



Introducción a la Ingeniería de Procesos Software

Francisco Ruiz
Universidad de Cantabria



Objetivos

- **Conocer** los principios e importancia de la **IPS**.
- Aprender las características del **estándar SPEM** como base para el modelado de ciclos de vida, metodologías, procesos, métodos, buenas prácticas, técnicas, etc.



•Introducción

- Procesos Software
- Perspectiva de Ingeniería
- Modelos de Procesos Software
- Principios de la Ingeniería de Procesos Software

•Fundamentos de SPEM

- Características básicas
- Metamodelo vs perfil UML
- Usos
- Características avanzadas



- OMG (2007): Software & Systems Process Engineering Metamodel Specification (**SPEM**); version 2.0, agosto 2007.
 - <http://www.omg.org/cgi-bin/doc?ptc/07-08-07>
- Ruiz, F. y Verdugo, J. (2008): Guía de Uso de SPEM 2 con EPF Composer, versión 3. UCLM.
 - Web de la asignatura.
- Software Process Engineering in the Real World
 - <http://www.softwareprocessengineering.com/>



•Introducción

- Procesos Software
- Perspectiva de Ingeniería
- Modelos de Procesos Software
- Principios de la Ingeniería de Procesos Software

•Fundamentos de SPEM

- Características básicas
- Metamodelo vs perfil UML
- Usos
- Características avanzadas



•Introducción

- Procesos Software
- Modelos de Procesos Software
- Principios de la Ingeniería de Procesos Software

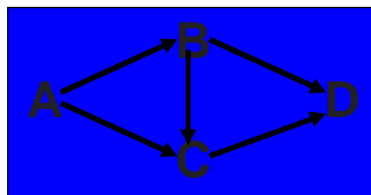
•Fundamentos de SPEM

- Características básicas
- Metamodelo vs perfil UML
- Usos



Un **Proceso Software** (PS) es

Un conjunto coherente de políticas, estructuras organizacionales, tecnologías, procedimientos y artefactos que son necesarios para concebir, desarrollar, instalar y mantener un producto software.
(Fugetta, 2000)



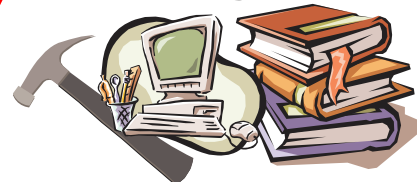
Métodos y Procedimientos que definen la relaciones entre las Tareas.

Personal



PROCESO SW

Herramientas y Metodologías.





- Son complejos:
- No son procesos de producción:
 - Dirigidos por excepciones,
 - Muy determinados por circunstancias impredecibles,
 - Cada uno con sus peculiaridades.
- No son procesos de ingeniería “*pura*”:
 - Desconocemos las abstracciones adecuadas,
 - Dependen demasiado de demasiada gente,
 - Diseño y producción no están claramente separados,
 - Presupuestos, calendarios, calidad no pueden ser planificados de forma fiable.



- No son (*completamente*) procesos creativos:
 - Algunas partes pueden ser descritas en detalle,
 - Algunos procedimientos han sido impuestos.
- Están basados en descubrimientos que dependen de la comunicación, coordinación y cooperación dentro de marcos de trabajo predefinidos:
 - Los entregables generan nuevos requerimientos,
 - Los costes del cambio del software no suelen reconocerse,
 - El éxito depende de la implicación del usuario y de la coordinación de muchos roles (ventas, desarrollo técnico, cliente, etc.).



Gestión y Mejora:

- Años 80
- Creciente Importancia en la industria Software por la calidad
- Aparecen estándares como la familia ISO 9000 y modelos de madurez como CMM (finales de los 80)
- Estándares ISO 9000
 - Certificación Calidad → Garantía de que una organización software entregará productos de calidad
- Estos estándares y modelos incluyen prácticas que facilitan la gestión de los procesos software
- Aparecen ciertas limitaciones:
 - ¿una organización con certificación de calidad obtendrá siempre productos de alta calidad?
 - Incremento de Burocracia

11



Utilidad en **Procesos Software**

- Para poder integrar varios PS, cada uno con su MP.
- Para la mejora de PS.
 - Permitiendo la evolución del modelo de un PS.
 - Pudiendo gestionar de forma integrada el proceso y su ciclo de vida (diseño, despliegue, ejecución, automatización, mejora, ..).
- Para construir Plataformas más potentes.
 - Haciendo que el repositorio sea más genérico y tenga más capacidad semántica.
 - Permitiendo que todas las herramientas (CASE, gestión de proyectos, ...) compartan los modelos.
 - Pudiendo realizar procesamiento y transformaciones directamente sobre los modelos.

12



Pasar de la **gestión de modelos** “contemplativa” a la “productiva” significa que la mayoría de los pasos de la cadena de producción y mantenimiento de software puede considerarse como operaciones definidas de forma precisa sobre artefactos del modelo.

Es muy diferente lo que se puede hacer con la descripción de una tarea según el formato en que esté:

- Texto en PDF o DOC
- Archivo XML basado en SPEM
- Web navegable
- ...

13



Tarea DSI 7.1: Verificación de las Especificaciones de Diseño

El objetivo de esta tarea es asegurar la calidad formal de los distintos modelos, conforme a la técnica seguida para la elaboración de cada producto y a las normas y estándares especificados en el catálogo de normas.

Productos

De entrada

- Catálogo de Requisitos (DSI 1.2)
- Catálogo de Excepciones (DSI 1.3)
- Catálogo de Normas (DSI 1.4)
- Diseño de la Arquitectura del Sistema (DSI 1.5)
- Entorno Tecnológico del Sistema (DSI 1.6)
- Diseño Detallado de Subsistemas de Soporte (DSI 2.1)
- Modelo Físico de Datos Optimizado (DSI 6.3)
- Esquemas Físicos de Datos (DSI 6.4)
- Asignación de Esquemas Físicos de Datos a Nodos (DSI 6.4)

En Diseño Estructurado:

- Diseño de la Arquitectura Modular (DSI 5.2)
- Diseño de Interfaz de Usuario (DSI 5.3)

En Diseño Orientado a Objetos:

- Diseño de la Realización de los Casos de Uso (DSI 3.4)
- Diseño de Interfaz de Usuario (DSI 3.3)
- Modelo de Clases de Diseño (DSI 4.6)
- Comportamiento de Clases de Diseño (DSI 4.4)

De salida

- Entorno Tecnológico del Sistema
- Diseño de la Arquitectura del Sistema
- Diseño Detallado de Subsistemas de Soporte
- Modelo Físico de Datos Optimizado
- Esquemas Físicos de Datos
- Asignación de Esquemas Físicos de Datos a Nodos
- Diseño de Interfaz de Usuario

En Diseño Estructurado:

- Diseño de la Arquitectura Modular
- En Diseño Orientado a Objetos:
- Diseño de la Realización de los Casos de Uso
- Modelo de Clases de Diseño
- Comportamiento de Clases de Diseño

Participantes

- Equipo de Arquitectura
- Equipo del Proyecto

PDF de METRICA 3 =>
Leerlo

14



```

<?xml version="1.0" encoding="UTF-8"?>
<org.eclipse.epf.uma:TaskDescription xmi:version="2.0"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:org.eclipse.epf.uma="http://www.eclipse.org/epf/uma/1.0.4/uma.ecore"
  xmlns:epf="http://www.eclipse.org/epf" epf:version="1.2.0" xmi:id="-
vF3xMvNwUJzWzNdl7nMXUCA"
  ...
  <mainDescription>&lt;p>&#xD;
  Identificador de tarea: 55&lt;br />&#xD;
  Origen de la tarea: &lt;em>Métrica 3.&lt;/em> Tarea &lt;em>DSI 6.1: Diseño
del modelo físico de datos.&lt;/em>&#xD;
&lt;/p>&#xD;
&lt;p align="center">&#xD;
  En esta tarea se determina cómo se van a convertir las clases en tablas,
considerando las relaciones existentes entre&#xD; ellas y los identificadores,
definiendo sus claves primarias, ajenas, alternativas u otros medios de acceso
en general.&#xD;
  ....
  así como para hacer una estimación del&#xD; espacio de
almacenamiento.&#xD;
&lt;/p></mainDescription>
  <sections xmi:id="-_9oDDkHMaEdyRedV3y8FdVg" name="El Administrador
de Bases de Datos y el Equipo de Desarrollo analizan el gestor de bases de
datos o el sistema de ficheros."
  guid="_9oDDkHMaEdyRedV3y8FdVg"/>
  <sections xmi:id="-_D_cDcHMBedyRedV3y8FdVg" name="El Equipo de
Desarrollo y el Equipo de Arquitectura analizan las estimaciones de utilización y
volumen de las ocurrencias de cada clase del modelo de clases."
  guid="-_D_cDcHMBedyRedV3y8FdVg"/>
  ...
  <purpose>&lt;p align="center">&#xD;
  El objetivo de esta tarea es realizar el diseño del modelo&nbsp; lógico de
datos a partir del modelo de clases.&#xD;
&lt;/p></purpose>
</org.eclipse.epf.uma:TaskDescription>

```

Especificación en XML de la misma tarea de METRICA 3 usando el metamodelo SPEM =>

ahora se puede realizar cualquier tipo de procesamiento automático que interese



Tarea: DSI 07.1: Diseño del modelo lógico de datos

En esta tarea se realiza el diseño del modelo lógico de datos.
Disciplinas: Diseño

Expandir todas las secciones | Contraer todas las secciones

Objetivo
El objetivo de esta tarea es realizar el diseño del modelo lógico de datos a partir del modelo de clases.
Volver al inicio

Relaciones		
Roles	Realizador principal: • Equipo de Desarrollo	Otras actividades adicionales que realiza: • Administrador de Bases de Datos • Equipo de Arquitectura
Entradas	Obligatoria: • Características específicas del SGBD o sistema de ficheros que se utilizará • PPL: Plan de migración y de carga inicial de datos • SDT: Especificaciones de migración y carga inicial de datos • SDT: Modelo de clases de diseño	Opcional: • Ninguno
Salidas	• SDT: Modelo lógico de datos	

Volver al inicio

Descripción principal
Identificador de tarea: 55
Origen de la tarea: *Métrica 3. Tarea DSI 6.1: Diseño del modelo físico de datos.*
En esta tarea se determina cómo se van a convertir las clases en tablas, considerando las relaciones



Los procesos de diferentes proyectos tienden a seguir patrones comunes.

Es necesario intentar capturar estos aspectos comunes en una **representación del proceso**, la cuál describe estas características comunes y fomenta la homogeneidad.

Modelo de Procesos (MP): representación abstracta de una familia de procesos expresada en una adecuada notación de modelado de procesos (formalismo).

17



Un **Modelo de PS**

Es una abstracción o representación (textual, gráfica o formal) en la que se capturan los aspectos más importantes de un PS

Es aplicable a un proyecto particular o a una familia de proyectos

Es una representación descriptiva de:

- las actividades,
- los recursos,
- los productos,
- los actores y
- las reglas que el proceso requiere para alcanzar sus objetivos.

Está basada en un lenguaje de modelado (metamodelo+sintaxis)

18

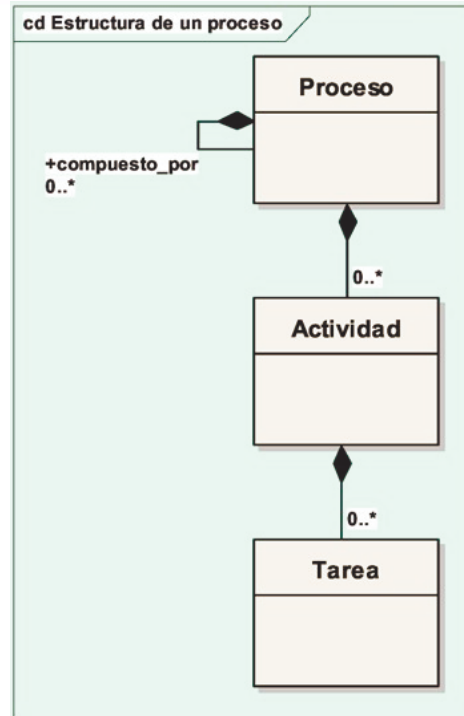


Según los estándares ISO, un PS tiene una **estructura jerárquica** con varios niveles de agregación:

- Subprocesos (opcionales)
- Actividades
- Tareas

Los procesos, subprocessos y actividades tienen asociado un flujo de trabajo.

Las tareas son las unidades básicas de trabajo (atómicas).



La disponibilidad de un MP (*computerizado*) proporciona capacidades para:

- Facilitar la comprensión y comunicación humana.
- Facilitar la reutilización.
- Dar soporte a la mejora de procesos.
- Dar soporte a la gestión de procesos.
- Guiar la automatización de procesos.
- Dar soporte para la ejecución automática.

Para poder ofrecer lo anterior, los MPS deben:

- Representar la forma en que el proceso es (o debería ser) realizado;
- Proveer un marco de trabajo flexible y fácil de comprender, aunque con potencia para representar y mejorar el proceso; y
- Permitir refinar hasta llegar al nivel de detalle que sea necesario.



Expresan un punto de interés particular en vez del MP completo (similar a vistas en BD):

- Sub-modelos (en modelado *bottom-up*).
- Modelos parciales (en modelado *top-down*).

Las más habituales son:

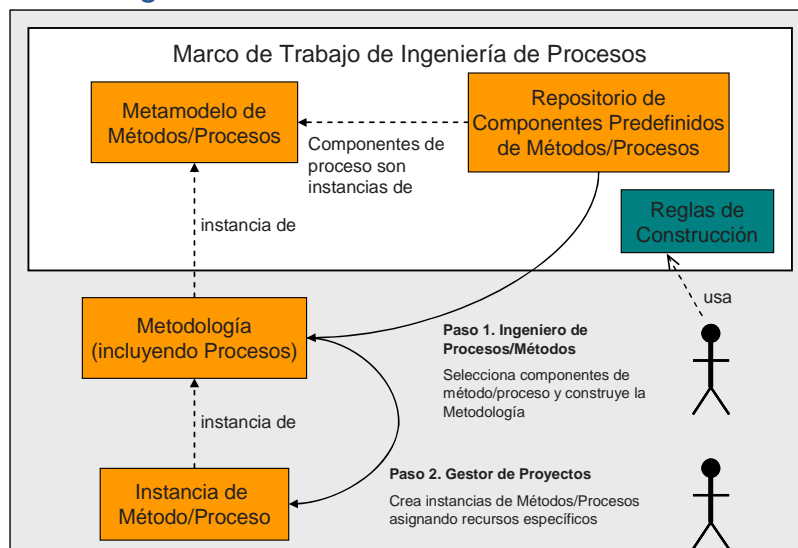
- **De actividades:** tipos, estructura y propiedades de las actividades y sus relaciones
- **De productos:** tipos, estructura y propiedades de los ítems software de un proceso;
- **De recursos:** describe los recursos que se necesitan o se suministran a los procesos;
- **De roles:** describe un peculiar conjunto de recursos, como son las habilidades que los desarrolladores suministran y las responsabilidades que aceptan.

Nos son disjuntas: una vista no puede ser definida sin usar conceptos de otras.



La **Ingeniería de Procesos** trata de aplicar a los procesos maneras y técnicas que antes han demostrado su utilidad en los productos (software).

Sinónimo: Ingeniería de Métodos





- Introducción

- Procesos Software
- Perspectiva de Ingeniería
- Modelos de Procesos Software
- Principios de la Ingeniería de Procesos Software

- Fundamentos de SPEM

- Características básicas
- Metamodelo vs perfil UML
- Usos
- Características avanzadas

23



SPEM 2

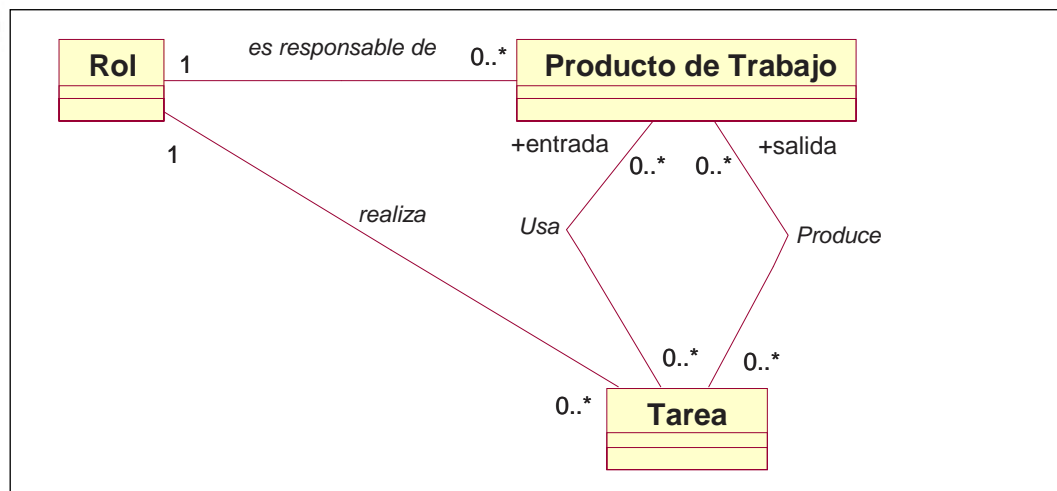
Software & Systems Process Engineering Metamodel Specification, v2.0

Metamodelo para modelos de procesos de ingeniería del software y de ingeniería de sistemas

- Se describe de dos maneras:
 - como un metamodelo MOF-compliant, y
 - como un perfil UML 2.

24

Idea básica de proceso:

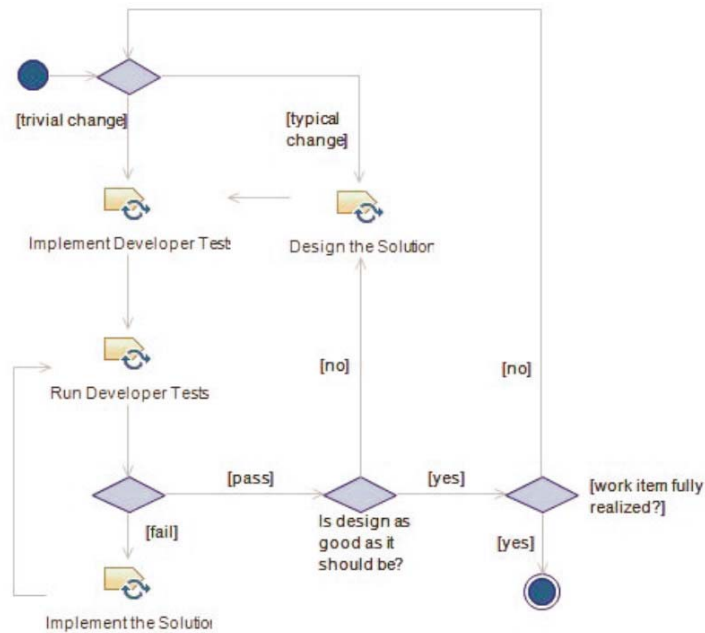


Las descripciones de trabajo tienen asociada una estructura de desglose de trabajo

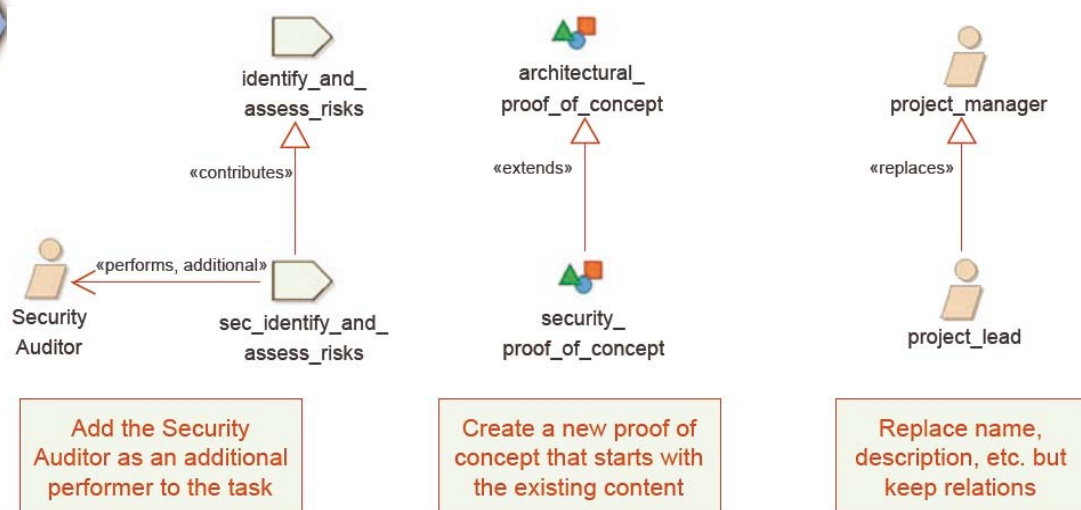
- Metrica 3 (Versión Estructurado)
 - Proceso ASI (Versión Estructurado)
 - ASI 1: Definición del Sistema (Versión Estructurado)
 - ASI 1.1: Determinación del Alcance del Sistema (Versión Estructurado)
 - ASI 1.2: Identificación del Entorno Tecnológico (Versión Estructurado)
 - ASI 1.3: Especificación de Estándares y Normas (Versión Estructurado)
 - ASI 1.4: Identificación de los Usuarios Participantes y Finales (Versión Estructurado)
 - ASI 2: Establecimiento de Requisitos (Versión Estructurado)
 - ASI 2.1: Obtención de Requisitos (Versión Estructurado)
 - ASI 2.2: Especificación de Casos de Uso (Versión Estructurado)
 - ASI 2.3: Análisis de Requisitos
 - ASI 2.4: Validación de Requisitos
 - ASI 3: Identificación de Subsistemas de Análisis (Versión Estructurado)
 - ASI 3.1: Determinación de Subsistemas de Análisis (Versión Estructurado)
 - ASI 3.2: Integración de Subsistemas de Análisis (Versión Estructurado)
 - ASI 6: Elaboración del Modelo de Datos
 - ASI 6.1: Elaboración del Modelo Conceptual de Datos
 - ASI 6.2: Elaboración del Modelo Lógico de Datos
 - ASI 6.3: Normalización del Modelo Lógico de Datos
 - ASI 6.4: Especificación de Necesidades de Migración y Carga de Datos
 - ASI 7: Elaboración del Modelo de Procesos
 - ASI 7.1: Obtención del Modelo de Procesos del Sistema
 - ASI 7.2: Especificación de Interfaces con Otros Sistemas
 - ASI 8: Definición de Interfaces de Usuario (Versión Estructurado)
 - ASI 8.1: Especificación de Principios Generales de la Interfaz
 - ASI 8.2: Identificación de Perfiles y Diálogos
 - ASI 8.3: Especificación de Formatos Individuales de la Interfaz de Pantalla (Versión Estructurado)
 - ASI 8.4: Especificación del Comportamiento Dinámico de la Interfaz (Versión Estructurado)
 - ASI 8.5: Especificación de Formatos de Impresión (Versión Estructurado)
 - ASI 9: Análisis de Consistencia y Especificación de Requisitos (Versión Estructurado)
 - ASI 9.1: Verificación de los Modelos (Versión Estructurado)
 - ASI 9.2: Análisis de Consistencia Entre Modelos (Versión Estructurado)
 - ASI 9.3: Validación de los Modelos (Versión Estructurado)
 - ASI 9.4: Elaboración de la Especificación de Requisitos Software (ERS) (Versión Estructurada)

**Work
Breakdown
Structure
(WBS)**

Las descripciones de trabajo también pueden tener asociado un flujo de trabajo



Existen mecanismos de **variabilidad** para modificar un proceso base



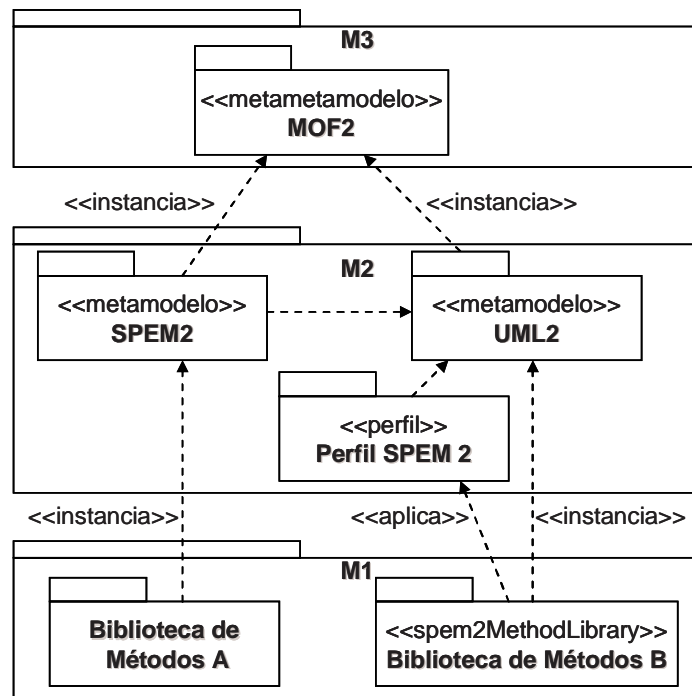


- SPEM 2 sirve para definir procesos de desarrollo de software y sistemas y sus componentes.
- =>
 - Su **alcance** se limita a los elementos mínimos necesarios para definir dichos procesos sin añadir características específicas de un dominio o disciplina particular.
 - Pero sirve para métodos y procesos de **diferentes** estilos, culturas, niveles de formalismo, o modelos de ciclos de vida.
 - No es un lenguaje de modelado de procesos en general.
 - No provee conceptos para **modelado del comportamiento**, pero incluye mecanismos para encajar el elegido (diagramas de actividad de UML 2, BPMN/BPDM, ..).

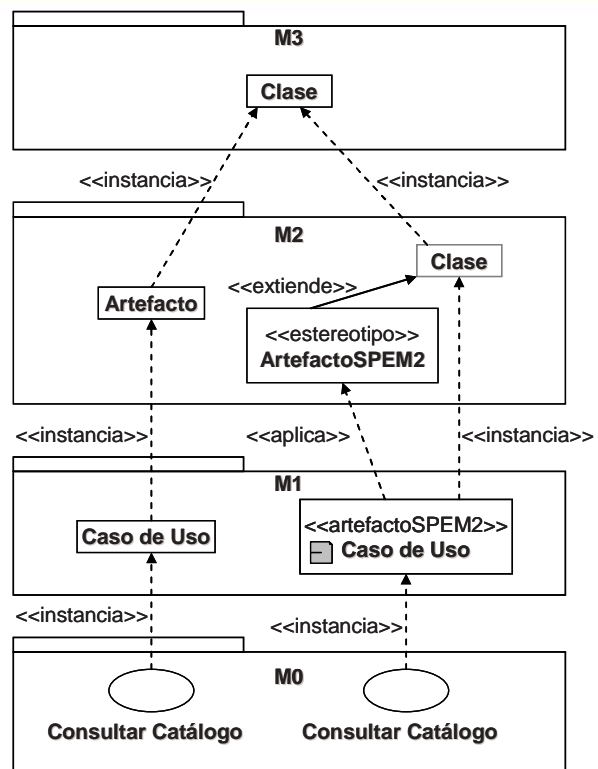


- **Metamodelo MOF-compliant**
 - Define todas las estructuras y reglas de estructuración para representar contenidos de métodos y procesos.
 - Es completo en sí mismo.
 - Está definido como un metamodelo del nivel M2 de MOF.
 - Reutiliza algunas clases de UML 2.
 - Define la notación de diagramas de proceso específicos.
- **Perfil de UML 2.**
 - Define un conjunto de estereotipos UML 2 que permiten representar métodos y procesos usando UML 2.
 - La definición sólo abarca la presentación, ya que las definiciones semánticas y restricciones están en el metamodelo.

- Capas de modelado



- Ejemplos de instancias





Problemas relacionados con los procesos, que enfrentan las organizaciones que desarrollan software:

- Miembros de los equipos no tienen acceso fácil y centralizado a la información de procesos que necesitan
 - Diferentes desarrolladores manejan versiones diferentes fuentes o versiones de la misma información.
- Es difícil combinar e integrar informaciones y procesos que están en formatos propietarios diferentes
 - Cada libro, manual , herramienta utiliza un lenguaje y estilo diferente.
- Es duro definir una aproximación de desarrollo organizada y sistemática que se adapte a las necesidades
 - Cultura, prácticas establecidas, requisitos de certificación, legales, etc.

33



- SPEM es un metamodelo para ingeniería de procesos y un marco de trabajo conceptual que provee los conceptos necesarios para **modelar, documentar, presentar, publicar, gestionar, intercambiar y realizar métodos y procesos software.**
- Está destinado a ingenieros de procesos, jefes de proyectos, gestores de proyectos y programas que son **responsables de mantener e implementar procesos** para sus organizaciones o para proyectos concretos.

34

Normalizar la representación y gestionar un repositorio de **Contenidos de Método** reutilizables

- Contenido sobre métodos ágiles
- Contenido sobre gestión del desarrollo iterativo
- Guías sobre java beans serializados



- Guía de usuario de JUnit
- Contenido sobre J2EE
- Directrices para gestión de configuración

Desarrollar y gestionar **Procesos** para llevar a cabo proyectos

- Proceso para desarrollar aplicaciones con J2EE
- Proceso para desarrollar sistemas embebidos
- Proceso basado en SOA

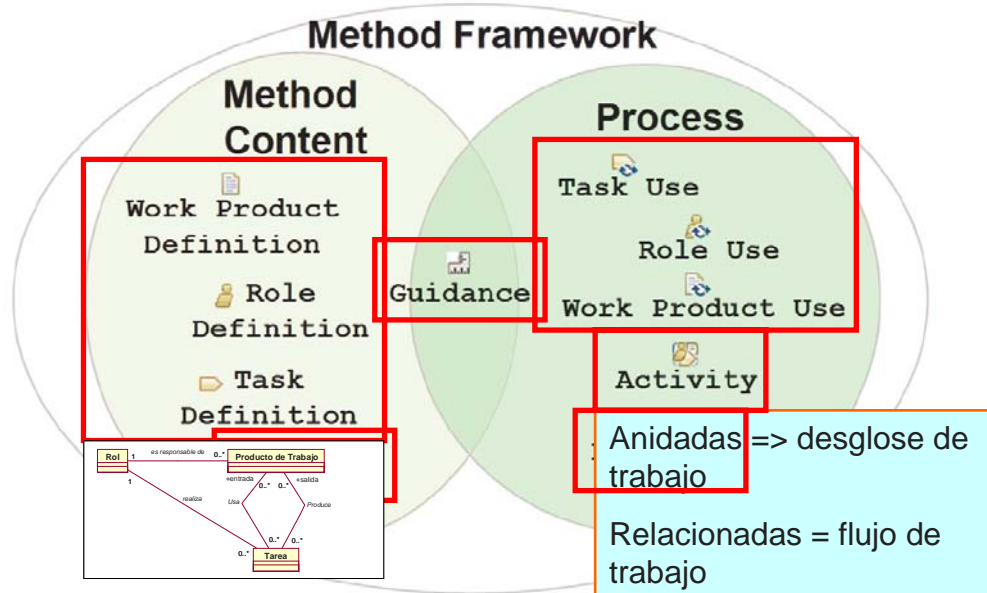


- Patrones de proceso
- Proceso estándar o de referencia
- Plantillas ejecutables para planes de proyectos
- Guías corporativas

Configurar un marco de trabajo con procesos integrado y adaptado para mis necesidades (proyectos)

Crear plantillas de planes de proyecto para la **Realización** de procesos en el contexto de mi proyecto

Separación clara entre la definición de contenidos de método y su aplicación en procesos.



Conceptos de contenido de método vs proceso



Mantenimiento consistente de muchos procesos alternativos.

- Para ello, SPEM incluye:
 - Un conjunto extendido de interrelaciones de reutilización y variabilidad con semántica de herencia y orientación a aspectos.
 - Conceptos de patrones de proceso, y
 - Plugins de métodos.
- Esto permite tener diferentes variantes de procesos específicos, basados en los mismos contenidos de método y estructuras de procesos, pero aplicados con diferente detalle y escala.

37



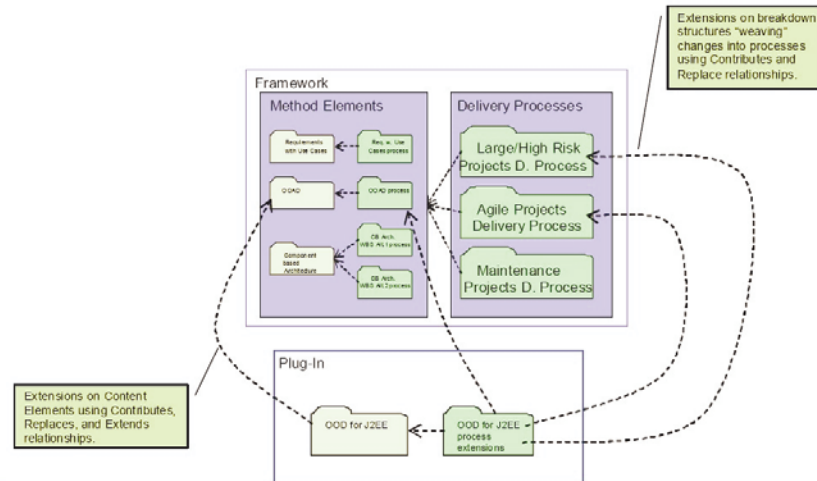
Muchos ciclos de vida diferentes.

- SPEM permite trabajar con distintos tipos de ciclos de vida del software:
 - Cascada, Iterativo, Incremental, Evolutivo, ..
- Para ello incluye un conjunto de atributos que permiten especificar aspectos temporales para los elementos de proceso que luego pueden ser asociados a los planes de proyectos.
- Ejemplo de atributo para clases de ciclos de vida:
 - Iteración => la ejecución de una o varias descripciones de trabajo se puede repetir más de una vez.

38

Variabilidad y extensibilidad.

- Para esto SPEM incluye un mecanismo de **plugins**:
 - **Method plug-ins**: para particularizar y adaptar contenidos de método sin modificar el original.
 - **Process plug-ins**: para procesos, pudiendo añadir o sustituir en el WBS sin afectar al original.



39

Patrones de proceso.

- Son bloques (trozos de proceso) reutilizables para crear nuevos procesos.
- La selección y aplicación de un patrón de proceso puede ser hecha de dos formas:
 - Puede ser **copiado y modificado**, permitiendo individualizar el contenido del patrón según las necesidades de cada momento.
 - Puede ser aplicado por medio del mecanismo de **Actividad en Uso**, que es una forma avanzada de reutilizar estructuras de proceso.
- Una **Actividad en Uso** define tipos de interrelaciones para que cuando el patrón esté siendo revisado o modificado, todos los cambios se reflejen automáticamente en todos los procesos en que se aplica el patrón.

40

Componentes de proceso.

- Piezas de proceso sustituibles y reutilizables basadas en los principios de **encapsulación** y **caja negra**:
 - No se especifica la descripción de trabajo interna del componente.
 - Sólo se especifican los productos de trabajo de entrada y salida que habrá (**puertos** de productos de trabajo).
- Permiten manejar las situaciones en que un proyecto requiere que partes del proceso no sean decididas hasta la ejecución (caso típico: outsourcing).

