



University of Twente
Enschede - The Netherlands

Solving the size estimation problem in ERP project context: the eEPC-COSMIC approach

(Solucionando los problemas de estimación en el contexto de los proyectos ERP: la propuesta eEPC-COSMIC)

Autor:

Francisco Martín Téllez

Supervisores:

Dr. Maya Daneva, University of Twente

Dr. Nelly Condori-Fernandez, University of Valencia

Universidad de Twente

Departamento:

EEMCS - Information Systems

Enschede (Holanda), 20 Marzo 2009

Cotutor UC3M

D. José Luis López Cuadrado, Universidad Carlos III de Madrid

Coordinador Académico

D. José María Sierra Cámara

Tribunal:

Dr. Maya Daneva, Universidad de Twente

Dr. Nelly Condori-Fernández, Universidad de Valencia

Mrs. Silja Eckartz, Universidad de Twente

Revisores del proyecto y referencias para la calificación de:

D. Luigi Buglione, Consultor de Atos Origin

D. Juan R. Cuadrado-Roura, Universidad de Alcalá de Henares

Dr. Olga Ormandjieva, Universidad Concordia de Canadá

D. José Luis López Cuadrado, Universidad Carlos III de Madrid

Calificación:

Sobresaliente 9

RESUMEN

El mundo donde vivimos está cambiando continuamente debido al aumento de las nuevas maneras de comunicación, haciendo el mundo más global y rompiendo con las barreras tradicionales como las diferentes lenguas, diferentes culturas, distancia. Las compañías tienen que responder rápidamente a los cambios que la globalización demanda y por lo tanto sus sistemas de información necesitan estar actualizados y tener la capacidad de responder a estos cambios. Es en este contexto donde los sistemas ERP (Enterprise Resource Planning) aparecen. Un ERP es un sistema de información corporativo diseñado para coordinar todos los recursos, información y actividades necesarias para completar los procesos de negocio de una compañía, como pueden ser la facturación o el proceso de una orden de compra. Un sistema ERP intenta ensamblar todos los sistemas diferentes de una organización en uno solo, logrando la convergencia del software, hardware, empleados y procesos de trabajo.

Pero para adoptar un sistema ERP por parte de una organización, hay numerosos pasos que tienen que ser desarrollados. Una de estas fases es la de planificación en la cual la organización estima el coste, el tiempo y los recursos necesarios para la implementación del nuevo paquete ERP. Para realizar un contrato con un vendedor de ERP y/o un consultor de implementación de ERPs, para la organización cliente es necesario tener el tamaño del futuro producto. Sin embargo, es en esta etapa cuando es muy común que soluciones aisladas sean ofrecidas a la compañía cliente, apareciendo grandes problemas de estimación de tamaño y esfuerzo. No hay un estándar o consenso en un método que permita determinar el tamaño de un paquete ERP. Los existentes métodos de medición funcionales (FSMM), muchos de los cuales son estándar ISO, son interpretados para medir el tamaño funcional usando los requisitos de usuario funcionales. El problema radica en la dificultad de la aplicación de estos métodos tradicionales en el contexto de los ERP. Esto es incluso más problemático en las fases tempranas de requisitos debido a la falta de información en ellas. Por lo que en la práctica, cada compañía vendedora o compradora de ERP usan sus propias técnicas de medición y estimación.

El objetivo de este proyecto es reducir el hueco existente sobre la estimación del tamaño del producto ERP. Para ello, en este presente proyecto hemos diseñado una aproximación basada en uno de los últimos modelos FSM en ser establecido como ISO estándar, el método COSMIC. Un método de medición del tamaño funcional del software en diferentes dominios como “aplicaciones de negocio” o “sistemas de información” y sistemas de en tiempo real así como híbridos de ambos. Para determinar el tamaño de un producto ERP, se han mapeado en los conceptos de CSOMIC los requisitos funcionales utilizados en las especificaciones de un ERP. En concreto se ha empleado las plantillas de SAP, la empresa más importante en el contexto de los ERPs y conocidas como Business Blueprint. Estas plantillas para los requisitos funcionales muestran cómo desarrollar los procesos de negocio de una compañía usando el sistema SAP.

En el desarrollo de esta tesis, se han empleado tres métodos de investigación: i) Literature survey (Revisión de la literatura): para familiarizar al autor del proyecto con la literatura existentes, los conceptos principales del trabajo y coleccionar el trabajo previo

de investigación sobre los problemas y posibles soluciones estimando el tamaño funcional de un producto ERP. ii) Proof-of-Concept (Prueba de concepto): para ilustrar cómo funciona la técnica propuesta; y iii) Perception-based evaluation (Evaluación de la percepción), un estudio llevado a cabo con expertos para evaluar la propuesta en cuanto a su facilidad de uso, su futura intención de uso y su usabilidad.

La estructura del PFC está organizada de la siguiente manera: El capítulo 1 introduce el tema; el capítulo 2 presenta los principales conceptos; el capítulo 3 analiza el problema de estimación del tamaño en los ERPs; el capítulo 4 presenta el estado del arte en las empresas; el capítulo 5 describe nuestra solución; el capítulo 6 presenta y justifica nuestra propuesta (eEPC-COSMIC) acorde al método estándar COSMIC; el capítulo 7 presenta un ejemplo de la aplicación de la propuesta; el capítulo 8 muestra un estudio empírico acerca de la propuesta mostrando las percepciones de un grupo de expertos en los modelos FSM y finalmente el capítulo 9 presenta las conclusiones del proyecto y las líneas de trabajo futuras.

Aclarar que este resumen no sigue la estructura de la tesis final.

Índice

1. INTRODUCCIÓN.....	8
1.1 El marco del proyecto	8
1.1.1 ¿Por qué es importante medir?	8
1.1.2 La Ingeniería del Software y su importancia en la medición de proyectos Software	8
1.1.3 ¿Qué ofrece la ingeniería del software para medir proyectos de software?	9
1.1.4 Enterprise Resource Planning (ERP) y sus principales características	9
1.1.5 La metodología EPC (Business Blueprint)	10
1.1.6 Notación de un diagrama EPC y eEPC.....	11
1.2 Motivación del proyecto	13
1.2.1 Dificultades estimando el tamaño de ERPs	13
1.2.2 ¿Cuál es el motivo de este proyecto?	14
1.2.3 Posicionando el problema de la estimación en la etapa temprana de la Ingeniería de Requisitos para los proyectos ERP	14
1.3 Objetivos del proyecto	15
1.3.1 Objetivos y preguntas de investigación	15
1.3.2 Método de investigación	16
1.3.2.1 Literature survey	17
1.3.2.2 Proof-of-Concept.....	17
1.3.2.3 Perception-based evaluation	17
2. PRINCIPALES PROBLEMAS ESTIMANDO EL TAMAÑO DE UN ERP.....	18
2.1 Problemas y dificultades de medir en el contexto de los ERPS	18
2.2 Práctica actual estimando el tamaño de un paquete ERP.....	20
3. EN BUSCA DE UNA SOLUCIÓN.....	23
3.1 Primer paso (tamaño como funcionalidad)	23
3.2 Segundo paso, métodos multi-dimensionales o de medición de la funcionalidad	23
3.3 Tercer paso, ¿Qué método FSM?	24
3.4 Cuarto paso, eligiendo el documento adecuado para establecer las mediciones	25
3.5 Solución final	25
4. LA PROPUESTA eEPC-COSMIC.....	26
4.1 La iniciativa COSMIC.....	26
4.2 El método COSMIC	27
4.2.1 Principios	27
4.2.2 El meta-modelo COSMIC	27
4.2.3 Fases del método COSMIC.....	29

4.3	Método eEPC-COSMIC	30
4.3.1	Fase de estrategia	30
4.3.2	Fase de mapeado	31
4.3.3	Fase de medición o conteo	33
5.	BIBLIOGRAFÍA.....	35

Ilustraciones

<i>Ilustración 1 : Módulos de un ERP.....</i>	<i>9</i>
<i>Ilustración 2 : Elementos de un proceso de negocio representados en un diagrama EPC.....</i>	<i>12</i>
<i>Ilustración 3 : Ejemplo de un diagrama extendido EPC.....</i>	<i>13</i>
<i>Ilustración 4 : Fases de estimación en un proyecto ERP.....</i>	<i>15</i>
<i>Ilustración 5 : Método de investigación.....</i>	<i>17</i>
<i>Ilustración 6 : Conclusiones propias sobre los problemas de estimación del tamaño y esfuerzo en el contexto de los ERP.....</i>	<i>20</i>
<i>Ilustración 7 : Conclusiones sobre el estado del arte estimando el esfuerzo y determinando el tamaño en el contexto de los ERPs</i>	<i>22</i>
<i>Ilustración 8 : Primera aproximación hacia una solución en los problemas de estimación en entornos ERPs</i>	<i>23</i>
<i>Ilustración 9 : Segunda aproximación hacia una solución en los problemas de estimación en entornos ERPs</i>	<i>24</i>
<i>Ilustración 10 : Tercera aproximación hacia una solución en los problemas de estimación en entornos ERPs</i>	<i>25</i>
<i>Ilustración 11 : Último paso en busca de una solución en los problemas de estimación en entornos ERPs .</i>	<i>25</i>
<i>Ilustración 12 : Solución propuesta para estimar el tamaño de un ERP</i>	<i>26</i>
<i>Ilustración 13 : Modelo genérico del flujo de los movimientos de datos desde una perspectiva funcional..</i>	<i>27</i>
<i>Ilustración 14 : Meta-modelo de COSMIC.....</i>	<i>28</i>
<i>Ilustración 15 : Fases del método COSMIC.....</i>	<i>29</i>
<i>Ilustración 16 : Propósito y alcance de medición de un proyecto SAP</i>	<i>30</i>
<i>Ilustración 17 : Principios del método eEPC-COSMIC (fase de estrategia).....</i>	<i>31</i>
<i>Ilustración 18 : Reglas del método eEPC-COSMIC (fase de mapeado).....</i>	<i>32</i>
<i>Ilustración 19 : Identificación de usuarios y frontera en un proceso funcional representado por un diagrama eEPC.....</i>	<i>32</i>
<i>Ilustración 20 : Reglas del método eEPC-COSMIC (fase de medición)</i>	<i>34</i>

1. Introducción

1.1 El marco del proyecto

El presente PFC se engloba dentro del contexto de los ERPs (Enterprise Resource Planning), del marco de la Ingeniería del Software (IS) y en concreto dentro de la fase de Ingeniería de Requisitos (IR) para ERPs.

1.1.1 ¿Por qué es importante medir?

Los humanos siempre tienen la necesidad de medir las cosas debido a que es muy útil en nuestra vida diaria y cada día encontramos muchísimas situaciones en las que necesitamos medir. Por ejemplo es importante medir las distancias entre puntos, el peso de los objetos, medir el tiempo que alguien necesita para realizar una actividad, saber el tamaño de una casa antes de construirla, coleccionar una serie de valores antes de emitir un diagnóstico ante un problema de salud o simplemente poder determinar los precios de los productos. Es decir, necesitamos medir en todas las facetas de la vida y para ello necesitamos sistemas de medida estándar, donde encontramos la primera dificultad en nuestro proyecto.

1.1.2 La Ingeniería del Software y su importancia en la medición de proyectos Software

La Ingeniería del software (IS) es una disciplina tecnológica y administrativa incluida en área de la Informática o Ciencias de la Computación, que ofrece métodos y técnicas para desarrollar y mantener software de calidad que resuelven problemas de todo tipo y en concreto problemas de estimación de tamaño y esfuerzo. Además ofrece actividades que incluyen la administración, el cálculo de costes, la planificación, modelado, análisis, especificación, diseño, implementación, pruebas y mantenimiento de un sistema software. En la Ingeniería del software, un problema de estimación de costes es el problema de precedir la cantidad de esfuerzo humano y material, así como el tiempo necesario para construir un sistema software

En el mundo de los proyectos software la dificultad de determinar las medidas de un proyecto software o su esfuerzo se incrementa. La realización de un programa y el marco en que se crea, un proyecto software, es un tipo especial de proyecto que tiene algunas características especiales que lo diferencian de otros proyectos de ingeniería:

- Dificultad de estimación de tiempo y coste.
- Metodologías y Estándares poco maduros.
- Uso de tecnologías de desarrollo, gestión, diseño, etc. en continuo cambio y siempre en evolución.
- Dificultad para determinar la calidad de un producto software, ya que no es un producto tangible.
- Coste muy elevado ante el absentismo y posibles bajas de personal.

1.1.3 ¿Qué ofrece la ingeniería del software para medir proyectos de software?

En respuesta a estos problemas, la organización ISO definió en 1999 un grupo de Métricas de tamaño funcional (Functional Size Measurement Method, FSMM), también conocidos como análisis de puntos de función (Function Point Analysis, FPA). Estos métodos permiten la medición del tamaño de un sistema de información expresado en funcionalidad y su unidad es de puntos de función. En concreto miden que es lo que se entrega al cliente no como es entregado. El método ISO es mantenido al día por un amplio grupo de cooperantes o grupos de usuarios como NESMA (Netherlands Software Metrics Users Association Function Point Analysis) [www.nesma.nl], COSMIC (Common Software Measurement International Consortium) [www.cosmicon.org], IFPUG (International Function Point Users Group) [<http://www.ifpug.org>] ó Finnish Software Measurement Association (FISMA) [[English Methods](#)]). Cada uno de estos grupos tiene su propio método estándar.

1.1.4 Enterprise Resource Planning (ERP) y sus principales características

Enterprise resource planning (ERP) es un sistema de información corporativo diseñado para coordinar todos los recursos, información y actividades necesarias para completar los procesos de negocio de una compañía, como pueden ser la facturación o el proceso de una orden de compra. El termino ERP fue introducido por el investigador y la firma analista Gartner, y surgió como una extensión de los MRP (material requirements planning) y CIM (Computer Integrated Manufacturing). Los sistemas ERP en la actualidad tratan de cubrir todas las funciones básicas de una organización.

Los sistemas ERP típicamente gestionan los procesos logísticos, de manufacturación, distribución, inventario, envío, facturación y cuentas de una compañía. Los ERPs ayudan en el control de muchas actividades de negocio y departamentos como ventas, marketing, producción, gestión de la calidad, recursos humanos, finanzas, ingeniería, etc.



Ilustración 1 : Módulos de un ERP

Hay diversas características que hacen de estos sistemas fundamentales en el mundo global de hoy: i) un sistema ERP está basado en una única base de datos común y con un diseño de software modular. Esta base de datos permite que cada departamento almacene y recupere información en tiempo real, y además permite que la información sea fácil de acceder y de compartir; ii) Esto a su vez permite que todas las sucursales y oficinas de

una empresa estén interrelacionadas en tiempo real; y iii) los ERPs permiten a las organizaciones propietarias la integración con proveedores, clientes y partners, lo que permite una comunicación permanente con ellos y el ahorro de gran cantidad de tiempo.

1.1.5 La metodología EPC (Business Blueprint)

El modelo Business Blueprint desarrollado por SAP está basado en el marco de la arquitectura de Sistemas de Información ARIS (Architecture of Integrated Information System) [1] para modelar los procesos de negocio. El concepto ARIS es ampliamente usado en proyectos comerciales en el campo de la gestión y análisis de sistemas de información. Permite separar los procesos de negocio en diferentes vistas y la integración de todas ellas da lugar a una visión completa de un proceso de negocio y que en nuestro proyecto consideraremos un requisito funcional. El marco ARIS presenta las siguientes vistas para un proceso de negocio:

- a) Vista funcional: responde a la pregunta ¿qué ocurre? y representa las funciones de un proceso de negocio y como se relacionan unas con otras.
- b) Vista organizativa: responde a la pregunta ¿Quién hace qué? Y muestra la estructura social de una organización y quién participa en un proceso funcional, representando a estas unidades como unidades organizativas o de la organización.
- c) Vista de los datos: responde a la pregunta ¿Qué se necesita? En un proceso de negocio y analiza como los objetos de información, es decir los datos, interactúan con las funciones. El modelo de datos además representa la información de los objetos y sus relaciones.

A estos modelos la propia SAP les añadió la siguiente vista al marco ARIS:

- d) Modelo de iteración: responde a la pregunta ¿Cómo los modelos de la compañía interactúan? Muestra las unidades de la organización involucradas un un proceso de intercambio de información y como son los flujos de información entre los destinatarios y los que envían esa información.

Estas cuatro vistas se resumen en una única, el modelo EPC, en ARIS llamado vista de control. La metodología Event-Driven Process Chain (EPC) que da nombre al diagrama de control o central de un proceso de negocio fue desarrollada por el instituto de Sistemas de información (IWi) de la universidad de Saarland, Alemania, en colaboración con SAP AG. Es el componente clave para modelar los conceptos de la ingeniería de negocios [2].

El modelo central EPC concentra los cuatro elementos explicados anteriormente representados con los siguientes conceptos: eventos (“Cuando se debería de hacer algo”); tareas o funciones (“Que debe ser realizado”); organización (“Quien debe hacer algo”); y comunicación o información (“Que información es requerida para hacer una tarea de forma adecuada”); En toda esta actividad se basa un EPC y representa un proceso de negocio y un requisito funcional dentro de los ERPs, permitiendo modelar una secuencia temporal o lógica de funciones.

1.1.6 Notación de un diagrama EPC y eEPC

La notación de la vista de control o central de un proceso de negocio se resume en la siguiente tabla:

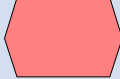


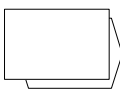

Notación de un ECP				
Nombre	Elemento	Convenciones de nombrado	Descripción	Ejemplo
Evento (event)		Objeto + verbo en participio describiendo el estado que ha alcanzado el objeto	Un evento puede ser el disparador de una función o el resultado después de una actividad.	“Una orden ha llegado” (“Order arrived”)
Función (function)		Verbo presente + objeto	Representa una cierta actividad o tarea que tiene que ser ejecutada por una persona o sistema, que requiera una entrada que alcanza un determinado estado como producto de su salida.	“Procesar la orden” (“Process order”)
Operadores lógicos (logical operators)		-	Se emplean para conectar funciones y eventos . Son usados para indicar decisiones (XOR, OR) o ejecuciones de una o más funciones en paralelo (AND).	Una orden puede ser procesada por teléfono XOR mail.
Camino de proceso (Process Path)		Verbo presente + objeto	Más o menos el mismo elemento que una función. Se emplea para describir las jerarquías en los diagramas (abstracto → detallado)	EPC detallado para la función “Procesar la orden”
Control de flujo (Control Flow)		-	Conectan eventos con funciones, caminos de proceso o conectores lógicos creando una secuencia cronológica e interdependencias lógicas entre ellos.	

Tabla 1 : Notación de un diagrama EPC

Las funciones están enlazadas por eventos y un evento se puede caracterizar como el estado en que se encuentra un objeto de información en un determinado momento. Los eventos también pueden ser el resultado de una función (evento de salida). De este modo se pueden ver como los elementos clave dentro de un proceso. Además múltiples funciones pueden ser resultado de un solo evento, y múltiples funciones pueden concluir en un único evento que puede ser disparado.

La siguiente ilustración explica la notación de una forma gráfica:

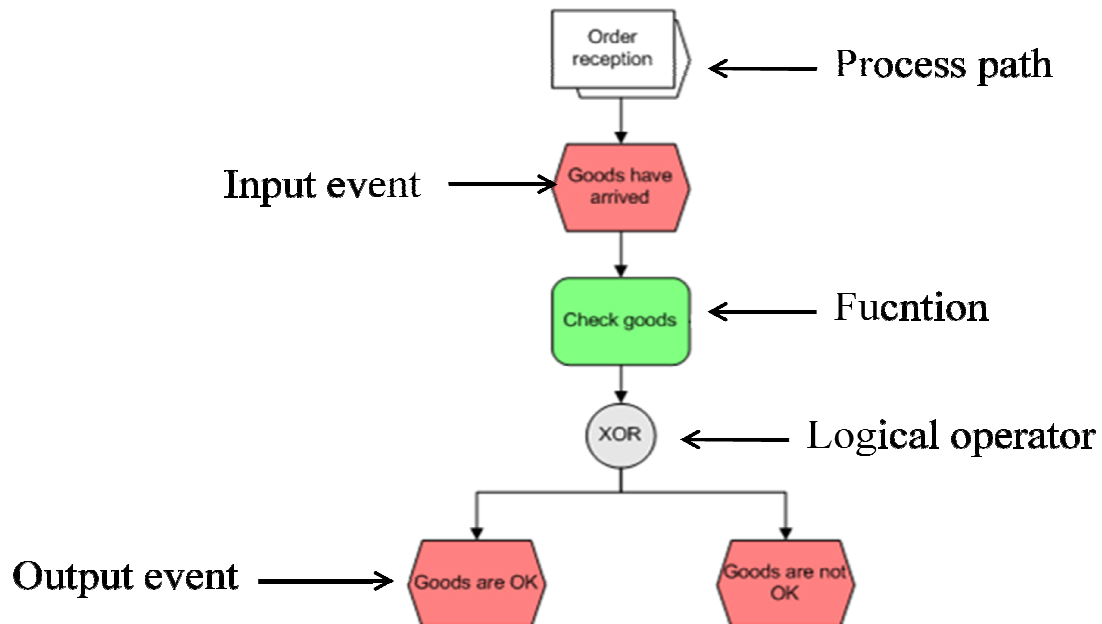


Ilustración 2 : Elementos de un proceso de negocio representados en un diagrama EPC

Ahora bien, acorde a lo explicado anteriormente en este tipo de diagramas faltaría mostrar que información necesita una función para ser procesada y quién son los responsables de una función o los destinatarios de su salida. Este tipo de información se encuentra representada en los diagramas eEPC, que son la versión extendida de los EPC y que son los que se utilizarán en nuestra propuesta eEPC-COSMIC.

Notación de un diagrama extendido EPC (eEPC)				
Nombre	Elemento	Convenciones de nombrado	Descripción	Ejemplo
Unidad de organización (Organization Unit)		Sustantivo	Representan roles o personas que son responsables de una cierta función.	Dep. de ventas es responsable de procesar la orden
Objeto de información (Information Object)		Sustantivo	Uno objeto de información puede ser visto como entrada o salida de una función.	Entrada: Orden de pedido Salida: Factura
Flujo de información (Information Flow)		-	Conectan las funciones con los objetos de información, mostrando si la información es entrada o salida de una función.	
Asignación de una unidad organizativa (Organization Unit Assignment)		-	Conectan las funciones con las unidades organizativas, señalando quien participa en la función.	

Tabla 2 : Notación de un diagrama extendido EPC

En la siguiente figura se representa un proceso (requisito funcional) con la notación extendida EPC y que muestra el proceso de compra de algunos artículos en una tienda online. El proceso empieza con la llegada de un cliente y finaliza con la entrega del material. Es importante indicar que dentro de los ERPs, suele haber unas “mejores

prácticas” para representar o llevar a cabo un proceso, es decir la mejor manera y más eficiente. En concreto SAP tiene unas 1000 plantillas como la siguiente para representar 1000 procesos de negocio, desde la facturación al envío de un pedido.

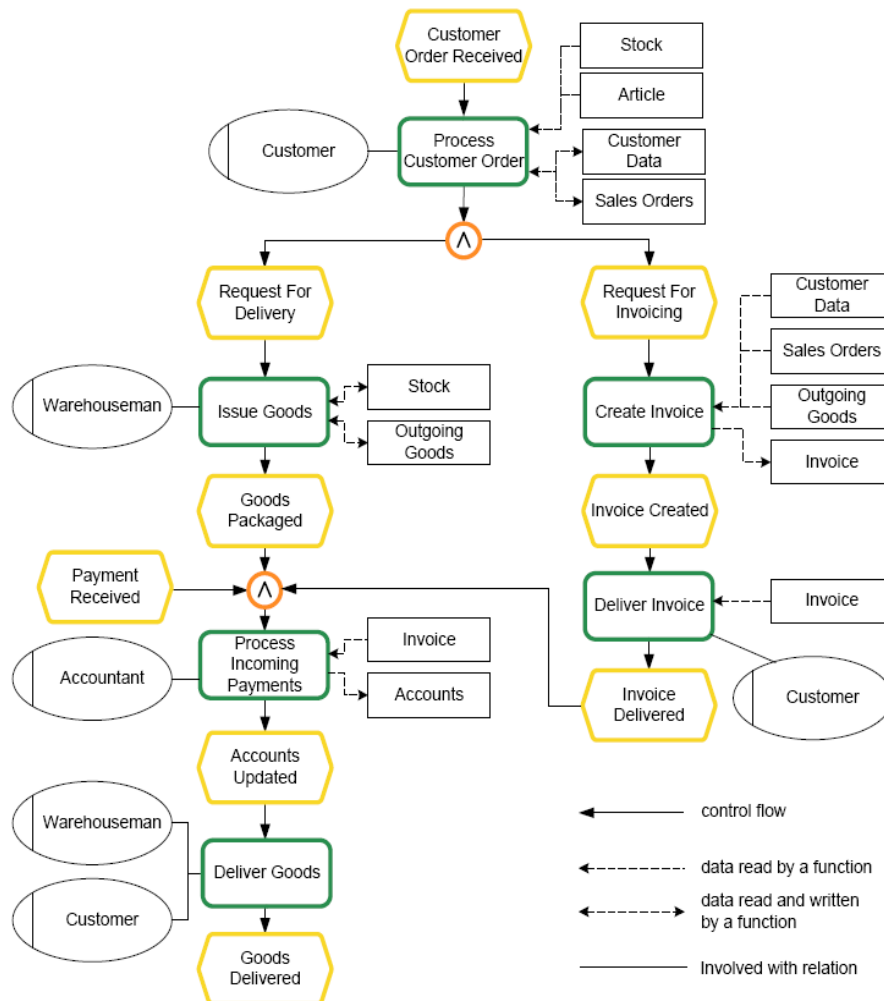


Ilustración 3 : Ejemplo de un diagrama extendido EPC

1.2 Motivación del proyecto

1.2.1 Dificultades estimando el tamaño de ERPs

En el contexto de los ERPs, el proceso de estimación es mucho más complicado que en un proyecto software tradicional. El tener una sola base de datos y tener tantos módulos tan diferentes, que a su vez pueden ser elegidos de varios vendedores, hacen que el proceso de desarrollo de este software sea muy complicado. A parte de ello, los modelos de medición existentes en la literatura no aportan soluciones completas para la estimación del tamaño de un ERP y por consiguiente su coste. Alguna de las razones por la que los FSM métodos no se adaptan bien al entorno ERP es porque fueron creados para el software tradicional y además su aplicación en la fase inicial de los requisitos es complicada.

1.2.2 ¿Cuál es el motivo de este proyecto?

La importancia de los ERPs en los últimos años se ha incrementado de forma notable, por ejemplo el año pasado los ERPs ocuparon la posición número 1 entre las prioridades de las compañías en cuanto a gasto de dinero en cuanto a nuevas tecnologías. Por otro lado la fase de estimación del tamaño en un proyecto software es fundamental. Como veremos más adelante, este problema de la medición de proyectos ERPs es un gran inconveniente en la fase de estimación del proyecto y requiere continua atención. Hay algunas propuestas para solucionar estos problemas pero son soluciones aisladas, sin gran consenso y muchas veces privadas y desarrolladas para las compañías, con lo cual al final cada empresa acaba empleando sus propias técnicas o acaba estimando según la experiencia de sus empleados, lo que trae numerosos problemas en el futuro ante cambios en los requisitos de los proyectos, bajas por parte de los empleados, necesidades específicas en el producto dependiendo de la compañía adoptante del ERP, etc. Por lo que es en la fase de estimación del ciclo de vida del software donde se detecta un hueco en el contexto de los ERPs, ya que como se ha mencionado anteriormente los modelos estándar no están adaptados a estos sistemas de información. Reducir ese hueco es el principal objetivo del proyecto fin de carrera. Para ello esta tesis introducirá una aproximación para evaluar el tamaño de un ERP, adaptando el método estándar COSMIC a los documentos de requisitos funcionales de un ERP, en concreto usando las plantillas (Business Blueprint) de la compañía SAP

1.2.3 Posicionando el problema de la estimación en la etapa temprana de la Ingeniería de Requisitos para los proyectos ERP

El objetivo de esta tesis se refiere a la etapa temprana de la ingeniería de requisitos de los ERPs. Esto es cuando hay se inicia el proceso de petición para propuestas (request-for-proposal, RFP) por parte de la compañía interesada en adoptar un ERP o bien cuando una compañía empieza a desarrollar los requisitos iniciales para un nuevo proyecto de ERP.

Desde el momento en que una compañía empieza el proceso de oferta e invita a las consultoras a enviar un RFP, o en el momento cuando el proyecto empieza, hay requisitos disponibles que permiten hacer una primera aproximación en el esfuerzo necesario y requerido para desarrollar un proyecto. Este proyecto se centra en la implementación de un paquete particular SAP [www.sap.com]. Se considera que la metodología estándar EPC (Event-driven Process Chain) para la implementación de un proyecto SAP, asume que los requisitos de SAP (representados mediante la Business Blueprint) están desarrollados cuando la etapa de RFP está terminada. La idea de esta tesis, es analizar una posible solución para estimar el tamaño de un proyecto desde este primer paso del ciclo de vida. Una vez con el tamaño del producto se podrá determinar el esfuerzo necesario para un proyecto y después su duración.



Ilustración 4 : Fases de estimación en un proyecto ERP

1.3 Objetivos del proyecto

Los principales objetivos del proyecto están desarrollados por las siguientes actividades de investigación:

1. Comprender la practica actual sobre como los proyectos ERP son estimados.
2. Entender porque es tan difícil la aplicación de los modelos de medición en los proyectos ERPs y porque hay una falta de credibilidad hacia ellos en este contexto.
3. Adaptar una técnica de medición existente en el contexto de los ERPs desde la fase de requisitos.
4. Para demostrar a través de expertos si la propuesta planteada cumple con los conceptos del estándar ISO COSMIC y cuáles son sus percepciones en cuanto a un futuro uso, su facilidad de uso y su usabilidad.

1.3.1 Objetivos y preguntas de investigación

El principal objetivo del proyecto es reducir el gap entre las técnicas de medición de software y las necesidades de estimación en los proyectos de ERP, proponiendo un método basado en el estándar COSMIC y que sea aplicable en la fase temprana de requisitos de un ERP. Con ello se podrá determinar el tamaño funcional de un producto ERP y acercarnos más a la estimación del esfuerzo y duración de un proyecto ERP.

La pregunta central que será respondida es la siguiente:

De que manera basándose en los requisitos funcionales se puede estimar el tamaño de un ERP en la fase inicial de su implementación?

Esta pregunta u objetivo inicial, se descompone en cinco sub-objetivos, que permiten formular las siguientes preguntas de investigación:

- 1) Sub-objetivo 1: Identificar las diferencias entre un proyecto ERP y un proyecto de software “tradicional”:
 - RQ1.1: ¿Cuales son las diferencias entre un proyecto ERP y un proyecto de software tradicional?
 - RQ1.2: ¿Que implicaciones tienen estas diferencias para determinar el tamaño y esfuerzo de un proyecto ERP?
 - RQ1.3: ¿Por qué es difícil medir un proyecto ERP?
 - RQ1.4: ¿A la hora de medir, que factores hacen diferente a un proyecto ERP de un proyecto de software estándar?

- 2) Sub-objetivo 2: Descubrir porque las técnicas existentes no solucionan el problema de estimación del tamaño funcional de un proyecto ERP:
 - RQ2.1: ¿Por qué es tan necesario estimar en la primera fase del ciclo de vida en el contexto de los ERPs?
 - o ¿Qué métricas existen en esta fase?
 - o ¿Son aplicables al contexto de los ERPs?
 - RQ2.2: ¿Las actuales métricas de tamaño proporcionan Buenos métodos para estimar proyectos ERPs?
 - o Si la respuesta es negativa, ¿Cuáles serían las razones?
- 3) Sub-objetivo 3: Conocer el estado del arte midiendo proyectos ERPs:
 - RQ3.1: ¿Cómo es un proyecto ERP medido?
 - o ¿Qué entienden las compañías por el término “SIZE”?
 - ¿Qué tipo de documentos usan para estimar?
 - RQ3.2: ¿Qué tipo de métricas o métodos emplean para estimar el tamaño de un paquete ERP?
- 4) Sub-objetivo 4: Diseñar una solución para los problemas sobre la estimación del tamaño en el contexto de un proyecto ERP:
 - RQ4.1: ¿Cómo desarrollar un procedimiento de medición?
 - RQ4.2: ¿Qué primitivas en los artefactos de requisitos funcionales en los ERPs (por ejemplo: dentro de la metodología EPC los eventos, funciones, objetos de información, conectores lógicos), podrían ser usados para el proceso de conteo del tamaño?
 - RQ4.3: ¿De qué manera se pueden mapear los requisitos de ERPs en las especificaciones de los conceptos de medición (p. ej: en COSMIC los movimientos de datos READ, WRITE, EXIT, ENTRY) de un FSM método estándar?
- 5) Sub-objetivo 5: Evaluar la nueva propuesta por expertos:
 - RQ5.1: ¿Podría ser integrada la propuesta en un proceso más largo de ingeniería de requisitos?
 - RQ5.2: ¿Es la propuesta fácil de usar, acorde a las percepciones de los expertos?
 - RQ5.3: ¿Es percibido el método como usable por los expertos en COSMIC?
 - RQ5.4: ¿Cómo es la intención de uso del método en futuros proyectos ERPs?
 - RQ5.5: ¿De qué manera el método podría ser mejorado basado en las sugerencias de los expertos?

1.3.2 Método de investigación

Esta investigación es conceptual, cualitativa e interdisciplinar. Es una investigación que necesita sintetizar e integrar información para desarrollar una propuesta de medición en proyectos ERPs basándose en los requisitos del producto. Las siguientes técnicas han sido empleadas.

1.3.2.1 Literature survey

El principal objetivo de la “revisión de la literatura” es familiarizar al autor del proyecto con la literatura existente, los conceptos principales del trabajo y coleccionar el trabajo previo de investigación sobre los problemas y posibles soluciones estimando el tamaño funcional de un producto ERP.

1.3.2.2 Proof-of-Concept

Prueba del concepto, empleada para ilustrar cómo funciona la técnica propuesta. En este caso se muestran especificaciones de requisitos SAP disponibles en libros de ERP [3] y ejemplos de la vida real proporcionados por la supervisora de la tesis (Maya Daneva). Este método está combinado con el anterior para solucionar el sub-objetivo 4.

1.3.2.3 Perception-based evaluation

La Evaluación de la percepción [4] sobre el método es un estudio llevado a cabo con un grupo de expertos en el ámbito de los métodos FSM y en COSMIC (dos de España, uno de Canadá y uno de Italia). El objetivo evaluar la propuesta presentada en esta tesis eEPC-COSMIC [5] en cuanto a su facilidad de uso, su futura intención de uso y su usabilidad. Para ello se ha realizado un cuestionario [6] y se ha empleado el Método de Adopción de [7] que incluye los tres constructores realacionados con la facilidad de uso, usabilidad e intención de uso.

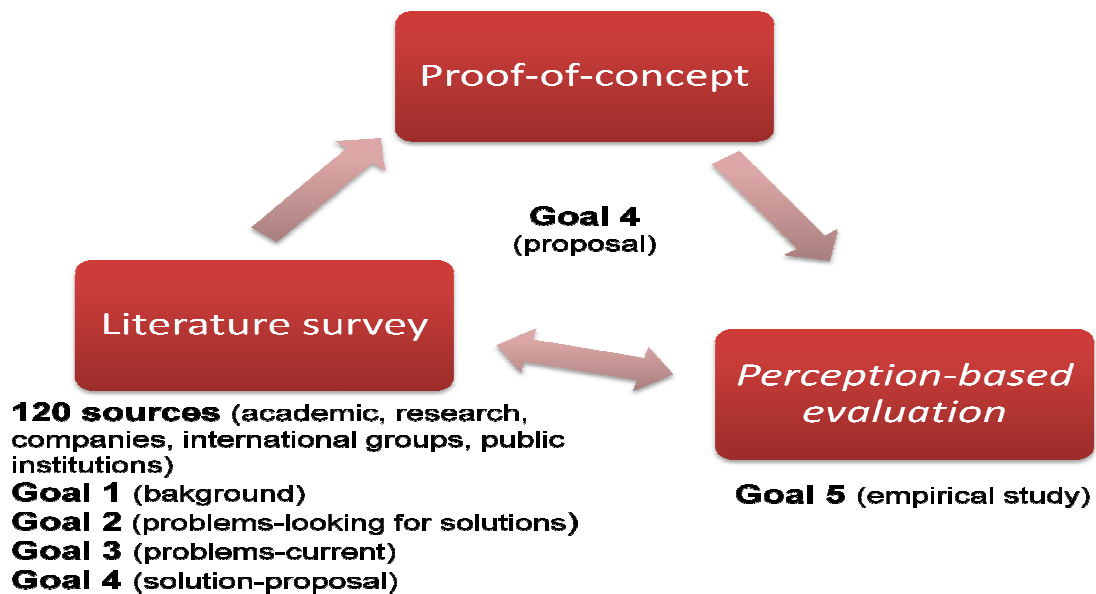


Ilustración 5 : Método de investigación

2. Principales problemas estimando el tamaño de un ERP

Esta sección se encuentra subdividida en dos. En la primera se muestra un resumen los problemas encontrados en la literatura, una serie de razones de porque los existentes métodos de medición de la funcionalidad del software no son válidos en el contexto de los ERPs y la importancia de medir en las fases iniciales del ciclo de vida, y en segundo lugar, se muestran nuestras propias conclusiones de cómo las compañías miden los proyectos ERPs en la actualidad y que entienden por tamaño.

2.1 Problemas y dificultades de medir en el contexto de los ERPS

Esta sección presenta un resumen con las conclusiones del autor las cuales son descritas con texto y también de una manera gráfica la cual se sitúa en medio de la explicación. Las diferentes conclusiones se van encadenando hasta alcanzar una conclusión central:
Es necesario medir un ERP desde los requisitos funcionales en las primeras fases del ciclo de vida.

Esta información se encuentra analizada a lo largo del capítulo 3 de la memoria final.

- **ERP presentan importantes diferencias con los proyectos de SW tradicional**

Hay importantes diferencias entre los proyectos ERP y los proyectos en los que se desarrolla software tradicional (sistemas de información o software personalizado). Algunas de estas diferencias son: la cantidad de funcionalidad que presenta un ERP; el entorno inter-organizacional en que se desarrolla; la dificultad de la entrega y puesta en marcha al ser un sistema compartido con proveedores, clientes y partners; las diversas configuraciones que se dan para una misma actividad funcional; la automatización de los procesos; ó entre otras, la existencia de una única base de datos para todas las actividades.

- **Métricas existentes creadas para proyectos de software tradicional**

Queda patente que los existentes métodos para la medición de la funcionalidad fueron creados para el software tradicional y en el presente las nuevas tecnologías son completamente diferentes. La falta de adaptación de estas métricas a estos nuevos entornos y tecnologías, hace que tampoco sean adaptables en el contexto de los ERPs. Además la mayoría de estas técnicas existentes no pueden ser aplicadas en las fases iniciales de los requisitos.

- **No hay estándares para determinar el tamaño de un ERP**

La reciente aparición de los ERPs, el hecho de que cada compañía utilice sus propias técnicas de desarrollo y la falta de adaptación de los métodos estándares a esta nueva tecnología, ha impedido que la industria de los ERPs desarrolle un estándar de medición.

- **Cada compañía tiene su propio método de medición**

La falta de estándares y el secretismo en muchos casos cuando un implementador de ERPs desarrolla una métrica de medición, hace que cada compañía finalmente aplique sus métodos de medición.

- **Las estimaciones no son creíbles ni fidedigna**

El hecho de que cada compañía emplee sus propias técnicas para contar el tamaño de un ERP hace que las medidas en muchos casos no sean muy exactas. En numerosas situaciones no hay ninguna métrica, y los jefes de proyecto y estimadores estiman el tamaño de un ERP basándose en su propia experiencia y/o usando datos de proyectos anteriores sin una validación de los datos y los cuales pueden diferir de forma notable en el nuevo proyecto.

- **Grandes proyectos (Consumen muchos recursos)**

La implantación de un ERP a veces suele durar varios años, y en su desarrollo y puesta en marcha es necesario el empleo de numerosos recursos, tanto humanos como materiales, debido precisamente a las características que se citan en el primer punto de esta subsección 2.1.

- **Directores y directivos asumen altos costes**

Otra asunción que se hace en el contexto de los ERPs, es entender que la implantación de un ERP es un proyecto de dimensiones gigantes respecto a un software tradicional. Esto es algo cierto, pero el hecho de que emplee numerosos recursos y tiempo para su desarrollo, no tiene que implicar que se asuman costes desproporcionados y que se tenga que gastar una elevada cantidad de dinero por parte de la empresa acogedora del ERP. Los directivos al no tener métricas creíbles tienen establecida la creencia de que implantar un ERP a veces es un cheque en blanco y asumen elevados riesgos.

- **Esencial medir en la primera etapa del ciclo de vida**

Si no hay buenas aproximaciones del tamaño de un ERP y una buena estimación del esfuerzo la planificación tiene muchas posibilidades de salir mal. En esta situación, un posible fallo de implementación arruinaría el proyecto y las pérdidas serían irreversibles e irrecuperables para las compañías. Es por esto, que cuanto antes se estime en el ciclo de vida mejor para anticipar posibles riesgos, esta fase podría considerarse como la “fase temprana de los requisitos funcionales”.

- **TAMAÑO desde los requisitos funcionales**

En esta fase inicial de los requisitos solo hay entregables del proyecto, es decir, documentos de requisitos o cualquier otro documento que describe el sistema por lo que se han de emplear los métodos que miden puntos funcionales de una aplicación, ya que en esta fase por ejemplo pantallas o líneas de código.

Es en este punto donde queda claro, la principal conclusión de nuestro análisis:

Es necesario medir un ERP desde los requisitos funcionales en las primeras fases del ciclo de vida.

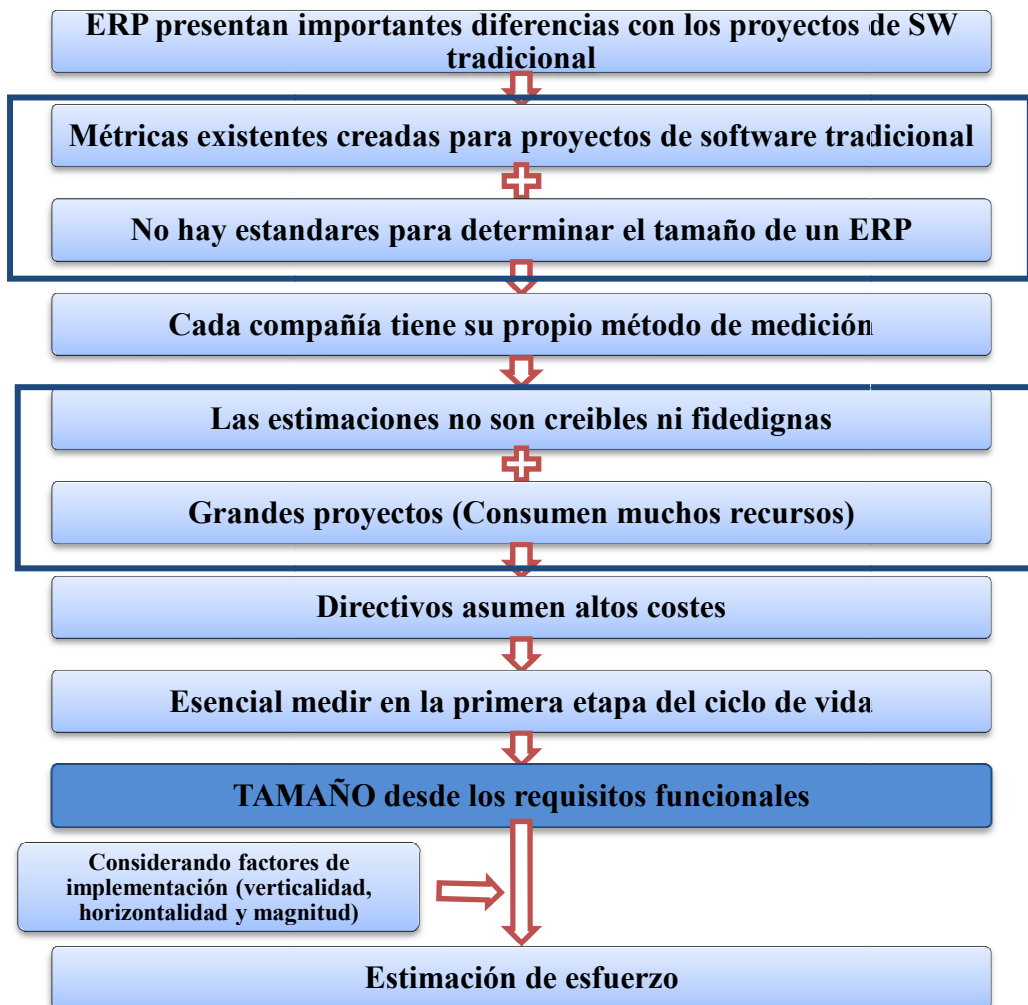


Ilustración 6 : Conclusiones propias sobre los problemas de estimación del tamaño y esfuerzo en el contexto de los ERP

- **Estimación de esfuerzo teniendo en cuenta los factores claves de implementación de un ERP (horizontalidad, verticalidad y magnitud)**

Finalmente, una vez que se tiene el tamaño funcional estaremos más cerca del tamaño final de la aplicación y por lo tanto se estará más cerca de estimar el esfuerzo necesario para implantar un ERP. En este punto de pasar de tamaño a esfuerzo se han de tener en cuenta los factores de implementación si se quieren lograr buenos resultados de esfuerzo. (Esto está fuera del alcance de este PFC, pero se deja a modo de introducción).

2.2 Práctica actual estimando el tamaño de un paquete ERP

Esta sección muestra el resumen del estado del arte de la fase de estimación de tamaño y esfuerzo de un ERP, información que se encuentra a lo largo del capítulo 4 de la memoria final.

Conclusión	Título	Descripción
A	<i>SLOC (tamaño estimado en líneas de código) no es válido para un proyecto ERP</i>	Con los métodos de medición SLOC, no se puede obtener un tamaño adecuado hasta que el diseño de una aplicación es completa y en los proyectos ERPs no hay aproximaciones válidas en este contexto para determinar el esfuerzo.
B	<i>Los modelos o SLOC determinan solo una medida</i>	Ambas técnicas no son completamente adecuadas para determinar el esfuerzo y tamaño en un proyecto ERP porque solo usan una medida como variable predictiva, y un proyecto ERP implica numerosas variables.
C	<i>Las estimaciones ERP no son exactas</i>	Las estimaciones de los paquetes ERPs están basadas en el propio conocimiento y experiencia de los estimadores. Por otro lado muchas veces la empresa que implanta el ERP no tiene expertos para leer e interpretar los modelos ERP. Además muchas veces, los implementadores de ERP usan datos de antiguos proyectos para validar los datos presentes de un proyecto en curso.
D	<i>Los modelos FSM pueden considerarse apropiados para determinar el tamaño funcional de un ERP</i>	Se han encontrado diferentes compañías que utilizan FSMM para determinar el tamaño funcional de un ERP. Si bien es cierto, estos análisis dependen de la habilidad de los expertos para interpretar los diagramas que describen la solución ERP y a menudo se miden otras variables no solo la funcionalidad.
E	<i>Los documentos de requisitos funcionales son considerados como un nivel adecuado para determinar el tamaño</i>	Este nivel de detalle se suele considerar apropiado para determinar el tamaño de un ERP. Las compañías que utilizan FSMM para determinar el tamaño, suelen utilizar los documentos de procesos de negocios que equivalen a los documentos de requisitos en el contexto de los ERPs.
F	<i>No hay suficiente trabajo entre todos los interesados en el ERP</i>	Esto produce que haya una falta de credibilidad y consenso al establecer el esfuerzo y coste de un proyecto ERP. En numerosas ocasiones las empresas que adoptan un ERP viven con estimaciones que reciben de consultores externos, ya que el vendedor de ERP suele ofrecer escasas estimaciones debido a razones de confidencialidad.
G	<i>Importantes factores no son considerados para medir</i>	Factores como la cultura de la empresa que adopta el ERP, prioridades de negocio, cantidad de módulos y componentes que necesitan ser diseñados a medida no son tenidos en cuenta a la hora de estimar, haciendo incompletas las estimaciones.
H	<i>Confusión sobre lo</i>	Hay una confusión sobre lo que el término

Conclusión	Título	Descripción
	<i>que el término “TAMAÑO” significa</i>	“TAMAÑO” es interpretado y como se determina dentro de las empresas relacionadas con los ERPs. Hay varias definiciones: “tareas del proyecto”, “número de trabajadores” y “Funcionalidad de un ERP”.
I	<i>Los proyectos ERP son tratados como proyectos tradicionales</i>	Un importante número de proyectos ERP son tratados como proyectos tradicionales, tanto en la implementación como en la estimación.

Ilustración 7 : Conclusiones sobre el estado del arte estimando el esfuerzo y determinando el tamaño en el contexto de los ERPs

3. En busca de una solución

En este capítulo tras analizar la problemática estableciendo el tamaño midiendo ERPs, se formula nuestra solución razonando paso a paso cada elección.

3.1 Primer paso (tamaño como funcionalidad)

La principal conclusión que se tiene sobre el problema de la medición de un producto ERP es la siguiente (sección 2.1):

Es necesario medir un ERP desde los requisitos funcionales en las primeras fases del ciclo de vida.

Esta suposición es corroborada con las conclusiones obtenidas en el estado del arte en el contexto de los ERPs. En el capítulo 5 de la memoria, se van demostrando como las hipótesis de la problemática de la medición de ERPs obtenidas por el autor son corroboradas con las conclusiones del estado del arte. Se desgana paso a paso hasta que obtiene la siguiente y primera aproximación a la solución:

Primera aproximación hacia la solución para solventar el problema de estimación de coste y esfuerzo en proyectos
Determinar el tamaño (funcionalidad de un ERP) desde un documento de requisitos funcionales aplicando un método FSM o una métrica Multi-dimensional.

Ilustración 8 : Primera aproximación hacia una solución en los problemas de estimación en entornos ERPs

Hemos visto que en el ámbito empresarial para definir el tamaño de un ERP hay varias definiciones: “tareas del proyecto”, “número de trabajadores” y “Funcionalidad de un ERP”. Nosotros en este primer paso hacia la solución establecemos que el tamaño será considerado la funcionalidad de cualquier aplicación ERP, ya que en la primera fase del ciclo de vida solo se tiene los requisitos del usuario, es decir la funcionalidad de la aplicación, antes de realizar cualquier planificación de tareas y estimación de número de trabajadores.

3.2 Segundo paso, métodos multi-dimensionales o de medición de la funcionalidad

En este paso, se presenta la respuesta de porqué usaremos un método FSM para determinar el tamaño de una aplicación ERP, en vez de una métrica multidimensional. Las principales razones son las siguientes:

- a) No hay ningún estándar de métricas multidimensionales. Sin embargo hay varios FSMM reconocidos como estándar por la organización ISO para medir el tamaño de los requisitos funcionales de usuario.
- b) No ay información pública da sobre las métricas multidimensionales, por el contrario hay elevada cantidad de información sobre los FSMM.
- c) Algunos de los modelos FSM pueden ser aplicados en las etapas iniciales de la toma de requisitos.

- d) La simplicidad es otra cualidad de este tipo de métricas (FSM). Desde su concesión el objetivo de los FSMM ha sido su facilidad para aplicarse, en cambio de las métricas multi-dimensionales tienen en cuenta muchísimos factores, que escapan a la funcionalidad de la aplicación.

En nuestra búsqueda hacia la solución se establece que se aplicará un método de medición de la funcionalidad del software.

Segunda aproximación hacia la solución para solventar el problema de estimación de coste y esfuerzo en proyectos

Determinar el tamaño (funcionalidad de un ERP) desde un documento de requisitos funcionales aplicando un método FSM.

Ilustración 9 : Segunda aproximación hacia una solución en los problemas de estimación en entornos ERPs

3.3 Tercer paso, ¿Qué método FSM?

En este paso, se decide que método estándar FSM se va a utilizar de los 5 presentes derivado, tres derivados de la metodología original Albrecht: IFPUG Function Point Analysis IFPUG [<http://www.ifpug.org>], NESMA Function Point Analysis [www.nesma.nl] y el método de la asociación Finesa de Medición de software, Finnish Software Measurement Association, FISMA [[English FISMA Methods](http://www.fisma.org)]; y otros dos extensiones derivadas de su método: MK II [www.ukσμα.co.uk] y COSMIC Functional Sizing Methods [www.cosmicon.org]. También se introduce el método PSU [[referencia al manual PSU](#)], que es un Nuevo método de medición funcional de software y que es considerado por algunas compañías importantes. Para ver información más detallada y consultar las principales características de cada método consultar el punto 5.2 de la memoria completa.

El método elegido en concreto ha sido COSMIC versión 3.0. A continuación se explica de manera breve porqué ha sido elegido este método.

La principal razón es el importante nivel de auge y aceptación en la industria y la comunidad científica que este método está adquiriendo en los últimos años. Los métodos FSM fueron diseñados para ser aplicados específicamente en la medición de software de gestión. Sin embargo, COSMIC fue diseñado para diferentes dominios de software, como tiempo real o también los sistemas de información, siendo completamente independiente del contexto y del entorno de desarrollo del software. Esta característica hace que la aplicación del método en un entorno tan variante como el de los ERPs sea más sencilla respecto a otros métodos. Además COSMIC hace posible la medición desde diferentes puntos de vista, el usuario final y el desarrollador, lo que lo hace muy flexible ante las diferentes formas de representación de los requisitos funcionales. Esta flexibilidad hace que se pueda emplear en las primeras fases del ciclo de vida de un ERP, empleándose en diagramas que no hay demasiado nivel de detalle.

Todos estos factores han contribuido a que COSMIC haya tenido una rápida aprobación como estándar respecto a otros métodos. También su elevado nivel de adaptabilidad a contextos particulares ha provocado que haya diferentes propuestas basadas en COSMIC (adaptaciones para diagramas de casos de uso, diagramas de secuencia, etc.). Estas propuestas establecen un mapeo entre las primitivas de los diferentes diagramas y

artefactos software y los conceptos relevantes de COSMIC, permitiendo adaptar COSMIC a ese tipo de diagramas y artefactos y poder establecer mediciones de los productos software a partir de ellos. Una importante característica para escoger COSMIC es su facilidad para representar conceptos modernos de la ingeniería del software. Finalmente otra característica para elegir COSMIC es la posibilidad de automatización de COSMIC integrándolo con programas de diseño de software, pudiendo automatizar el método de conteo.

Tercera aproximación hacia la solución para solventar el problema de estimación de coste y esfuerzo en proyectos

Determinar el tamaño (funcionalidad de un ERP) desde un documento de requisitos funcionales aplicando el método COSMIC.

Ilustración 10 : Tercera aproximación hacia una solución en los problemas de estimación en entornos ERPs

3.4 Cuarto paso, eligiendo el documento adecuado para establecer las mediciones

El último paso es determinar que documento representa el nivel adecuado de los requisitos funcionales de un ERP y que esté disponible en la fase inicial del ciclo de vida del producto ERP.

El documento de requisitos de negocio es el nivel adecuado. Como se describe en la sección 2.3.4 de la memoria, “SAP Business Blueprint” describe la funcionalidad de un ERP. Extendiendo esta afirmación al resto de sistemas ERP, el elemento más importante de representación de requisitos funcionales son los escenarios de negocio. Un escenario puede ser modelado usando diferentes procesos de negocio, los cuales en nuestra propuesta serán representados con nivel de detalle equivalente a la vista de un analista. Estos procesos se representarán usando diagramas de proceso de negocio y representan los requisitos funcionales utilizando la metodología Event-driven Process Chain (EPC).

Por lo tanto ahora las palabras “documento de requisitos funcionales”, son remplazadas por “El conjunto de diagramas de Procesos de Negocio de un ERP (Metodología EPC)”.

Solución para solventar el problema de estimación de coste y esfuerzo en proyectos

Determinar el tamaño (funcionalidad de un ERP) utilizando el conjunto de diagramas de Procesos de Negocio de un ERP (Metodología EPC) aplicando el método COSMIC.

Ilustración 11 : Último paso en busca de una solución en los problemas de estimación en entornos ERPs

3.5 Solución final

La idea final para solventar el problema de estimación del tamaño de un producto ERP y sus problemas derivados de estimación de esfuerzo y coste, es usar el método estándar COSMIC en su versión 3.0 (la más reciente) adaptándolo a los diagramas de procesos de negocio que representan la funcionalidad de un ERP. Para determinar el tamaño del producto final ERP habría que analizar y medir todos los escenarios de negocio.

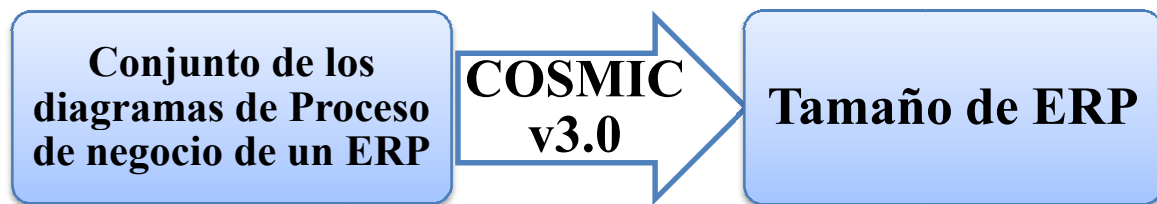


Ilustración 12 : Solución propuesta para estimar el tamaño de un ERP

4. La propuesta eEPC-COSMIC

Este capítulo presenta una introducción al método COSMIC y muestra un resumen del mapeo de los conceptos del método adaptados a la metodología EPC. Este mapeo viene dado por una serie de principios y reglas.

4.1 La iniciativa COSMIC

La versión 3 de COSMIC [8] define un método estandarizado de medición del tamaño funcional de un producto software en diferentes dominios comúnmente referidos como aplicaciones de negocio (MIS) y software en tiempo real incluidos híbridos de ambos. El método es aplicable a lo largo de todo el ciclo de vida de un producto, desde la fase de requisitos hasta la fase de implementación y mantenimiento. Es posible encontrar sus conceptos y directrices en su manual de medición [8].

Al final de los noventa, dada el explosivo crecimiento y diversidad de software, las empresas desarrolladoras, proveedores y clientes se vieron con la necesidad de contar con métodos de estimación cada vez más precisos que los tradicionales métodos de Analisis de Puntos de Función (FPA). La primera generación de métodos para la medición del tamaño del software no siempre era lo que el mercado demandaba, o trabajaban solo para determinados tipos de software. La industria necesitaba urgentemente técnicas de estimación que fueran más adecuadas y más fáciles de usar. Con este propósito surgió en 1998 el grupo COSMIC (the COmmon Software Measurement International Consortium), una iniciativa voluntaria llevada a cabo por un grupo de expertos de estimación de software a nivel empresarial y académico de Asia, Europa y Norte América. El primer paso fue desarrollar, probar, traer al mercado y buscar la aceptación de un nuevo método de medición de software. El primer método con los principios de COSMIC surgido en 1999, fue dejado a un lado, hasta que los intentos realizados en 2000/01 por introducirlo en diferentes compañías internacionales y el mundo académico fueron satisfactorios. El proceso de desarrollar un estándar internacional para COSMIC fue iniciado en 2001, y en diciembre de 2002 fue aprobado, para ser publicado al principio de 2003 como International Standard ISO/IEC 19761 ‘Software Engineering – COSMIC-FFP – A functional size measurement method’ (versión 2.2).

COSMIC continuó trabajando para refinar la definición y explicación del método, pero sin cambiar la base del método de medición, que no ha cambiado desde su primera publicación en 1999. La versión más reciente 3.0 es el último paso en este proceso de refinamiento, y es compatible con el estándar ISO/IEC 19761 y se está a la espera que esta nueva versión sea declarada estándar.

4.2 El método COSMIC

4.2.1 Principios

COSMIC es un método FSM de segunda generación. Esta generación se caracterizó por la extensión del alcance del punto de vista de medición. Es decir, COSMIC, a diferencia de los métodos tradicionales, no sólo mide la funcionalidad desde el punto de vista del usuario final (persona que interactúa con el sistema), sino que, al ampliar el concepto de usuario, la medición del sistema es realizada desde diferentes puntos de vista. De este modo, toda la funcionalidad es cubierta, la cual era muchas veces parcialmente invisible por los métodos tradicionales.

El modelo general de software COSMIC distingue cuatro tipos de movimientos de datos: entrada, salida, lectura y escritura. Un movimiento de dato transfiere uno o más atributos de datos que pertenecen a un único grupo de dato. Así mismo, COSMIC establece como principio de medición que el tamaño funcional del software es directamente proporcional al número de movimientos de datos.

- **Entrada (E):** Una entrada es un movimiento de datos que mueve un grupo de datos que es introducido por un usuario y cruza la frontera del software hacia un proceso funcional que lo requiere.
- **Escritura (W):** Una escritura es un movimiento de datos que mueve un grupo de datos desde un proceso funcional hacia almacenamiento persistente.
- **Lectura (R):** Una lectura es un movimiento de datos que mueve un grupo de datos desde el almacenamiento persistente hacia un proceso funcional que lo requiere.
- **Salida (X):** Una salida es un movimiento de datos que mueve un grupo de datos desde un proceso funcional cruzando la frontera del software hasta un usuario que lo requiere

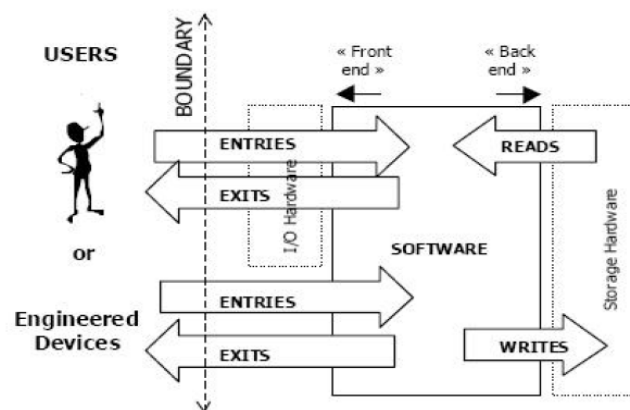


Ilustración 13 : Modelo genérico del flujo de los movimientos de datos desde una perspectiva funcional

4.2.2 El meta-modelo COSMIC

El meta-modelo de COSMIC fue diseñado con el fin de representar de manera más clara los diversos conceptos genéricos que son presentados en su manual de medición, así como también poder identificar las diferentes relaciones existentes entre dichos conceptos. La ilustración situada a continuación muestra el meta-modelo de COSMIC,

representado mediante un diagrama de clases de UML, que fue seleccionado por su sencillez, expresividad y popularidad.

Como se puede observar en este meta-modelo, el objeto de interés a ser medido puede ser identificado desde muchos puntos de vista de medición. El *punto de vista* determina el nivel de detalle con que puede ser enfocado un objeto de interés; por ejemplo el detalle observado bajo el punto de vista de medición de un programador será mayor que la de un usuario final. Tal como se observa en el diagrama, este punto de vista es determinado por el propósito de la medición. El *propósito de la medición* define el porqué de la medición y para qué será utilizada. Este propósito ayuda también a determinar el *alcance de la medición*; es decir, determinar la funcionalidad a ser incluida en una medición particular.

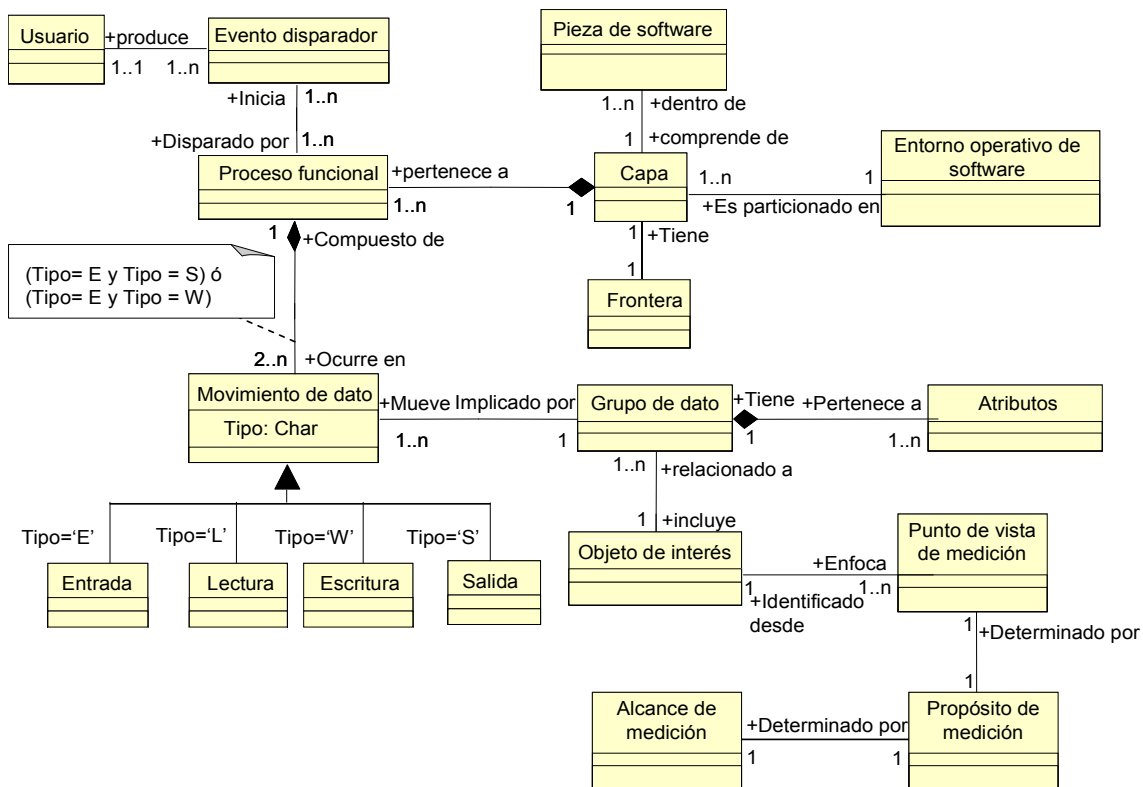


Ilustración 14 : Meta-modelo de COSMIC

El *objeto de interés*, identificado desde un punto de vista de medición, puede ser alguna entidad física u objeto conceptual descrito por un conjunto de atributos que pertenecen a un *grupo de datos*, siendo un *atributo de datos* definido como la unidad de información más pequeña. De este modo, una relación de agregación ha sido utilizada entre los conceptos grupo de datos y atributo de datos. Además, uno o más grupos de dato están directamente relacionados a un solo objeto de interés y están siempre implicados en algún movimiento de dato. Tal como se observa en la Ilustración 10, este *movimiento de dato* ocurre en un proceso funcional y puede ser de cuatro tipos: *entrada*, *lectura*, *escritura* y *salida*. Por ejemplo, un movimiento de dato de tipo entrada mueve un grupo de dato, a través de la frontera, desde un usuario hasta el proceso funcional que requiere dicho grupo de dato. Cada *proceso funcional* está compuesto de un mínimo de dos movimientos de datos: una entrada y una salida o una escritura. Esto es representado por medio de la cardinalidad mínima de dos en el rol "Ocurre en" de la relación entre proceso funcional y movimiento de dato.

El conjunto de procesos funcionales ejecutados en el mismo nivel de abstracción constituye el concepto de capa. Una *capa* es el resultado de la partición funcional del entorno operativo de software. Así mismo, esta capa puede ser dividida en uno o más *piezas de software*. El *entorno operativo de software* es el conjunto de software que opera concurrentemente sobre un sistema específico. En un entorno de software multicapa, cada capa es un usuario de otra capa, ya que una capa utiliza los servicios funcionales proporcionados por otras capas subordinadas. Entre dos capas existe una *frontera*, la cual es definida como una interfaz conceptual entre el software bajo estudio y sus usuarios. Un *usuario* es definido como la persona o cosa que comunica o interactúa con el software. Retomando el concepto de proceso funcional, se observa en la **¡Error! No se encuentra el origen de la referencia.** 9 que este proceso es disparado por uno más eventos disparadores. Consecuentemente, un *evento disparador* es un evento que inicia uno o más procesos funcionales, que son provocados directa o indirectamente por un usuario.

4.2.3 Fases del método COSMIC

El proceso de medición de COSMIC consiste en tres fases como se muestra en la ilustración de abajo.

- ◆ Fase de estrategia (Measurement Strategy), llevada a cabo antes de empezar la medición, en el cual el modelo de contexto del software es aplicado al software que va a ser medido. Es decir se determinan como los principios de COSMIC son referidos en el contexto del software que va a ser medido, así como se fijan los objetivos de la medición. En concreto el propósito de la medición, el alcance, los componentes que se van a medir, las diferentes capas del software y se establece el detalle con el que se va a realizar la medición.
- ◆ Fase de mapeo (Mapping Phase), en la cual los conceptos genéricos del modelo de software son aplicados a los documentos y artefactos que contienen los requisitos funcionales de usuario. Se identifican en el diagrama eEPC (que representa un requisito funcional dentro de un ERP) los procesos funcionales, los grupos de datos, los usuarios y la frontera del software.
- ◆ Fase de medición (Measurement Phase) en la cual el tamaño del producto es obtenido, aplicando una serie de reglas para identificar los movimientos de datos y establecer el tamaño final del producto o componente a ser medido.

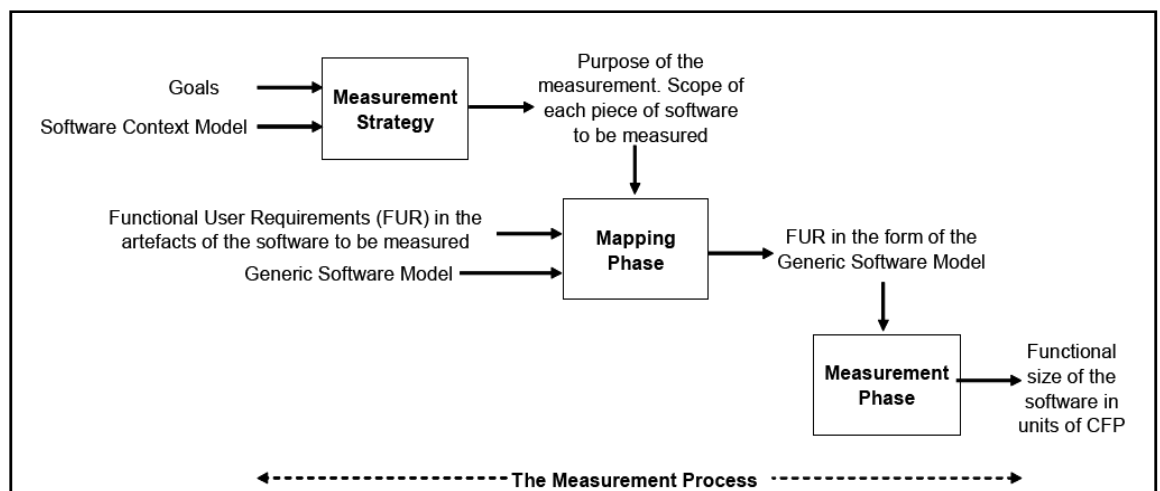


Ilustración 15 : Fases del método COSMIC

4.3 Método eEPC-COSMIC

Finalmente en este capítulo presentamos la adaptación de COSMIC a la metodología EPC. La propuesta recibe su parte del nombre correspondiente a eEPC, porque es el nombre de los diagramas en su versión extendida (extended EPC) que representan los procesos de negocio de un ERP y que consideramos como requisitos funcionales de usuario.

4.3.1 Fase de estrategia

El objetivo de este trabajo es: *definir un procedimiento de medición para el propósito de medir una especificación de requisitos con respecto a su tamaño funcional desde el punto de vista del analista en el contexto de la metodología EPC que permite representar procesos de negocio en el ámbito de un desarrollo ERP.*

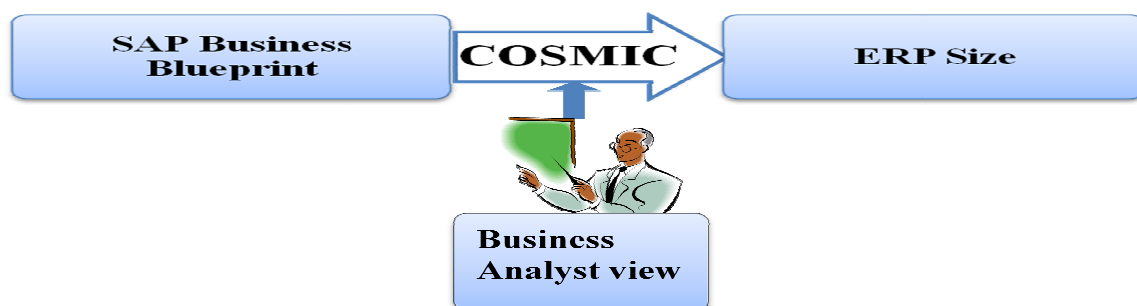


Ilustración 16 : Propósito y alcance de medición de un proyecto SAP

En la siguiente tabla se muestran los principios de esta fase. Indicar que la tabla está organizada por Nomenclatura del concepto, su nombre, si el concepto o regla puede automatizarse (en esta fase solo son principios no reglas de aplicación por lo que este campo no necesita ser rellenado), descripción del principio o de la regla y la sección en que se puede encontrar en la memoria final.

Reglas y principio del método eEPC-COSMIC				
Número	Concepto	Automático	Descripción	Sección en la memoria final
P-01	Propósito	-	Desde el punto de vista del analista de procesos de negocio, el propósito es determinar el tamaño de la funcionalidad de un nuevo producto ERP como previo paso para establecer el esfuerzo necesario para su desarrollo.	Sección 6.2.1 (Defining the purpose of the measurement)
P-02	Alcance	-	Funcionalidad de un paquete o componente de un ERP que son representados por un conjunto completo de escenarios de negocio.	Sección 6.2.2.1 (Defining the scope of the measurement)
P-03	Capa	-	No hay partición funcional; solo una capa va a ser identificada.	Sección 6.2.2.2 (Defining the Layers)
P-04	Componente	-	Un módulo ERP o cualquier pieza de software de un ERP puede ser	Sección 6.2.2.3 (Defining the Peer

Reglas y principio del método eEPC-COSMIC				
Número	Concepto	Automático	Descripción	Sección en la memoria final
			medida si el conjunto de modelos EPC que lo representa está complete y toda su funcionalidad está presente.	Component)
P-05	Nivel de granularidad	-	Alto nivel de abstracción, representado por la técnica de modelado Event-driven process chain. El conjunto completo de procesos de negocio representado por el diagrama extendido EPC es necesario así como el modelo de datos.	Sección 6.2.3 (Defining the Level of granularity)

Ilustración 17 : Principios del método eEPC-COSMIC (fase de estrategia)

4.3.2 Fase de mapeado

En esta fase como se indicó anteriormente se identifican los procesos funcionales representados por los diagramas EPC y los grupos de datos, así como los usuarios y la frontera del software.

En esta fase se las siguientes reglas:

Reglas y principio del método eEPC-COSMIC				
Nombre	Concepto	Automático	Descripción	Sección en la memoria final
MaR-01	Usuario funcional (Functional User)	Si	a) Aceptar como usuario funcional cada sistema externo que interactúa con el sistema a ser medido. b) Aceptar cada unidad organizacional de un diagrama eEPC como un usuario del sistema.	Sección 6.3.1.1 (Defining the Functional Users)
MaR-02	Frontera (Boundary)	Si	El borde entre los usuarios funcionales y el resto del diagrama de proceso de negocio exceptuando los caminos de proceso.	Sección 6.3.1.1 (Defining the Boundary)
MaR-03	(Proceso funcional) Functional process	Si	a) Aceptar cada proceso de negocio representado por un diagrama eEPC como un proceso funcional. b) Si un diagrama eEPC tiene caminos de proceso considerar cada uno de ellos como un proceso funcional diferente. c) Si un escenario de negocio se divide en más procesos de	Sección 6.3.2 (Identifying the functional process)

Reglas y principio del método eEPC-COSMIC				
Nombre	Concepto	Automático	Descripción	Sección en la memoria final
			negocio, cada uno de ellos será considerado un proceso funcional.	
MaR-04	Grupo de dato (Data group)	Si	Aceptar cada objeto de información que aparece en el modelo de datos como un grupo de datos.	Sección 6.3.3 (Identifying data groups)

Ilustración 18 : Reglas del método eEPC-COSMIC (fase de mapeado)

A continuación se presenta un proceso funcional (tramitado de una recepción de productos), representado en un diagrama eEPC y donde se mapean los elementos de esta fase.

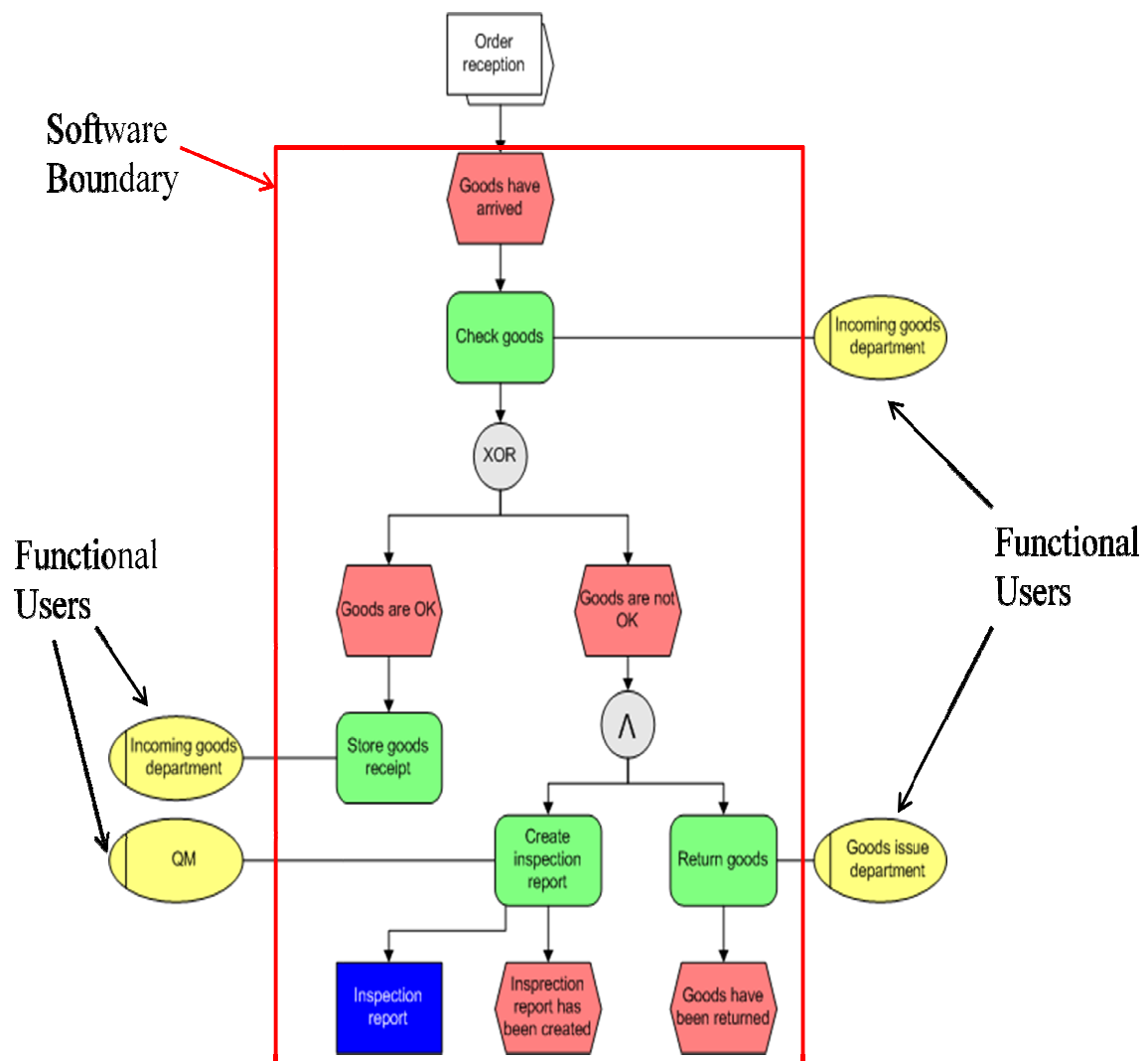


Ilustración 19 : Identificación de usuarios y frontera en un proceso funcional representado por un diagrama eEPC

4.3.3 Fase de medición o conteo

Es en esta fase cuando se lleva a cabo la medición de un producto o componente ERP. Para ello definen una serie de reglas que permiten identificar los movimientos de datos en aquellos diagramas eEPC que se consideren oportuno.

El conjunto de reglas en esta fase se dividen en dos, aquellas para identificar los movimientos de datos y otro para la función de medición.

En este caso las reglas en su gran mayoría se pueden automatizar, en aquellos casos en que su automatización sea semi-automática es decir se necesite la presencia del medidor se establecerá que la regla es “semi”.

Reglas y principio del método eEPC-COSMIC				
Nombre	Concepto	Automático	Descripción	Sección en la memoria final
MeR-01	Entrada	Si	<i>Para cada evento de entrada que no procede de ninguna función, aceptar como candidato a ENTRADA su objeto de información si previamente ha sido identificado como grupo de dato.</i>	Sección 6.4.2.1. i) (Identifying the ENTRY data movements I)
MeR-02	Entrada	Semi	<i>Aceptar como candidato a ENTRADA cada evento de entrada que tiene un objeto de información con un estado que muestra que el elemento ha entrado al sistema como por ejemplo “ha llegado”, “es alcanzado”, etc.</i>	Sección 6.4.2.1. ii) (Identifying the ENTRY data movements II)
MeR-03	Salida	Si	<i>Para cada evento de salida que no sea usado como entrada en ninguna función del resto del diagrama, aceptar como un candidato de Salida su objeto de información presente si se ha identificado previamente como grupo de dato.</i>	Sección 6.4.2.2. i) (Identifying the EXIT data movements I)
MeR-04	Salida	Semi	<i>Aceptar como candidato de SALIDA cada evento de salida que tiene un objeto de información cuyo estado muestra que el elemento ha salido del sistema, como por ejemplo “es enviado”, es entregado”, etc.</i>	Sección 6.4.2.2. ii) (Identifying the EXIT data movements II)
MeR-05	Lectura	Si	<i>Aceptar cada objeto de información que es leído por una función como un movimiento de LECTURA</i>	Sección 6.4.2.3. i) (Identifying the READ data movements I)
MeR-06	Lectura	Si	<i>Aceptar como LECTURA cada grupo de datos diferente del conjunto de eventos de entrada que participan en una precondition.</i>	Sección 6.4.2.3. ii) (Identifying the READ data movements II)
MeR-07	Escritura	Si	<i>Aceptar cada objeto de información que es escrito por una función como</i>	Sección 6.4.2.4 (Identifying the

Reglas y principio del método eEPC-COSMIC				
Nombre	Concepto	Automático	Descripción	Sección en la memoria final
			<i>un movimiento de ESCRITURA.</i>	WRITE data movements)
MeR-08	Duplicados	-	<i>Si el mismo tipo de movimiento de datos afecta más de una vez al mismo grupo de datos en el mismo proceso funcional, solo ICFP tiene que ser identificado para ese movimiento de datos.</i>	Sección 6.4.3.1 (Identifying the duplicate data movements)
MeR-09	Duplicados	-	<i>Si un escenario de negocio tiene diferentes caminos, el diagrama de negocio que representa a dichos caminos, tiene que ser contado una vez, independientemente de que aparezca diferentes veces en la misma capa.</i>	Sección 6.4.3.2 (Duplicate occurrences for the process paths)
MeFR-01	-	Si	<i>El tamaño funcional de un proceso de negocio representado por un diagrama eEPC es igual a la suma de todos los movimientos de datos identificados.</i>	Sección 6.4.4.1.i) Aggregation function (Counting in a functional process)
MeFR-02	-	Si	<i>El tamaño funcional de un escenario base de negocios es igual a la suma de los procesos de negocio que lo componen.</i>	Sección 6.4.4.1.ii) Aggregation function (Counting in a business scenario)
MeFR-03	-	Si	<i>El tamaño funcional del un proceso de negocio principal extendido por otro proceso secundario, es igual a la suma del tamaño de estos sub-procesos de negocio además del tamaño del principal proceso.</i>	Sección 6.4.4.1.iii) Aggregation function (Counting in a main business process)
MeFR-04	-	Si	<i>El tamaño funcional de una capa de software es igual a la suma de los tamaños de todos sus Escenarios de Negocio.</i>	Sección 6.4.4.2 Aggregation function in the software layer

Ilustración 20 : Reglas del método eEPC-COSMIC (fase de medición)

5. Bibliografía

- [1] Ayers, James B. *Making Supply Chain Management Work (Best practices series)*. Auerbach.
- [2] Indihar-Stemberger, Mojca, Vesna Bosilj-Vuksic, y Ales Popovic. «Simulation and information systems modelling: a framework for business process change.» *Proceedings European Simulation Symposium XV* (2003).
- [3] Curran, T., y A. Ladd. *SAP SAP R/3 Business Blueprint: Understanding Enterprise Supply Chain Management*. Prentice, May1999.
- [4] Gaiser, T. «Conducting online focus groups: A methodological discusión.» *Social Science Computer Review XV*, nº 2 (1997): 135-144.
- [5] The Common Software Measurement International Consortium. *The COSMIC Functional Size Measurement Method (ISO 19761)*. Version 3.0. September 2007.
- [6] Condori Fernández, Olinda Nelly. *PhD Thesis. A measuring procedure of functional measurement for the Requirements Specifications*. Valencian University of Technology, Spain, May 2007.
- [7] Moody, D.L. *Dealing with Complexity: A Practical Method for Representing Large Entity Relationship Models*. Editado por Department of Information Systems, University of Melbourne PhD. Thesis. Australia, 2001.
- [8] COSMIC. *COSMIC Implementation Guide to ISO 19761 version 3.0*. 2007. <http://www.lrgl.uqam.ca/cosmic-ffp/>.