

Proceso Software y Gestión de Conocimiento

Aurora Vizcaíno

Departamento de Tecnologías y Sistemas de
Información

Escuela Superior de Informática
Universidad de Castilla-La Mancha

Necesidad de KM en la mejora de Procesos

- ◆ En una Ingeniería es fundamental la idea de **Proceso**
 - El cómo hacer.
 - Existe una correlación directa entre la calidad del proceso y la calidad del producto obtenido.

Necesidad de KM en la mejora de Procesos

- ◆ Ingeniería del Software - Problemática actual:

El desarrollo y mantenimiento de software es un **trabajo altamente complejo**.

Contar con información y conocimiento que describan las actividades y tareas que deben realizarse de forma sistemática facilitaría el trabajo de los ingenieros y el software resultante tendría mayor calidad



Necesidad de KM en la mejora de Procesos



- Los proyectos software son **difíciles de gestionar.**

Reutilizar la experiencia obtenida del desarrollo de previos proyectos ayuda a gestionar nuevos proyectos.




Necesidad de KM en la mejora de Procesos

- ◆ Los sistemas de Gestión del Conocimiento pueden realizar de forma automática procesos enteros y además pueden aprender del usuario
- ◆ La automatización de actividades o/y procesos conlleva importantes beneficios para una organización:
 - ◆ Reducción de Costes
 - ◆ Mejora en la gestión y mayor calidad en el producto final



Metodología para la mejora de procesos usando gestión del conocimiento



Metodología propuesta por Oscar Mario Rodríguez en su tesis doctoral



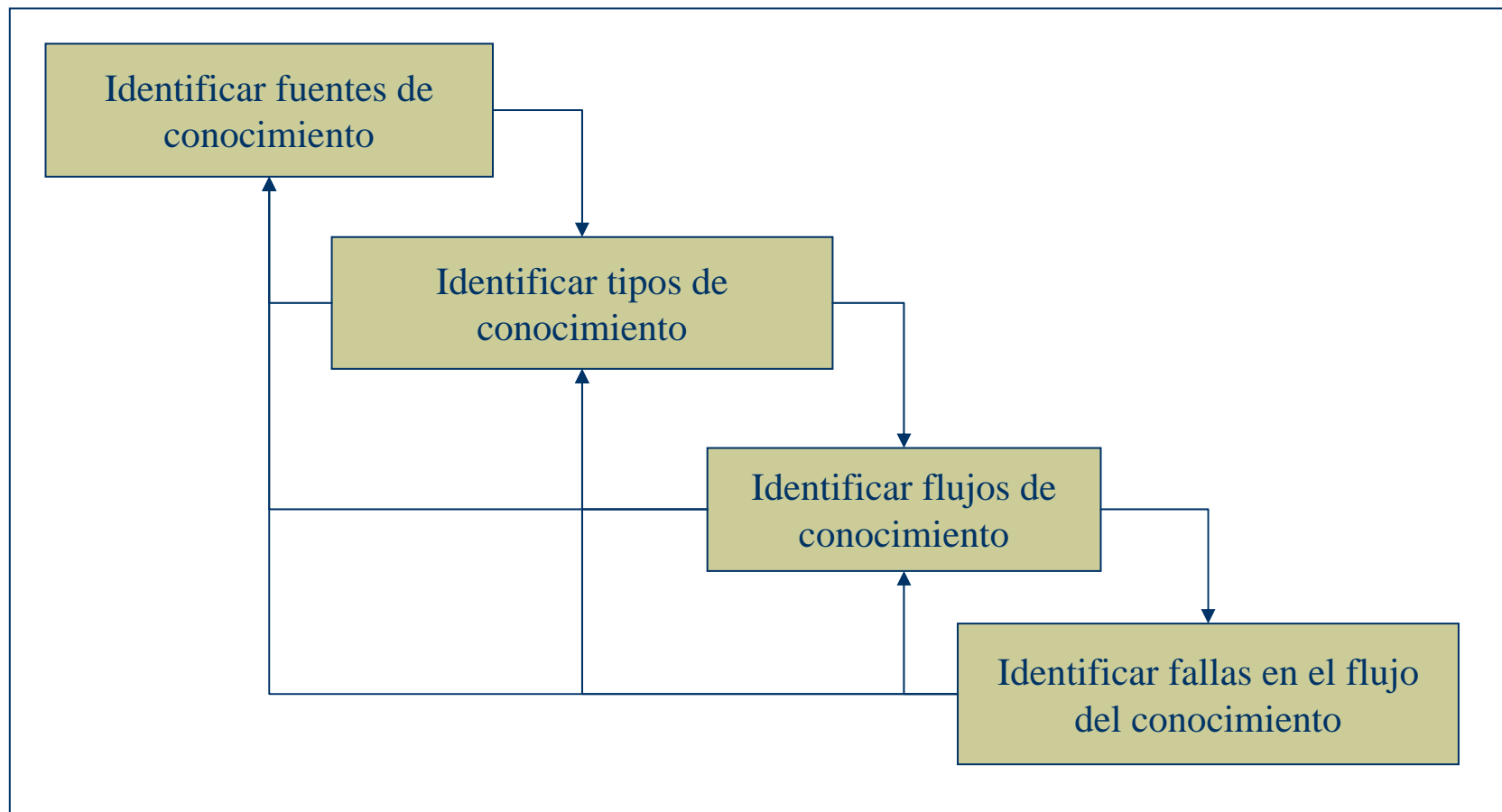
Metodología para la Identificación de Flujos de Conocimiento



Motivación

- ◆ Proporcionar herramientas de gestión del conocimiento
- ◆ ¿Qué tipo de problemas dentro del proceso software pueden ser resueltos o disminuidos por medio de la gestión del conocimiento?
 - ¿Qué tipo de conocimiento puede ser útil de almacenar?
 - ¿En qué fuentes se localiza éste conocimiento?
 - ¿Cómo fluye el conocimiento dentro una organización?

Etapas de la Metodología





Identificar fuentes de conocimiento



- ◆ Identificar los documentos que son utilizados o generados dentro de una organización, las personas involucradas en el proceso, etc.
 - Taxonomía de fuentes de conocimiento
 - Esquema para la definición de fuentes de conocimiento
 - Ontología de fuentes de conocimiento

Taxonomía de fuentes de conocimiento

◆ Documentación

- De sistema
- Técnica
- De usuario
- Organizacional
- Del proceso de mantenimiento

◆ Sistema o producto

- Sistema ejecutable
- Bases de datos
- Código fuente

◆ Personas

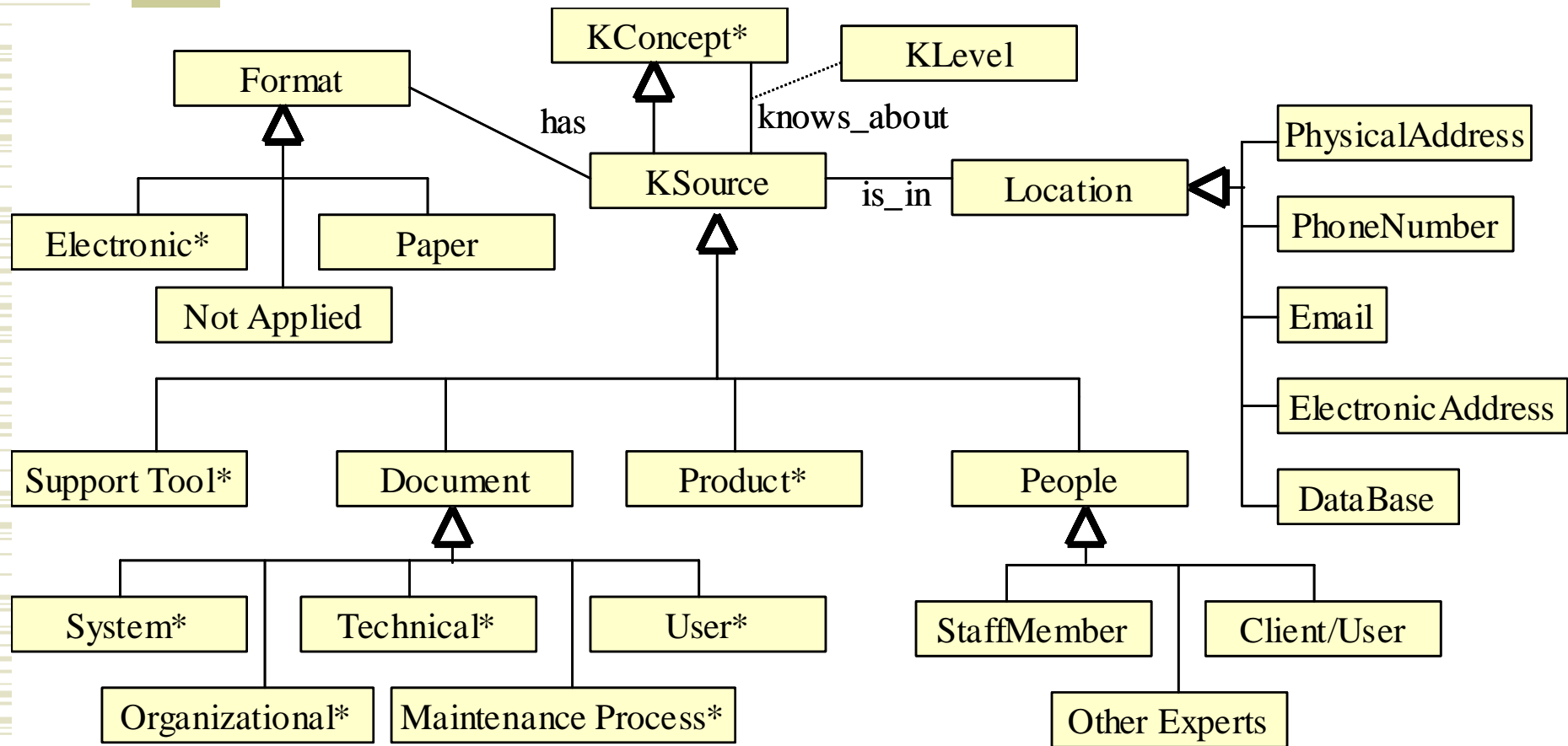
- Miembros del personal
- Usuarios o clientes
- Otros expertos

◆ Herramientas de apoyo

Esquema para definición de fuentes de conocimiento

Nombre:		
Clase:		
Super Clase:		
Formato primario:		
Formato Secundario:		
Descripción:		
Localización:	Tipo	Descripción
Conoce a cerca de:	Concepto	Nivel

Ontología de fuentes de conocimiento



Nombre Elemento de la ontología

Nombre* Sub-ontología



Identificar tipos de conocimiento



- ◆ Identificar el conocimiento y conceptos involucrados dentro del proceso de mantenimiento, y sus relaciones.
- ◆ Definir o elegir una ontología.
 - Taxonomía de clases de conocimiento
 - Esquema para la definición de conocimientos
 - Ontología de tópicos de conocimiento



Taxonomía de clases de conocimiento...



- ◆ Conocimiento sobre computación
 - Generales:
 - Sistemas operativos; redes de computadoras; bases de datos; aplicaciones diversas.
 - Ingeniería del software:
 - Programación; lenguajes de programación; paradigmas y metodologías de programación; lenguajes y herramientas de modelado; ambientes de desarrollo; herramientas de apoyo.



...Taxonomía de clases de conocimiento...



- ◆ Conocimiento dependiente de la aplicación
 - Requerimientos; arquitectura o estructura interna del sistema; artefactos que componen la aplicación; historia y evolución de la aplicación; dominio de la aplicación; conocimiento sobre los usuarios; lenguajes y herramientas con las que la aplicación ha sido desarrollada y mantenida.

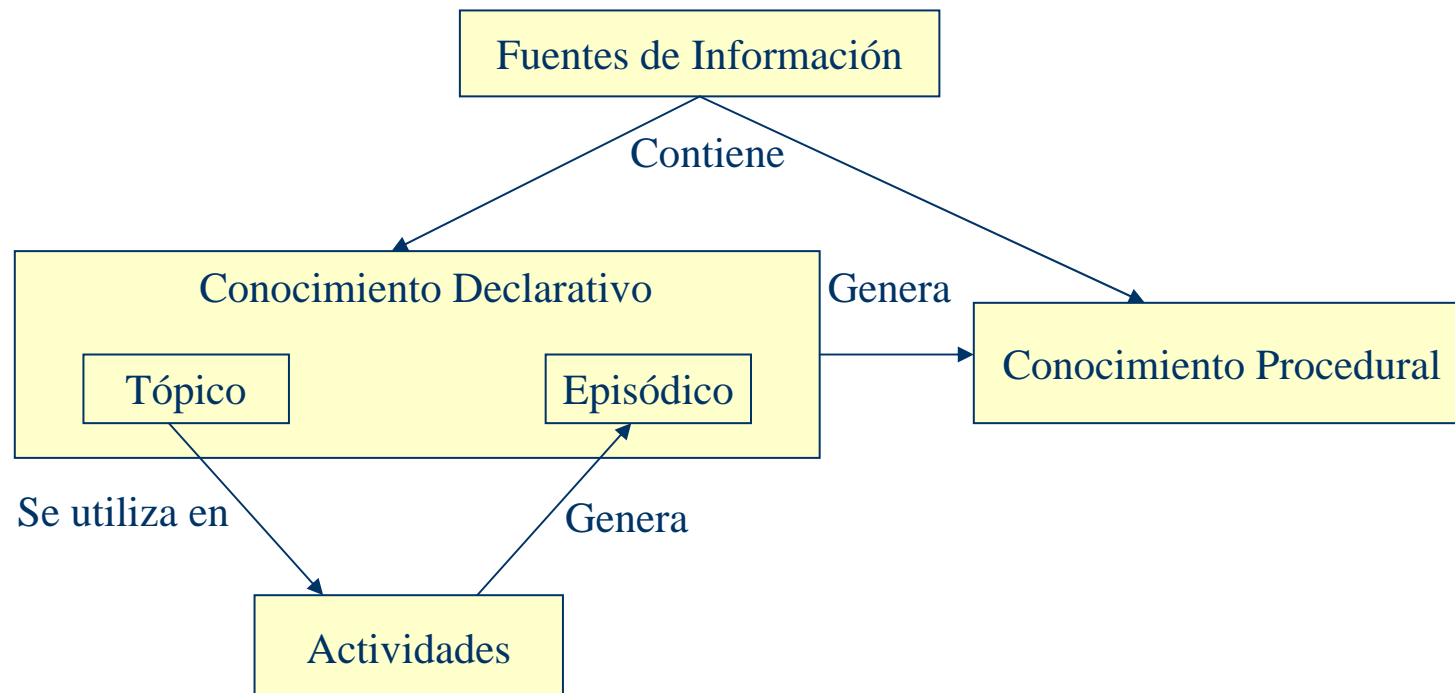


...Taxonomía de clases de conocimiento

- ◆ Conocimiento sobre la organización
- ◆ Procesos y actividades
- ◆ Conocimientos generales

Esquema para la definición de conocimientos...

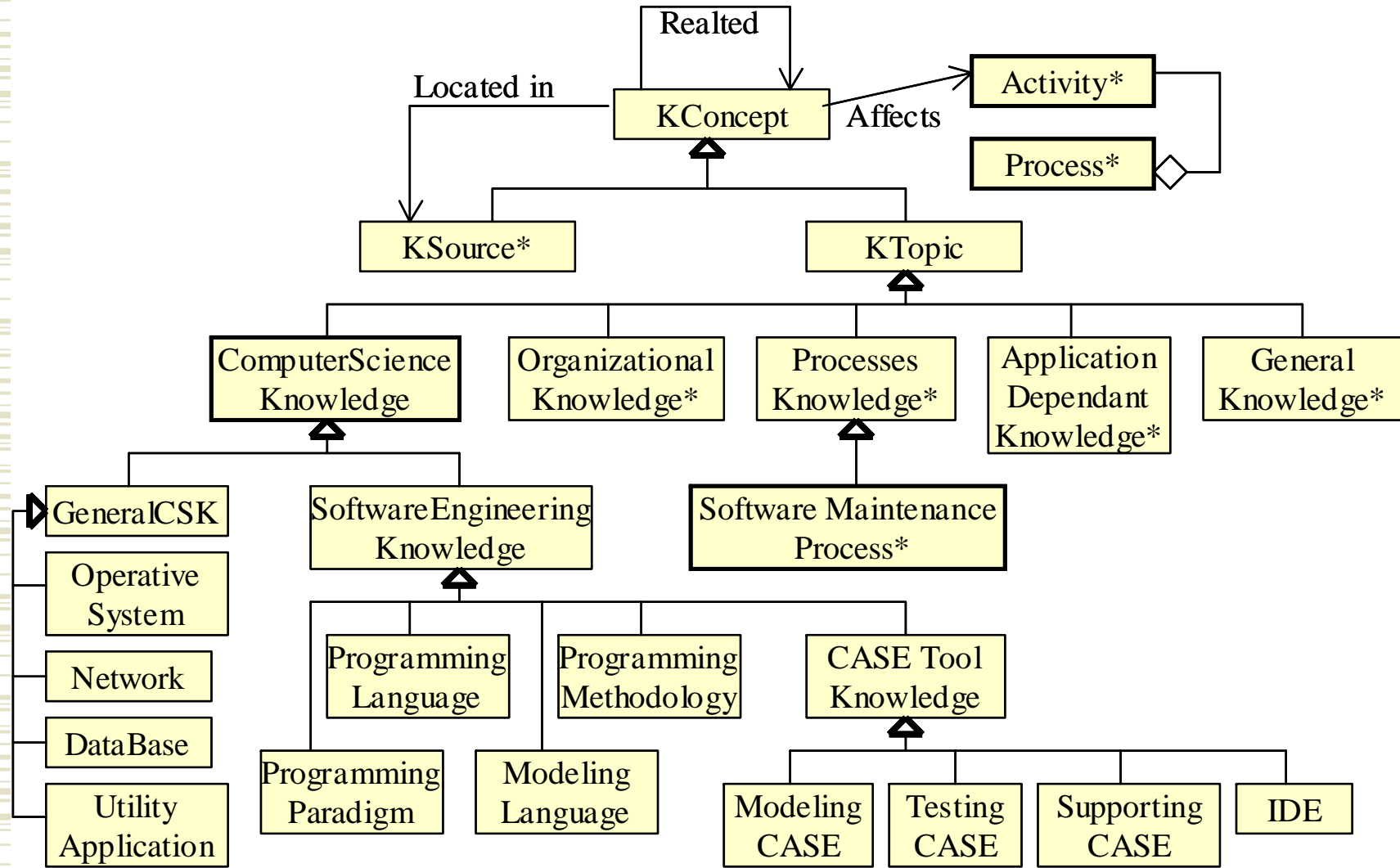
- ◆ Modelo conceptual para la identificación de conocimiento



...Esquema para la definición de conocimientos

Nombre:		
Descripción:		
Tópico:		
Episódico:		
Procedural:		
Afecta en:	Proceso/Actividad	Descripción
Localizado en:	Nombre de fuente	Grado de conocimiento
Conocimiento relacionado	Nombre del concepto	Tipo de relación

Ontología de tópicos de



Nombre Elemento de la ontología

Nombre* Sub-ontología

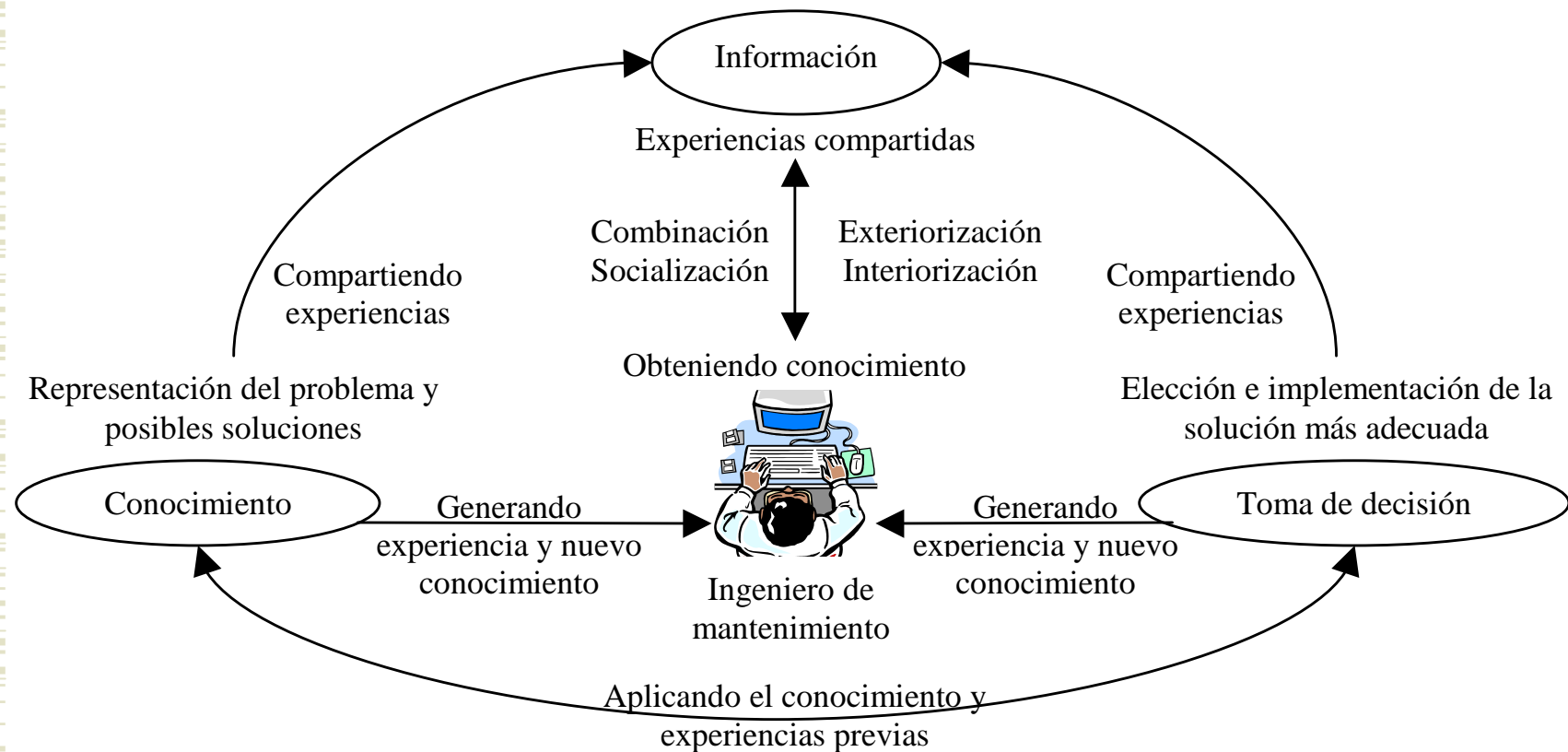


Identificar flujos de conocimiento

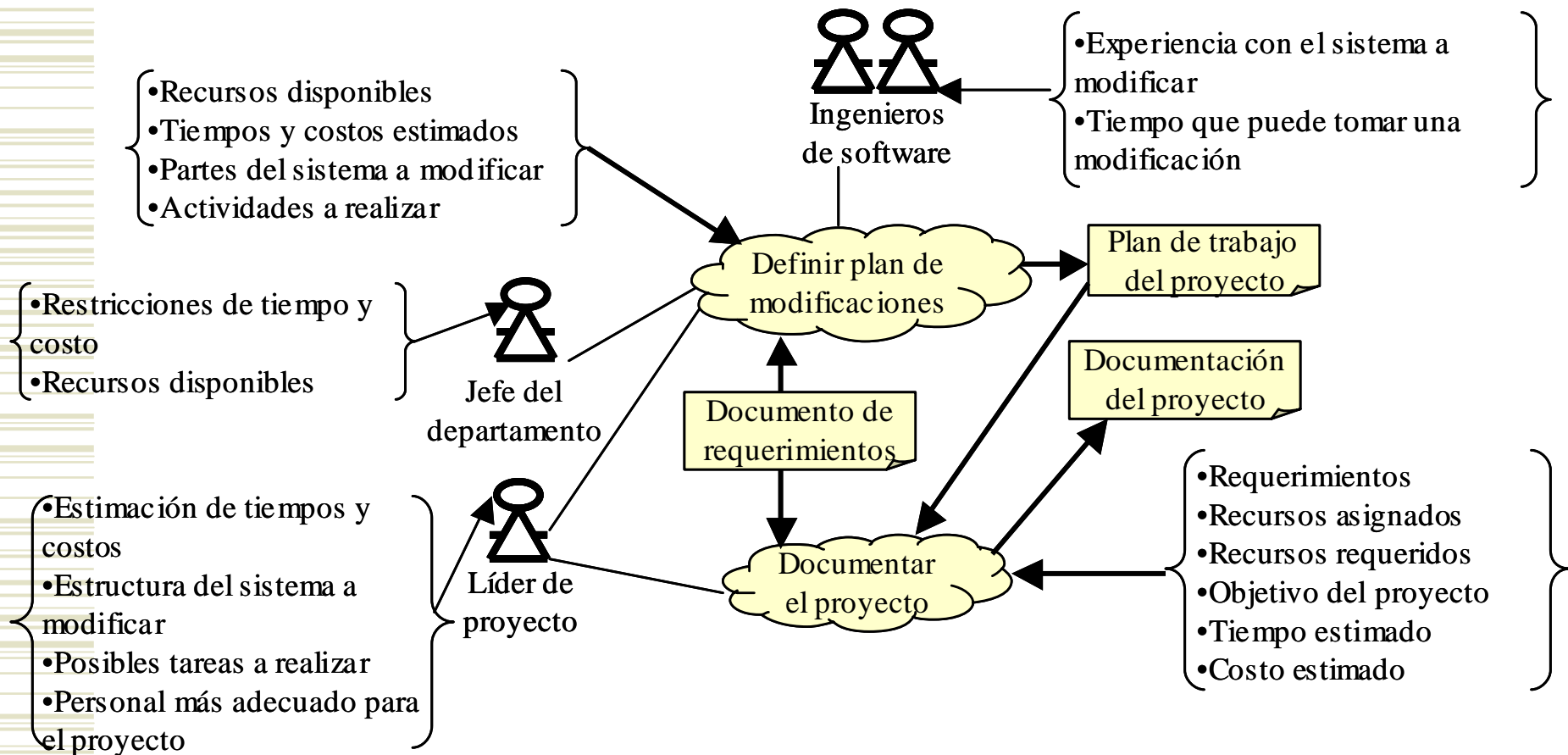


- ◆ Identificar la manera en que los conceptos y fuentes interactúan dentro de los distintos procesos y actividades.
 - Modelo genérico del flujo de conocimiento
 - Elementos para el modelado gráfico de procesos
 - Esquema de descripción de procesos
 - Esquema de toma de decisiones

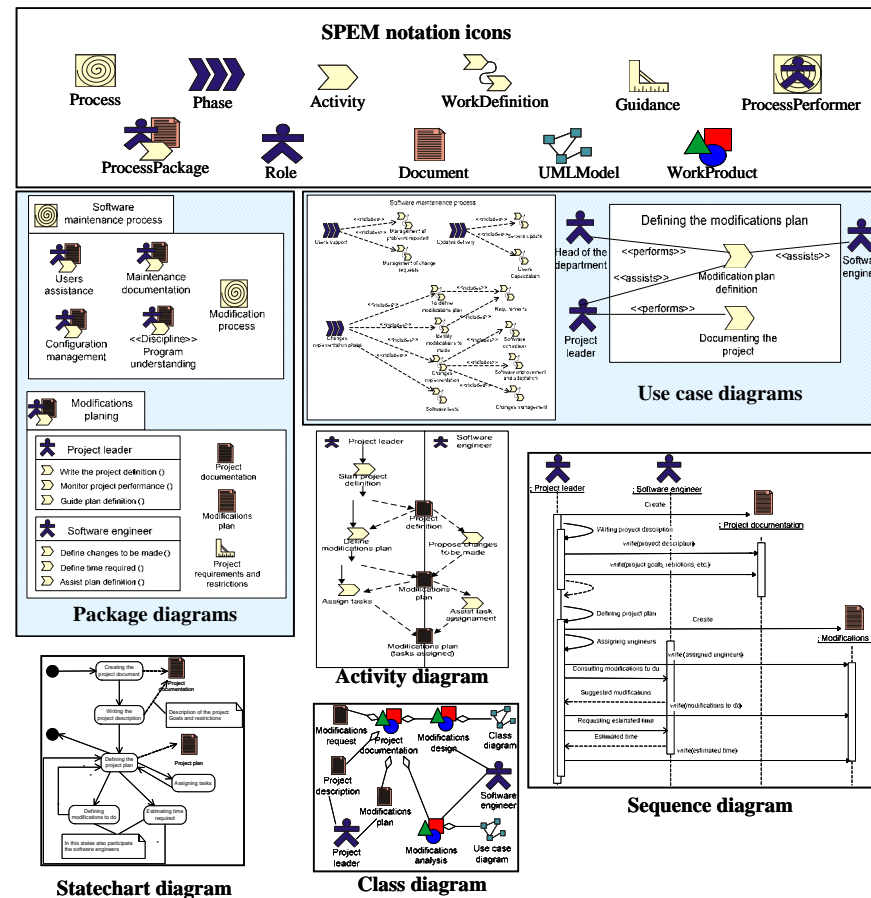
Modelo genérico del flujo del conocimiento



Modelado gráfico de procesos



Modelado gráfico de procesos



Modelado gráfico de procesos

Extensión de SPEM



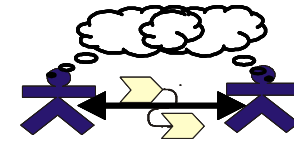
KTopic



KSource



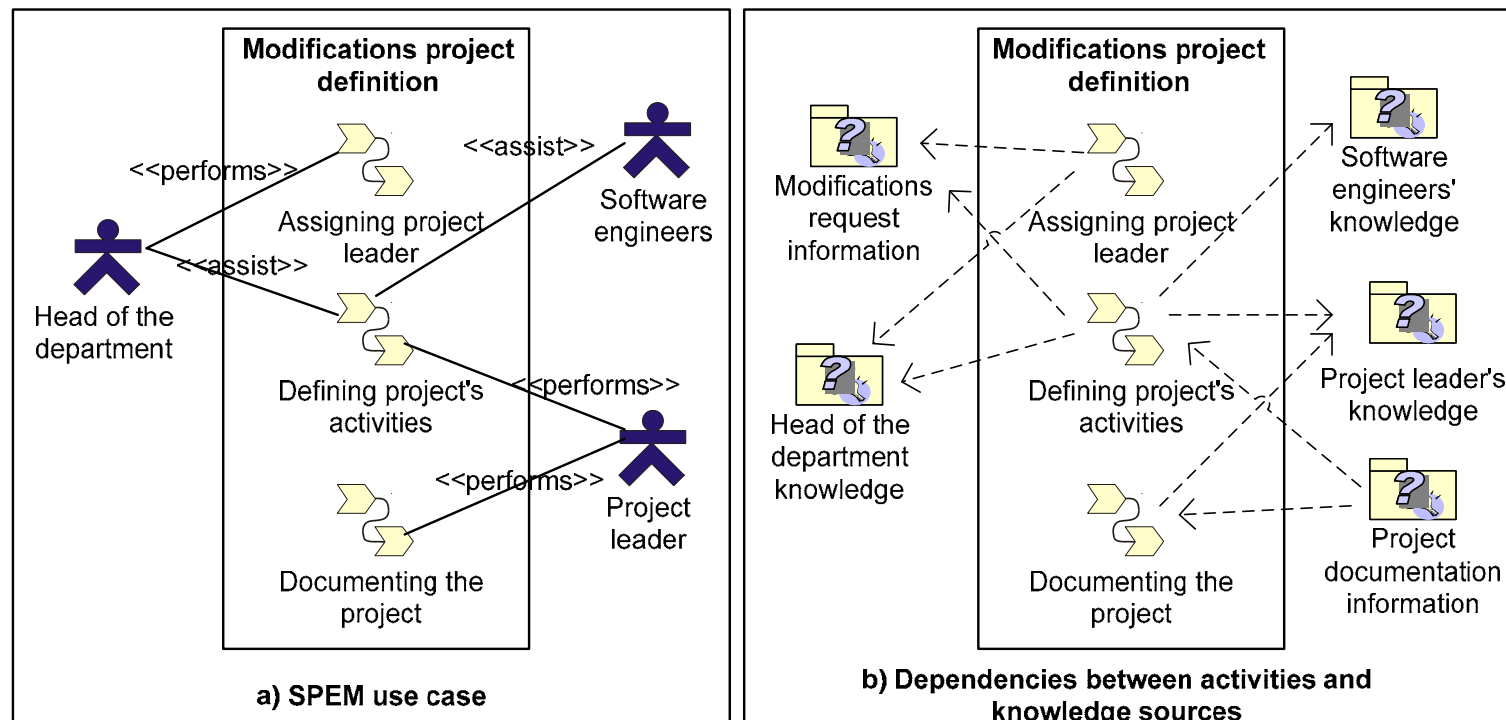
GroupedKnowledge



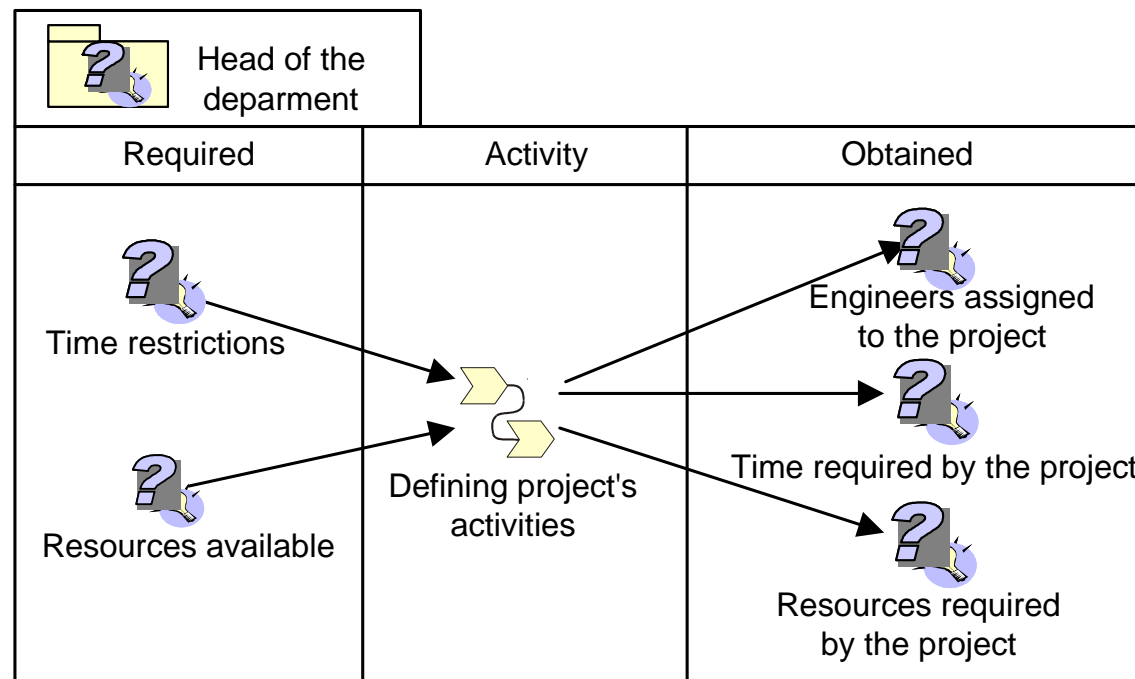
KnowledgeTransfer

Modelado gráfico de procesos

Ejemplo que muestra las actividades y las fuentes de conocimiento

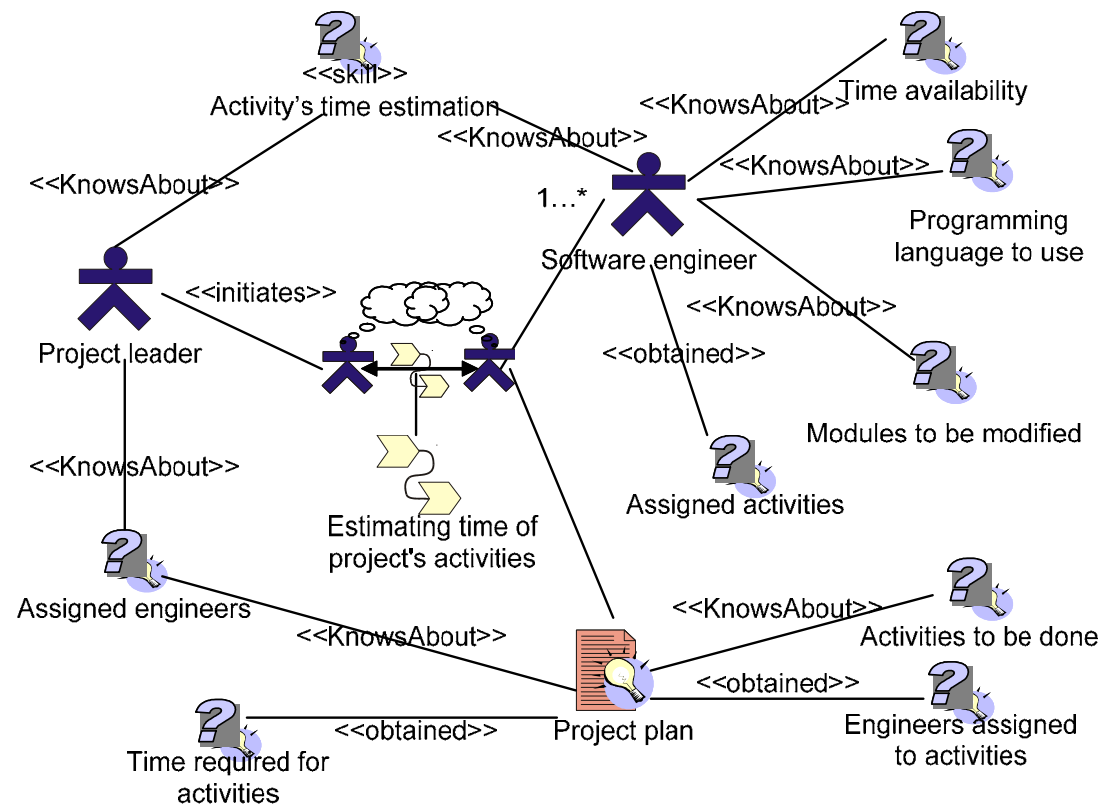


Modelado gráfico de procesos



Ejemplo de modelo de fuente de conocimiento y su relación con actividades

Modelado gráfico de procesos



Ejemplo de diagrama de transferencia de Información

Esquema de descripción de procesos

Nombre del proceso	Definición del plan del proyecto
Descripción	<p>Una vez que se ha recibido una solicitud de modificaciones, el jefe del departamento se reúne con quien será el líder del proyecto de modificaciones, para definir el plan del proyecto. Esta actividad consiste en determinar el tiempo estimado y los recursos que requerirá el proyecto, con lo que se elabora un plan de trabajo. En caso de ser necesario, el líder del proyecto consulta con los ingenieros que participarán en el proyecto para determinar cuáles serán las posibles modificaciones y cuanto tiempo podrían llevar. Una vez definido el plan de trabajo del proyecto, el líder del proyecto desarrolla un documento donde se define este plan, así como los recursos que serán requeridos por el proyecto, el objetivo y alcance del mismo, entre otra información relevante para la organización.</p>
Principales actividades	Definición del plan de trabajo del proyecto Documentación del plan del proyecto
Roles involucrados	Jefe del departamento Líder del proyecto Ingenieros de mantenimiento
Fuentes involucradas	Solicitud de modificaciones Plan de trabajo del proyecto Documentación del proyecto

Esquema de toma de decisiones

Rol	Líder del proyecto	
Actividad	Definir plan de trabajo del proyecto	
Decisiones	Definir recursos requeridos	
	Definir principales tareas a realizar	
	Asignar tareas a los participantes del proyecto	
	Definir el tiempo requerido por el proyecto	
Conocimientos	Experiencia con proyectos previos	
	Requerimientos y restricciones del proyecto	
	Habilidades y experiencia de los posibles participantes en el proyecto	
Fuentes de información		
Nombre	Información	Consultado en
Jefe del departamento	Recursos disponibles; Restricciones de tiempo y costo	Teléfono, Dirección Física; Email
Ingenieros de mantenimiento	Experiencia con el sistema a modificar; tiempo que les puede llevar hacer una modificación; disponibilidad de tiempo.	Teléfono, Dirección Física; Email
Documentación de proyectos previos	Recursos requeridos en proyectos similares	Archivo de documentos; Bitácora de modificaciones



Identificar fallos en el flujo del conocimiento



- ◆ Identificar los problemas específicos relacionados con la utilización del conocimiento dentro de los procesos y actividades que realiza el grupo de mantenimiento.
 - Identificación de escenarios
 - Definición de alternativas de solución



Identificar fallos en el flujo del conocimiento



- Hubs: En un punto llegan muchos flujos (congestión, cuello de botella)
- Black holes: Proceso que no origina flujos (nunca consultado)
- Springs: Se originan muchos flujos pero no entra ninguno (no recibe feedback)
- Missing links: No están o no funcionan

Escenarios

- ◆ **Escenario 1: Localización de expertos.** Rosario, un ingeniero de mantenimiento (IM), debe implementar ciertos cálculos dentro del sistema de finanzas. Ya que su conocimiento en el área no es suficiente, las modificaciones le han tomado alrededor de una semana más del tiempo programado. Al cabo de esta semana, Susana, la jefa del departamento (JD), en una revisión del avance de los proyectos, detecta el retraso. Cita a Rosario para preguntarle la razón de dicho retraso. Cuando Rosario le comenta a Susana cuál es el problema, Susana se da cuenta de que es algo en lo que ella tiene experiencia, por lo que puede ayudar a Rosario a solucionar el problema ese mismo día.



Discusión

- ◆ **Discusión.** Al analizar este caso de estudio, podemos ver que si Rosario hubiera tenido la manera de saber desde un principio, que Susana tenía la experiencia para ayudarle a resolver el problema, posiblemente el retraso nunca se hubiera dado.



Conclusiones



- ◆ Se identificaron cinco tipos de escenarios donde se muestra cómo la gestión del conocimiento puede ayudar a resolver problemas durante el mantenimiento de software.
- ◆ Se obtuvieron requerimientos para un sistema de gestión del conocimiento.
- ◆ Se desarrolló una arquitectura multi-agentes para la gestión del conocimiento
- ◆ Se implementó un prototipo de un sistema de gestión del conocimiento basado en la arquitectura propuesta.

Ejemplo de Herramienta para la Gestión de Procesos

- ◆ GENESIS
- ◆ <http://www.genesis-ist.org/english/default.htm>

**Generalized Environment for procESs
management in cooperative Software engineering**

Es una plataforma de código abierto que da soporte a los
Procesos de ingeniería del software en entornos distribuidos



Ejemplo de Herramienta para la Gestión de Procesos

La vista global del proyecto es modelada en el sitio coordinador (por el líder técnico del proyecto distribuido) mientras que los subprocessos pueden ser modelados y ejecutados de forma autónoma en diferentes sitios organizativos

Ejemplo de Herramienta para la Gestión de Procesos

GENESIS incluye los siguientes componentes:

- ◆ Un sistema de gestión de flujos de trabajo para modelar procesos
- ◆ Un sistema de gestión de artefactos para almacenar y recuperar los artefactos producidos en un proceso
- ◆ Un sistema de gestión de recursos, para asignar recursos, en particular humanos, a un proyecto
- ◆ Un motor de eventos y un sistema de comunicación para recoger y enviar eventos ocurridos durante la gestión del procesos
- ◆ Un motor de métricas para representar informes sobre el estado de proyecto

Referencias

- ◆ Briand, L. C., V. R. Basili, Y.-M. Kim y D. R. Squier (1994). A Change Analysis Process to Characterize Software Maintenance Projects. International Conference on Software Maintenance (ICSM'1994), Victoria, BC, Canada, IEEE Computer Society Press.
- ◆ Curtis, B., H. Krasner y N. Iscoe (1988). "A Field Study of the Software Design Process for Large Systems." Communications of the ACM 31(11): 1268-1287.
- ◆ Deridder, D. (2002). A Concept-Oriented Approach to Support Software Maintenance and Reuse Activities. Joint Conference on Knowledge-Based software Engineering, Maribor, Eslovenia.
- ◆ Dias, M. G. B., N. Anquetil y K. M. d. Oliveira (2003). "Organizing the Knowledge Used in Software Maintenance." Journal of Universal Computer Science 9(7): 641-658.
- ◆ ISO/IEC (1999). ISO/IEC FDIS 14764:1999, Software Engineering - Software Maintenance, Secretariat : Standard Council of Canada.
- ◆ Kitchenham, B. A., G. H. Travassos, A. v. Mayrhauser, F. Niessink, N. F. Schneidewind, J. Singer, S. Takada, R. Vehvilainen y H. Yang (1999). "Towards an ontology of software maintenance." Journal of Software Maintenance: Research and Practice 11(6): 365-389.
- ◆ Mayrhauser, A. v. y A. M. Vans (1995). "Program Comprehension During Software Maintenance and Evolution." IEEE Computer 28(8): 44-55.
- ◆ Mayrhauser, A. v. y A. M. Vans (1996). "Identification of Dynamic Comprehension Processes During Large Scale Maintenance." IEEE Transactions on Software Engineering 22(6): 424-437.

Referencias

- ◆ Oliveira, K. M. d., N. Anquetil, M. G. B. Dias, M. Ramal y R. Meneses (2003). Knowledge for Software Maintenance. Proceedings of the Fifteenth International Conference on Software Engineering and Knowledge Engineering, San Francisco, California, USA, Knowledge Systems Institute.
- ◆ Pigoski, T. M. (2001). Software Maintenance. SWEBOK: Guide to the Software Engineering Body of Knowledge. Abran, A., J. W. Moore, P. Bourque yR. Dupuis. Los Alamitos, California, IEEE Computer Society: 87-98.
- ◆ Polo, M., M. Piattini y F. Ruiz (2002). "Using a Qualitative Research Method for Building a Software Maintenance Methodology." Software Practice & Experience 32(13): 1239-1260.
- ◆ Ruiz, F., A. Vizcaíno Barceló, M. Piattini y F. García (2004). "An Ontology for the Management of Software Maintenance Projects." International Journal of Software Engineering and Knowledge Engineering.
- ◆ Seaman, C. (2002). The Information Gathering Strategies of Software Maintainers. Proceedings of the International Conference on Software Maintenance.
- ◆ Walz, D. B., J. J. Elam y B. Curtis (1993). "Inside a Software Design Team: Knowledge Acquisition, Sharing, and Integration." Communications of the ACM 36(10): 63-77.