

Arquitectura para la Gestión del Proceso de Documentación

**Emilio Ernica Lafuente, Antonio de Amescua Seco, Yaser Rimawi,
Javier García Guzmán**

Universidad Carlos III de Madrid

Avda. Universidad, 30, 28911 Leganés -Madrid- España

correo electrónico: eernica@inf.uc3m.es, amescua@inf.uc3m.es, yaser@ie.inf.uc3m.es, jgarcia@inf.uc3m.es

Resumen

Las aplicaciones de software son cada vez más complejas y necesitan dar solución a una gran cantidad de problemas, ello obliga a prestar más atención al proceso de documentación del software para mejorar en el mantenimiento, la calidad, la reutilización y a su vez disminuir los costes. A continuación se propone una arquitectura para la gestión del proceso de documentación en las Pequeñas y Medianas Empresas (PYMES.), lo cual obliga de forma intrínseca a que esté enfocada a sistemas abiertos que permitan la utilización del software y hardware ya existente en la empresa. De este modo se reduce de forma decisiva la inversión a realizar, se conservan todas las capacidades de los recursos humanos asociadas al uso del software y no se compromete la capacidad de evolución del sistema de información de la empresa. Finalmente, se detallan los principios en los que se basa la arquitectura propuesta, para mejorar la gestión del proceso de documentación software, y se describe la contribución de la utilización de esta arquitectura frente a las existentes actualmente.

1 Introducción

La complejidad alcanzada hoy en día en las aplicaciones de software obliga inexorablemente a prestar más atención al proceso de documentación del software para mejorar la labor de mantenimiento y su calidad [Emam et al., 1998; ISO, 1991; ISO, 1998; Paulk et. al., 1995].

Como bien indican algunos autores, los que desarrollan sistemas de información se enfrentan a diseñar sistemas cada vez más complejos; y por tanto, a desarrollar y tratar con un mayor volumen de información en ciclos de desarrollo más reducidos, requiriendo para ello utilizar una metodología de desarrollo de información para gestionarla

La documentación software va a ser el puente de comunicación entre usuario, diseñador y creador del sistema.

Esta arquitectura que se propone está orientada hacia las PYMES, ya que según la Unión Europea, una empresa es

[Ingram, 1997]. A pesar de ellas al ser las aplicaciones de software más complejas, la labor de mantenimiento crece y se requiere no sólo un funcionamiento adecuado a las necesidades del usuario, sino un software de calidad. Para mejorar la calidad del software y su labor de mantenimiento, es necesario considerar la documentación como uno de los factores claves que permitirán reducir los costes [Visconti, 1993] y mejorar el proceso de desarrollo de software.

Aunque el proceso de documentación se empieza desde las primeras fases y sólo se considera finalizado cuando lo es la vida del software [ISO, 1990], sigue infravalorándose, ya que se suele dejar este proceso de documentación como una de las últimas actividades de desarrollo del software.

Sin embargo, prácticamente todo el mundo está de acuerdo de que una buena documentación es importante para todas las etapas del proceso de desarrollo de software [ISO, 1995; ISO, 1990; IEEE, 1991] y que además es un producto de software importante, pero nadie reconoce cuan de importante es la documentación.

La documentación es la parte más importante para la fase de mantenimiento la cual constituye el 60-75% del coste total del software [Hager, 1991]. Además, una documentación no actualizada es más costosa que si no hubiera documentación, en este punto expertos en la Ingeniería del Software [Sommerville, 1992] han reconocido que documentos mal escritos, incompletos, obsoletos, difíciles de entender, han degradado su utilidad en etapas como el mantenimiento del software.

Para poder hacer una correcta gestión del proceso de documentación software, tenemos que tener siempre en mente cuales son los principios de este proceso en concreto, los cuales podríamos sintetizar en los siguientes:

- Mejora la comunicación entre tareas o fases
- Control de calidad y control de proyecto
- Permite una referencia histórica
- Permite una referencia instruccional

PYME si tiene menos de 250 trabajadores. De forma más concreta, la Unión Europea distingue en su definición entre pequeñas y medianas empresas. Así considera que una empresa es mediana si tiene más de 49 y menos de 250

trabajadores; pequeña si tiene menos de 50 trabajadores y muy pequeña si tiene menos de 10 trabajadores. El 93% de las empresas de la Unión Europea tiene menos de 10 empleados, el 5,9% representan las empresas entre 10 y 49 empleados, y al 0,9% de empresas medianas. Además existe una resistencia al cambio debido entre otros motivos[Chen and Price, 1994] a :

- * Muchas de las actividades las realiza una sola persona.
- * El cambio es rápido y hace que los procedimientos queden anticuados en breve tiempo.
- * Limitada experiencia organizativa o carencia de toda experiencia.
- * Amenaza del puesto de trabajo por miedo a dejar de ser indispensable, etc.

2 Arquitectura propuesta

Esta arquitectura esta dividida en tres niveles, de tal manera que todos los niveles permiten cubrir la gestión del proceso de documentación de una forma más clara e intuitiva. En el primer nivel se refleja el elemento de interfaz que utilizará el usuario para la gestión completa de la documentación, delegándose luego interna o externamente las funciones a otros componentes para facilitar todas las tareas posteriores de una forma sencilla y estándar, ver figura 1.

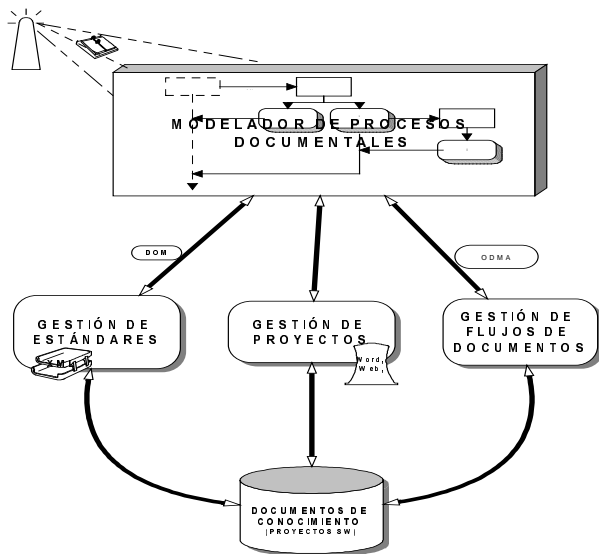


Figura 1 Arquitectura Propuesta

- En el primer nivel se encuentra el modelador del proceso de documentación en el cual se permitirá:
 1. Definir un proceso de documentación
 - Con ello se podrá identificar de una forma visual y formal cual es el proceso de documentación que debe seguir la empresa, es decir, se deberán indicar las actividades a realizar y las tareas que conlleva cada actividad.
 2. Definir la documentación que se debe crear.
 - Cada una de las tareas definidas anteriormente tendrán asociadas una serie de documentos, los cuales constituyen o son parte de un documento 'estándar'. De esta forma se realiza una planificación de qué estándares se utilizarán (para así poder mejorar la

comunicación entre los usuarios actores) y además se obtiene una lista de la documentación que se va a tener que realizar en cada actividad y la generada hasta llegar a una actividad.

3. Establecer los responsables de la documentación
 - Cada uno de los documentos tendrá un responsable de creación o revisión, permitiendo al equipo de documentación controlar la calidad y la utilidad de los documentos de una forma más exhaustiva
4. Determinar las dependencias entre los documentos para que se actualicen después de cada cambio (trazabilidad)

Asociado a cada documento se indicarán de qué documentos depende, en caso de que no este claramente indicado mediante el modelo del proceso de documentación. De esta forma se puede saber las dependencias de los documentos y los procesos que se deberán de seguir ante las modificaciones de mantenimiento.

5. Conocer el estado del proceso de documentación
 - En cualquier momento es posible visualizar el estado en que se encuentra el proceso de una forma visual, permitiéndose obtener una información más detallada una vez seleccionado cualquier estado.
6. Introducir medidas para hacerlos más formal y animar a su reusabilidad

Finalmente se introducirán valoraciones de pesos a los distintos arcos o transiciones de estados dentro del proceso para poder medir el proceso y ver que se realiza en los tiempos planificados o el porque no, y así permitir detectar cuellos de botellas.

Este componente del primer nivel será visual y estará constituido por un lenguaje de modelado visual(basado en alguno de los más 'estándares' de la ingeniería del software; ejem. UML [UML, 1997]), ver figura 2.

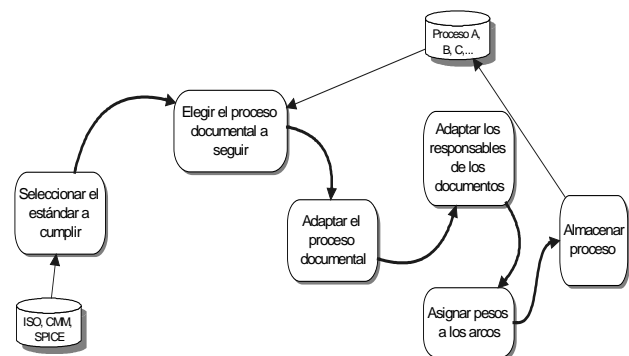


Figura 2 Modelador del Proceso de Documentación

- En el segundo nivel se desglosa los distintos elementos que forman el motor de trabajo.
 - El primer elemento que describiremos es el Gestor de Estándares. En el se permitirá:
 1. Definir el contenido de los distintos documentos del estándar.
 - Con esto se pretende que un documento deba contener una información básica

2. Definir el formato de los documentos del estándar

Vamos a poder indicar como se va a presentar y en que sitio va a estar la información. De esta manera todos los documentos de un mismo tipo van a tener el mismo aspecto, y el cambio de formato de un documento se puede automatizar para el resto de documentos que deben de cumplir ese formato.

3. Clasificar los documentos del estándar.

Siguiendo unas normas definidas se intenta indexar los documentos para así poder clasificarlos.

4. Buscar, almacenar y recuperar los documentos del estándar.

Vamos a poder recuperar o almacenar nuevos estándares, para sí tener un repositorio de estándares actualizado y útil para los usuarios

5. Modificar documentos del estándar

Se debe permitir la actualización de los estándares almacenados, así como, la adaptación de estándares existentes como nuevos estándares de la organización.

6. Definir un procedimiento que compruebe que el contenido de un documento satisface el estándar.

Para evaluar que un documento contiene la información básica requerida y cumple los objetivos marcados, es necesario introducir un marco formal que permita automatizar esta tarea, ver figura 3.

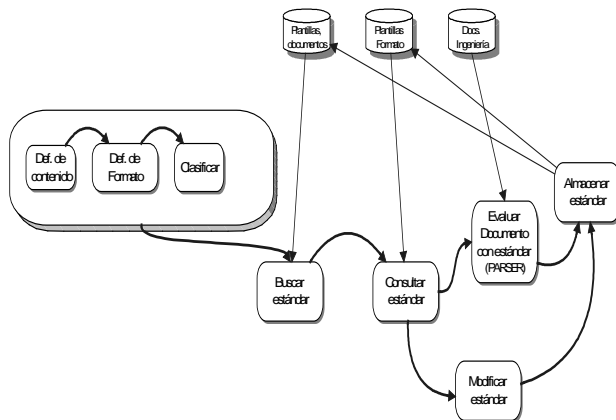


Figura 3 Gestión de Estándares

- El segundo elemento es el Gestor de Documentos de Proyectos Software. Este será un producto Add-in que trabaja dentro de las aplicaciones estándares de creación de documentos, tales como Word y WordPerfect. Se permitirá:

1. Obtener la información de las distintas herramientas CASE más utilizadas en el desarrollo del software.

Consiste en capturar la información de la herramienta y trasladarla al documento que estamos generando, para ello será necesario que se disponga de unos dominios de herramientas en la que vendrá especificado cómo se puede captar la información y cómo está almacenada.

2. Actualización automática de la información dependiente de las herramientas CASE.

Se debe permitir que exista la posibilidad de actualizar la información del documento ante variaciones realizadas en la herramienta de ingeniería.

3. Control de la versión de los documentos

De forma automatizada siempre se pedirá que un documentos se guarde con otra versión o no, para así tener un histórico de versiones, con la fecha del creador

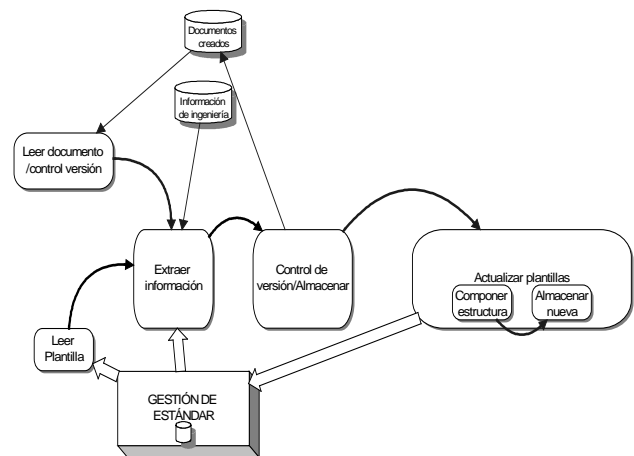


Figura 4 Gestión de Documentos de Proyectos

- El tercer elemento es el Gestor Corporativo o de Flujos de Documentos. Este elemento permitirá:

1. Tener un control sobre las responsabilidades de cada documento.

Con ello se podrá visualizar el estado en que se lleva cada documento por su responsable, sabiendo por lo tanto si ha existido alguna incidencia, si ha sido leído el documento o realizada su elaboración

2. Automatizar el reenvío de los documentos a los responsables

Con esta posibilidad una vez introducidos los datos del creador del documento y las modificaciones o actualizaciones pertinentes, el usuario al cerrarlo como revisado o finalizado se enviará automáticamente al siguiente responsable definido en el modelo del proceso documental

3. Generar eventos de aviso personalizado a los distintos responsables.

Ante incidencias en el proceso de documentación identificado inicialmente, se deben permitir la generación de mensajes a los responsable afectados por las modificaciones o por las nuevas variables de entorno.

- En el tercer nivel se encuentra el repositorio, almacén, o base de los documentos, el cual nos permitirá:
 1. Administración de los privilegios de los distintos usuarios que hacen uso de la documentación
 2. Importar/Exportar los distintos objetos documentales almacenados.

Mediante esta utilidad se permitirá la utilización del repositorio en nuevos entornos de gestión.
 3. Monitorizar todo el flujo de intercambio de documentos.

Poder en cierta medida ver las cargas de uso de documentos específicos, para obtener estadísticas de uso por los distintos usuarios y responsables.

3 Principios para la gestión del proceso de documentación software

A continuación se detallan los principios en los que nos hemos basado para la realización de la arquitectura comentada anteriormente

3.1 Principios basados en los estándares del ciclo de vida de software: IEEE 1074-1991 e ISO/IEC 12207 [IEEE,1991; ISO, 1995].

- Se requiere una comunicación efectiva y una respuesta oportuna entre el personal de software y el de documentación
- Hay que planificar, implementar producir y distribuir la documentación. La planificación consiste en identificar la documentación y sus responsables lo que nos permitirá la creación de calendarios.
- Verificar y validar la documentación. El medio más natural consiste en la realización de revisiones.
- La gestión de la configuración es imprescindible para llevar a cabo el mantenimiento de una forma correcta.

3.2 Principios basados en la gestión de la documentación software. [ISO, 1990]

- Evidencia del compromiso y soporte directivo a la documentación
 - Existencia de estándares y de guías para todos los aspectos de la documentación software
 - Compromiso de una revisión continua para asegurar la conformidad con la política, los estándares, procedimientos y planes de documentación
- Funciones de la documentación software
 - Comunicación a la gestión:
 - Evaluar el progreso, los problemas, y las expectativas. Se deben realizar informes periódicos que muestren el progreso respecto al calendario planificado.
 - Comunicación entre tareas:
 - Los proyectos de desarrollo de software se dividen en tareas que son realizadas a menudo por diferentes grupos de personas. Estas personas necesitan algún medio de

comunicación con las de otros grupos a los que les ofrece información, por ello la mayoría de las metodologías de desarrollo de software establecen documentos formales para la comunicación.

- Referencia histórica
 - Se necesita esta documentación como referencia histórica para permitir asistir en la transferencia y conversión del software a nuevos entornos.
- Establecer políticas de documentación:
 - La documentación debe ser adecuada a su lector.
 - Se deben identificar y utilizar estándares de documentación.
 - Se deben usar herramientas que ayuden a la documentación, siempre que sea asequible económicamente.
- Establecer estándares y guías de documentación:
 - Se deben adoptar estándares o si no existen crearlos. La aplicación de un estándar permite responder a las siguientes preguntas:
 1. ¿Qué tipo de documentación se requiere?
 2. ¿Cuánta documentación se va a proporcionar?
 3. ¿Qué van a contener los documentos, formatos.
 4. ¿Qué nivel de calidad se alcanzará.
 5. ¿Cuándo se producirán los documentos.
 6. ¿Cómo se almacena , mantiene y comunica la documentación?
 - Definir un sistema de identificación de documento para un control efectivo de la documentación.
- Establecer procedimientos de documentación
- Planificación de la documentación
 - Se establece qué se va a hacer, cómo, cuándo y quién lo hará.

3.3 Principios basados en el trabajo colaborativo

- Modelos de seguridad orientados al negocio con implantación de roles de puesto de trabajo.
- Administración y control de versiones de objetos que contiene documentos y tareas.
- Compartición bajo demanda de objetos de acuerdo al role del puesto.
- Firmas Electrónicas
- Notificación permanente e histórica de eventos y acciones
- Flujo de trabajo gráfico:
 - Rutas Manuales /Ad Hoc
 - Rutas basadas en modelos
 - Fechas de finalización
 - Notificación de eventos
 - Estado en tiempo real del flujo

4 Contribuciones de esta arquitectura frente a otras

4.1 La situación actual

Podemos decir que existen dos grandes grupos de arquitecturas que tratan la gestión de la documentación. Por un lado nos encontramos con aquellas que gestionan la documentación software y por otro las que gestionan los documentos en el trabajo corporativo.

Al primer grupo correspondería [SoDA, 1999; DocEXPRESS, 1998] a aquellas que basan toda su funcionalidad en permitir extraer la información a partir de las herramientas de desarrollo del software. Así pues, la primera función consiste en reconstruir la información procedente de varias fuentes de información y usar esta para generar documentos y listados de acuerdo a una plantilla seleccionada por el usuario.

La arquitectura se basa en entornos add-in con otras herramientas de visualización y de edición de los documentos que se interconectan con programas especiales llamados domains, los cuales contiene el conocimiento de cómo la información es modelada en la fuente de ingeniería del software, permitiendo la posterior recuperación de la información de las herramientas de desarrollo del software.

Al segundo grupo [Documentum, 1998; Work Expeditor, 1998; DOCSFusion, 1998; CyberDOCS, 1997; DOCS Open, 1996; Papiro, 1998] corresponde a aquellas que se basan en una infraestructura de gestión de los documentos y comunicación empresariales. Esta infraestructura se implanta de forma poco estructurada, utilizando el sistema de correo electrónico corporativo como columna vertebral de información asíncrona de las aplicaciones que contiene la lógica del negocio y el flujo de los procesos. Con esto permiten manejar todo tipo de información incluso la poco estructurada (documentos, formularios) y favorece las labores de comunicación más complejas del personal y las aplicaciones de colaboración.

La arquitectura posee dos entornos de trabajo, por un lado el tratamiento y gestión del ciclo de vida de los documentos y por otro un entorno add-in con otras herramientas de visualización y edición de los documentos.

4.2 Novedades

Basándonos en principios de la ingeniería del software, del trabajo colaborativo y de la gestión de documentos, la arquitectura propuesta tiene las siguientes novedades respecto a las existentes:

Se introduce una capa visual de control y gestión del proceso de documentación, permitiendo el acceso a la información dependiendo del perfil del usuario y la trazabilidad de los documentos.

Se realiza una gestión específica de los estándares existentes y de los creados por la organización, posibilitando un control total sobre ellos tanto de seguridad como de acceso a su información mediante técnicas de recuperación y clasificación de la información.

Se integra la comunicación entre la gestión corporativa y la gestión de documentos software.

Se recoge todo el conocimiento de los documentos en un repositorio o base de documentación para tener una mayor flexibilidad de gestión de los datos y documentos generados.

Con esta nueva arquitectura se pretende en definitiva dar el soporte adecuado para la realización de las

siguientes prácticas claves que permiten mejorar el proceso de documentación [Visconti, 1993; Cook and Visconti, 1996]:

1. Creación consistente de documentos básicos de desarrollo de software
2. Reconocimiento general de la importancia de la documentación
3. Registro de políticas sobre la importancia de la documentación
4. Tiempo y recursos adecuados para la documentación
5. Seguimiento de estándares de documentación
6. Utilización de una lista sobre la documentación requerida
7. Utilización de herramientas simples de documentación
8. Exactitud, precisión y fiabilidad, seguridad, veracidad
9. Mecanismos para actualizar la documentación
10. Mecanismos para controlar la calidad de la documentación
11. Métodos para evaluar la utilidad de la documentación
12. Utilización de conjuntos de herramientas de documentación comunes
13. Utilización de herramientas de documentación avanzadas
14. Tecnología y formación relacionada con la documentación
15. Medidas de la calidad del proceso de documentación
16. Análisis del uso y de la utilidad de la documentación
17. Bucle de realimentación de mejora del proceso
18. Integrar CASE y herramientas de documentación

5 Conclusiones

Con esta arquitectura se pretende abordar la gestión del proceso documental dentro de las PYMES de una forma invisible, es decir, el propio uso de este soporte permite la realización de los objetivos necesarios para mejorar la documentación y gestionarla de forma adecuada, minimizando el coste de formación y el de inversión económica, al utilizar herramientas y entornos muy extendidos entre esta comunidad de organizaciones. Así pues, con esta arquitectura se gestionará el proceso tanto en el sentido de la estructura y operación del proceso en sí, como en la calidad del producto o de la información que fluye por dentro de la estructura, es decir los documentos o unidades documentales. Finalmente recabamos un conjunto de los requisitos que cumpliríamos con esta herramienta desglosándolos en, generales cuando hacen referencias a ideas plasmadas en otros procesos del desarrollo de software; estándares cuando pretendemos reflejar todo aquello que se debe llevar a cabo en el uso de los estándares, proyectos mostrando a nivel de proyectos informáticos que es lo que vamos a cumplir y de documentos, identificando los principios que se contemplaran referentes a estos.

6 Bibliografía

[Chen and Price, 1994] Chen, E; Price J : *"La gestión de la calidad en la pequeña empresa"*. Harvard Deusto Business Review nº 59, 1994.

- [Cook and Visconti, 1996] Cook, C.R. and Visconti, M.: New and improved documentation process model, Fourthen Annual Pacific Northwest Software Quality Conference, Portland, Or, USA, pp.364-380, 1996
- [CyberDOCS, 1997] CyberDOCS, The right technology, to deliver the right solution, at the right time, WhitePaper, PC DOCS, Inc. Copyright 1997.
- [DocEXPRESS, 1998] DocEXPRESS/Factory ,© 1998 ATA, Incorporated 3528 Torrance Blvd. #214Torrance, CA
- [DOCSFusion, 1998] DOCSFusion Architecture, Advanced Document Management, WhitePaper, PC DOCS, Inc. Copyright 1998
- [DOCS Open, 1996] DOCS Open Release, Architected for the Enterprise and Beyond, WhitePaper, PC DOCS, Inc. Copyright 1996
- [Documentum, 1998] The Documentum Enterprise Document Management System 98 (EDMS 98) Copyright ©1998 by Documentum, Inc. Gibraltar Drive, Pleasanton, CA
- [Emam et al., 1998] Emam, Khaled El; Drouin, Jean-Normand; Melo, Walcelio: SPICE: The Theory and Practice of Software Process Improvement and Capability Determination; IEEE Computer Society, Los Alamitos, California, 1998
- [Fayad, 1997] Fayad, M. E.: Software Development Process: A Necessary Evil, Communications of the ACM, September 1997, vol. 40, n. 9, pp.101-103
- [Hager,1991] Hager, J.A.: Software cost reduction methods in practice: a post mortem analysis. Journal of systems and software, Vol. 14, Nº.2 °,1991
- [IEEE,1991] The Institute of Electrical and Electronics Engineers. ANSI/IEEE Std. 1074-1991, IEEE Standard for Software Life Cycle Processes. The Institute of Electrical and Electronics Engineers, Nueva York: IEEE Computer Society, 1991
- [Ingram, 1997] Ingram, J.: From the real toward the ideal: a case study in virtual document development. 15th Annual International Conference on Computer Documentation Conference Proceedings. SIGDOC '97.Crossroads in Communication. 19-22 Oct. 1997.Salt Lake City, UT, USA. ACM. pp. 137-144
- [ISO, 1985] International Organization for Standardization, ISO/IEC 6592:1985 Information processing – Guidelines for the documentation of computer-based application systems., International Organization for Standardization, Ginebra(Suiza), 1985.
- [ISO, 1988] International Organization for Standardization, ISO/IEC 9127:1988 Information processing systems – User documentation and cover information for consumer software packages, International Organization for Standardization, Ginebra(Suiza), 1988.
- [ISO, 1990] International Organization for Standardization, ISO/IEC TR 9294:1990 Information Technology – Guidelines for the management of software documentation, International Organization for Standardization, Ginebra(Suiza),
- [ISO, 1991] "ISO 9000-3, Quality management and quality system elements - Guidelines for development, supply and maintenance of software", Ginebra (Suiza): International Standards Organization, 1991
- [ISO, 1995] International Organization for Standardization, ISO/IEC Std. 12207:1995 Information Technology – Software Life Cycle Processes, International Organization for Standardization, Ginebra(Suiza), 1995.
- [ISO, 1998] ISO/IEC TR 15504-1:1998 Information technology - Software process assessment - Part 1 : Concepts and introductory guide ; Ginebra (Suiza): International Standards Organization, 1998
- [Jose and Viswanathan, 1992] Jose, J.M., Viswanathan, T.: Software Documentation and Standards. Annals of Library Science and Documentation, 1992, 39(4), 123-133
- [Papiro, 1998] Papiro "Documentos Electrónicos", Acimut Integración de Sistemas S.L., acimut@acimut.com , (Valencia), 1998
- [Paulk et. al., 1995]: Paulk, M., Weber, C., Curtis, B., y Chrissis, M.: The Capability Maturity Model: Guidelines for Improving the Software Process, ISBN 0-201-54664-7, Addison-Wesley Publishing Company, Reading, MA, 1995.
- [SoDA, 1999] SoDA, Automating the Creation and Maintenance of Project DocumentationRational, SoDA for Word 3.5 , Rational Software Corporation, 1999 <http://www.rational.com/products/soda/index.jtmpl>
- [Sommerville, 1992] Sommerville, I.: Software Engineering, Reading, MA (EE.UU.), Addison Wesley, 1992
- [UML, 1997] UML Notation Guide, version 1.1 , 1 September 1997, Rational Software Corporation et al.
- [Visconti, 1993] Visconti, M.A.: Software System Documentation Process Maturity Model, Dissertation (Director: Cook, C.R.), Oregon, Oregon State University, 1993
- [Work Expeditor, 1998] Compaq Work Expeditor, @ Compaq, 1998