Modelo de información (PSI 4.2) - Catálogo de requisitos (PSI 4.3)	- Modelo de Procesos de la Organización (PSI 4.1) - Sesiones de Trabajo - Modelo Entidad / Relación extendido (PSI 4.2) - Diagrama de Clases (PSI 4.2) - Catalogación (PSI 4.3)	Especificación detallada de los requisitos generales de información y la clasificación por prioridad. Requisitos de los procesos afectados por el PSI
EVS 3 Definición de Requisitos del Sistema: -EVS 3.2 Identificación de Requisitos -EVS 3.3 Catalogación de Requisitos	Catálogo de requisitos	- Sesiones de trabajo (EVS 3.2) - Catalogación (EVS 3.3)	Generación del Catálogo de Requisitos del proyecto
ASI 2 Establecimiento de Requisitos: ASI 2.1 Obtención de Requisitos ASI 2.2 Especificación de Casos de Uso ASI 2.3 Análisis de Requisitos ASI 2.4 Validación de Requisitos	- Catálogo de Requisitos - Modelo de Casos de Uso - Especificación de Casos de Uso	- Catalogación - Sesiones de Trabajo - Casos de Uso	Definición, análisis y validación de los requisitos del sistema. Generación de la Línea Base
DSI 1.2 Identificación de requisitos de Diseño y Construcción	Catálogo de requisitos	- Sesiones de Trabajo - Catalogación	Se actualiza el Catálogo de Requisitos con aspectos técnicos
DSI 1.7 Especificación de Requisitos de Operación y Seguridad	- Procedimientos de Seguridad y Control de Acceso - Procedimientos de Operación y Administración del	- Sesiones de Trabajo - Catalogación	Se generan los procedimientos de seguridad para garantizar la protección del sistema
DSI 11 Establecimiento de Requisitos de Implantación - DSI 11.1 Especificación de Requisitos de Documentación de Usuario - DSI 11.2 Especificación de Requisitos de Implantación	Catálogo de requisitos	- Sesiones de Trabajo - Catalogación	Se actualiza el Catálogo de Requisitos con los datos relativos a la implantación y documentación de usuario
DSI-SEG 2 Especificación de Requisitos de Seguridad del Entorno Tecnológico	Requisitos de Seguridad del Entorno Tecnológico		El equipo de seguridad estudia el entorno tecnológico
DSI-SEG 3 Requisitos de Seguridad del Entorno de Construcción.	Requisitos de Seguridad del Entorno de Construcción	Sesiones de Trabajo	El equipo de seguridad estudia el entorno de construcción

Tabla 1: Resumen de los Requisitos en Métrica v3

3.3 CMMI

Para no perder capacidad competitiva, las empresas y organizaciones deben ofrecer sus productos y servicios mejores que la competencia, más rápidos y más económicos. Si a esta circunstancia, unimos las características actuales del mercado donde el *outsourcing* a nivel contractual y la integración de los componentes y/o sistemas a nivel de producto juegan un papel fundamental; los procesos de control sobre la gestión se hacen más necesarios [22].

El plan Avanza [23][24] es la iniciativa del Ministerio de Industria, Comercio y Deporte para fomentar la adopción de CMMI en el sector TI español. Como ejemplo, podemos citar al proyecto Vulcano y el trabajo realizado en definir los requisitos necesarios que tiene que satisfacer Vulcano para dar soporte a la adopción del modelo [25]. El proyecto Vulcano, perteneciente a la Iniciativa Española de Software y Servicios (Ines [26]) tiene como objetivo la promoción del Software Libre para facilitar la adopción del mismo por la industria [27].

3.3.1 Conceptos Generales de CMMI

CMMI (Capability Maturity Model Integration) [28] consiste en una serie de prácticas y procesos de alto nivel que ayudan a las organizaciones a construir un modelo para mejorar sus procesos.

El Software Engineering Institute (en adelante SEI) de la Carnegie Melton University [29] es el instituto que promueve CMMI tiene la siguiente premisa: “la calidad de un sistema o un producto está fuertemente influenciada por la calidad de los procesos empleados para desarrollarlo o para mantenerlo”.

Las tres dimensiones que las organizaciones comparten son: Personas, Herramientas y Procedimientos. Dado el dinamismo actual de las personas y de las herramientas, son los procesos a los que las organizaciones les pueden dotar de mayor independencia ante el cambio. Tal y como otras áreas de la Ingeniería llevan años realizando, será la calidad en los procedimientos los que tengan por objetivo el mantener y aumentar la productividad en las herramientas que utilizamos y los productos que se ofrecen.

CMMI se basa en *constelaciones*, que agrupa componentes CMMI en un modelo, materiales de aprendizaje y documentos de evaluación. En la actualidad, en la versión 1.2 [30], dispone de las siguientes *constelaciones*:

- CMMI for Acquisition v1.2 (2007). Aborda la cadena de suministro, adquisición y contratación externa.
- CMMI for Services v1.2 (2007). Aborda la gestión ofrecida por el servicio.
- CMMI for Development v1.2 (2006). Aborda los procesos de desarrollo (de productos y servicios).

En este trabajo nos centraremos en CMMI for Development, en adelante CMMI-DEV.

3.3.2 Elementos Principales de CMMI

CMMI tiene dos representaciones: la escalonada y la continua.

- La **Representación Continua** se basa en el conocimiento de los procesos y las dependencias entre procesos para llevarla a cabo. Ofrece mayor flexibilidad al permitir adaptar CMMI a los procesos elegidos. Cuando es elegida esta representación por parte de

una organización se dice que tiene Niveles de Capacidad. Existen 6 niveles de Capacidad: 0-5:

- 0- Incomplete (Incompleta). Se dice que un proceso alcanza el nivel incompleto cuando un proceso o bien está parcialmente implementado o no está realizado en su totalidad.
 - 1- Performed (Realizado). Representa un proceso que satisface los objetivos específicos pero que no está *institucionalizado* en la organización.
 - 2- Managed (Gestionado). Representa los procesos que estando en nivel 1, son además monitorizados, controlados y revisados.
 - 3- Defined (Definido). Son aquellos procesos que habiendo alcanzado el nivel 3, son gestionados de una forma proactiva utilizando el conocimiento de las interrelaciones con el resto de procesos de la organización. Son medidos y controlados los productos a generar por el proceso y los servicios a ofrecer.
 - 4- Quantitatively Managed (Cuantitativamente Gestionados). Son los procesos de nivel 3 que además son gestionados usando estadísticas y otras medidas cuantitativas que ayuden a determinar la calidad y el rendimiento del proceso.
 - 5- Optimizing (Optimizadores). Son aquellos que además de estar en nivel 4 tienen como foco la mejora continua y aplican medidas innovadoras.
- Para una organización la **Representación Escalonada** ofrece un camino estructurado y sistemático para adaptar todos los procesos a un determinado nivel del modelo. Cuando es elegida esta representación por parte de una organización se dice que tiene Niveles de Madurez. Existen 5 niveles de Madurez: 1-5:
 - 1- Inicial (Inicial). Los resultados dependen totalmente de la capacidad de los trabajadores. Son las organizaciones que no son capaces de repetir sus éxitos al no disponer de procesos.
 - 2- Managed (Gestionados). La organización tiene procesos establecidos que aseguran una planificación y ejecución. Estos procesos son monitorizados, controlados y revisados.
 - 3- Defined (Definidos). Los procesos están bien caracterizados y son estandarizados para toda la organización, con herramientas y métodos. Proporcionan consistencia a la organización.
 - 4- Quantitatively Managed (Cuantitativamente Gestionados). Se establecen medidas cuantitativas para poder medir el proceso y poder predecir rendimientos en los procesos. Es decir, las medidas cuantitativas se utilizan para conocer con mayor exactitud el proceso y poder realizar predicciones cuantitativas y no sólo cualitativas como en el nivel anterior.
 - 5- Optimizing (Optimizadores). Se centran en la mejora continua de los procesos que rigen la organización y en la aplicación de medidas innovadoras para alcanzar la optimización de los procesos.

CMMI utiliza en su framework los siguientes elementos principales (Figura 11: Elementos de CMMI 1.2):

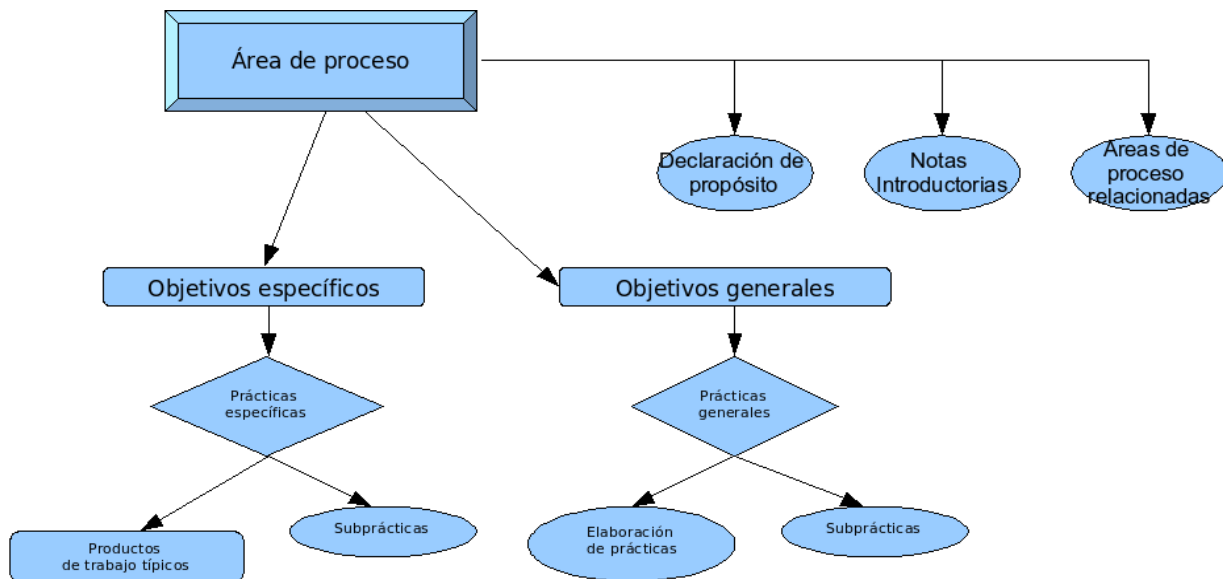


Figura 11: Elementos de CMMI 1.2

- **Áreas de Procesos (Process Area):** Es el conjunto de prácticas relacionadas con una área, que cuando se implementan conjuntamente, satisfacen una serie de objetivos para mejorar esa área.
- **Declaración de propósito (Purpose Statement):** Elemento informativo que nos indica el objetivo o finalidad del área de proceso.
- **Notas introductorias (Introductory Notes):** Describe los conceptos involucrados.
- **Áreas de proceso relacionadas (Related Process Areas):** Referencia a otras áreas de proceso relacionadas con el área de proceso en cuestión.
- **Objetivos específicos (Specific Goals):** Describe las características que se deben cumplir. Es un elemento obligatorio, que es además usado en las evaluaciones para validar si se ha alcanzado el nivel de una área de proceso.
- **Objetivos genéricos (Generic Goals):** Son características comunes que deben ser satisfechas por más de un área de proceso.
- **Prácticas específicas (Specific Practices):** Son actividades que se consideran importantes para conseguir el objetivo específico.
- **Prácticas genéricas (Generic Practices):** Son recomendaciones aplicables en más de un proceso para conseguir los objetivos.

Las 22 áreas de procesos que intervienen en el desarrollo, están agrupadas en cuatro categorías:

- **Gestión del Proceso (Process Management).** Contiene las actividades relacionadas con la definición, planificación, desarrollo, implementación, monitorización, control, evaluación, medición y mejora de los procesos. Se proporcionan dos formas interacción de los procesos: básica y la avanzada, en función de la capacidad de mejora de la organización. La figura 12 muestra las relaciones básicas entre las áreas de procesos que componen la Gestión del

Proceso:

- Organizational Process Focus (OPF)
- Organizational Process Definition + IPPD (OPD+IPPD)
- Organizational Process Performance (OPP)
- Organizational Training (OT)
- Organizational Innovation and Deployment (OID) (En la representación avanzada de la Gestión del Proceso aparece esta área)

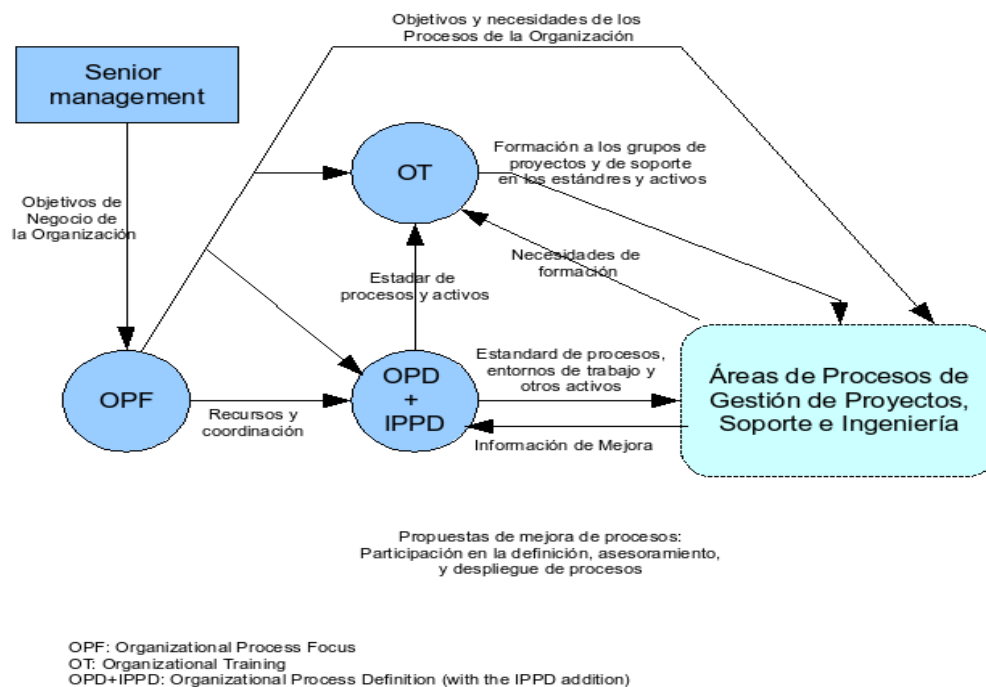


Figura 12: Área de Proceso Gestión de Proyectos Básico

- **Gestión del Proyecto (Project Management).** Esta categoría cubre las actividades de gestión del proyecto relacionadas con la planificación, monitorización y control del proyecto. La figura 13 muestra las relaciones básicas entre las siguientes áreas de procesos:
 - Project Planning (PP)
 - Project Monitoring and Control (PMC)
 - Supplier Agreement Management (SAM)
 - Risk Management (RSKM)
 - Quantitative Project Management (QPM)
 - Integrated Project Management + IPPD (IPM+IPPD) (En la representación avanzada interviene esta área)

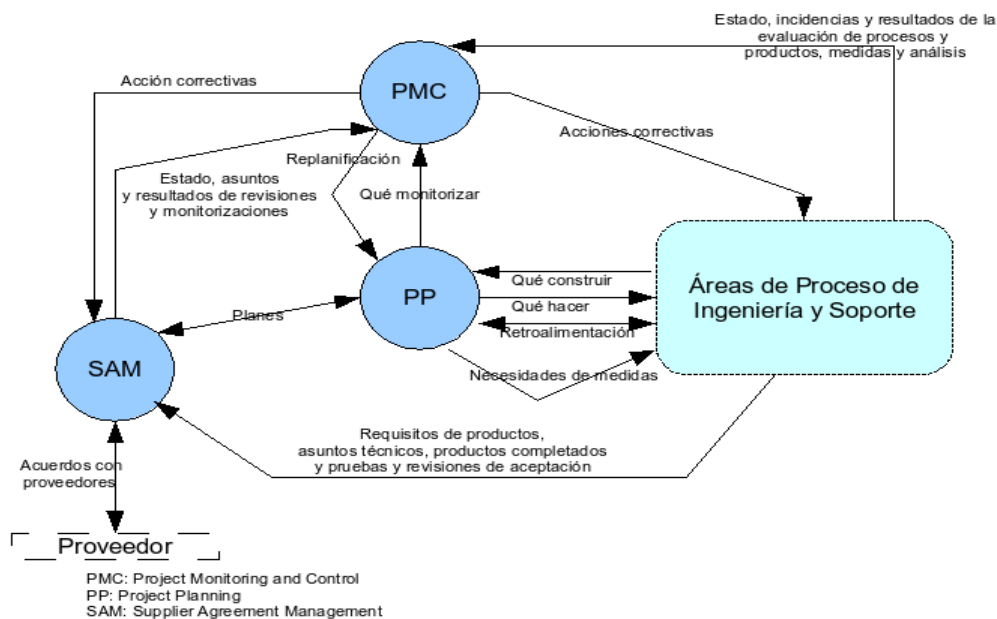


Figura 13: Área de Procesos Gestión de Proyecto Básica

- **Ingeniería (Engineering)**. Cubre el desarrollo y las actividades de mantenimiento que son compartidas a través de las disciplinas de ingeniería. En la figura 14 se muestra las relaciones entre las áreas de procesos:
 - Requirements Development (RD). Este proceso proporciona los requisitos para que la Solución Técnica (TS) pueda implementarlos y para que la Integración del Producto (PI) pueda combinar los productos obtenidos.
 - Requirements Management (REQM). Este proceso gestiona los requisitos a lo largo del ciclo de vida del proyecto.
 - Technical Solution (TS). Selecciona y desarrolla los productos que serán usados por el proceso Integración del Producto.
 - Validation (VAL). Este proceso valida que las necesidades de los clientes son satisfechas. Este proceso incluye la validación de los productos intermedios y los procesos seguidos.
 - Verification (VER). Este proceso asegura que los productos obtenidos satisfacen las necesidades de los usuarios. Es un proceso incremental que comienza con la verificación de los productos y termina con la verificación del producto ensamblado final. Ejemplos de criterios de verificación pueden ser: claridad, trazabilidad, consistencia, completitud, etc.
 - Product Integration (PI). Este proceso utiliza las prácticas específicas tanto de Verificación como de la Validación, para realizar la integración del producto. Las prácticas de Verificación comprueban las interfaces y los requisitos de dichas interfaces para que la integración pueda realizarse. Durante la integración del

producto en el entorno operacional, las prácticas específicas de la Validación también son utilizadas.

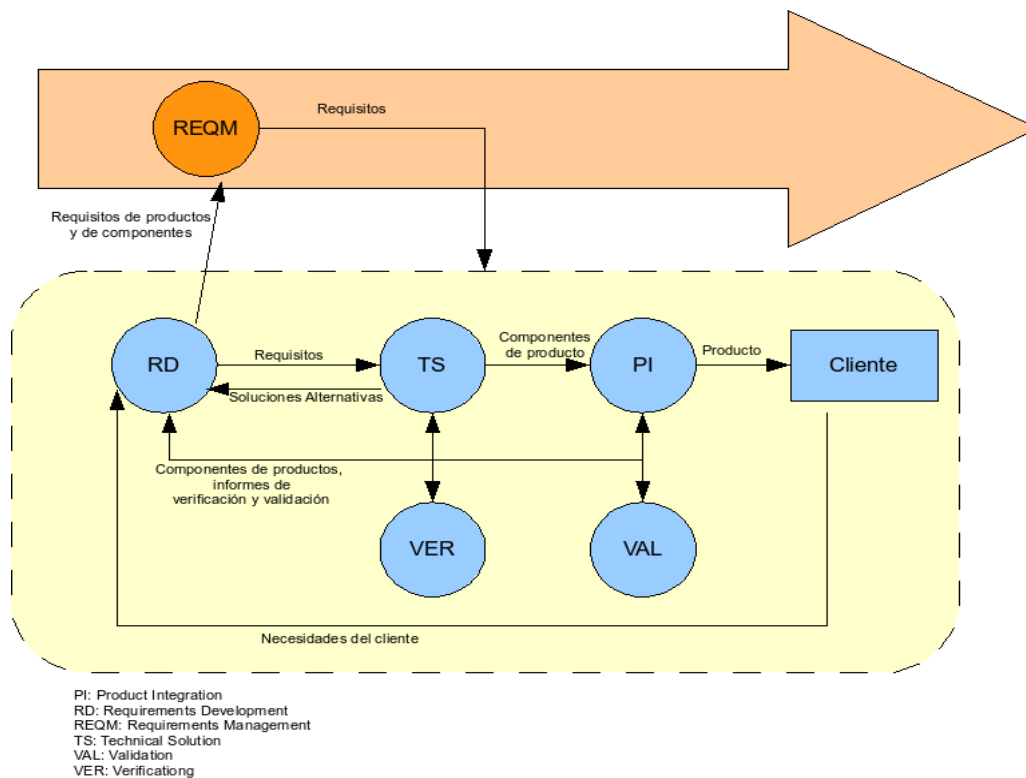


Figura 14: Área de Proceso Ingeniería

- **Soporte (Support).** El área de proceso de soporte cubre las actividades de soporte del producto tanto para el desarrollo y el mantenimiento. La figura 15 nos muestra las relaciones entre las áreas de procesos relativas al soporte:
 - Causal Analysis Resolution (CAR)
 - Decision Analysis and Resolution (DAR)
 - Configuration Management (CM)
 - Measurement and Analysis (MA)
 - Process and Product Quality Assurance (PPQA)

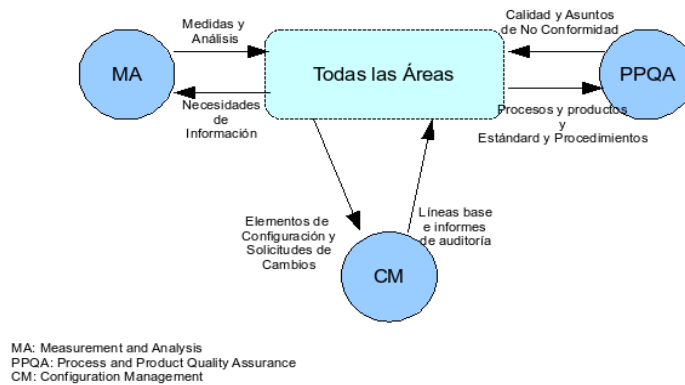


Figura 15: Área de Proceso Soporte Básica

Para terminar de tener una visión general de CMMI, falta por enlazar qué procesos son los requeridos para alcanzar un determinado nivel de madurez. La Tabla 2: Niveles de Madurez y Procesos nos muestra la correspondencia entre los procesos y los niveles.

Áreas de Procesos	Abreviatura	Nivel de Madurez
Requirements Management	REQM	2
Project Planning	PP	2
Project Monitoring and Control	PMC	2
Supplier Agreement Management	SAM	2
Measurement and Analysis	MA	2
Process and Product Quality Assurance	PPQA	2
Configuration Management	CM	2
Requirements Development	RD	3
Technical Solution	TS	3
Product Integration	PI	3
Verification	VER	3
Validation	VAL	3
Organizational Process Focus	OPF	3
Organizational Process Definition + IPPD	OPD + IPPD	3
Organizational Training	OT	3
Integrated Project Management + IPPD	IPM + IPPD	3
Risk Management	RSKM	3
Decision Analysis and Resolution	DAR	3
Organizational Process Performance	OPP	4
Quantitative Project Management	QPM	4
Organizational Innovation and Deployment	OID	5
Causal Analysis and Resolution	CAR	5

Tabla 2: Niveles de Madurez y Procesos

3.3.3 Requisitos en CMMI

Una vez resumidos los conceptos generales de CMMI, en esta sección se describen las áreas que se enfocan en el tratamiento de los requisitos.

3.3.3.1 Desarrollo de Requisitos (*Requirement Development RD*)

El desarrollo de Requisitos está englobado en el área de nivel madurez 3 y su propósito es identificar, desarrollar, generar, determinar y analizar las necesidades de los usuarios. Dichas necesidades se transformarán en las características que tendrá el producto final. CMMI distingue entre *producto* y *componentes del producto*. El segundo término alude a las posibles partes en las que se dividirá el producto final, en forma de subsistemas u otro mecanismo de división acordado previamente en la organización.

Los requisitos son la base del diseño. Dentro del desarrollo de los requisitos nos encontramos las siguientes actividades:

- Elicitación, análisis, validación y comunicación de las necesidades de los clientes, de lo que esperan y de las restricciones.
- Recopilación y coordinación de las necesidades de los involucrados en el negocio.
- Desarrollo del ciclo de vida de los requisitos.
- Establecimiento de los requisitos de cliente.
- Establecimiento de los requisitos de producto y de los componentes de forma consistente con los requisitos de los clientes.

Es importante recoger también los requisitos específicos de diseño que el cliente pudiera tener.

Los requisitos del cliente se irán refinando en requisitos del producto y de sus componentes.

Los objetivos más importantes de esta área de proceso son:

- Desarrollo de los Requisitos del Cliente, para que sean utilizados en el desarrollo de los requisitos del producto. Se definen los requisitos del cliente como el resultado de elicitar, consolidar, y resolver conflictos entre las necesidades, restricciones y expectativas de los participantes relevantes del producto. Estos requisitos se deben definir entendiendo que deben ser aceptables para el cliente, es decir en un lenguaje natural y sin descripciones técnicas.
- Desarrollo de los Requisitos del Producto, para que sean utilizados en el diseño del producto y de sus componentes. Los requisitos del producto son un refinamiento de los requisitos del cliente, expresados en el lenguaje técnico. Se derivan de los requisitos del cliente, los aspectos tanto explícitos como implícitos. En definitiva, es la solución que el ingeniero de requisitos propone para que el software desarrollado ayude a alcanzar los objetivos de negocio.
- Analizar y Validar los requisitos.

El grado de interrelación con el área de proceso Solución Técnica (TS) es muy alto. De los requisitos se obtendrá el **análisis funcional**. También será necesario la definición de la **arquitectura funcional**, que comprende la agrupación de la funcionalidad y sus relaciones. Este análisis es un proceso iterativo hasta que se consiga suficiente detalle como para que otras áreas, como el diseño o el testing, dispongan de los elementos necesarios con el nivel de calidad aceptable

para realizar sus actividades. Como resultado de este análisis iterativo, obtendremos requisitos del tipo restricciones, limitaciones tecnológicas, conceptos operacionales etc.

Dentro del desarrollo de Requisitos, CMMI propone los siguientes objetivos y prácticas relacionadas:

- **SG 1 Desarrollar los requisitos del cliente.** El término cliente acuña todos los perfiles involucrados en el desarrollo del Software: desde los propios clientes, hasta usuarios finales pasando por proveedores. El objetivo es recoger toda la información necesaria para conseguir un conjunto de requisitos del cliente, donde las expectativas, restricciones, interfaces, conceptos operacionales, legales, etc estén presentes.
 - **SP 1.1 Elicitar las necesidades.** Se deberá tener en cuenta que esta práctica debe ir más allá de una simple recopilación, ya que deben obtenerse a través de las técnicas disponibles aquellos requisitos que no se indican directamente.
 - **SP 1.2 Desarrollar los requisitos del cliente.** Toda la información recogida en la práctica anterior, debe ser consolidada, eliminando los posibles conflictos.
- **SG 2 Desarrollar los requisitos del producto.** En el proceso iterativo pasaremos del conjunto de los requisitos del cliente a un conjunto de requisitos del producto y de sus componentes Este refinamiento trata de mapear los requisitos del cliente a un nivel de producto. Este mapeo incluye procesos, objetos y personal. Esto constituye la base para la solución técnica. Conviene documentar la trazabilidad de este mapeo.
 - **SP 2.1 Establecer los requisitos del producto y de los componentes.** En este proceso expresaremos los requisitos en terminología técnica desde el punto de vista del sistema.
 - **SP 2.2 Asignar los requisitos a cada componente.** En el área de proceso de TS (Technical Solution) proporciona información acerca de cómo establecer la asignación.
 - **SP 2.3 Identificar los requisitos de interfaz.** Las interfaces entre la funcionalidad debe especificarse. Como la práctica anterior se debe consultar al área de proceso TS.
- **SG 3 Analizar y validar los requisitos.** Una vez que se dispone de un conjunto de requisitos, éstos deben ser analizados y validados para tener la garantía de que los requisitos son coherentes con las intenciones de los Casos de Uso y no hay conflictos entre ellos.
 - **SP 3.1 Establecer conceptos y escenarios operacionales.** Los escenarios son las secuencias que nos muestran cómo el sistema se va a comportar. Los conceptos operacionales van a depender del diseño elegido y del escenario. Los conceptos operacionales son refinados como soluciones, las decisiones se van tomando y los requisitos con un mayor grado de detalle son desarrollados.
 - **SP 3.2 Establecer una definición de la funcionalidad requerida.** Se realizada el análisis funcional del sistema.
 - **SP 3.3 Analizar los requisitos.** Los requisitos son analizados para tener la garantía de que los objetivos marcados se están satisfaciendo sin conflictos entre ellos.
 - **SP 3.4 Analizar los requisitos para conseguir un equilibrio.** Una vez que los requisitos están analizados, hay que revisar si las restricciones y las necesidades conllevan un riesgo, marcan la planificación, etc.
 - **SP 3.5 Validar los requisitos.** Los requisitos son validados para asegurarse que el

producto obtenido será el necesario.

3.3.3.2 *Gestión de Requisitos (Requirement Management REQM)*

El objetivo de la Gestión de Requisitos es gestionar los requisitos del producto y de sus componentes e identificar las posibles inconsistencias entre estos y la planificación realizada y los productos a obtener dentro del proyecto.

Pertenece al nivel 2 de madurez y debe recoger toda clase de requisitos que se vayan generando a lo largo de la ejecución del proyecto. Los requisitos deben ser revisados y aprobados constituyendo una línea base, antes de ser incorporarlos y el área de Gestión de Proyectos debe revisar la planificación.

Parte de esta gestión consiste en documentar los cambios y mantener la trazabilidad bidireccional entre los requisitos y los productos que se generan.

Esta área de proceso tiene importantes relaciones con otras áreas de proceso: Desarrollo de Requisitos, Solución Técnica, Planificación del Proyecto, Gestión de la Configuración y el Control y la Monitorización de Proyectos. Es la trazabilidad de los requisitos la que va a determinar parte de la calidad del producto, al controlar que se cumplen e implementan las necesidades del negocio.

- **SG 1 Gestión de Requisitos.** Se debe mantener el conjunto de requisitos aprobados y actualizados a lo largo de todo el proyecto, realizando:
 - Gestionando las peticiones de cambio y su resolución
 - Gestionando las relaciones y las posibles inconsistencias entre los requisitos, la planificación del proyecto y los productos o entregables definidos.
 - Tomando acciones correctivas.

Las prácticas asociadas con este objetivo son:

- **SP 1.1 Obtener acuerdo e interpretación de los requisitos.** Hay que definir los canales (vías de comunicación) por dónde los nuevos requisitos se van a obtener. Todo cambio debe ser aprobado y debe ser analizado dentro del conjunto del proyecto.
- **SP 1.2 Obtener un compromiso con los requisitos.** Mientras que la práctica anterior se centra en recoger las nuevas peticiones, en entenderlas y atenderlas, esta práctica debe trasladar los requisitos para que puedan ser implementados. (proceso de derivación)
- **SP 1.3 Gestionar los cambios de los requisitos.** Tanto las actividades de añadir requisitos como un cambio en ellos debe ser controlado de una forma eficaz y efectiva y establecerse las actividades necesarias para controlar el impacto del cambio.
- **SP 1.4 Mantener una trazabilidad bidireccional de los requisitos.** Identificar tanto el requisito como los productos en los que se ve reflejado es necesario para verificar que que el requisito ha sido realizado así como en el control del impacto.
- **SP 1.5 Identificar inconsistencias entre los requisitos y los productos del proyecto,** para que se puedan llevar a cabo las acciones correctivas.

3.3.4 Conclusiones

Siendo interesantes las recomendaciones de CMMI en cuanto a los requisitos, es necesario establecer los procesos que se van a llevar a cabo para alcanzar los objetivos que proponen las áreas de proceso. Como recomendaciones que son, no proponen una taxonomía de requisitos, los atributos que estos deben contemplar, ni el ciclo de vida que deben seguir.

De la misma forma, la validación y verificación, como actividades fundamentales en el desarrollo y gestión de los requisitos, tampoco son expuestas con detalle.

CMMI no es una metodología, sino un conjunto de buenas prácticas. Por su gran aceptación, hay varios trabajos que proponen o adaptan de forma metodológica un proceso concreto, como Xavier Franch en [31].

En la figura 16 se resumen las recomendaciones que CMMI propone acerca del tratamiento de los requisitos en las dos áreas de proceso.

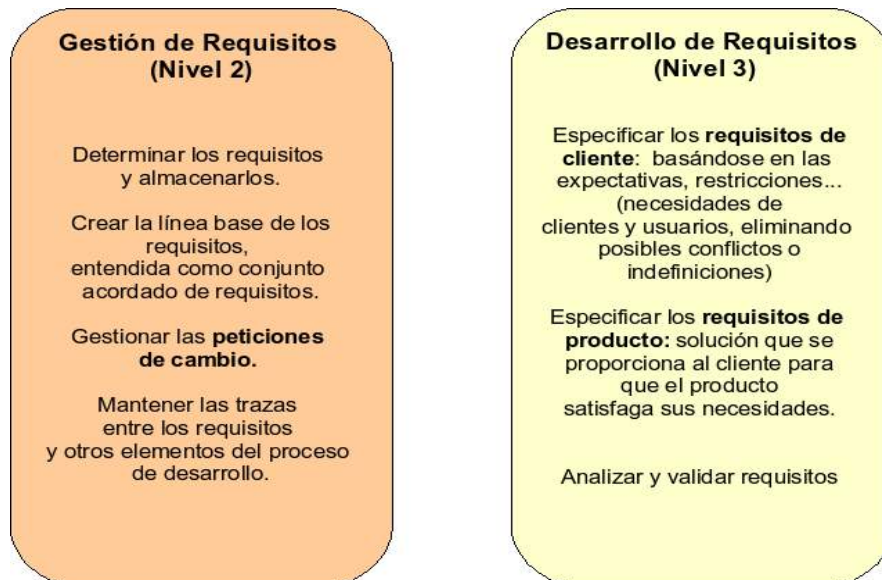


Figura 16: Resumen tratamiento Requisitos en CMMI v1.2

3.4 ITIL

Hoy en día todo tipo de organización tiene una gran dependencia de la tecnología, como es sobradamente conocido. ITIL proporciona un marco de referencia para alinear la tecnología con las líneas estratégicas de negocio de dicha organización.

Hoy en día ITIL es el estándar de facto en la Gestión del Servicio.

3.4.1 Conceptos Generales de ITIL

ITIL fue creado a finales de la década de los ochenta por la Central Computer and Telecommunication Agency (CCTA) del Reino Unido. A día de hoy está regulado y patentado por el Ministerio de Comercio (OCG) pero externalizado también a las empresas privadas por el itSMF (Information Technology Service Management Forum) [32]. Dicha organización centra sus actividades, para el ámbito privado, en la difusión de ITIL.

En el 2000 fue publicada la versión 2 de ITIL, teniendo en el 2007 ITIL V3 [33].

ITIL se define como una biblioteca que describe las buenas prácticas en la Gestión de Servicios. Al igual que CMMI, no es una metodología ya que no definen el cómo sino que atienden al qué.

El valor de ITIL radica en que enfoca los servicios tecnológicos como activos de la organización, formando parte fundamental del plan estratégico de la organización. Es decir, los servicios que ofrece cualquier organización a sus usuarios y/o clientes, se basan en una infraestructura tecnológica. Dicha infraestructura hay que dotarla y prepararla para que sea capaz de responder ante las prioridades y retos que vaya marcando las directrices estratégicas, tal y como se ilustra en la figura 17.

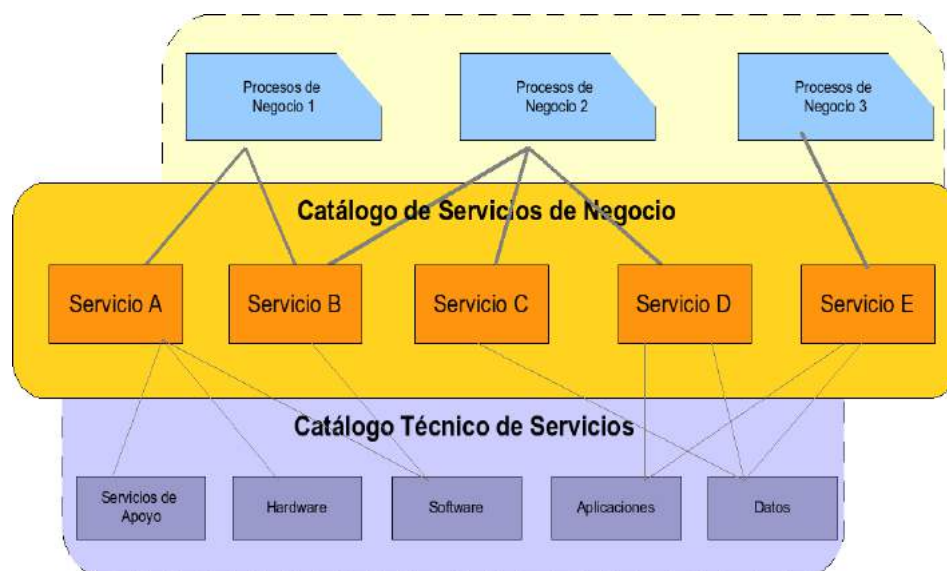


Figura 17: Servicios en ITIL

ITIL v2 proporciona estructura en los procesos, mientras que la evolución a v3 proporciona mejora continua en los servicios. Si se piensa en una implantación de ITIL en una organización, primero habría que comenzar por implantar los procesos de la v2, para poder pasar a la mejora continua propuesta por la v3.

Esta última versión integra los procesos y funciones de la ISO 20.000, ya que es el estándar de la

Gestión del Servicio de TI. Dichos procesos son cubiertos dentro de los procesos definidos en ITIL en la Estrategia del Servicio y Diseño del Servicio, que profundizaremos más adelante.

Es decir, aumentando el grado de abstracción y teniendo en cuenta que ITIL no es una metodología, tendríamos en la base los procedimientos y procesos implantados en la organización, guiados por las buenas prácticas de ITIL. Teniendo en la parte superior, la ISO 20.000, tal y como se muestra en la figura 18. La certificación de la organización se realiza conforme a la ISO 20.000, siendo las personas las que se certifican en ITIL.

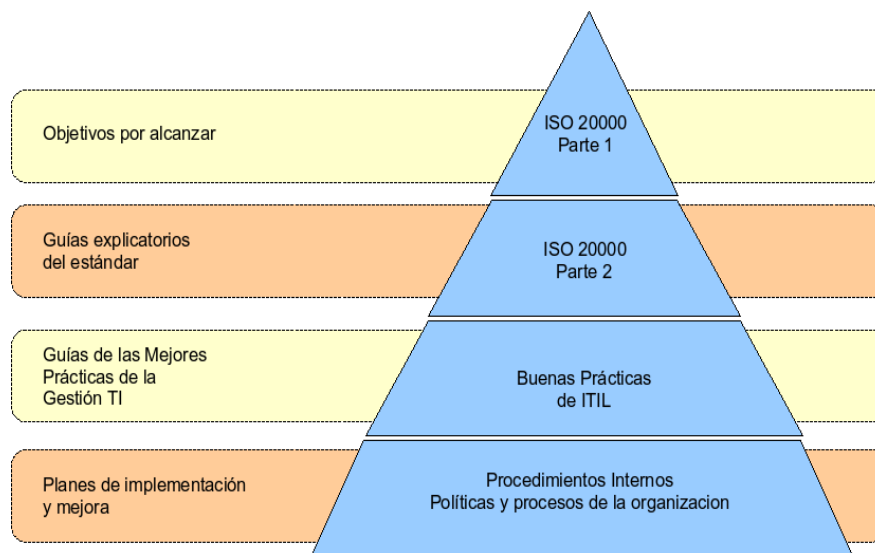


Figura 18: Relación entre ITIL v3 e ISO 20.000

Una de las características que enfatiza ITIL es la **orientación al proceso** que debe tener una organización. Esta orientación al proceso debe estar embebida en la organización, para que independientemente del producto a obtener, los procesos garanticen un producto que se pueda medir y comparar con productos anteriores. Con esta perspectiva es donde la mejora continua puede llevarse a cabo.

El **proceso** es, por tanto, el conjunto de actividades que se desarrollan para alcanzar un objetivo concreto. Estos objetivos tienen que aportar valor dentro de la organización, ya sea por su impacto en la mejora interna de la organización, ya sea por la mejora de los servicios ofrecidos a los clientes. Los procesos son un medio para la prestación de servicios.

Los procesos tienen una entrada y una salida. La salida del proceso es el producto. Para estructurar el proceso y poder definir los roles que intervienen, los procesos se componen de actividades. Las actividades desempeñadas en el proceso, están basadas en procedimientos, a los que se les asocian los distintos roles y responsabilidades.

Las actividades son el conjunto de procedimientos, ejecutados de forma orquestada. Los procedimientos son el conjunto de tareas y de información ejecutadas siguiendo un flujo de trabajo.

Un **Servicio de TI**, consiste por tanto, en el conjunto de personas, procesos y la tecnología que satisfacen un requisito del negocio. En el libro “Implementing Service and Support Management Processes: A Practical Guide” [34], un servicio se define como “función empresarial proporcionada por uno o más componentes del Servicio de TI (hardware, software e instalaciones) para uso del negocio”. A partir de este momento, siempre nos referiremos a servicio como servicio TI.

Hay una herramienta que se utiliza de apoyo para la definición de roles: Modelo RACI (Responsible, Accountable, Consulted, Informed). El objetivo es ayudar a definir qué personas están encargadas de qué actividades, quienes son los responsables del servicio, a quienes se tiene que informar y consultar.

3.4.2 Elementos Principales de ITIL

La versión 2 de ITIL consta de 7 libros, aunque dos de ellos son el núcleo. Estos dos libros son los siguientes:

- **Service Support** (Provisión del Servicio) [35]. Proporciona una visión táctica del negocio. Engloba los siguientes procesos:
 - Service Level Management (Gestión de Niveles del Servicio): Determina y se responsabiliza de los niveles de servicio que la organización requiere para cumplir con los objetivos del negocio.
 - Financial Management (Gestión Financiera): Gestión financiera de los servicios.
 - Capacity Management (Gestión de la Capacidad): Gestiona que la infraestructura sea la óptima para las necesidades definidas.
 - Continuity Management (Gestión de la Continuidad): Gestiona la continuidad de los servicios.
 - Availability Management (Gestión de la Disponibilidad): Optimiza la disponibilidad de los sistemas.
- **Service Delivery** (Soporte al Servicio). Proporciona una visión operacional. Tiene un componente llamado Centro de Servicios o ServiceDesk, que debe ser el punto único de entrada y gestión de los servicios. El soporte al servicio, engloba los siguientes procesos:
 - Incident Management (Gestión de Incidentes): Tiene por objetivo restablecer la operativa normal de la organización, tan rápido como sea posible. Un incidente es cualquier circunstancia que provoca que la operativa normal o prestación de servicios no se produzca con la calidad esperada.
 - Problem Management (Gestión de Problemas): Minimiza el impacto de las incidencias y los problemas, buscando las causas a todas las incidencias y previniendo su aparición, tanto de forma proactiva como reactiva.
 - Configuration Management (Gestión de la Configuración): Implementa la CMDB (Configuration Management Data Base) que contiene a los distintos CI (Configuration Items). Los CI son los elementos de la infraestructura que son necesarios para proporcionar el servicio, desde una máquina hasta la definición del responsable del servicio de negocio. Facilita el control del impacto de los CI en la organización, ya que las relaciones entre CI son diseñadas en la CMDB. Gracias a estas relaciones entre estos elementos se puede calcular el impacto en la organización de cada CI ante un posible estado inoperativo del mismo.
 - Change Management (Gestión de Cambios): Gestiona los cambios necesarios en la infraestructura, de forma que se hagan de manera controlada, justificada y autorizada.
 - Release Management (Gestión de Entregas): Gestiona la puesta en producción de los

servicios y de los cambios en los CI.

Si los libros de la versión anterior acercaban la perspectiva del negocio con la infraestructura, ITIL v3 amplía la biblioteca mejorando el ciclo de vida de los servicios. Es en esta versión de ITIL donde el trabajo se va a centrar, al ser la más actual.

Para disponer de una visión completa, se van a enumerar todos los procesos, aunque se tratarán aquellos que sean de interés manifiesto para el alcance del trabajo.

Se puede observar, a continuación, cómo la v3 engloba y amplía los procesos descritos anteriormente. De hecho, la v2 sigue estando operativa, y se recomienda que se empiece por ella si se desea implantar ITIL en cualquier organización.

- **Service Strategy** (Estrategia del Servicio) [36]: Su objetivo principal es diseñar, desarrollar e implantar la gestión del servicio como un valor de la estrategia corporativa. Se compone de los siguientes procesos:
 - Financial Management (Gestión Financiera).
 - Demand Management (Gestión de la Demanda).
 - Service Portfolio Management (Gestión del Portfolio de Servicios). El portfolio de servicios es el catálogo donde residen todos los servicios de la organización, independientemente de su estado (en explotación o en desarrollo). La parte visible a los clientes del Portfolio es el Catálogo de Servicios, que es el compendio de los Servicios activos (en producción) de la organización.
- **Service Design** (Diseño del Servicio): Su objetivo es diseñar y desarrollar los servicios que cumplan con la estrategia anterior. Se compone de los siguientes procesos:
 - Service Catalogue Management (Gestión del Catálogo del Servicio). Define detalles de los servicios existentes y de los procesos de negocio a los que da soporte.
 - Service Level Management (Gestión de Niveles de Servicio). Es el proceso responsable de que los niveles pactados del servicio se cumplan.
 - Capacity Management (Gestión de la Capacidad). Asegura que la capacidad de los recursos necesarios para proporcionar el servicio con los niveles pactados, sean los adecuados. La capacidad engloba a todos los recursos técnicos, tanto servidores, memoria, etc.
 - Availability Management (Gestión de la disponibilidad). Asegura que la disponibilidad de los servicios sean los acordados.
 - IT Service Continuity Management (Gestión de la continuidad del Servicio). Asegura la recuperación de los servicios en los niveles pactados.
 - Information Security Management (Gestión de la Seguridad de la Información). Se ocupa del cumplimiento de las políticas de seguridad.
 - Supplier Management (Gestión de proveedores). Gestiona a los distintos proveedores.
- **Service Transition** (Transición del Servicio): Su objetivo principal es el desarrollo y mejora de los servicios que se desean llevar a producción. Se compone de los siguientes procesos:
 - Transition Planning and Support (Planificación y Apoyo de la Transición). Gestiona los recursos involucrados en la puesta en marcha del servicio.

- Change Management (Gestión de Cambios). Gestiona que los cambios estén controlados.
- Service Asset and Configuration Management (Gestión de Activos y Configuración del Servicio). Gestiona y controla los distintos activos necesarios en los servicios.
- Release and Deployment Management (Gestión de Versiones y Despliegue). Construye, libera y despliega los servicios.
- Service Validation and Testing (Validación y Prueba del Servicio). Valida que el servicio cumple con los objetivos establecidos en la definición del servicio.
- Evaluation (Evaluación). Controla si el rendimiento acordado del servicio es el adecuado.
- Knowledge Management (Gestión del Conocimiento). Gestiona que la información sobre el servicio sea conocimiento para todos los involucrados.
- **Service Operation (Operación del Servicio):** Su objetivo es la eficiencia y efectividad de los servicios implantados. Se compone de los siguientes procesos:
 - Event Management (Gestión de Eventos). Gestiona los eventos producidos y que son importantes para el servicio.
 - Incident Management (Gestión de Incidencias). Gestiona las incidencias producidas para un restablecimiento lo más pronto posible del servicio o un impacto menor.
 - Request Fulfillment (Gestión de Peticiones). Gestiona las peticiones de los usuarios.
 - Problem Management (Gestión de Problemas). Analiza las causas de las incidencias e intenta que no se vuelvan a producir.
 - Access Management (Gestión de Acceso). Gestiona los accesos a los servicios.
 - Monitoring and Control (Monitorización y Control). Monitoriza y controla continuamente los servicios.
 - IT Operations (Operaciones de TI). Gestiona las actividades necesarias diariamente para la infraestructura del servicio.

- **Continual Service Improvement** (Mejora Continua del Servicio): Enlaza los procesos de Diseño, Transición y operación, con un enfoque de mejora continua. En lugar de un proceso propiamente dicho, son los pasos que hay que realizar para la mejora continua: ¿Qué medir?, ¿Qué se puede medir?. Medir, Procesar, Analizar, Informar y ejecutar acciones correctivas. La figura 19 muestra el ciclo del proceso de mejora continua :

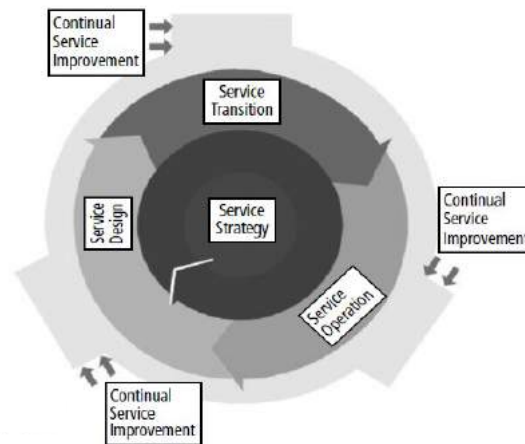


Figura 19: Ciclo de Mejora Continua en ITIL v3

3.4.3 Requisitos en ITIL

Cuando se analiza ITIL para integrar esta propuesta dentro de este trabajo, se observa inicialmente que ITIL se centra en los procesos del servicio. Aun así, sí que incluye los requisitos como mecanismo de explicitar las necesidades de los clientes. Los requisitos son tratados fundamentalmente en el Diseño del Servicio aunque es la Estrategia del Servicio la que genera las necesidades en la Gestión del Portfolio de Servicios. Hay que resaltar que ITIL en su nueva versión ha integrado la gestión de la aplicación (Application Management) en la vida del ciclo del servicio, aunque no es un modelo de procesos centrado en la gestión del desarrollo [37][38].

¿Qué son los objetivos de un proceso sino los requisitos del proceso? ¿qué es la definición de los servicios que quiero ofrecer sino los requisitos del mismo? ¿qué son los acuerdos de nivel de servicio sino requisitos?

En el nivel de definición en el que se mueve ITIL, hay dos tipos de requisitos:

- Requisitos de usuario. Es decir, las necesidades que tiene el usuario o los procesos de negocio.
- Requisitos de los procesos. Es decir, qué exigencias tienen los distintos procesos entre sí al interrelacionar en su funcionamiento.

Veamos cómo se tratan y qué papel juegan. En la figura 20 se observa la relación que tienen los procesos anteriormente definidos.

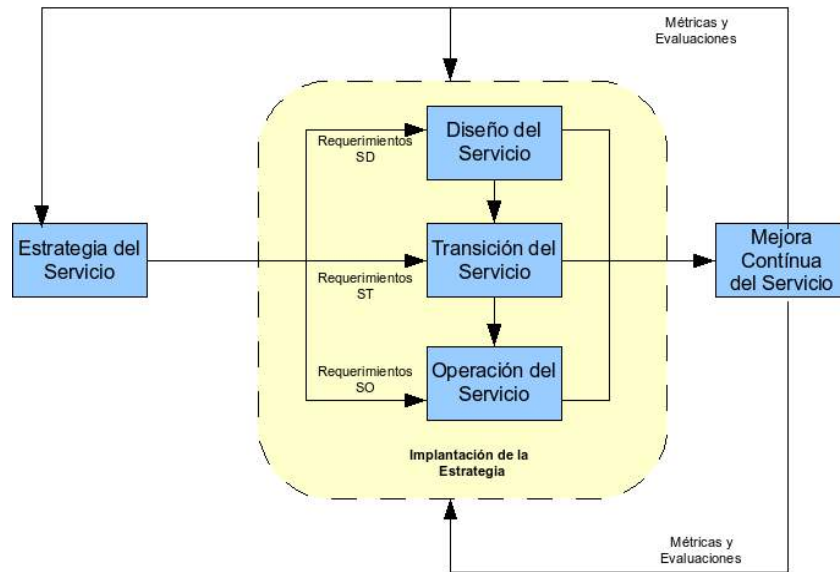


Figura 20: Relaciones entre procesos y los Requisitos

Conforme un servicio se va definiendo, construyendo e implementando va atravesando los procesos que se definen del ciclo de vida del servicio. Para cada etapa que va atravesando se deben definir los requisitos que se tienen que cumplir. La definición de estas necesidades, es la clave para que se pueda medir y evaluar si se está cumpliendo con lo acordado o necesario. Esta definición de las necesidades es la base para se pueda llevar a cabo un control , y así ante las posibles desviaciones entre lo solicitado y lo realizado se pueda realizar la mejora continua.

Antes de entrar en ciertos elementos claves de ITIL, veamos cómo los requisitos no son elementos sólo internos a los procesos. La figura 21 nos muestra que la entrada a la Estrategia del Servicio, son, por un lado de los requisitos del negocio y/o del cliente y por otro de las retroalimentaciones del resto de procesos.

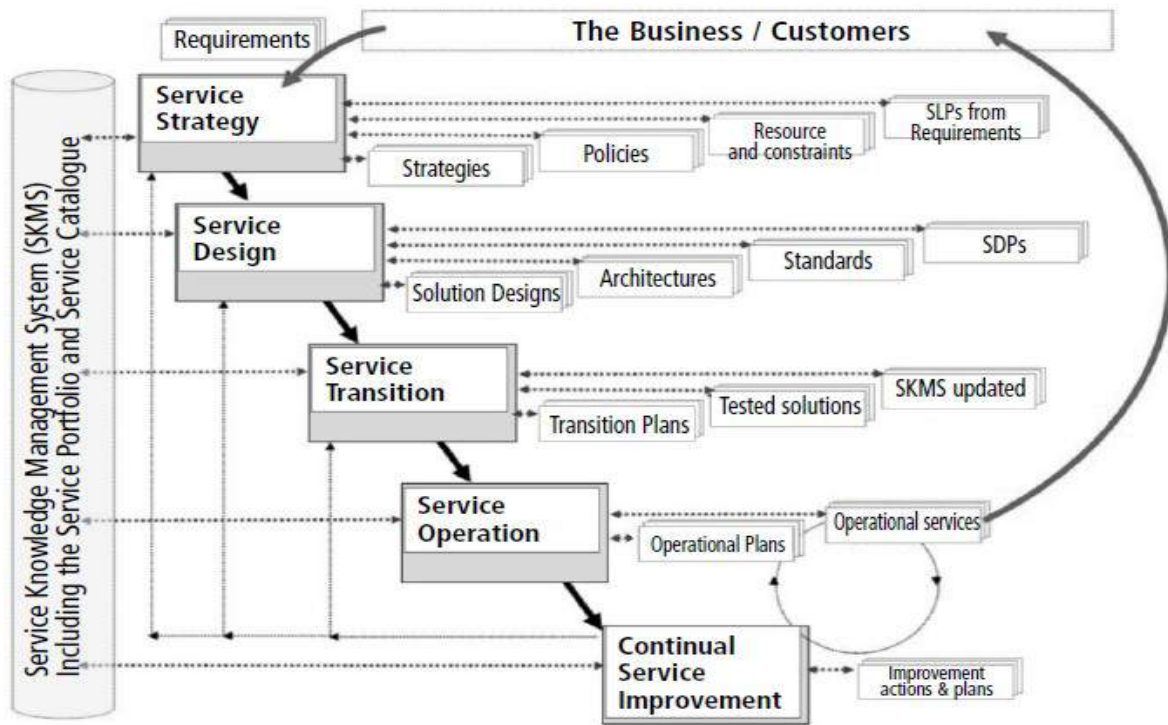


Figura 21: Componentes de ITIL v3

Los elementos destacados en este ciclo de vida del servicio son conforme a [39]:

- El **Paquete de Nivel del Servicio** (Service Level Package-SLP) procede directamente de los requisitos tanto del negocio como del cliente y consiste en una descripción detallada de los servicios ofrecidos. Estos requisitos deben estar alineados con la estrategia del negocio. Los SLP son la entrada al Service Design, donde se encontrará una implementación a los requisitos planteados.
- El **Paquete del Diseño del Servicio** (Service Design Package-SDP) son los documentos que describen las características de un Servicio y sus requisitos según el estado en que se encuentren. El ciclo de vida de los servicios, se define en el proceso de Gestión del Catálogo de Servicio, dentro del Service Design. Veremos en puntos posteriores del trabajo, qué elementos dentro del Software Design son los más relevantes desde el punto de vista de este trabajo.

- Los SDP son la entrada al Service Transition, donde la implementación anterior, es evaluada, probada y validada. Se actualiza el Service Knowledge Management System (SKMS).
- El servicio se pasa a producción por el Service Operation, que es el proceso que se va a encargar de su mantenimiento y explotación. Cuando un servicio está en explotación, es cuando se va a poder interactuar con el usuario (gestionado a través de las incidencias y las peticiones). Se necesita la relación directa entre Service Operation y la entrada de los requisitos del negocio/ cliente.

Dentro del **Service Design** existe un proceso fundamental desde el punto de vista de este trabajo: Gestión de Niveles de Servicio. Este proceso acuerda y negocia el nivel de servicio, y se responsabiliza de que dichos niveles sean medidos, documentados y que concuerden con la estrategia del negocio definida.

ITIL considera los requisitos en los siguientes documentos:

- **Requisitos de Nivel de Servicio** (Service Level Requirement-SLR), que es un documento formal donde se recogen requisitos del Cliente.
- **Criterios de Aceptación del Servicio** (Service Acceptance Criteria-SAC), representan el conjunto de criterios para garantizar que un servicio cumple con la funcionalidad y requisitos de Calidad.
- **Acuerdos de Nivel de Servicio** (Service Level Agreement-SLA), acuerdo entre el proveedor y el cliente
- OLA (Operational Level Agreement-OLA) y UC (Underpinning Contract-UC). Son respectivamente los SLA realizados entre partes diferentes dentro de la organización y los UC, son SLA firmados por la organización y un tercero.

Estos documentos son fundamentales a la hora de que la provisión del servicio sea traduzca en un proceso formal y no se base en percepción o expectativas subjetivas. La formalización de cualquier elemento dentro del proceso de productos de TI, su elaboración y explotación, son decisivos a la hora de establecer la informática como una ingeniería real. De la misma manera, transformar estos elementos de lenguaje natural a cuestiones computables y medibles dentro del producto es también tarea dentro de la ingeniería.

Las actividades desarrolladas dentro del Service Transition vienen activadas por una **Petición de Cambio (Request for Change-RFC)** cuando el servicio está activo.

En la figura 22 se muestran cómo se van generando los distintos tipos de documentos y por tanto definiendo y ampliando la definición del servicio, en función de la fase en la que se encuentre el producto.

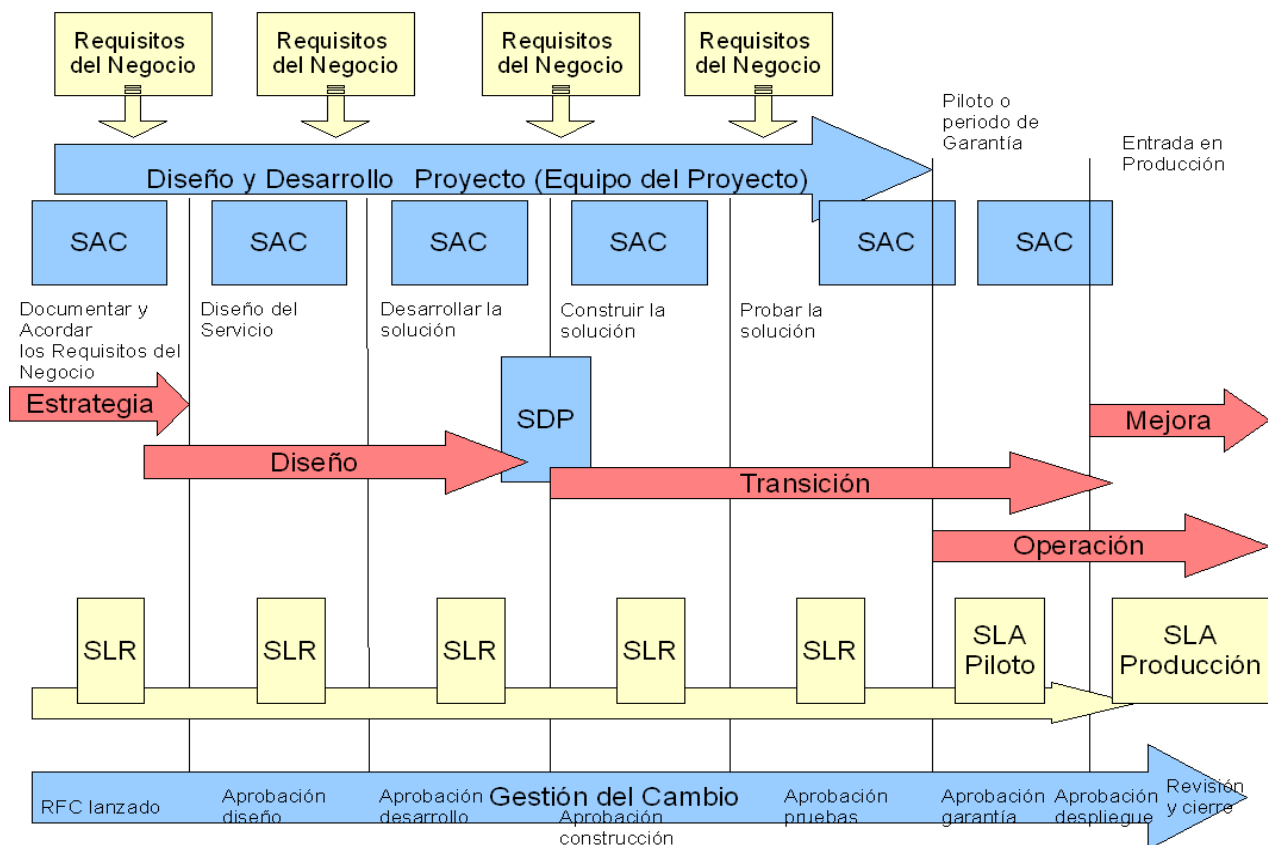


Figura 22: Relaciones entre los productos y procesos en ITIL v3

Una RFC procede por alguna de estas causas:

- Incidencia (Gestión de incidencias)
- Petición. Por cambios en los requisitos, ya sean solicitados por el usuario, mantenimiento adaptativo o perfectivo, cambios en normativa, etc. (Gestión del Servicio, de la Capacidad, de la Disponibilidad, etc)
- Consecuencia de solventar un error conocido (Gestión de Problemas)

Dentro de los datos que se exijan en las RFC, debería aparecer, entre otros:

- Código que identifica a la RFC y al Problema, en caso de proceda de la Gestión de Problemas.
- Descripción del cambio, identificando los CI afectados por el. Evaluación del impacto.
- Razón del Cambio.
- Riesgos y consecuencias de no realizar el cambio.
- Prioridad del cambio.
- Recursos necesarios en el cambio
- Aprobación del cambio.

- Plan de vuelta-atrás.

La exigencia que tienen los usuarios de los servicios, le confiere a este proceso una gran importancia, ya que un cambio incontrolado tiene una alta probabilidad de una reducción en la calidad de los servicios, ya sea por un fallo a nivel de infraestructura o por requisitos mal implementados.

Es crucial que dichos cambios sean aprobados, para que se tenga control y respaldo de aquello que se está realizando. Una buena implementación de los procesos de Gestión del Cambio y Gestión de la Configuración reducirán tantos los riesgos, como los costes e impacto que toda evolución de los sistemas conlleven.

Por último, ITIL Application Management [40] , que es el libro de ITIL dónde se aborda el ciclo de desarrollo, proporcionando detalles de cómo ante los cambios del negocio se deben producir definiciones claras de los requisitos y las soluciones aportadas ante las necesidades de los usuarios, clasifica los requisitos de la siguiente forma:

- Functional Requirements (Requisitos Funcionales)
- Non-Functional Requirements (Requisitos No Funcionales)
- Usability Requirements (Requisitos de Usabilidad)
- Change Cases (Casos de Cambio)
- Testing Requirements (Requisitos de Pruebas)
- Requirements management checklist (Lista de gestión de requisitos)
- Organization of the requirements team. (Organización del equipo de requisitos)

3.5 CMMI-ITIL

Existe una fusión de CMMI e ITIL realizada por la empresa Wibas [41] en colaboración con la universidad alemana de Darmstadt [42], estando autorizado tanto por el SEI [29] como por OCG [43].

Según el trabajo que han realizado [44], han integrado ITIL en CMMI bajo los principios que se observan en la figura 23. Esta muestra cómo se complementan CMMI-DEV e ITIL, al cubrir CMMI-DEV el desarrollo del producto e ITIL la producción del producto.

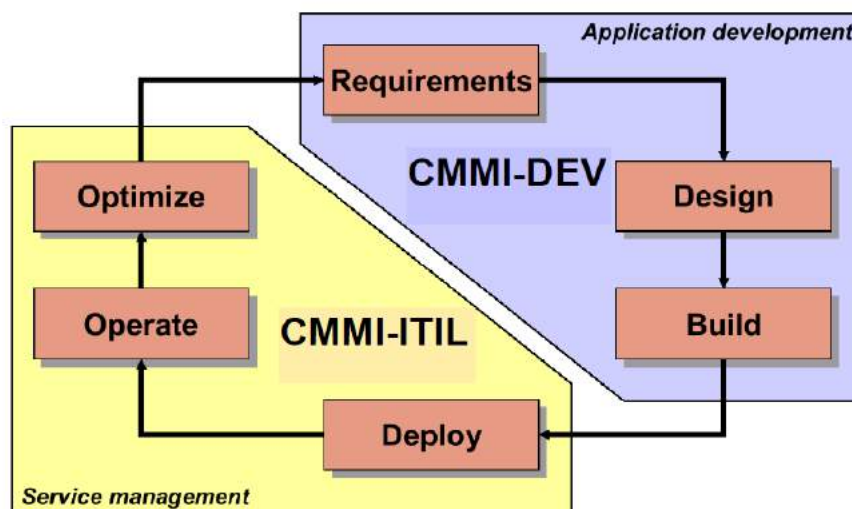


Figura 23: Relación entre CMMI-ITIL y CMMI-DEV

La arquitectura CMMI 1.2 ha hecho posible que se pueda integrar ITIL como parte de la familia CMMI. De esta fusión, han obtenido un modelo CMMI especializado en operaciones TI. Han desarrollado un metamodelo que cubre tanto los elementos de ITIL como de CMMI.

Destacan las diferencias con CMMI for Services (en adelante CMMI-SVC), ya que:

- CMMI-ITIL es una integración de ITIL en CMMI, de tal manera que los esfuerzos realizados independientes en ambas direcciones, no son incompatibles. Esta integración permite tener un modelo especializado en TI. Se basa en Service Delivery de ITIL, por lo que es bastante completo.
- CMMI-SVR no se centra en servicios TI.

CMMI-ITIL está estructurada en procesos, al igual que CMMI, habiendo sido añadidos las referencias de ITIL.

Tiene cuatro categorías, tal y como se observa en la Tabla 3. Desde el foco de este trabajo, el área de proceso de la Gestión de Requisitos se relaciona con la Gestión de los Niveles de Servicio y con la Planificación de Proyectos. Es importante resaltar que en esta integración no ha sido considerado el proceso de Desarrollo Requisitos (RD)

Category	Process Areas
Process Management	Organizational Process Focus Organizational Process Definition + IPPD Organizational Training Organizational Process Performance Organizational Innovation and Deployment
Project Management	Requirements Management Project Planning Project Monitoring and Control Supplier Agreement Management + IPPD Risk Management Quantitative Project Management
Service Delivery	Service Level Management Financial Management

	Capacity Management IT Service Continuity Management Availability Management
Service Support	Incident Management including Service Desk Problem Management Change Management Release Management Configuration Management Process and Product Quality Assurance Decision Analysis and Resolution Causal Analysis and Resolution

Tabla 3: Categorías y Áreas de Proceso en CMMI-ITIL

La gestión de procesos y de proyectos han sido adoptados directamente de CMMI, mientras que las otras categorías han sido obtenidas de ITIL: Provisión y Soporte al Servicio.

Al incorporar en la estructura de CMMI los aspectos de ITIL, han tenido que definir tanto Objetivos como Prácticas específicas en los procesos identificados.

Los Niveles de Madurez han sido definidos específicamente para CMMI-ITIL, como se observa en la Tabla 4.

Maturity Level	Process Area	
5: Optimizing	Organizational innovation and resolution Causal Analysis and Resolution	
4: Quantitatively Managed	Organizational Process Performance Quantitative Project Management	
3: Defined	Organizational Process Focus Organizational Process Definition Organizational Training Risk Management Integrated Project Management Decision Analysis and Resolution	Problem Management Release Management Financial Management Capacity Management IT Service Continuity Management Availability Management
2: Managed	Requirements Management Project Planning Project Monitoring and Control Supplier Agreement Management Measurement and Analysis Process and Product Quality Assurance Configuration Management	Incident Management Change Management Service Level Management
1: Initial	-	

Tabla 4: Niveles de Madurez en CMMI-ITIL

3.6 COBIT

Tras los conceptos vistos hasta ahora, falta un componente importante. Si se dispone de una metodología, de unos procesos, de una orientación al servicio, ¿cómo se puede evaluar los avances? Como es sobradamente conocido, sin mediciones no se pueden realizar diagnósticos objetivos [45].

Las siglas COBIT, corresponden a **C**ontrol **O**bjectives for **I**nformation and related **T**echnology [46]. Ha sido creado por la Asociación para la Auditoría y Control de Sistemas de Información (ISACA [47]) y el Instituto de Administración de las Tecnologías de la Información (ITGI [48]) surgiendo por tres motivos fundamentalmente :

- ante la complejidad de los sistemas TI,
- la dependencia que de ellos se tiene por parte de la organización
- las expectativas que la dirección adquiere.

COBIT nace como un marco genérico que proporciona herramientas para evaluar en qué grado se alcanzan los objetivos planteados en la organización, tal y como se muestra en la figura 24.

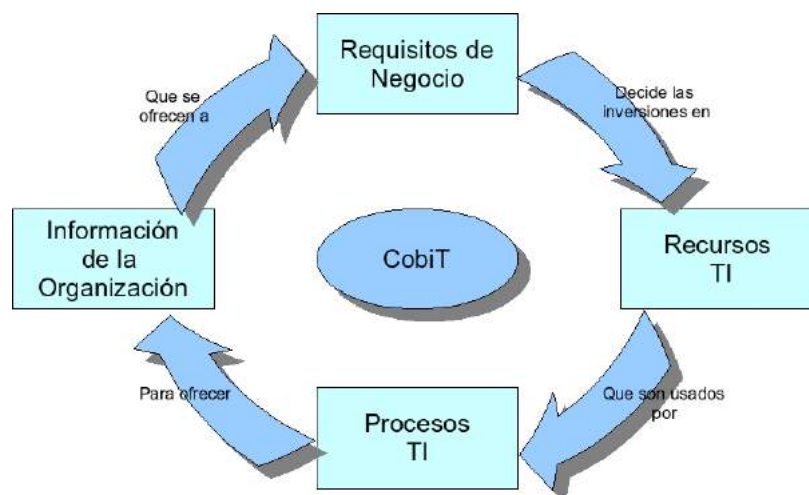


Figura 24: Visión de COBIT

Para ello COBIT integra normas y reglamentos, códigos de conducta, criterios de calificación, prácticas en la industria y requisitos emergentes de sectores específicos como : ISO (9000-3) COSO, IFAC, IIA, ISACA, AICPA, SPICE, EDIFACT, NIST SP 800-18, ISO IEC 13335n, etc.

La primera edición se publicó en 1996 hasta la edición v4.1 publicada en Septiembre del 2007 y tiene como misión: “Investigar, desarrollar, publicar y promover un conjunto de objetivos de control autorizados, actualizados, y de carácter internacional generalmente aceptados para ser usados, en el día a día por directivos y auditores y profesionales de TI.”

La principal característica de COBIT es la sinergia entre las actividades de las organizaciones TI y los objetivos que éstas deben cumplir. Para ello, define una serie de indicadores (318 en total) asociados a unos procesos (34) agrupados por dominios (4):

- Plan and Organise (Planificación y Organización) compuesto por 11 procesos.
- Acquire and Implement (Adquisición e Implementación) compuesto por 6 procesos.
- Deliver and Support (Entrega y Soporte) compuesto por 13 procesos.
- Monitor and Evaluate (Supervisión) compuesto por 4 procesos.

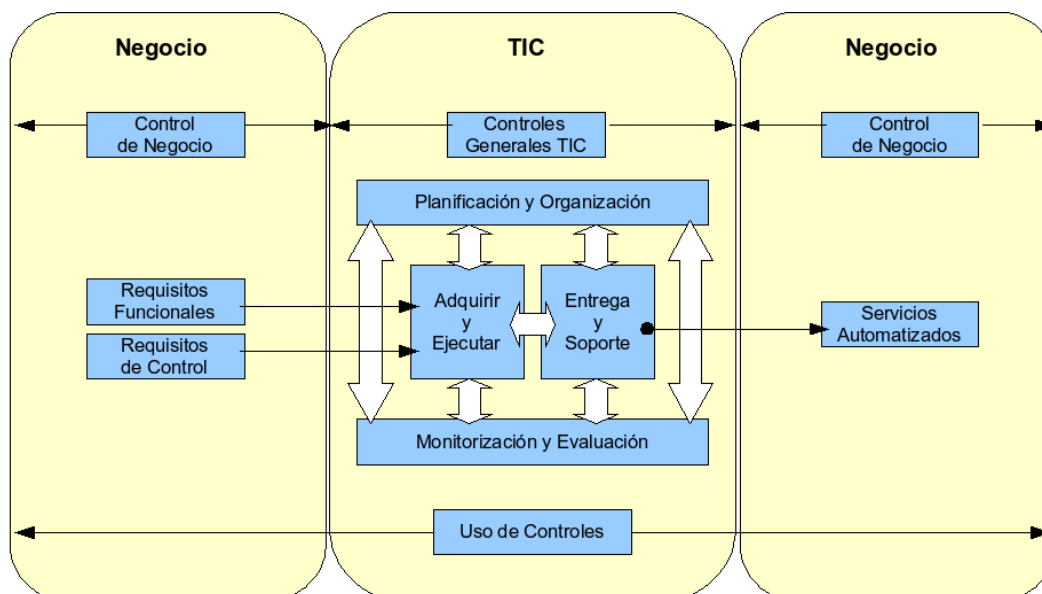


Figura 25: Responsabilidades y Procesos en Cobit v 4.1

La figura 25 muestra la relación de los procesos y su distribución en los roles de la empresa. Es interesante el hecho de que tanto el control de los requisitos, como la propia definición sea responsabilidad del negocio y no del departamento de TI.

Los indicadores definidos para realizar el control, monitorización y evaluación propuestos son:

- *Outcome measures*, en versiones anteriores conocidos por Key Goal Indicator (KGI). Es la medida que indica el grado de cumplimiento de los objetivos. Son llamados “lag indicators”, ya que son medidas a posteriori.
- *Performance indicators*, anteriormente conocidos como Key Performance Indicator (KPI). Es la medida que indican cómo se están alcanzando los objetivos. Son llamados “lead indicators”.

Estas métricas tienen en común el hecho de proporcionar información a la organización de forma eficaz, eficiente, confidencial, íntegra, conforme y fiable. Se tiene que medir la organización si se quiere tener control, mejorar la productividad, reducir riesgos, etc.

La relación de los indicadores se puede observar en la figura 26, con un ejemplo proporcionado por el propio Framework. En dicha figura se muestra cómo los indicadores de rendimiento a nivel de Negocio, se convierten en las KGI a nivel de proceso, ya que los objetivos del proceso (KGI), son los indicadores del rendimiento a nivel de TI (KPI), y a su vez los objetivos de TI (KGI) son los indicadores del rendimiento del negocio (KPI).

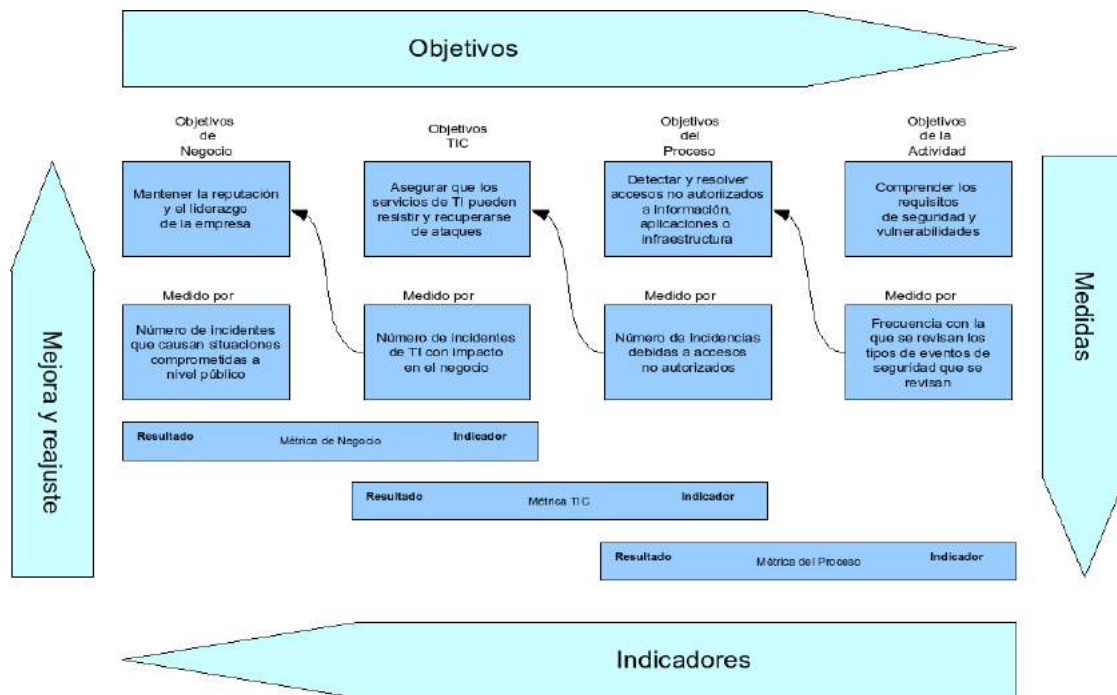


Figura 26: Ejemplo de relación de los indicadores en Cobit 4.1

Uno de los aspectos más valorados es, no sólo su interrelación directa con CMMI o ITIL, sino la definición de las medidas para obtener la información necesaria en la realización de las auditorías. ([49])

3.7 SOMA

SOA (Service Oriented Architecture) es la adecuación de la infraestructura de soporte de TI con la exposición de servicios que pueden ser consumidos por aquellos otros servicios que los requieran.

Una de las razones que refuerzan e impulsan la adopción de SOA es precisamente tener una mayor capacidad de respuesta ante los requisitos de negocio, que redundan en un aumento de la agilidad del negocio. Según [50] las razones por las que las organizaciones adoptan SOA son:

- Arquitectura escalable y flexible al separar el proceso del servicio.
- Reutilización de la inversión, aunque uno o varios procesos de negocios cambien, no implica que el esfuerzo de la exposición del servicio se vea alterado o inutilizado.
- Modelado de Procesos de Negocio, mientras el negocio está en continuo cambio, una infraestructura de TI rígida puede ser un gran inconveniente para que la organización ofrezca aquello que demanda el mercado.
- Incrementar la agilidad del negocio, poder dar respuesta a los requisitos del negocio que se debe adecuar a las exigencia cambiantes de la demanda.

Para alcanzar los objetivos anteriores, existe un concepto llamado SOA Governance que recomienda que las personas, la tecnología y los procesos deben coordinarse para crear y administrar las políticas y acciones necesarias que aseguren la alineación con las líneas de negocio. [51]

La elección de una metodología de desarrollo o proceso que tenga en cuenta estos conceptos es necesaria.

SOMA es el acrónimo de “Service-oriented modeling and architecture” [52] siendo una **metodología** que propone un ciclo de vida de desarrollo para el **diseño de soluciones SOA**, que proporciona las **técnicas y normas** necesarias para el **análisis, diseño, implementación, pruebas y despliegue** de **servicios, componentes, procesos, información y políticas** necesarios para el diseño y la construcción de soluciones robustas y reutilizables de soluciones SOA en la empresa.

Este enfoque hace que se asimile el paso que se dio en la orientación a objetos, en la que se pasó de pensar en la programación a diseñar las interfaces, se aplique a los servicios, asimilando que la capa de negocio debe describir un conjunto de servicios, diseñando de esta manera la arquitectura del software.

Pero estos métodos tradicionales de abstracción aplicados a SOA son insuficientes por todas las implicaciones que tiene una arquitectura SOA, en la que se deben identificar mucho más que interfaces, como por ejemplo: definición de servicios, componentes que implementan dichos servicios, identificación del modelo de negocio subyacente en los servicios, la relación o composición de servicios (coreografía), los requisitos de información asociados a los servicios y las políticas asociadas a los mismos.

Hay un nuevo conjunto de elementos que se deben identificar y definir que no hacen posible una adopción de Object Oriented Analysis and Design (en adelante OOAD) para la definición de SOA:

- Los participantes en los proyectos SOA se dividen en dos perfiles, los tecnológicos y los de negocio.

- Detrás de cada definición de servicio de una organización hay un componente de negocio relacionado de forma directa (uno o varios)
- La decisión tecnológica no es la única que prima en su diseño y concepción, sino también las circunstancias del negocio, lo que propicia que la arquitectura SOA deba ser diseñada con la capacidad adecuada de cambio que tenga el proceso de negocio que representa.

El cambio desde un enfoque OOAD a un enfoque SOA es significativo, y lleva a la definición de nuevos elementos de diseño.

Frente a la identificación de casos de uso, el enfoque SOA utiliza para la la definición de su arquitectura los *casos de servicio*, que identifican la colaboración reconfigurable de un conjunto de operaciones de servicio (unidades de servicio), este enfoque proporciona la flexibilidad necesaria y la adaptabilidad ante los cambios en los procesos de negocio.

Esta aproximación requiere de pautas para identificar y especificar:

- los servicios,
- los componentes,
- la colaboración entre ellos

La metodología SOMA se divide en un conjunto de fases:

- Modelado y Transformación del Negocio: Fase del modelado, simulación y optimización del negocio completo.
- Gestión de la solución: Como las soluciones SOA son híbridas por naturaleza, es conveniente tener unas plantillas donde se encuadre el proyecto.
- Identificación: La fase de identificación tiene como objetivos principales el descubrimiento de los servicios, los componentes y su proceso de integración. Se inicia alineando los servicios con los objetivos de negocio, una etapa denominada Modelado de Servicios con Objetivos (GSM – Goal Service Modelling). La identificación de servicios candidatos y la creación de un Portfolio de servicios (concepto de ITIL) es la primera fase. Es importante destacar que no todo el porfolio de servicios se implementará debido a diversos criterios y a la priorización que se realice. Pueden ser asumidos en distintas fases o distintos proyectos, o bien quedar como APIS internas del sistema y no llegar a exponerse como servicios.
 - GSM: Se usa una aproximación basada en las Opciones y Oportunidades de Negocio, Estrategia Corporativa y Objetivos de Negocio. Hace una descomposición jerárquica de estos elementos, descomponiéndolos hasta llegar a un nivel de propuesta de servicios. En este proceso de subdivisión es importante identificar los KPIs que permitan luego medir, monitorizar y cuantificar el grado de éxito de la solución SOA de acuerdo a los principios del negocio.
 - Descomposición del dominio: Análisis de arriba a abajo (top-down) centrado en el modelo del proceso de negocio, reglas, información y análisis de orientada a la variación.
 - Análisis de principios existentes: Aproximación de abajo arriba (down-top) que parte de los sistemas de información ya existentes en la organización para determinar potenciales servicios y reutilizar los componentes ya existentes y adaptarlos.
- Especificación: Se elabora el modelo de los servicios en términos de las dependencias de servicios, eventos, reglas, operaciones, mensajes, requisitos no funcionales, etc.

- Realización: Se valida el desarrollo realizando una exploración de viabilidad de la solución.
- Implementación y ensamblado: Se construye la solución.
- Despliegue, monitorización y gestión: Se despliega la solución, se valida por los usuarios y se pone en producción.

El ciclo de Vida puede observarse en la figura 27, que expresa las diferentes fases y actividades que se llevan a cabo:

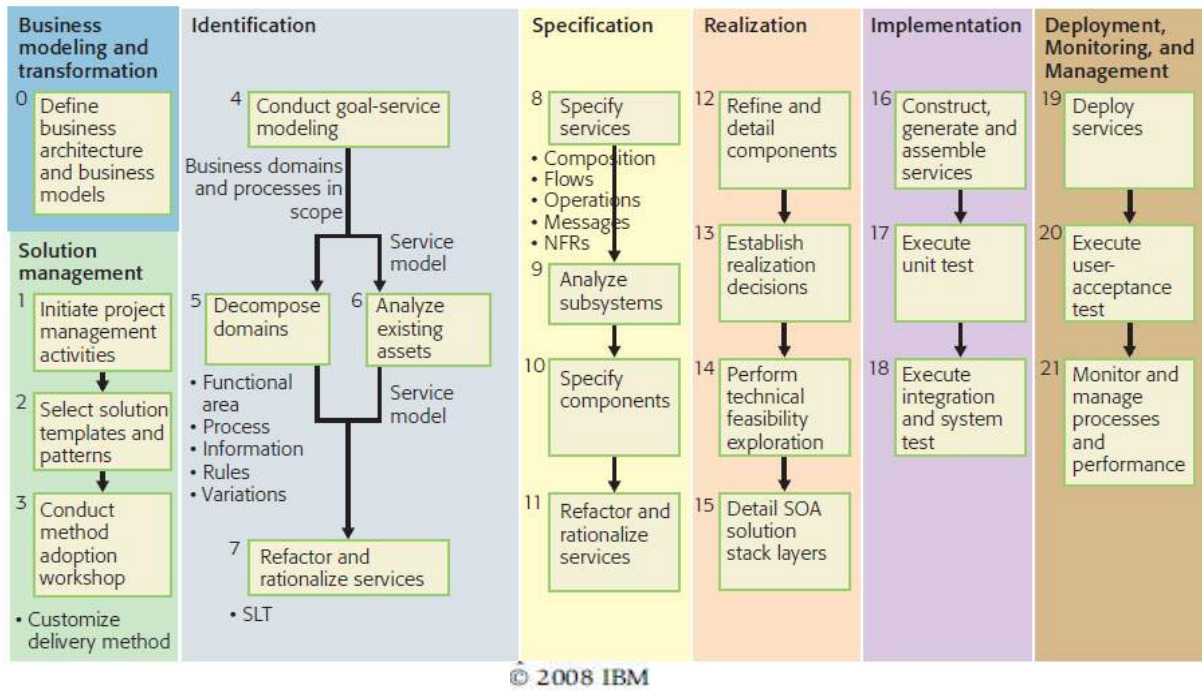


Figura 27: Ciclo de Vida SOMA

Estas fases no se deben entender de un modo secuencial ni iterativo, sino que se deben aplicar de una manera fractal y a medida de cada solución (ver figura 28), proporcionando valores unas fases a otras de forma bidireccional, y aplicándose en el momento en el que resulten de valor al proyecto.

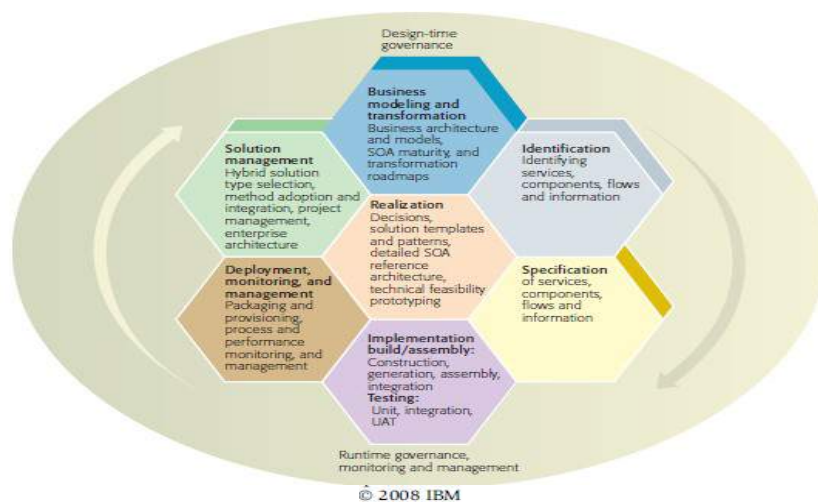


Figura 28: Enfoque fractal de SOMA

El enfoque fractal de la metodología es una metáfora para representar la forma de composición y aplicación de las técnicas, normativas y fases en la construcción de los sistemas, comparándolo con la manera en la que funcionan los sistemas fractales, cuyas principales características se podrían resumir en:

- Aplicación de métodos a ámbitos diferentes. En SOMA, la misma técnica o norma se puede estar empleando para hacer definiciones a alto o bajo nivel, y aplicadas en mayor o menor detalle en función del momento y los escenarios que se estén evaluando y diseñando.
- La iteración en la aplicación de los modelos, hace que las representaciones fractales se compongan y se doten de completitud. En SOMA, la evolución de los servicios es un requisito de todo sistema, e implica no sólo el diseño orientado a la mitigación del riesgo (concepto que gobierna la programación incremental e iterativa) sino también el diseño de los servicios con las dependencias asociadas al portfolio ya vigente y en servicio.(Ver apartado de ITIL sección 3.4)

IBM [53]ha desarrollado también un framework de desarrollo SOMA-ME [54], Service-Oriented Modeling Environmet. El objetivo de dicho framework es el diseño de soluciones SOA utilizando SOMA basado en MDD (Model-Driven Design). Iniciativas en esta misma línea se están desarrollando y responde a un área en sus inicios. [55]

4 Conclusiones y trabajo futuro

Tengo la suerte de poder trabajar en la gestión de proyectos, donde el re-trabajo que supone no establecer claramente las necesidades del producto se observan. Esto no sólo no se traduce a que los costes del proyecto se disparen y que las entregas no lleguen a tiempo, sino a factores psicológicos como es la pérdida de motivación del equipo.

Hay factores diversos que afectan a esta situación, siendo aspectos como es la dificultad en la comunicación, que caracterizan de una forma peculiar a esta rama del conocimiento. A los usuarios les cuesta estructurar y definir sus necesidades. Además de la dificultad de obtener el conocimiento no explícito, el que realiza la elicitación no siempre tiene las actitudes y aptitudes recomendables. Formación y cultura faltan en este sentido, tanto por parte de los usuarios, equipos directivos, como de los propios profesionales. Si en el resto de ingenierías, se entiende que, por ejemplo, un arquitecto realice distintos planos en función de la vista o sección que se tenga que representar, ¿por qué no se exige que el ingeniero del software tiene que realizar distintos modelados en función de con quien habla o qué comportamiento tiene que representar? Respecto al enfoque o no de la informática en ingeniería, es interesante las conferencias realizadas por D. Miguel Toro y el material de sus transparencias.

La aplicación o la elección de una metodología así como de una herramienta de soporte, no siempre es fácil. Bien porque la metodología sea genérica y cueste su adopción o bien porque las herramientas no se integren con el resto de tareas a realizar dentro del ciclo de vida del proyecto, o sean excesivamente caras y su justificación no tenga el suficiente respaldo directivo.

El reconocimiento del papel de la ingeniería de requisitos, su integración en el resto del ciclo de vida del producto y la explotación a nivel de gestión de la trazabilidad de los elementos del producto, es una situación que debe ser mejorada. Por un lado se dispone de normas como la IEEE 830 que tiene como foco los requisitos, por otro lado recomendaciones y buenas prácticas de gestión de procesos y de servicios como las vistas con anterioridad en el trabajo.

Una organización que desee mejorar y evitar una problemática ya conocida, tiene ante sí un reto, tanto mayor cuanto menor sea la misma. ¿Hay alguna metodología o recomendación que vincule todas las actividades del ciclo de vida de un producto suficientemente actualizada, incluyendo testing? [56] ¿qué importancia tienen los requisitos en las pruebas y calidad del producto final? ¿cuáles son los procedimientos y automatismo disponibles? ¿qué indicadores puedo obtener para comprobar la mejora del proceso y justificación del gasto en la implantación?.

El mundo empresarial de hecho, ha tenido que dar parte de esas respuestas, generando estándares de facto, como son CMMI e ITIL. No obstante, se están produciendo convergencias con el mundo académico, como por ejemplo podemos encontrar III Congreso Interacadémico 2008-itSMF España, realizado en la Universidad Carlos III de Madrid [57]o la colaboración de la universidad alemana de Darmstadt y la empresa Wibas con CTIL [44]. Ejemplo de demanda de esta

convergencia es la vivencia en primera persona en el congreso internacional del IEEE Ingeniería de Requisitos en Barcelona 2008 [58]. En ese congreso asistí a una conferencia sobre el proyecto SecSE (Service-Centric Systems) [59] and Requirements Engineering y vi el trabajo a realizar en la integración de todos los aspectos que intervienen en el ciclo de vida de vida del software y cómo la orientación SOA no puede dejarse atrás como parte integrante de la ingeniería de requisitos.

Después de haber realizado este compendio en la materia, las posibles líneas de trabajo futuras para abordar de una forma unificada el proceso de Ingeniería de Requisitos son:

- Definición de unos patrones lingüísticos que faciliten la definición de una taxonomía de requisitos y atributos:
 - acordes a las nuevas necesidades tecnológicas como es la arquitectura orientada a servicios (SOA).
 - acordes a las necesidades de trazabilidad y de utilidad en el resto de vida del proyecto, atendiendo a la gestión del mismo.
 - acordes a los procesos recomendados en los estándares de facto presentados.
 - que refuercen la orientación del servicio prestado dentro del negocio
- Definición de los elementos necesarios tanto en la ingeniería de requisitos como en etapas sucesivas del ciclo de vida, para resolver aspectos como la trazabilidad y relaciones entre los mismos.
- Definición del proceso o procesos para que de forma iterativa se vayan completando los elementos necesarios en función de la etapa o actividad en la que se esté en el desarrollo.
- Definición de un modelo de procesos detallado y de los elementos para que el mantenimiento del servicio prestado sea mejorado.
- Evaluación de la idoneidad de la solución aplicada dentro de una batería de proyectos.

5 Bibliografía

- [1] B.W Boehm, Some Experience with Automated Aids to the Design of Large-Scale Reliable Software., IEEE Transactions on Software Engineering, 1975, 125-133
- [2] E.Daly, Management of Software Development, IEEE Transactions on Software Engineering, 1977, 199-242
- [3] Richard W. Selby, Barry W. Boehm, Software engineering: Barry W. Boehm's lifetime contributions to software development, management, and research, Wiley-IEEE, 2007, 75-78, 978-0470148730
- [4] Manchester Metropolitan University , Introduction to Requirements Analysis , disponible en: <http://www.doc.mmu.ac.uk/online/SAD/T03/reqanal.htm>
- [5] Ye Yang, Mei He, Mingshu Li, Qing Wang, Barry Boehm, Phase distribution of software development effort, ACM, 2008, 61-69, 978-1-59593-971-5
- [6] Information Technology Research Institute, Software Maintenance Costs, 2004, disponible en: <http://www.cs.jyu.fi/~koskinen/smcosts.htm>
- [7] Standish Group , Página Oficial , disponible en: <http://standishgroup.com>
- [8] David Rubinstein , Standish Group Report: There is Less Development Chaos Today , disponible en: <http://www.sdtimes.com/content/article.aspx?ArticleID=30247>
- [9] IEEE Computer Society, Software Engineering Body of Knowledge, 2004, disponible en: <http://www.swebok.org/>
- [10] Amador Durán, Un entorno metodológico de Ingeniería de Requisitos para Sistemas de Información, Tesis Doctoral, Universidad de Sevilla, 2000
- [11] G.Nistal, Calidad del software, punto de vista y experiencias en la administración pública, Novática, 1999, 137:50-53
- [12] Beatriz Bernárdez Jiménez, Una aproximación empírica al desarrollo de Heurísticas basadas en Métricas para Verificación de Requisitos, Tesis Doctoral, Universidad de Sevilla, 2004
- [13] Matt Light, The first Key to Project Success Is Collaborative Requirements Definition and Management, Gartner, 2008, ID Number: G00160211
- [14] Ian Sommerville, Pete Sawyer, Requirements Engineering: A Good Practice Guide, John Wiley & Sons, 1997, , 0-471-97444-7
- [15] Karl E. Wiegers, More About Software Requirements, Microsoft Press, 2006, , 2005936071
- [16] Mathias Weske, Business Process Management: Concepts, Languages, Architectures, Springer-Verlag, 2007, , 978-3-540-73521-2
- [17] Gerald Kotonya, Ian Sommervilles, Requirements Engineering. Processes and Techniques, John Wiley & Sons, 2001, , 0-471-97208-8
- [18] Betty H.C Cheng, Joanne M. Atlee, Reserch Directions in Requirements Engineering, IEEE. Futere of Software Engineering (FOSE`07), 2007,
- [19] Thomas E. Murphy, Requirements Form the Foundation of Software Quality, Gartner, 2009, ID Number: G00165755
- [20] Fundación Orange, eEspaña 2008. Informe anual sobre el desarrollo de la sociedad de la información en España, 2009, disponible en: <http://www.fundacionorange.es/fundacionorange/analisisprospectiva.html>
- [21] Consejo Superior de Administración Electrónica , Métrica v3. Metodología de Planificación, Desarrollo y Mantenimiento de sistemas de Información , disponible en: <http://www.csi.map.es/csi/metrica3/index.html>
- [22] Barry Boehm, Unifying Software Engineering and Systems Engineering, IEEE Computer

Society Press, 2000, 114 - 116, 0018-9162

[23] Ministerio de Industria, Comercio y Deporte , Plan Avanza , disponible en:

<http://www.planavanza.es/>

[24] Ministerio de Industria, Comercio y Deporte , CMMI en Plan Avanza , disponible en:

<http://www.planavanza.es/Noticias/Modelo+de+Mejora+de+Procesos+CMMI.htm>

[25] Proyecto Vulcano , Informe sobre CMMI y los requisitos necesarios para Vulcano , disponible en: www.ines.org.es/vulcano/.../d31-informacion-sobre-cmmi-y-los-requisitos-necesarios-para-vulcano.doc

[26] Ministerio de Industria, Comercio y Deporte , INES- Plataforma Tecnológica Española de Software y Servicios , disponible en: <http://www.ines.org.es/>

[27] Ministerio de Industria, Turismo y Comercio , Proyecto Vulcano , disponible en:

<http://www.ines.org.es/vulcano/>

[28] Software Engineering Institute , Página oficial de CMMI , disponible en:

<http://www.sei.cmu.edu/cmmi/>

[29] Software Engineering Institute , Página Oficial , disponible en: <http://www.sei.cmu.edu/>

[30] Carnegie Mellon University , CMMI v1.2 , disponible en:

<http://www.sei.cmu.edu/publications/documents/06.reports/06tr008.html>

[31] Juan Pablo Carvallo, Xavier Franch and Carme Quer, Supporting CMMI Level 2 SAM PA with non-technical features Catalogues, John Wiley & Sons, 2008, 171-182

[32] itSMF , Página Oficial , disponible en: <http://www.itsmfi.org/>

[33] ITIL , Página Oficial , disponible en: <http://www.itsmfi.org/>

[34] Carrie Higday-Kalmanowitz, Sandra E. Simpson, Implementing Service and Support Management Processes: A Practical Guide, van Haren Publishing, 2005, , 9789077212431

[35] OGC, Mejores Prácticas para el Soporte de Servicio, TSO, 2006, , 139780113309818

[36] OGC, Service Strategy, TSO, 2007, , 9780113310456

[37] Machteld Meijer, Mark Smalley and Sharon Taylor , ITIL V3 and ASL , disponible en: http://www.best-management-practice.com/gempdf/ITILV3_ASL_Sound_Guidance_White_Paper_Jan08.pdf

[38] ASL BiSL Foundation , Página Oficial , disponible en: <http://www.aslbislfoundation.org/>

[39] Alison Cartlidge, Ashley Hanna, Colin Rudd, Ivor Macfarlane, John Windebank, Stuart Rance , An Introductory Overview of ITIL V3 , disponible en: http://www.best-management-practice.com/gempdf/itSMF_An_Introductory_Overview_of_ITIL_V3.pdf

[40] OGC, ITIL: Application Management, TSO (The Stationery Office), 2002, , 9780113308668

[41] wibas GmbH , Página Oficial , disponible en: http://www.wibas.com/home/index_en.html

[42] Universidad de Darmstadt , Página de la Universidad , disponible en: <http://www.tu-darmstadt.de/>

[43] Office of Government Commerce , Página Oficial , disponible en:

<http://www.ogc.gov.uk/index.asp>

[44] Wibas y Universidad de Darmstadt , CMMI-ITIL. ITIL integrated into CMMI. , disponible en:

http://www.wibas.com/knowledge_base/articles__links/articles__presentations/index_en.html

[45] Marcela Genero, Mario Piattini, Coral Calero, Metrics for Software Conceptual Models, Imperial College Press, 2005, , 1-86094-497-3

[46] ISACA , Cobit 4.1 , disponible en: [http://www.isaca.org/Template.cfm?](http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981)

[Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981](http://www.isaca.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981)

[47] ISACA , Página Oficial , disponible en: <http://www.isaca.org/>

[48] IT Governace Institute , Página Oficial , disponible en: <http://www.itgi.org/>

[49] ITGI, OCG , Aligning CobiT® 4.1, ITIL® V3 and ISO/IEC 27002 for Business Benefit , disponible en: [http://www.itgi.org/Template.cfm?](http://www.itgi.org/Template.cfm?Section=COBIT6&Template=/TaggedPage/TaggedPageDisplay.cfm&TPLID=55&ContentID=7981)

Section=COBIT_Mapping1&Template=/ContentManagement/ContentDisplay.cfm&ContentID=45932

- [50] The Standish Group , Trends in Service-Oriented Architecture (SOA) , disponible en: http://www1.standishgroup.com/market_research/index.php
- [51] Oracle , Right from the Start: SOA Lifecycle Governance , disponible en: <http://www.oracle.com/technologies/soa/soa-governance.html>
- [52] Arsanjani et al., SOMA: a method for developing service-oriented solutions, IBM Systems Journal , Volume 47 , Issue 3 (July 2008), 377-396, 0018-8670
- [53] , Página Oficial de IBM , disponible en: <http://www.ibm.com/>
- [54] Zhang et Al., SOMA-ME: a platform for the model-driven design of SOA solutions, IBM Systems Journal , Volume 47 , Issue 3, 397-413, 0018-8670
- [55] P. Jamshidi et Al., An automated method for service specification, ACM, 2009, 25-28
- [56] Carey Schwaber , The Changing Face Of Application Life-Cycle Management , disponible en: <http://www.forrester.com/Research/Document/Excerpt/0,7211,37653,00.html>
- [57] IV Congreso Académico: Evolucionando de la Gestión al Gobierno de TI, disponible en: <http://www.itsmf.es/news.asp?ID=67>
- [58] 16th IEEE International Requirements Engineering Conference. Requirements engineering for a sustainable world, disponible en: <http://sites.upc.edu/~www-gessi/re08/>
- [59] SeCSE Consortium , SecSe: Service Centric , disponible en: <http://www.secse-project.eu/>

Todas las direcciones electrónicas han sido comprobadas a Junio del 2009.

6 Índice de Figuras

Figura 1: Proporción de Errores en cada una de la fases de desarrollo [4].....	5
Figura 2: Evolución temporal del coste de Mantenimiento [6].....	5
Figura 3: Resultados de los Informe CHAOS [7].....	6
Figura 4: Dimensiones del Trabajo.....	8
Figura 5: Actividades en la Ingeniería de Requisitos.....	9
Figura 6: Los requisitos en los procesos de la ISO/IEC 12.207.....	13
Figura 7: Plan de Sistemas de Información en Métrica v3.....	17
Figura 8: Estudio de viabilidad del sistema en Métrica v3.....	19
Figura 9: Análisis del Sistema de Información en Métrica v3.....	20
Figura 10: Diseño del Sistema de Información en Métrica v3.....	22
Figura 11: Elementos de CMMI 1.2.....	29
Figura 12: Área de Proceso Gestión de Proyectos Básico.....	30
Figura 13: Área de Procesos Gestión de Proyecto Básica.....	31
Figura 14: Área de Proceso Ingeniería.....	32
Figura 15: Área de Proceso Soporte Básica.....	33
Figura 16: Resumen tratamiento Requisitos en CMMI v1.2.....	37
Figura 17: Servicios en ITIL.....	38
Figura 18: Relación entre ITIL v3 e ISO 20.000.....	39
Figura 19: Ciclo de Mejora Continua en ITIL v3.....	43
Figura 20: Relaciones entre procesos y los Requisitos.....	44
Figura 21: Componentes de ITIL v3.....	45
Figura 22: Relaciones entre los productos y procesos en ITIL v3.....	47
Figura 23: Relación entre CMMI-ITIL y CMMI-DEV.....	49
Figura 24: Visión de COBIT.....	52
Figura 25: Responsabilidades y Procesos en Cobit v 4.1.....	53
Figura 26: Ejemplo de relación de los indicadores en Cobit 4.1.....	54
Figura 27: Ciclo de Vida SOMA.....	57
Figura 28: Enfoque fractal de SOMA.....	57

7 Índice de Tablas

Tabla 1: Resumen de los Requisitos en Métrica v3.....	26
Tabla 2: Niveles de Madurez y Procesos.....	33
Tabla 3: Categorías y Áreas de Proceso en CMMI-ITIL.....	51
Tabla 4: Niveles de Madurez en CMMI-ITIL.....	51