

# **GUÍA DE TAILORING DEL PROCESO UNIFICADO A PROYECTOS ESPECÍFICOS PARA LAS PYMES DESARROLLADORAS DE SOFTWARE**



**Pablo Hernando Ruiz Melenje  
Carlos Ernesto Maya Vallejo**

Director: Ing. Esp. Wilson Libardo Pantoja  
Co-director: Dr. Julio Ariel Hurtado

**Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Programa de ingeniería de Sistemas  
Grupo IDIS -Investigación y Desarrollo en Ingeniería de  
Software  
Popayán, 2009**

# **GUÍA DE TAILORING DEL PROCESO UNIFICADO A PROYECTOS ESPECÍFICOS PARA LAS PYMES DESARROLLADORAS DE SOFTWARE**



**Pablo Hernando Ruiz Melenje  
Carlos Ernesto Maya Vallejo**

Trabajo de investigación para optar al título de Ingenieros de Sistemas

Director: Ing. Esp. Wilson Libardo Pantoja  
Co-director: Dr. Julio Ariel Hurtado

**Universidad del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Programa de ingeniería de Sistemas  
Grupo IDIS -Investigación y Desarrollo en Ingeniería de  
Software  
Popayán, 2009**

## **Agradecimientos**

A la Universidad del Cauca, al Programa de Ingeniería de Sistemas y a los docentes por sus enseñanzas y consejos durante toda la carrera de Ingeniería de sistemas, los cuales desarrollaron en nosotros un alto grado de ética, mejora continua y la Ingeniería como profesión y método de vida.

A Parquesoft, en especial a la empresa INPUT Technologies y a los estudiantes del programa de ingeniería de sistemas de la Universidad del Cauca, por la colaboración prestada en la ejecución del proyecto.

A nuestras familias por su apoyo y cariño incondicional.

Mil Gracias a todos.

## Dedicatoria

*A mis padres por ser personas maravillosas que incondicionalmente han estado conmigo en las tristezas y alegrías durante el transcurso de mi vida, y que me han inculcado principios y valores para ser una mejor persona.*

*A mis hermanos por ser un ejemplo de superación y apoyo que inspiran en mí el deseo de fijarme metas cada vez más altas. A mi sobrina por infundir en mi cariño, ternura y responsabilidad.*

*A mi novia por compartir su vida y por ser muestra de progreso y tenacidad.*

*A mis compañeros y amigos, que de alguna forma han colaborado en el alcance de esta meta.*

**Pablo Hernando Ruiz Melenje.**

## Tabla de contenido

<b>AGRADECIMIENTOS</b> .....	<b>3</b>
<b>DEDICATORIA</b> .....	<b>4</b>
<b>TABLA DE CONTENIDO</b> .....	<b>5</b>
<b>LISTA DE FIGURAS</b> .....	<b>7</b>
<b>LISTA DE TABLAS</b> .....	<b>8</b>
<b>CAPITULO 1. MARCO TEORICO.</b> .....	<b>9</b>
<b>1.1. EL PROCESO DE SOFTWARE</b> .....	<b>9</b>
<b>1.2. ¿QUÉ ES TAILORING?</b> .....	<b>9</b>
1.2.1 Aplicaciones del Tailoring .....	10
1.2.2 ¿Por qué es importante el Tailoring en las PyMEs_DS ? .....	10
<b>1.3 REVISIÓN DE LITERATURA</b> .....	<b>11</b>
1.3.1 Tailoring and Verifying software Process [17]. .....	11
1.3.2 Tailoring Software Evolution Process [19]. .....	12
1.3.3 Modelo de Adaptación de procesos software (MAPS) [25]. .....	14
1.3.4 Process Tailoring and the Software Capability Maturity Model (CMM) [18]. .....	15
1.3.5 A Matrix Approach to Software Process Definition (Centro de Vuelos Espaciales Goddard) [23].	16
1.3.6 Disciplina de ambiente de RUP [10]. .....	17
1.3.7 Tailoring a Software Process for Software Project Plans [26]. .....	18
1.3.8 SPEM 2.0 (Software and Systems Process Engineering Meta-model) [12]. .....	20
1.3.9 La plataforma EPFC [27]. .....	23
<b>CAPITULO 2 ADAPTACIÓN DE UP</b> .....	<b>25</b>
<b>2.1 EL PROCESO UNIFICADO (UP) [10].</b> .....	<b>25</b>
2.1.1 ¿Qué es el Proceso Unificado (UP Unified Process)? .....	25
2.1.2 Mejores Prácticas.....	25
2.1.3 Dos dimensiones.....	26
<b>2.2 ADAPTACIÓN DE UP (¿POR QUÉ ES IMPORTANTE LA ADAPTACIÓN?)</b> .....	<b>34</b>
<b>2.3 EL PROCESO UNIFICADO DESCRITO A TRAVÉS DE SPEM 2.0</b> .....	<b>35</b>
2.3.1 Aspectos utilizados para implementar UP con SPEM 2.0.....	35
2.3.2 Método Plugin del Proceso Unificado .....	36
<b>2.1.1.3 CONTENIDO DEL MÉTODO</b> .....	<b>36</b>
<b>CAPITULO 3. DESCRIPCIÓN DEL PROYECTO.</b> .....	<b>40</b>
<b>3.1 EVALUACIÓN DE ESTUDIOS</b> .....	<b>40</b>
3.1.1 Criterios de evaluación.....	41
<b>3.2 ENFOQUE DE LA GUÍA DE TAILORING</b> .....	<b>43</b>
<b>3.3 CARACTERIZACIÓN DEL PROYECTO</b> .....	<b>44</b>
3.3.1 Variables para realizar la caracterización de los proyectos. ....	44
<b>3.4 SELECCIONAR LAS ENTIDADES DEL PROCESO</b> .....	<b>51</b>
<b>3.5 CONFIGURACIÓN DEL PROCESO</b> .....	<b>54</b>
3.5.1 Operación de Eliminación .....	55
3.5.2 Operación de adición .....	56
3.5.3 Operación de Reemplazo. ....	62
<b>3.6 DESPLEGAR EL PROCESO ADAPTADO</b> .....	<b>69</b>
<b>3.7 RESUMEN DE LA GUÍA DE TAILORING</b> .....	<b>70</b>
3.7.1 Elementos de la guía .....	71
3.7.1.1 Patrón de procesos:.....	71
3.7.1.3 Rol encargado de realizar la adaptación: .....	72
3.7.2 Pasos para realizar la adaptación.....	72
<b>3.8 PASOS DE LA GUÍA DE TAILORING DEL PROCESO UNIFICADO.</b> .....	<b>72</b>
3.8.1 Primer paso - Caracterización del Proyecto. ....	72
3.8.2 Segundo paso – Selección de las entidades del proceso. ....	75

---

3.8.3	Tercer paso – Realizar la configuración del proceso. ....	80
3.8.4	Cuarto paso- Desplegar el proceso.....	83
<b>CAPITULO 4. APLICACIÓN DE GTUP – AMBIENTE ACADEMICO Y EMPRESARIAL.....</b>		<b>86</b>
<b>4.1</b>	<b>AMBIENTE ACADÉMICO .....</b>	<b>87</b>
4.1.1	PRUEBA DE CONCEPTO .....	87
4.1.1.1	Contextualización de la guía.....	87
4.1.1.2	Ejecución prueba de concepto.....	87
4.1.2	AMBIENTE ACADÉMICO EN UN PROYECTO.....	88
4.1.2.1	Primer paso - Caracterización del Proyecto.....	88
4.1.2.2	Segundo paso - Selección de las entidades del proceso .....	89
4.1.2.3	Tercer Paso – Configurar el Proceso.....	89
4.1.2.4	Cuarto Paso - Desplegar el proceso.....	89
<b>4.2</b>	<b>AMBIENTE EMPRESARIAL .....</b>	<b>90</b>
4.2.1	Contextualización de la guía .....	90
4.2.2	Primer paso - Caracterización del Proyecto.....	91
4.2.3	Segundo paso - Selección de las entidades del proceso.....	91
4.2.4	Tercer Paso – Configurar el Proceso.....	92
4.2.5	Cuarto Paso - Desplegar el proceso.....	93
<b>4.3</b>	<b>CONCLUSIONES DE LA APLICACIÓN DE GTUP.....</b>	<b>93</b>
<b>CAPITULO 5. LECCIONES APRENDIDAS, CONCLUSIONES Y TRABAJOS FUTUROS. ....</b>		<b>94</b>
<b>5.1</b>	<b>LECCIONES APRENDIDAS .....</b>	<b>94</b>
<b>5.2</b>	<b>CONCLUSIONES GENERALES.....</b>	<b>95</b>
<b>5.3</b>	<b>TRABAJOS FUTUROS.....</b>	<b>96</b>
<b>REFERENCIAS.....</b>		<b>97</b>

## LISTA DE FIGURAS

Figura 1	Bloque de Secuencia.....	12
Figura 2	Bloque de Concurrencia .....	12
Figura 3	Bloque de Selección.....	13
Figura 4	Bloque de Iteración.....	13
Figura 5	Adición de la actividad <i>ep</i> dentro de un bloque de secuencia.....	13
Figura 6	Modelo de Adaptación de Procesos.....	14
Figura 7	Modelo conceptual de una Metodología.....	16
Figura 8	Matriz de representación de procesos. ....	17
Figura 9	Resumen de la adaptación del proceso software.....	20
Figura 10	Escenarios de trabajo de SPEM.....	21
Figura 11	Entorno del EPFC.....	23
Figura 12	Estructura de UP .....	27
Figura 13	Fases e hitos en UP .....	27
Figura 14	Representación de Roles, actividades y artefactos en UP.....	31
Figura 15	Ortogonalidad en SPEM y UP .....	36
Figura 16	Paquetes de Contenido definidos para el UP.....	36
Figura 17	División del Paquete de Contenido Modelado del Negocio.....	37
Figura 18	Categorías Estándar utilizadas en UP.....	37
Figura 19	Estructura de las Categorías Personalizadas .....	38
Figura 20	Publicación de las Categorías Personalizadas .....	38
Figura 21	Paquetes de Proceso definidos para el Plugin.....	39
Figura 22	Patrones de proceso .....	39
Figura 23	Proceso para despliegue.....	40
Figura 24	Elementos necesarios en la definición del Rol .....	58
Figura 25	Elementos necesarios para la definición de una tarea.....	59
Figura 26	Elementos necesarios para adicionar un artefacto. ....	60
Figura 27	Representación gráfica de una actividad .....	63
Figura 28	Actividades a ser ensambladas.....	64
Figura 29	Ensamble de las Actividades Previa, Posterior y Nativa. ....	65
Figura 30	Actividad reemplazante o nativa, requiere de una entrada no suministrada por la actividad previa del proceso. ....	66
Figura 31	<i>La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa. ....</i>	66
Figura 32	<i>La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso. ....</i>	67
Figura 33	<i>La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior .....</i>	67
Figura 34	Guía de tailoring del Proceso Unificado - GTUP. ....	71
Figura 35	Flujo de trabajo para el primer paso - Caracterización del Proyecto.....	73
Figura 36	Flujo de trabajo para el segundo paso - Selección de las entidades del proceso.....	76
Figura 37	Flujo de trabajo para el tercer paso - Configuración del proceso.....	81
Figura 38	Flujo de trabajo para el cuarto paso - Desplegar el proceso.....	84

## **LISTA DE TABLAS**

Tabla 1 Nomenclatura de criterios .....	42
Tabla 2. Evaluación de estudios e investigaciones de adaptación.....	43
Tabla 3. Clasificación de Artefactos.. .....	52
Tabla 4. Ejemplo de Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización.....	53
Tabla 5. Actividad vs Componente .....	63

## **CAPITULO 1. MARCO TEORICO.**

En el presente capítulo, se dan a conocer algunos conceptos importantes relacionados con el tailoring de procesos, así como una descripción de algunos estudios encontrados en la literatura sobre este tema y por último se presentan SPEM (Software & Systems Process Engineering Meta-Model Specification) versión 2.0 y EPFC (Eclipse Process Framework Composer) como lenguaje y herramienta de modelado de procesos.

### **1.1. El proceso de software**

En [18] el proceso de software se define como un conjunto de actividades, métodos, prácticas y transformaciones que las personas usan para desarrollar y mantener software, como también sus productos asociados (p.ej., planes, especificaciones, diseños y pruebas), en [15] un proceso de software es un conjunto de actividades necesarias para transformar los requerimientos del usuario en un sistema software, y [24] se refiere a un proceso software como el conjunto de herramientas, métodos, y prácticas usadas para producir un artefacto software.

Una ilustración formal de los procesos software para el desarrollo y mantenimiento de software ofrece potencialidad para entender, mejorar y automáticamente ejecutar estos procesos. Cuando los procesos de desarrollo de software son informales, ingenieros de software tienen sus propias presunciones en cuanto a la forma de llevar a cabo el desarrollo de software. Estos procesos implícitos y descripciones de procesos informales son imposibles de analizar, difíciles de discutir, poco prácticos para enseñar y problemáticos de cambiar. Los procesos implícitos o descripciones informales pueden fingir entornos estables, pero cuando la empresa esté sometida a cambios importantes como: rotación de personal, dominio de aplicación, y/o tecnología; los problemas serán evidentes [20].

Los estándares de procesos software promueven el uso de procesos sistemáticos y de prácticas administradas para la ingeniería del software, permitiendo a las organizaciones entender, controlar y mejorar los procesos de desarrollo, con el fin de ayudar en el manejo y empleo de métodos que refuerzan la ingeniería del software como una disciplina de “ingeniería” en lugar de una “artesanía” [21].

### **1.2. ¿Qué es Tailoring?**

El Tailoring es definido como: “todas las actividades para borrar o modificar una parte del estándar de procesos, o para adicionar nuevas entidades<sup>1</sup> al estándar de procesos sin violar las dependencias entre entidades considerando la relación del entorno” [17]; también es definido como “El acto de adecuación y/o particularización de los términos de una descripción general para deducir una descripción aplicable a un entorno alterno (menos general)” [18]. En otras palabras, el Tailoring es el acto de adaptar un estándar de procesos software para satisfacer las necesidades de una organización o proyecto.

---

<sup>1</sup> ENTIDAD forma general de referirse a un artefacto, tarea, rol

La actividad de crear una descripción de procesos derivada de adaptar, y/o complementar los detalles de elementos de procesos o otras especificaciones incompletas de un proceso es la definición del Proceso de Tailoring [18].

### **1.2.1 Aplicaciones del Tailoring**

El Tailoring de procesos puede tomar lugar en dos diferentes niveles: el nivel organizacional y el nivel de proyecto [4]

#### **2.1.1.1 Tailoring a nivel organizacional**

Toma lugar en la definición y desarrollo del proceso software estándar organizacional (OSSP Organization's Standard Software Process), es decir, cuando no hay un proceso software definido en la Empresa; el esfuerzo es adaptar un estándar de procesos para generar un conjunto de requisitos adaptables para el OSSP de cada empresa.

#### **2.1.1.2 Tailoring a nivel de proyecto**

El Tailoring a nivel de proyecto se efectúa cuando la empresa tenga definido su OSSP, es decir, adaptar el OSSP en una definición de procesos que cumplan con las características de los proyectos que se ejecutan al interior de la empresa.

### **1.2.2 ¿Por qué es importante el Tailoring en las PyMEs\_DS ?**

Debido a que la industria del software en su mayoría está compuesta por PyMEs\_DS (Pequeñas y Medianas Empresas Desarrolladoras de Software) [22] con características limitadas en tamaño, tiempos y recursos, aplicar estándares de procesos software sin un ajuste adecuado puede incurrir en trabajar con un proceso inadecuado generando más complejidad, costos y tiempos de desarrollo, los cuales las PyMEs\_DS no pueden soportar.

Bajo esta problemática es necesario para las PyMEs\_DS adecuar los procesos desde patrones de procesos y estándares de desarrollo de software existentes como ISO/IEC 12207[14], IEEE/IEA 12007[16] y UP[10], porque sus contenidos son enormes y pueden fácilmente abrumar a las empresas. Además, cada proyecto es único en términos de su dominio de negocio, requerimientos, tecnología, etc. Cada organización exige ajustar su estructura y procesos para sus entornos internos y externos; así, cada proyecto necesita ajustar el estándar de procesos software antes de adoptarlos [15].

El Tailoring de procesos software en patrones y estándares de procesos puede efectivamente mejorar el funcionamiento de los procesos de trabajo asegurando la calidad de los productos finales, evitando riesgos y reduciendo el re-trabajo [19]. Es importante enfatizar que los ajustes a los procesos de software proporcionan a las empresas, prácticas convenientes que ayudarán adecuadamente al desarrollo de proyectos software.

Se puede concluir que el Tailoring es una actividad obligatoria, debido a que las definiciones formales de procesos, en su intento de ser aplicadas a múltiples contextos, son generales y definen procesos que no se ajustan a las necesidades de los proyectos de las PyMEs\_DS. Usar una buena definición de procesos es un

acercamiento ampliamente reconocido para incrementar la calidad y la productividad en el desarrollo de software [15].

### **1.3 Revisión de literatura**

A pesar que el Tailoring de procesos es una actividad obligatoria en muchas de las propuestas de procesos software la cantidad de investigación hecha sobre este tópico hasta la fecha puede ser considerada poca [4] [13]. En esta sección se realiza una revisión de los estudios afines con el Tailoring de procesos software encontrados en la literatura consultada.

#### **1.3.1 Tailoring and Verifying software Process [17].**

El ETRI (Institute Research Electronic and Telecommunications) y KAIST (Korea Advanced Institute of Science and Technology), proponen un método sistémico para formalizar claramente un estándar de procesos y un prototipo de herramienta para dar soporte a éste; el método se basa mediante módulos de procesos reusables encapsulados, para adaptar módulos de procesos y para verificar los procesos adaptados. El AAG (Activity Artifact Graph) es usado para representar y adaptar cada módulo de procesos, es una notación simple para modelar procesos estándares y procesos adaptados, la cual representa las relaciones entre actividades y artefactos del correspondiente módulo de procesos.

El método planteado contiene un modelo de procesos, que consiste de módulos de procesos definidos como: *una unidad reusable encapsulada de fragmentos de procesos*, y *cuatro operaciones para llevar a cabo la adaptación de los procesos* las cuales son:

- Adición: Es definida como la adición de una entidad al estándar de procesos, esta operación puede tener dos casos, adición de entidades definidas por el estándar de procesos y adición de entidades definidas por el usuario.
- Borrado: En esta operación se considera el borrado de artefactos y actividades del estándar de procesos teniendo en cuenta cómo afectará al estándar de procesos.
- División: Es definida como la división de actividades y artefactos en el estándar de procesos para tener una gestión detallada del proceso. Cuando se realiza la división sobre una actividad o artefacto la entidad original será borrada y las divisiones de esta deben conservar la naturaleza de la actividad o artefacto.
- Fusión: Esta operación es contraria a la división, no conserva la naturaleza original de las entidades, si antes de la fusión existe una relación de dependencia entre entidades éstas pueden ser removidas. En esta operación solo los artefactos intermedios son borrados.

Además, se proponen varias condiciones para asegurar la exactitud del proceso y una métrica para evaluar el cumplimiento del estándar; se describe una técnica de verificación estática que es muy importante para los procesos adaptados antes de su divulgación, la cual consiste en revisiones de correcciones sintácticas, revisiones de conformidad de tipo y evaluación de conformidad del estándar. El método formulado es útil en la adaptación de un estándar de procesos y la verificación del proceso adaptado.

### 1.3.2 Tailoring Software Evolution Process [19].

La escuela de ciencias de la información y Ingeniería de la universidad de Yunnan China, propone un método sistémico para el Tailoring de procesos de desarrollo de software enfocado en la adaptación de actividades, basado sobre el meta modelo EPMM (Evolution Process Meta-Model ) que tiene cuatro niveles de arquitectura: nivel global, nivel de procesos, nivel de actividades y nivel de tareas. Este método se basa sobre cuatro bloques básicos los cuales son: Secuencia, Concurrencia, Selección, e Iteración. Basado en estos bloques básicos se definen cuatro operaciones: adición, Borrado, división y fusión para efectuar el Tailoring del proceso.

#### Bloques Básicos

El término bloque básico denota una estructura PETRI NET (herramienta de modelado gráfico y matemático), sin marcas o etiquetas iniciales. Un bloque básico puede ser descrito como una 5-tupla así:

$B=(C, A; F, Ae, Ax)$  donde: C, A y F son el conjunto de condiciones, actividades y arcos respectivamente. Ae, Ax están contenidas en A y de denominan entrada y salida de b respectivamente.

#### Bloque de Secuencia

El bloque de secuencia está encerrado por líneas punteadas, describe actividades  $e_i$  y  $e_j$  las cuales son ejecutadas secuencialmente (ver Figura 1). Formalmente  $C=\{c\}$ ;  $A=\{e_i, e_j\}$ ;  $F=\{(e_i, c), (c, e_j)\}$ ;  $Ae=\{e_i\}$ ;  $Ax=\{e_j\}$ .

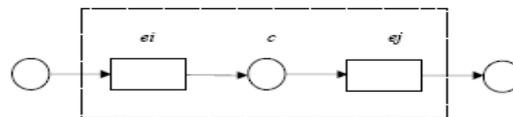


Figura 1 Bloque de Secuencia

#### Bloque de Concurrencia

El bloque de concurrencia está encerrado por líneas punteadas, describe actividades  $e_i$  y  $e_j$ , las cuales son ejecutadas concurrentemente (Ver Figura 2). Formalmente  $C=\{c_1, c_2, c_3, c_4\}$ ;  $A=\{e_0, e_i, e_j, e_n\}$ ;  $F=\{(e_0, c_1), (e_0, c_2), (c_1, e_i), (c_2, e_j), (e_i, c_3), (e_j, c_4), (c_3, e_n), (c_4, e_n)\}$ ;  $Ae=\{e_0\}$ ;  $Ax=\{e_n\}$ .

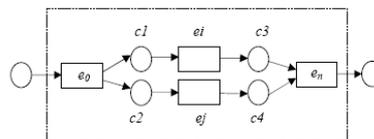
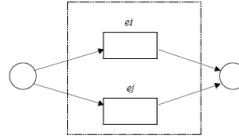


Figura 2 Bloque de Concurrencia

#### Bloque de Selección

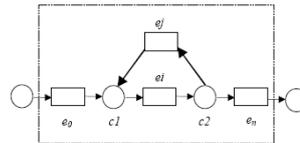
Este está encerrado por líneas punteadas, describe actividades  $e_i$  y  $e_j$ , las cuales son ejecutadas selectivamente (Ver Figura 3). Formalmente  $C=\{\}$ ;  $A=\{e_i, e_j\}$ ;  $F=\{\}$   $Ae=\{e_i\}$ ;  $Ax=\{e_j\}$ .



**Figura 3 Bloque de Selección**

**Bloque de Iteración**

Describe actividades  $e_i$  y  $e_j$ , las cuales son ejecutadas repetidamente (Ver Figura 4). Formalmente  $C=\{c_1, c_2\}$ ;  $A=\{e_0, e_i, e_j, e_n\}$ ;  $F=\{(e_0, c_1), (c_1, e_i), (e_i, c_2), (c_2, e_j), (e_j, c_1), (c_2, e_n)\}$ ;  $Ae=\{e_0\}$ ;  $Ax=\{e_n\}$ .



**Figura 4 Bloque de Iteración**

**Operaciones de Adaptación de Procesos**

Basado en los cuatro bloques básicos se definen cuatro operaciones llamadas adición, Borrado, división y fusión.

**Operación de Adición**

La operación de adición está definida como la adición de un bloque básico dentro del correspondiente proceso de desarrollo software. Se considera que el proceso de adición de actividades no viola el estándar de conformidad [17]. Cuando un diseñador de procesos desea adicionar una actividad dentro de un modelo de procesos de desarrollo de software, tiene que considerar dónde la actividad es adicionada y qué tipo de relación básica hay entre la nueva actividad adicionada y la actividad adyacente.

La Figura 5 ilustra un bloque de secuencia mostrado en la Figura 1 después de adicionar la actividad **ep**.



**Figura 5 Adición de la actividad ep dentro de un bloque de secuencia**

**Operación de Borrado**

La operación de borrado especifica cómo eliminar una actividad del correspondiente proceso de desarrollo de software. Esta operación puede ser la más común del proceso de adaptación. Para eliminar una actividad de un modelo de procesos de desarrollo de software, se debe considerar qué actividades son afectadas por la eliminación.

**Operación de División**

La operación de división está definida como la división de una actividad en el correspondiente proceso de desarrollo de software dentro de un bloque básico. Si un

proyecto es grande, se puede usar esta operación para gestionar actividades en detalle. Cualquier actividad en un proceso de desarrollo de software puede ser reemplazada por un bloque básico. El nivel inferior de un bloque básico es el detalle del nivel alto de una actividad. Así, un proceso de desarrollo de software puede ser construido nivel por nivel, de tal forma que el proceso más refinado puede ser conseguido continuamente con el aumento de profundidad hasta estar satisfecho con la granularidad del proceso.

### Operación de Fusión

La operación de Fusión está definida como la fusión de un bloque básico en el correspondiente proceso de desarrollo de software dentro de una actividad. Si un proyecto es muy pequeño, se puede usar esta operación para ignorar los detalles de algunas actividades. Cualquier bloque básico en el proceso de desarrollo de software puede ser reemplazado por una actividad. El nivel alto de una actividad es una abstracción especificación abstracta del nivel inferior de un bloque básico.

### 1.3.3 Modelo de Adaptación de procesos software (MAPS) [25].

Es un modelo genérico para la adaptación de procesos software, este modelo propone los siguientes pasos para hacer la adaptación del proceso a las características de los proyectos, ver Figura 6.

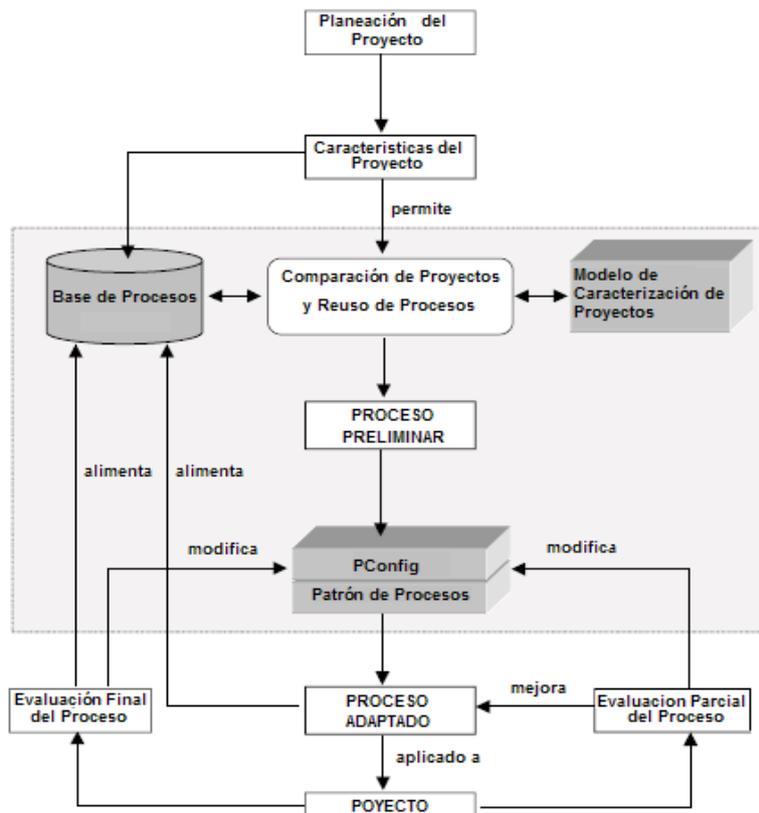


Figura 6 Modelo de Adaptación de Procesos.

1. Determinación de características. Aquí se identifican las características del proyecto que pueden ser: de desarrollo, restrictivas y las que priorizan el proyecto.
2. Comparación de proyectos. Las características del proyecto sirven de base para identificar proyectos similares, almacenados en una base de procesos, utilizando un modelo de comparación por características del proyecto.
3. Reutilización de los procesos. Se utiliza una lista de proyectos identificados como semejantes con el proyecto actual y se realiza una evaluación de los procesos utilizados en previos proyectos para brindar al ejecutor de la adaptación opciones para decidir qué procesos serán reutilizados.
4. Complementar el proceso. La reutilización de partes de los procesos en proyectos anteriores puede dar un proceso preliminar; el cual puede ser incompleto. Para terminar el proceso de desarrollo de software se utiliza el patrón de procesos y un proceso de configuración (PConfig), que ayudan a terminar el proceso de adaptación de aquellas partes del proceso que no fueron reutilizadas. El resultado de complementar el proceso es un proceso completo que será utilizado para desarrollar el proyecto.
5. Evaluación del Proceso. Periódicamente, durante el proyecto, se realizan evaluaciones parciales del proceso; las evaluaciones se pueden hacer por ejemplo, al final de cada iteración. Dentro de las evaluaciones del proceso se puede incluir evaluaciones parciales o mejoras en el proceso de adaptación, y también se pueden realizar evaluaciones finales al proceso.
6. Alimentación de la base de procesos. La evaluación final del proceso y del propio proceso adaptado, con las mejoras, se almacena junto con las características del proyecto, en la base de procesos, de modo que el proceso puede ser reutilizado posteriormente.

#### **1.3.4 Process Tailoring and the Software Capability Maturity Model (CMM) [18].**

El modelo de madurez de capacidades SW-CMM (Software Capability Maturity Model) desarrollado por el SEI (Software Engineering Institute), es reconocido por su valiosa contribución hacia la mejora de procesos software; El SW- CMM está compuesto de dos volúmenes: El Modelo de Madurez de Capacidades y las Prácticas Claves del Modelo de Madurez de Capacidades. El primer volumen contiene una descripción de los cinco niveles del modelo y una definición operacional. El segundo volumen contiene prácticas claves que corresponden a las áreas claves de procesos KPAs (key Process Areas) en cada nivel de madurez del modelo.

Las prácticas claves del SW-CMM están expresadas en términos generales; de esta manera pueden ser usadas en una amplia variedad de entornos. La naturaleza general del modelo y su presunción específica requiere que cada organización ajuste a su medida las prácticas claves a su propio entorno específico, necesidades de negocio y objetivos.

Por estas razones, la población de producción de software y organizaciones necesitan interpretar y/o adaptar las prácticas claves de este modelo antes de ser aplicadas. Este reporte técnico explora algunas de las áreas donde el ajuste de las prácticas de SW-CMM probablemente sean requeridas y sugiere como prerrequisito para la utilización de las prácticas claves determinar las similitudes y diferencias entre el

entorno expresado en la terminología de SW-CMM y el entorno de la organización. Además, provee un framework completo y una guía para el tailoring de procesos en organizaciones certificadas en CMM. Identifica los diferentes contextos donde el tailoring es usado y asocia actividades de tailoring para el proceso de administración de software; también describe los puntos claves que deben ser considerados antes de adaptar las prácticas en el SW-CMM y explora la naturaleza de varios tipos de tailoring usados en la definición y desarrollo de descripciones de procesos software.

### 1.3.5 A Matrix Approach to Software Process Definition (Centro de Vuelos Espaciales Goddard) [23].

El Laboratorio de Ingeniería de Software (SEL) realizó un trabajo para el Centro de Sistemas de Información (ISC) del Centro de Vuelos Espaciales Goddard (GSFC). Este trabajo incluyó, entre otras cosas, ayudar al equipo de desarrollo de software GFSC en la selección y adaptación de procesos de software. Uno de los objetivos era que los procesos fueran compatibles con las normas ISO 9001 y CMM.

Los proyectos fueron clasificados en nuevo desarrollo, mantenimiento, alta reutilización y prototipos. Además de estos tipos de proyectos se definieron tres criterios para la adaptación:

- Tamaño del equipo de desarrollo: pequeño, mediano, grande.
- Sistemas críticos, sistemas no críticos.
- Cronograma: lento, normal, rápido

Los procesos software están constituidos de actividades, organizadas en grupos de actividades. Cada actividad utiliza un conjunto de técnicas. Como se muestra en la Figura 7.

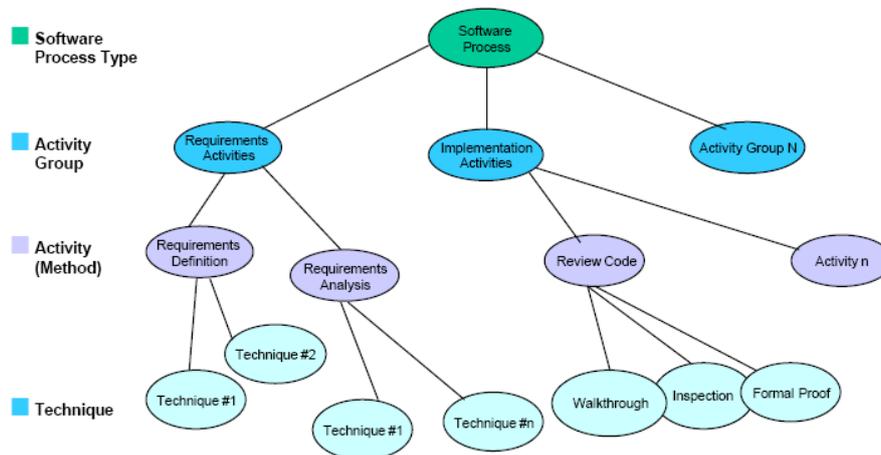


Figura 7 Modelo conceptual de una Metodología.

Para especificar en qué condiciones, determinadas actividades no se deben realizar, se utiliza un lenguaje estructurado, una especie de pseudocódigo. El problema es que cada especificación en pseudocódigo corresponde a una única instancia del proceso. Dado que hay cuatro tipos de procesos y tres criterios de adaptación; como dos de estos criterios pueden tomar tres valores distintos, y el tercero puede tomar dos

valores diferentes, sería necesario setenta y dos especificaciones diferentes (4x3x3x2) para cubrir todos las instancias posibles para este proceso. Este gran número de especificaciones hace que sea difícil el desarrollo y el análisis de los procesos.

Para resolver este problema, se adoptó un enfoque basado en una matriz de representación de los procesos Figura 8. La columna de la izquierda de la matriz identifica el pseudocódigo y los pasos que serán seguidos. Las otras columnas representan las diversas combinaciones de criterios de adaptación. Cada actividad se identifica como obligatoria u opcional y las revisiones se han identificado como formal o informal.

**Legend:** X = Perform this activity  
 O = This review or activity is optional but recommended  
 Rv (in 1st column) = Review  
 This font and shading indicates logic

F = Perform this review formally  
 I = Perform this review informally  
 \* (in last column) = This activity is associated with an ISO "shall"  
 This font and no shading is used for activities

X.0 Activity Group	X.X Major Activity	Tailoring Drivers	Critical Software				Non-Critical Software				ISO	
			Normal Schedule		Aggressive Schedule		Normal Schedule		Aggressive Schedule			
			Small/Medium Team	Large Team	Small/Medium Team	Large Team	Small/Medium Team	Large Team	Small/Medium Team	Large Team		
	Determine the selection criteria for COTS/GOTS software.		Perform for all project types									
	Identify potential COTS/GOTS candidates.		Perform for all project types									
	Obtain demo versions and evaluate the candidate COTS/GOTS.		X	X	X	X	X	X	O	O		
	Select COTS/GOTS.		Perform for all project types									
	Document the selection criteria and evaluation results.		X	X	X	X	X	X	O	X		
	Document the risks (cost, schedule, reliability, etc.) associated with use of the selected COTS/GOTS.		X	X	X	X	X	X	O	X		
1.4	Requirements Documentation and Review		Always perform this activity									
	Document requirements.		Always perform this activity									
	IF the software to be developed is part of a larger system or project (e.g., a spacecraft or instrument and high level requirements were provided), THEN											
Rv	Conduct a Software Specifications Review (a.k.a. Software Requirements Review). (SSR)		I	F				O	O		*	
	Conduct a combined Software Specifications Review/Preliminary Design Review (SSR/PDR)				I	F			O	O	*	
	ELSE (a complete software system is to be developed and high level requirements were developed)											
Rv	Conduct a System Requirements Review (SRR)		I	F				O	O		*	
	Conduct a combined System Concept Review/System Requirements Review (SCR/SRR)				I	F			O	O	*	

**Figura 8 Matriz de representación de procesos.**

El enfoque basado en la matriz de desarrollo, compara, presenta y revisa los procesos. Además, abre el camino para que en un futuro la automatización sea posible, a partir de la selección del tipo de proceso y los valores de los criterios de adaptación, generando una lista de actividades para el proyecto.

### 1.3.6 Disciplina de ambiente de RUP [10]

La disciplina de ambiente de RUP describe a grandes rasgos la configuración del proceso software, sin embargo, no define claramente cómo las disciplinas deben ser ajustadas de acuerdo a las características de los proyectos, limitándose solo a brindar información general que puede ser útil, pero no es suficiente para guiar la adaptación. Según la disciplina de ambiente de RUP, diversos factores influyen la adaptación del proceso; estos factores son ilustrados a continuación:

### **Contexto del negocio**

Según la disciplina de ambiente existen diferentes tipos de contextos del negocio que afectan la configuración del proceso. Ejemplos de estos contextos son:

- Contrato de trabajo: el software es producido para un cliente específico.
- Desarrollo comercial o especulativo: este tipo de contexto sucede cuando el desarrollador del producto cubre los costos de poner el software en el mercado.
- Proyectos internos: donde el cliente y el desarrollador son la misma organización, el producto es desarrollado para el uso de la propia organización.

### **Tamaño del esfuerzo en el desarrollo**

El tamaño de un proyecto puede ser descrito por algunas métricas como, LOC (líneas de código) y PM (puntos de función). Cuanto mayor es el proyecto, mayor es el tamaño de equipo y, en consecuencia, el nivel de formalidad es mayor. La participación de más personas en un equipo implica una forma más formal de comunicación y, por tanto, un proceso más pesado.

### **Grado de innovación**

El grado de innovación se refiere a la experiencia de la organización con el proceso de desarrollo y el tipo de producto que se desarrollará. Un nuevo proyecto, sin antecedentes, requiere más atención en las fases de inicio y elaboración. Es necesario hacer mayor énfasis en la licitación de requisitos.

### **Tipo de aplicación**

El tipo de aplicación afecta el proceso de desarrollo, ya que cada tipo de sistema impone diferentes restricciones en el desarrollo. Estas restricciones se refieren a la complejidad técnica del software, restricciones de tiempo, de desempeño, necesidad de cumplir con normas, criticidad del sistema. Una aplicación crítica por ejemplo, requiere un mayor grado de formalidad en el desarrollo.

### **Factores de la organización**

Para establecer un proceso en una organización, se debe tener en cuenta aspectos como la estructura organizativa y la cultura, habilidades y actitudes de las personas implicadas y la previa experiencia de la organización.

### **La complejidad técnica y de gestión**

En general, la necesidad de un proceso más formal crece de acuerdo a la complejidad técnica y de gestión del proyecto. La complejidad de gestión afecta directamente el nivel de la ceremonia del proceso debido al aumento de las revisiones formales. A mayor complejidad técnica, a su vez, aumenta el número de herramientas, técnicas y funciones especializadas, generando un número mayor de actividades.

#### **1.3.7 Tailoring a Software Process for Software Project Plans [26].**

El laboratorio Charles Stark Draper en Cambridge, preparó un informe para organizaciones que necesitan adaptar un modelo de proceso de software a las necesidades específicas de proyectos software. En concreto, incluye organizaciones de software que tienen previsto cumplir los requisitos del SEI CMM Nivel 3.

Este informe aborda los problemas más evidentes que surgen en la adaptación de un conjunto de bloques de construcción (building blocks) a las características exclusivas de un proyecto de software.

Los proyectos software tienen necesidades únicas. Los productos (o modificaciones a los productos) resultados de un proyecto por lo general son únicos. Tradicionalmente, los aspectos singulares de un proyecto más que un conjunto común de bloques de construcción ha sido la fuerza detrás de la planificación en el desarrollo de software.

Uno de los factores que apoya el desarrollo y la utilización del proceso software estándar organizacional es el uso de bloques de construcción comunes para todos los proyectos dentro de la organización, el cual ayudará a hacer la planificación y la gestión más fácil, mediante el apoyo del uso de materiales, métodos y la experiencia adquirida en anteriores proyectos.

Sin embargo, dada la naturaleza singular de la mayoría de los proyectos de software, algunas adaptaciones necesitan ser hechas al proceso software estándar organizacional para su aplicación a nuevos proyectos. Los pasos utilizados para adaptar el proceso software organizacional a las necesidades de un proyecto concreto será algo diferente para cada organización. Sin embargo, los siguientes conceptos se pueden utilizar para ayudar a considerar cómo desarrollar la adaptación del proceso software a un proyecto.

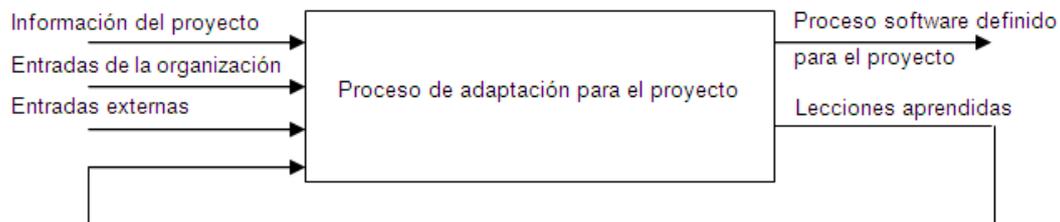
En la definición de un nuevo proyecto software se suele incluir la siguiente información:

- Los objetivos del proyecto.
- Requerimientos técnicos del proyecto
- Desempeño de requisitos y normas.

Esta información se combina con las entradas de la organización y las entradas externas, tales como:

- Los objetivos del negocio de la organización.
- Proceso software estándar de la organización.
- Reglas para adaptar el proceso software estándar de la organización.
- Adquisición de reglamentos, y
- La adquisición de la filosofía de gestión de la organización.

En su forma más simple, la adaptación de un proceso software estándar organizacional para un determinado proyecto de software puede ser como se muestra en la Figura 9.



**Figura 9 Resumen de la adaptación del proceso software.**

Este informe comprende la adaptación para un conjunto pequeño de áreas que tienen una alta rentabilidad en términos de coste y calendario. Las características que rigen el método de adaptación descrito en este informe es el siguiente:

- Tamaño y complejidad del Proyecto.
- Formalidad.
- Control.

### **1.3.8 SPEM 2.0 (Software and Systems Process Engineering Meta-model) [12].**

En la industria del software hay suficientes ideas y conocimiento disponible acerca de cómo desarrollar software efectivamente. Actualmente, los equipos de desarrollo necesitan y tienen acceso a una amplia variedad de información. No solo se necesita tener información detallada de las tecnologías de desarrollo específicas como Java, Java EE, Eclipse, SOA Technologies, .NET, etc. si no también se necesita entender cómo organizar el trabajo a lo largo de las mejores prácticas del desarrollo de software moderno como son el desarrollo ágil, iterativo, centrado en la arquitectura, gestión de riesgos y calidad.

Algunos problemas que enfrentan las organizaciones de desarrollo de software cuando dejan a sus desarrolladores gestionar tal información por ellos mismos son:

- Los miembros del equipo no tendrán acceso fácil y centralizado al mismo cuerpo de información cuando ellos la necesiten. Por ejemplo, diferentes desarrolladores pueden depender de fuentes y versiones diferentes de la misma información
- Es difícil combinar e integrar contenido de procesos de desarrollo que están disponibles en su propio formato, como libros y publicaciones que representan contenido de los métodos y procesos usando una representación y estilo diferente.
- Es difícil definir un método de desarrollo organizado y sistemático que sea de tamaño adecuado a las necesidades, por ejemplo, tener en cuenta la cultura específica, prácticas estandarizadas, y el cumplimiento de requerimientos.

SPEM 2.0<sup>2</sup> es un meta modelo de ingeniería de procesos así como también un framework conceptual, el cual puede proveer conceptos necesarios para modelar, documentar, presentar, administrar, intercambiar e implementar procesos y métodos de desarrollo. La implementación de este meta modelo podría estar dirigida a

<sup>2</sup> <http://www.omg.org/cgi-bin/doc?ptc/07-08-07>

ingenieros de procesos, líderes de proyectos, gestores de proyectos quienes son responsables por el mantenimiento y implementación de procesos para la organización de desarrollo o proyectos individuales.

SPEM 2.0 sirve para representar procesos de ingeniería de software, entendiendo un Proceso Software (PS) como “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). SPEM 2.0 se encuadra dentro de la Ingeniería de Procesos de Software (Software Processes Engineering - SEP), que es un área nueva de la Ingeniería de Software dedicada a la definición, implementación, medición y mejora de los procesos de Ingeniería de Software.

Gracias al uso de SPEM 2, se puede disponer de modelos de PS en formato procesable por computador, lo que 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.

SPEM es a los procesos software lo mismo que UML es a los sistemas software. UML es un metamodelo que sirve para representar modelos de sistemas software y SPEM es un metamodelo que sirve para representar modelos de procesos software.

Cuando se trabaja con SPEM existen cuatro escenarios fundamentales que se explican a continuación, ver Figura 10.

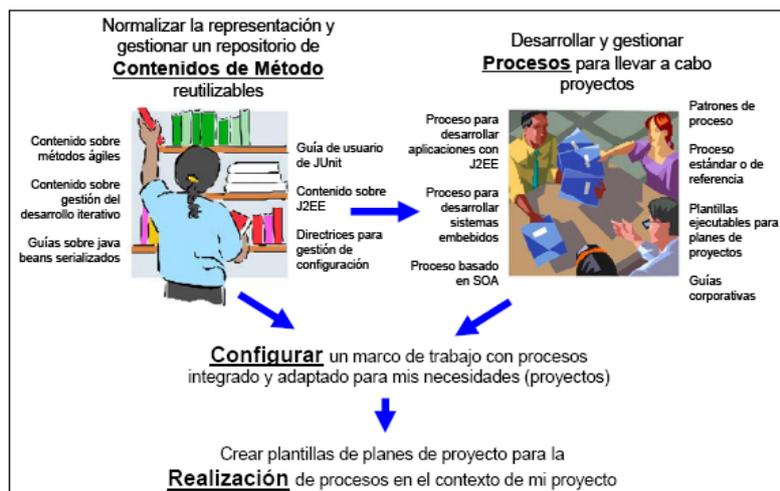


Figura 10 Escenarios de trabajo de SPEM.

- Crear un repositorio de “contenidos de método” reutilizables, es decir, una colección organizada de roles, tareas, productos de trabajo, guías, fragmentos

de método y procesos, etc. para ayudar a los profesionales del desarrollo en el establecimiento de una base de conocimiento de activos intelectuales para el desarrollo de software y sistemas que les permitan gestionar y desplegar su contenido con un formato uniforme. Esto en sí solo ya es un valor para cualquier organización software, ya que supone disponer de un repositorio de conocimiento sobre procesos en un formato estandarizado. Además, es de gran ayuda a los desarrolladores de software porque en su trabajo necesitan conocer cómo hacer las tareas de desarrollo y mantenimiento de software, cómo gestionar el proyecto, cómo comprender los productos de trabajo que se deben crear en cada tarea, cuales son las habilidades requeridas en cada rol, y disponer de las guías, directrices, plantillas, etc. adecuadas en cada momento. Además, el repositorio basado en SPEM es una base de conocimiento ideal para la formación en procesos.

- Dar soporte al desarrollo, gestión y crecimiento de procesos software. Los equipos de desarrollo necesitan definir la forma de aplicar sus métodos de desarrollo y mejores prácticas a lo largo del ciclo de vida del proyecto. En otras palabras, es necesario definir o seleccionar un proceso de desarrollo. Esto implica combinar, reutilizar y extender los elementos de método anteriores para configurar los procesos que sirven para guiar los proyectos. Desde fragmentos de proceso elementales se puede llegar a generar todo un proceso completo o toda una metodología, incluyendo varios procesos. SPEM 2 ayuda a los equipos de desarrollo a definir o seleccionar un proceso, incluyendo opciones para que los mismos métodos puedan ser aplicados de forma diferente en momentos distintos o en proyectos distintos, para comprender claramente como se relacionan unas tareas con otras, o para que los procesos puedan ser vistos como flujos de trabajo o como estructuras de desglose de trabajo (WBS), según interese en cada momento. Para lo anterior, SPEM soporta la creación sistemática de procesos basada en la reutilización de contenidos de método. También provee el fundamento conceptual para que los ingenieros de procesos y gestores de proyectos seleccionen, adapten y rápidamente ensamblen procesos para sus proyectos concretos. El ensamblado rápido de procesos es posible gracias a la implementación de catálogos de procesos predefinidos, “trozos” de procesos o patrones de proceso.
- Establecer un marco de trabajo general de la organización a partir de los procesos y los elementos definidos anteriormente. Para ello, SPEM permite dar soporte al despliegue del contenido de método y proceso que justo se necesita en cada caso, teniendo en cuenta que ningún proyecto es exactamente como el anterior y nunca exactamente el mismo proceso software se ejecuta dos veces. En este punto es importante recordar que el nivel 3 de CMMI (proceso definido) necesita disponer de procesos estándares en la organización y de mecanismos de particularización (tailoring); SPEM 2 provee de ambas cosas. Entre otras capacidades adicionales, SPEM 2 incorpora conceptos para:
  - Reutilización de procesos o patrones de procesos,
  - Variabilidad (procesos que incluyen partes alternativas configurables), y

- Particularización (los usuarios definen sus propias extensiones, omisiones y puntos de variabilidad sobre procesos estándares reutilizados)
- Generar plantillas para planes de proyecto concretos. Esto supone que a partir de ahora los jefes de proyecto pueden contar con mucha más información y disponible de manera automática a la hora de definir los planes de los proyectos. Para darles auténtico valor, las definiciones de los procesos deben ser desplegadas en formatos que permitan su realización automática (sistemas de gestión de proyectos y recursos, motores de flujos de trabajo). Para ello, SPEM incluye estructuras de definición de procesos que permiten expresar cómo un proceso será realizado de forma automática con estos sistemas. Ejemplos de ello son las iteraciones (una o varias definiciones de trabajo serán repetidas varias veces en un proyecto) y las ocurrencias múltiples (varias instancias de una definición de trabajo pueden llevarse a cabo a la vez de forma paralela).

### 1.3.9 La plataforma EPFC [27].

Eclipse Process Framework Composer (EPFC)<sup>3</sup>, es una herramienta gratuita y de fuente abierta, desarrollada dentro del entorno ECLIPSE, ver Figura 11 que sirve para documentar y publicar fragmentos de método, procesos o metodologías, y generar automáticamente la documentación adecuada en formato para la web. Dichos fragmentos se almacenan en formato XMI<sup>4</sup> y, al estar basados en el estándar SPEM 2.0, pueden ser reutilizados por cada vez más herramientas CASE. En realidad, EPFC utiliza la "Unified Method Architecture" (UMA), que a su vez está basada en el metamodelo SPEM 2.

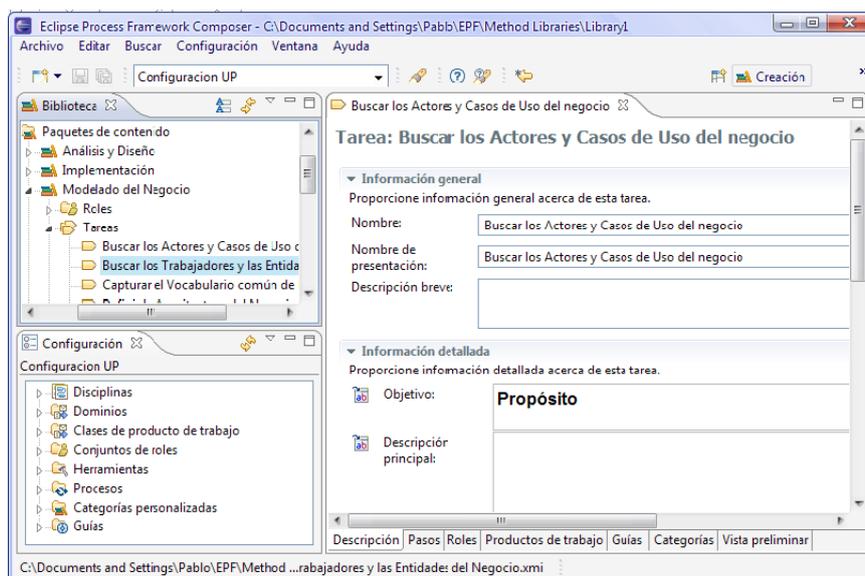


Figura 11 Entorno del EPFC

<sup>3</sup> <http://www.eclipse.org/epf/>

<sup>4</sup>XMI es un acrónimo para el intercambio de Metadatos XML.

El EPFC es una herramienta para ingenieros de procesos, líderes y administradores de proyectos, quienes son responsables de mantener e implementar procesos para organizaciones dedicadas al desarrollo o para proyectos individuales. Hay dos problemas típicos que deben ser tratados a la hora de llevar a cabo la implementación de un proceso.

- Primero, los desarrolladores deben entender los métodos y las prácticas del desarrollo de software. Ellos deben estar familiarizados con las tareas básicas del desarrollo, tales como la licitación y manejo de requisitos, el análisis y el diseño, etc. Para lograr esto muchas empresas se basan en la documentación explícita de los procesos y la enseñanza de tales métodos para establecer prácticas comunes y reguladas.
- Segundo, los equipos de desarrollo deben definir cómo se aplican sus métodos de desarrollo y mejores prácticas a través del ciclo de vida de un proyecto, es decir, ellos deben definir o seleccionar un proceso de desarrollo. Además deben entender claramente cómo las diferentes tareas definidas en los métodos se relacionan entre sí: Por ejemplo, cómo el método de administración de cambios impacta el método de administración de requisitos. Incluso los equipos de trabajo dentro de una misma organización necesitan definir un procesos que les de alguna guía sobre cómo será delimitado el desarrollo a través del ciclo de vida, cuándo se lograrán y verificarán los hitos<sup>5</sup>, etc.

El proyecto de Eclipse Process Framework tiene como objetivo satisfacer algunas necesidades comunes que los líderes y equipos de desarrollo enfrentan cuando están asimilando y administrando métodos y procesos. Por ejemplo:

- Los equipos de desarrollo necesitan un acceso fácil y centralizado a la información.
- Los contenidos de los procesos de desarrollo deben estar en formatos estándar que permitan una fácil integración.
- Los equipos de desarrollo requieren una base de conocimiento actualizada para que ellos mismos aprendan sobre métodos y mejores prácticas.
- Los equipos de desarrollo necesitan soporte para dimensionar correctamente sus procesos.
- Los equipos de desarrollo requieren la habilidad de estandarizar prácticas y procesos dentro de las organizaciones.
- Los equipos de desarrollo necesitan cerrar la brecha entre la ingeniería de procesos y el establecimiento de los mismos en las organizaciones por medio del uso de representaciones y terminologías similares.

---

<sup>5</sup> MILESTONE es un punto de verificación de cumplimiento de los objetivos que puede coincidir con el final de una fase.

## **CAPITULO 2 ADAPTACIÓN DE UP**

### **2.1 EL Proceso Unificado (UP) [10].**

#### **2.1.1 ¿Qué es el Proceso Unificado (UP Unified Process)?**

El proceso unificado en adelante UP (Unified Process), es un proceso de ingeniería del software, provee un método disciplinado para asignar tareas y responsabilidades dentro de una organización de desarrollo de software, su objetivo es asegurar la producción de software de alta calidad que reúna las necesidades de los usuarios finales, dentro de una planificación y presupuesto predecible [9][11].

Un proceso de software se define como: “un conjunto de actividades, métodos, prácticas y transformaciones que las personas usan para desarrollar y mantener software y sus productos asociados (p.ej., planes, especificaciones, diseños y pruebas) [10].

El UP es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyectos, el cual debe ser adaptado para específicas organizaciones o proyectos.

#### **2.1.2 Mejores Prácticas**

UP identifica 6 mejores prácticas, (desarrollo de software iterativo, gestión de Requerimientos, desarrollo basado en componentes, modelado visual, verificación de la calidad del software, gestión de los cambios), con las que define una forma efectiva de trabajar para los equipos de desarrollo de software.

##### **2.1.2.1 Desarrollo de software iterativo**

Debido a la complejidad de los sistemas software, no es posible definir totalmente el problema, diseñar una solución y luego construir el software para probar el producto final. Un método iterativo es necesario, que permita un entendimiento y solución progresiva y efectiva del problema, a través de múltiples iteraciones con sucesivos refinamientos. Un método iterativo hace más fácil adecuarse a los cambios estratégicos en los requerimientos, características o en la planificación [3]. UP soporta un método iterativo de desarrollo que direcciona ítems de alto riesgo en cada fase del ciclo de vida, esta característica permite reducir significativamente el riesgo.

##### **2.1.2.2 Gestión de Requerimientos.**

UP brinda una guía para encontrar, organizar, documentar y seguir los cambios de los Requerimientos funcionales y restricciones; utiliza una notación de Casos de Uso y escenarios que han probado ser una excelente forma de capturar requerimientos funcionales que conducen el diseño, implementación y pruebas del software haciendo probable que el sistema final cumpla a cabalidad con las necesidades del usuario final. Los casos de uso y escenarios proveen un hilo coherente de trazabilidad a través del desarrollo y la entrega del sistema [2].

### **2.1.2.3 Desarrollo basado en componentes**

La creación de sistemas intensivos en software requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema; esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o se desarrollan sus componentes.

### **2.1.2.4 Modelado visual**

UML (Unified Modeling Language) es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema software. Utilizar herramientas de modelado visual facilita la gestión de dichos modelos, permitiendo ocultar o exponer detalles cuando sea necesario; el modelado visual también ayuda a mantener la consistencia entre los artefactos del sistema: Requerimientos, diseños e implementaciones. En resumen, el modelado visual ayuda a mejorar la capacidad del equipo para gestionar la complejidad del software, donde UML es la base del éxito [1].

### **2.1.2.5 Verificación de la calidad del software.**

La calidad debe ser revisada con respecto a los requerimientos en base a la fiabilidad, funcionalidad y desempeño del sistema. UP brinda mecanismos para planear, diseñar, implementar, ejecutar y evaluar este tipo de pruebas.

Es importante que la calidad de todos los artefactos se evalúen en varios puntos durante el proceso de desarrollo, especialmente al final de cada iteración. En esta verificación las pruebas juegan un papel fundamental y se integran a lo largo de todo el proceso; para todos los artefactos no ejecutables las revisiones e inspecciones también deben ser continuas.

### **2.1.2.6 Gestión de los cambios**

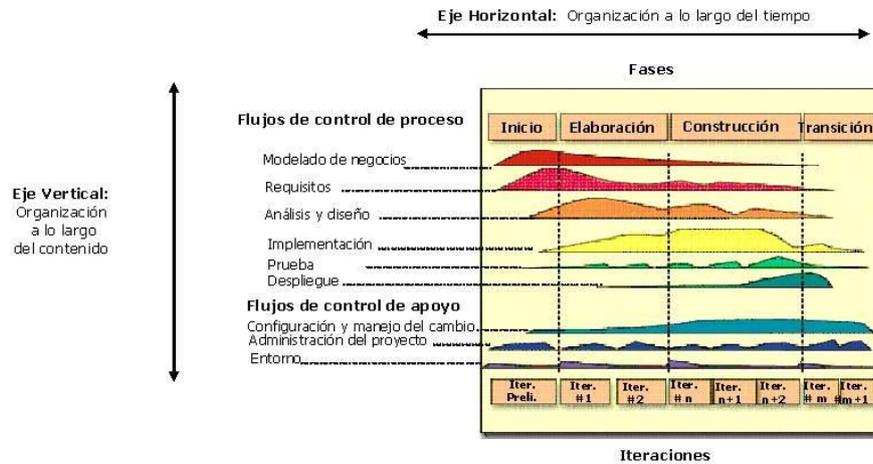
Asegurarse de que cada cambio es admisible, y ser capaz de seguir los cambios, es esencial en un entorno donde el cambio es inevitable [9]. UP describe cómo controlar, rastrear y monitorear los cambios para permitir un desarrollo iterativo exitoso. El cambio es un factor de riesgo crítico en los proyectos de software. Los artefactos software cambian no sólo debido a acciones de mantenimiento posteriores a la entrega del producto, sino que durante el proceso de desarrollo, especialmente importantes por su posible impacto son los cambios en los Requerimientos.

## **2.1.3 Dos dimensiones**

El proceso puede ser descrito en dos dimensiones o ejes [8]:

Eje horizontal: representa el tiempo y es considerado el eje de los aspectos dinámicos del proceso. Indica las características del ciclo de vida del proceso expresado en términos de fases, iteraciones e hitos. Se puede observar en la Figura 12 que UP consta de cuatro fases: inicio, elaboración, construcción y transición. Cada fase se subdivide a la vez en iteraciones.

Eje vertical: representa los aspectos estáticos del proceso. Describe el proceso en términos de componentes de proceso, disciplinas, flujos de trabajo, actividades, artefactos y roles.



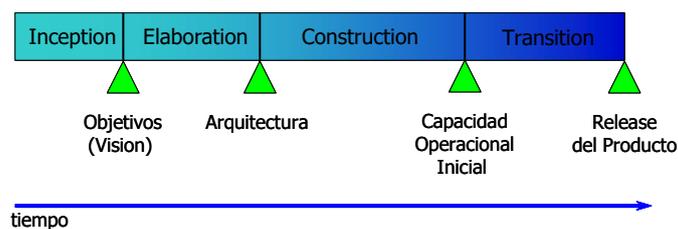
**Figura 12 Estructura de UP**

**Eje horizontal fases e iteraciones – La Dimensión dinámica del proceso.**

Eje horizontal es la organización dinámica del proceso a lo largo del tiempo. UP se repite a lo largo de una serie de ciclos que constituyen la vida de un producto. Cada ciclo concluye con una generación del producto para los clientes. Cada ciclo consta de cuatro fases consecutivas [6]:

- Fase Inicio.
- Fase Elaboración.
- Fase Construcción.
- Fase Transición.

Cada fase se concluye con un hito bien definido. Los hitos para cada una de las fases son: Inicio – *objetivos (visión)*, Elaboración - *Arquitectura*, Construcción – *Capacidad operacional inicial*, Transición – *versión del producto*. Las fases y sus respectivos hitos se ilustran en la Figura 13.



**Figura 13 Fases e hitos en UP**

**Fase de Inicio.**

Durante la fase de inicio se define el modelo del negocio y el alcance del proyecto. Para cumplir con esto se deben identificar todas las entidades externas con las cuales el sistema interactúa (actores) y se define la naturaleza de la interacción a un alto nivel. Esto involucra identificar todos los casos de uso y describir los más importantes. También se desarrolla, un plan de negocio para determinar qué recursos deben ser

asignados al proyecto y un plan que muestre las fechas importantes de los principales hitos [7].

Los resultados de la fase de inicio son [8]:

- Un documento de visión: una visión general de los requerimientos del proyecto, características clave y restricciones principales.
- Modelo inicial de casos de uso (10-20% terminado).
- Un glosario inicial: terminología clave del dominio.
- El caso de negocio.
- Lista de riesgos y plan de contingencia.
- Plan del proyecto, mostrando fases e iteraciones.
- Modelo de negocio, si es necesario.
- Prototipos exploratorios para probar conceptos o la arquitectura candidata.

Al final de la fase de inicio se debe alcanzar el hito: objetivos del proyecto.

### **Fase de Elaboración.**

El propósito de la fase de elaboración es analizar el dominio del problema, establecer los cimientos de la arquitectura, desarrollar el plan del proyecto y eliminar los riesgos mayores del proyecto. Para cumplir con estos objetivos, se debe tener una buena visualización de lo que será el sistema. Para la toma de decisiones arquitecturales se debe tener una comprensión completa del sistema: su alcance, principales requerimientos funcionales y no funcionales.

En esta fase se construye un prototipo de la arquitectura, que debe evolucionar en iteraciones sucesivas hasta convertirse en el sistema final; este prototipo debe contener los Casos de Uso críticos identificados en la fase de inicio; también debe demostrarse que se han evitado los riesgos más graves.

Al final de la fase de elaboración la arquitectura de referencia debe demostrar que soportará las funcionalidades claves del sistema y exhibir un buen comportamiento en términos de desempeño, escalabilidad y costo.

Al terminar deben obtenerse los siguientes resultados [8]:

- Un modelo de casos de uso (completo al menos hasta el 80%): todos los casos y actores identificados, la mayoría de los casos de uso han de ser desarrollados.
- Requerimientos adicionales que capturan los Requerimientos no funcionales y cualquier requisito no asociado con un caso de uso específico.
- Descripción de la arquitectura software.
- Un prototipo ejecutable de la arquitectura.
- Lista de riesgos y caso de negocio revisados.
- Plan de desarrollo para el proyecto.
- Un caso de desarrollo actualizado especificando el proceso a seguir.
- Un manual de usuario preliminar (opcional).

Al final de la fase de elaboración se debe alcanzar el hito: arquitectura del proyecto.

### **Fase de Construcción.**

La finalidad principal de esta fase es alcanzar la capacidad operacional del producto de forma incremental a través de las sucesivas iteraciones. Durante esta fase todos los componentes, características y Requerimientos deben ser implementados, integrados y probados en su totalidad, obteniendo una versión aceptable del producto.

Los resultados de la fase de construcción deben ser [8]:

- Modelos Completos (Casos de Uso, Análisis, Diseño, Despliegue e Implementación)
- Arquitectura íntegra (mantenida y mínimamente actualizada)
- Riesgos Presentados Mitigados
- Plan del Proyecto para la fase de Transición.
- Manual Inicial de Usuario (con suficiente detalle)
- Prototipo Operacional – beta
- Caso del Negocio Actualizado

Al final de la fase de construcción se debe alcanzar el milestone: capacidad operacional inicial.

### **Fase de Transición.**

La finalidad de la fase de transición es poner el producto en manos de los usuarios finales, para lo que se requiere desarrollar nuevas versiones actualizadas del producto, completar la documentación, entrenar al usuario en el manejo del producto, y en general tareas relacionadas con el ajuste, configuración, instalación y facilidad de uso del producto.

Los resultados de la fase de transición son [8]:

- Prototipo Operacional
- Documentos Legales
- Caso del Negocio Completo
- Directriz completa y corregida del producto, que incluye todos los modelos del sistema
- Descripción de la arquitectura completa y corregida
- Las iteraciones de esta fase irán dirigidas normalmente a conseguir una nueva versión.

Al final de la fase de transición se debe alcanzar el milestone: versión del producto

### **Iteraciones.**

Es práctico dividir el trabajo en partes más pequeñas o en mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos en el flujo de trabajo, y los incrementos al crecimiento del producto. Como resultado de una iteración se tiene una versión (interna o externa) del producto o partes del producto final en desarrollo, los cuales crecen incrementalmente de iteración a iteración hasta llegar a ser el sistema final [6].

Las ventajas de un método iterativo comparado un proceso tradicional en cascada son:

- Mitigación de riesgos temprana.

- El cambio es controlable.
- Alto nivel de reuso.
- El equipo de proyecto puede aprender a lo largo del método.
- Mejor calidad.

### **Eje vertical aspectos estáticos del proceso – Dimensión estática del proceso.**

Un proceso describe **quién** está haciendo **qué, cómo y cuándo**. UP es representado por cuatro elementos primarios de modelamiento.

- Roles, 'Quién'.
- Actividades, 'Cómo'.
- Artefactos, 'Qué'.
- Flujos de trabajo, 'Cuándo'.

### **Roles, Actividades y Artefactos**

#### **Roles.**

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Una persona puede desempeñar diversos roles, así como un mismo rol puede ser representado por varias personas. Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el dueño de un conjunto de artefactos [8].

#### **Actividades.**

Una actividad en concreto es una unidad de trabajo que una persona que desempeñe un rol puede realizar. Las actividades tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto. Cada actividad es asignada a un rol específico, y afecta a uno o un número pequeño de artefactos. Una actividad debe ser usada como un elemento de planeación y progreso.

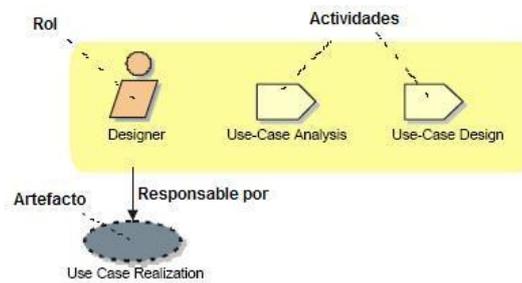
#### **Artefactos**

Un producto o artefacto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final [8]. Los roles utilizan los artefactos como entrada para desarrollar una actividad, y también son el resultado o salida de dichas actividades.

Un artefacto puede tomar varias formas como [5]:

- Un documento, como el documento de la arquitectura del software.
- Un modelo, como el modelo de casos de uso o el modelo de diseño.
- Un elemento del modelo, un elemento que pertenece a un modelo como una clase, un caso de uso o un subsistema.
- Código fuente.
- Ejecutables.

La representación grafica de los roles, actividades y artefactos se plasman en la Figura 14.



**Figura 14 Representación de Roles, actividades y artefactos en UP.**

### **Disciplinas.**

Con la enumeración de roles, actividades y artefactos no se define un proceso, necesitamos contar con una secuencia de actividades realizadas por los diferentes roles, así como la relación entre los mismos. Una disciplina es una secuencia de actividades que producen un resultado de valor observable. Los siguientes son las disciplinas centrales de UP.

### **Modelado del Negocio**

Uno de los mayores problemas en las comunidades de ingeniería del software y de ingeniería del negocio es la inapropiada comunicación; UP direcciona este problema dando un proceso y lenguaje común para las dos comunidades, así como también muestra cómo crear y mantener una trazabilidad directa entre el negocio y los modelos software.

Los objetivos del modelado de negocio son [5][4]:

- Entender la estructura y la dinámica de la organización para la cual el sistema va a ser desarrollado (organización objetivo).
- Entender el problema actual en la organización objetivo e identificar potenciales mejoras.
- Asegurar que clientes, usuarios finales y desarrolladores tengan un entendimiento común de la organización objetivo.
- Derivar los Requerimientos del sistema necesarios para apoyar a la organización objetivo.

Para lograr estos objetivos, el modelo de negocio describe cómo desarrollar una visión de la nueva organización, basado en esta visión se definen procesos, roles y responsabilidades de la organización por medio de un modelo de casos de uso del negocio y un modelo de objetos del negocio; complementario a estos modelos, se desarrollan otras especificaciones tales como un glosario.

### **Requerimientos.**

Esta es una de las disciplinas más importantes, porque se establece qué debe hacer el sistema a construir; los Requerimientos se pueden ver como un contrato que se debe cumplir, de modo que los usuarios finales tienen que comprender y aceptar la especificación de los Requerimientos.

Los objetivos del flujo de Requerimientos son [5][4]:

- Establecer y mantener un acuerdo entre clientes y otros stakeholders<sup>6</sup> sobre lo que el sistema podría hacer.
- Proveer a los desarrolladores un mejor entendimiento de los requerimientos del sistema.
- Definir el ámbito del sistema.
- Proveer una base para la planeación de los contenidos técnicos de las iteraciones.
- Proveer una base para estimar costos y tiempo de desarrollo del sistema.
- Definir una interfaz de usuario para el sistema, enfocada a las necesidades y metas del usuario.

Los requerimientos se dividen en dos grupos. Los requerimientos funcionales representan la funcionalidad del sistema. Se modelan mediante diagramas de casos de uso. Los Requerimientos no funcionales representan aquellos atributos que debe exhibir el sistema, pero que no son una funcionalidad específica; por ejemplo requerimientos de facilidad de uso, fiabilidad, eficiencia, portabilidad, etc.

En este flujo de trabajo, y como parte de los requerimientos de facilidad de uso, se diseña la interfaz gráfica de usuario, para ello habitualmente se construyen prototipos de la interfaz gráfica de usuario que se contrastan con el usuario final.

### **Análisis y Diseño**

El objetivo del flujo de trabajo de análisis y diseño es traducir los requerimientos a una especificación que describe cómo implementar el sistema en la fase de implementación.

Los objetivos del análisis y diseño son [5][4]:

- Transformar los requerimientos al diseño del futuro sistema.
- Desarrollar una arquitectura para el sistema.
- Adaptar el diseño para que sea consistente con el entorno de implementación, diseñando para el rendimiento.

El análisis consiste en obtener una visión del sistema que se preocupa de ver qué hace, de modo que sólo se interesa por los requerimientos funcionales. Por otro lado el diseño es un refinamiento del análisis que tiene en cuenta los requerimientos no funcionales, en definitiva cómo cumple el sistema sus objetivos.

Los resultados del análisis y diseño son un modelo de diseño y opcionalmente un modelo de análisis. El modelo de diseño sirve como un abstracción del código fuente; es decir, actúa como un 'anteproyecto' de cómo el código fuente debe ser estructurado y escrito, este modelo consiste en un diseño de clases estructuradas, que pueden ser agregadas en paquetes y subsistemas con interfaces bien definidas.

---

<sup>6</sup> STAKEHOLDERS Personas interesadas en el proyecto, por ejemplo: clientes, usuarios finales, persona de marketing, etc.

Otro producto importante de este flujo es la documentación de la arquitectura de software. Las actividades de diseño son centradas en torno a la noción de la arquitectura, la producción y la validación de ésta es el foco principal de iteraciones tempranas en el diseño. La arquitectura está representada por un número de vistas arquitecturales [5][4]; estas vistas capturan las decisiones trascendentales del diseño, en esencia, las vistas arquitecturales son abstracciones o simplificaciones del diseño en su totalidad, en las cuales las características importantes son más visibles y los detalles son dejados a un lado.

### **Implementación**

En este flujo de trabajo se implementan las clases y objetos en archivos fuente, binarios, ejecutables y demás. Además se deben hacer las pruebas de unidad: cada implementador es responsable de probar las unidades que produzca. El resultado final de este flujo de trabajo es un sistema ejecutable.

Los objetivos de implementación son [8]:

- Definir la organización del código, en términos de implementación de subsistemas organizados en capas.
- Implementar clases y objetos en términos de componentes (archivos fuente, binarios, ejecutables y otros)
- Probar los componentes desarrollados como unidades.
- Integrar los resultados producidos por implementaciones individuales (o equipos), en un sistema ejecutable.

La estructura de todos los elementos implementados forma el modelo de implementación. La integración debe ser incremental, es decir, en cada momento sólo se añade un elemento, de este modo es más fácil localizar fallos y los componentes se prueban más a fondo; en fases tempranas del proceso se pueden implementar prototipos para reducir el riesgo, su utilidad puede ir desde ver si el sistema es viable desde el principio, probar tecnologías o diseñar la interfaz de usuario. Los prototipos pueden ser exploratorios (desechables) o evolutivos, estos últimos llegan a transformarse en el sistema final.

### **Pruebas**

Este flujo de trabajo es el encargado de evaluar la calidad del producto que estamos desarrollando, pero no para aceptar o rechazar el producto al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

Esta disciplina brinda soporte a las otras disciplinas. Sus objetivos son [5]:

- Verificar la interacción entre objetos.
- Encontrar y documentar defectos en la calidad del software.
- Verificar la apropiada integración de todos los componentes del software.
- Verificar que todos los requerimientos han sido implementados correctamente.
- Identificar y garantizar que los defectos son gestionados previamente al despliegue del software.

Las actividades de este flujo comienzan en el proyecto con el plan de prueba (el cual contiene información sobre los objetivos generales y específicos de las pruebas en el

proyecto, así como las estrategias y recursos con que se dotará a esta tarea), o incluso antes con alguna evaluación durante la fase de inicio, y continuará durante todo el proyecto.

El desarrollo del flujo de trabajo consistirá en planificar qué es lo que hay que probar, diseñar cómo se va a hacer, implementar lo necesario para llevarlos a cabo, ejecutarlos en los niveles necesarios y obtener los resultados, de forma que la información obtenida sirva para ir refinando el producto a desarrollar.

### **Despliegue**

El objetivo de esta disciplina es producir con éxito distribuciones del producto y distribuirlo a los usuarios. Las actividades implicadas incluyen:

- Probar el producto en su entorno de ejecución final.
- Empaquetar el software para su distribución.
- Distribuir el software.
- Instalar el software.
- Proveer asistencia y ayuda a los usuarios.
- Formar a los usuarios y al cuerpo de ventas.
- Migrar el software existente o convertir bases de datos.

Este flujo de trabajo se desarrolla con mayor intensidad en la fase de transición, ya que el propósito del flujo es asegurar una aceptación y adaptación sin complicaciones del software por parte de los usuarios. Su ejecución inicia en fases anteriores, para preparar el camino, sobre todo con actividades de planificación, en la elaboración del manual de usuario y tutoriales.

### **2.2 Adaptación de UP (¿por qué es importante la adaptación?)**

El Proceso Unificado por ser un framework genérico que intenta ser aplicado a múltiples contextos de proyectos software y organizaciones de desarrollo debe ser adaptado para cumplir con las diversas características tanto de las empresas como de los proyectos, además un proceso que ha sido efectivo para un proyecto no será necesariamente efectivo para otro, puesto que los proyectos dentro de la misma organización pueden ser diferentes debido a que no existen proyectos ni procesos similares en el mundo [29].

Jacobson, Booch y Rumbaugh dicen: “UP es un framework, el cual ha de ser adaptado teniendo en cuenta un número de variables como: el tamaño del sistema, el dominio del sistema, la complejidad del sistema, la experiencia de las personas y el nivel del proceso de la organización”. Es claro que UP necesita ser adaptado considerando estas variables. Según [28] en la literatura no hay las suficientes directrices para hacer la adaptación y definitivamente es latente la necesidad de buenas directrices y consejos para realizarla.

Para entender mejor la necesidad de la adaptación de UP, se puede utilizar una analogía donde se puede decir que UP es como el proceso descrito para hacer un viaje en general, el cual implica todas las actividades posibles para realizar un viaje de forma efectiva, podríamos decir que la descripción general de un viaje puede contener

las siguientes actividades: elegir el destino, el cual puede ser nacional o internacional; preparar la maleta que puede implicar, escoger prendas adecuadas para el tipo de clima del destino, cálido, frío, templado, nieve y también la duración del viaje; verificar documentos personales que puede implicar de acuerdo al destino, pasaporte, visa y otros documentos exigidos por el destino escogido y hasta el tipo de moneda; realizar la compra de pasajes que puede implicar, compra de pasajes aéreos o terrestres; realizar la reserva del sitio de hospedaje etc. Entonces si este es el proceso general para realizar un viaje efectivo, una persona que decida realizar un viaje con unas determinadas características como: salir en su propio carro a su finca de la misma localidad por un fin de semana va a efectuar algunas de las actividades generales del proceso pero debe tener en cuenta las características de su viaje, de manera que no está obligado a realizar algunas actividades que para él no tienen sentido como por ejemplo: en la verificación de los documentos personales él no tendría en cuenta los documentos necesarios para hacer un viaje a otro país o el tipo de moneda, realizar la compra de pasajes y la reserva del sitio de hospedaje. Entonces, de esta manera la persona debe adaptar el proceso genérico de acuerdo a sus necesidades y características, pues carecería de sentido realizar actividades del proceso que no aporten en la ejecución exitosa del proceso. Esta situación es similar a la que las empresas de desarrollo de software se enfrentan cuando utilizan UP como proceso de desarrollo, ellas necesitan saber qué elementos del proceso realmente aportan de acuerdo a las necesidades y características del proyecto.

### **2.3 El Proceso Unificado descrito a través de SPEM 2.0**

Para efectos de modelar y realizar la adaptación del Proceso Unificado (UP), en este proyecto, se implementó UP a través del lenguaje SPEM 2.0 (Software & Systems Process Engineering Metamodel), que sirve para la representación de modelos de procesos de ingeniería del software e ingeniería de sistemas. Como editor de procesos se utilizó la herramienta EPFC (Eclipse Process Framework Composer) que se basa en SPEM y permite definir, gestionar y reutilizar un repositorio de fragmentos de métodos y procesos.

En las siguientes secciones se presenta la implementación de UP en EPFC.

#### **2.3.1 Aspectos utilizados para implementar UP con SPEM 2.0**

En SPEM se distinguen dos aspectos en el momento de implementar una metodología, primero, se define y se estructura el contenido del método con elementos de contenido, es decir, los elementos primarios o constructores básicos y luego, se definen los procesos mediante la reutilización de los elementos del método.

Para implementar la dimensión estática de UP a través de SPEM los elementos del contenido del método fueron categorizados por las disciplinas de UP, donde los elementos básicos (actividades, roles y artefactos) de las disciplinas de UP pasaron a ser los elementos del contenido de método (task definition, role definition, work product definition) de SPEM y para implementar la dimensión dinámica de UP se utilizaron los elementos del Process (process, activity, work product use, role use, task use) definidos por SPEM, todo lo anterior debido a la ortogonalidad del contenido del método y de los procesos de SPEM y UP, ver Figura 15.

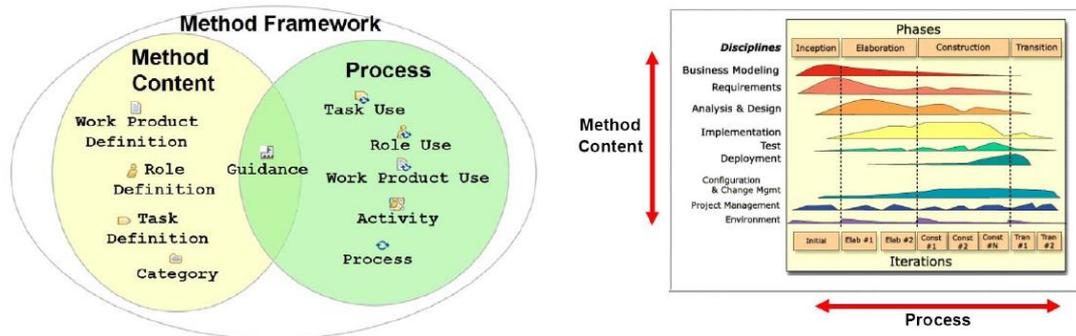


Figura 15 Ortogonalidad en SPEM y UP

### 2.3.2 Método Plugin del Proceso Unificado

Un método plugin es un paquete que representa un contenedor físico para paquetes de contenido y procesos. El contenido del método define un nivel de granularidad para la modularización y organización de contenido del método y procesos [12]. Para la definición de UP se implementó un plugin llamado *Unified Process*.

#### 2.1.1.3 Contenido del método

Los elementos de contenido del método del plugin fueron organizados mediante una jerarquía de paquetes de contenido. Los cinco paquetes de contenido de acuerdo con las disciplinas de UP son: modelado del negocio, requerimientos, análisis y diseño, implementación y pruebas. Ver Figura 16.

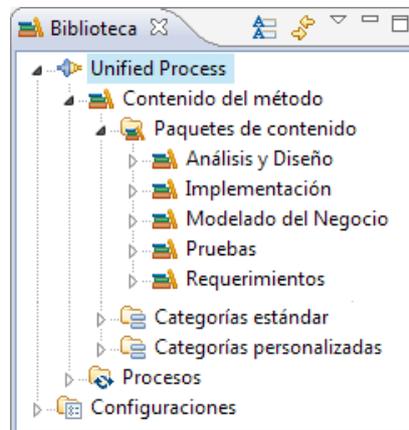


Figura 16 Paquetes de Contenido definidos para el UP.

#### 2.3.2.1.1 Paquetes de Contenido

En cada paquete de contenido se definieron los elementos de contenido del método básicos: roles, tareas y productos de trabajo involucrados en cada una de las disciplinas de UP. La Figura 17 muestra la división del paquete de contenido modelado del negocio y sus elementos básicos; por cuestiones de espacio solo se muestran algunos roles, tareas y productos de trabajo. Es importante destacar que cada uno de los cinco paquetes de contenido están estructurados de la misma forma.

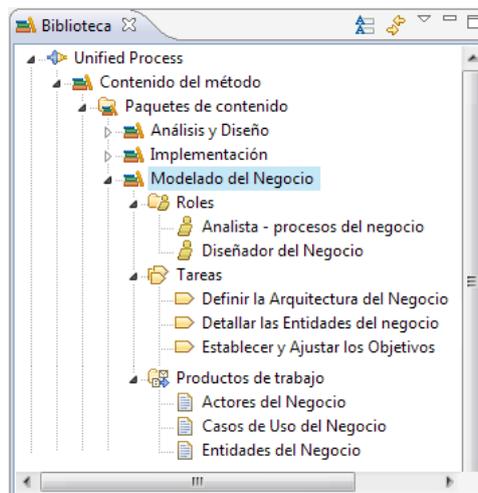


Figura 17 División del Paquete de Contenido Modelado del Negocio.

### 2.3.2.1.2 Categorías

Una categoría es un elemento de contenido, o de proceso, usado para categorizar [12], es decir, clasificar o agrupar dichos elementos en base a los criterios del usuario. Una categoría puede tener 0..\* (cero o muchas) subcategorías; esto permite establecer cualquier tipo de jerarquía de agrupamiento de elementos. SPEM 2 distingue dos clases de categorías:

### 2.3.2.1.3 Categoría Estándar

Los elementos del contenido de método se pueden categorizar mediante su asociación con categorías estándar (disciplinas, dominios, clases de productos de trabajo, conjuntos de roles y herramientas). Las categorías estándar vienen predefinidas en SPEM. Para el plugin de UP se utilizaron tres tipos de categorías estándar, disciplinas, clases de productos de trabajo y conjunto de roles. Ver Figura 18.

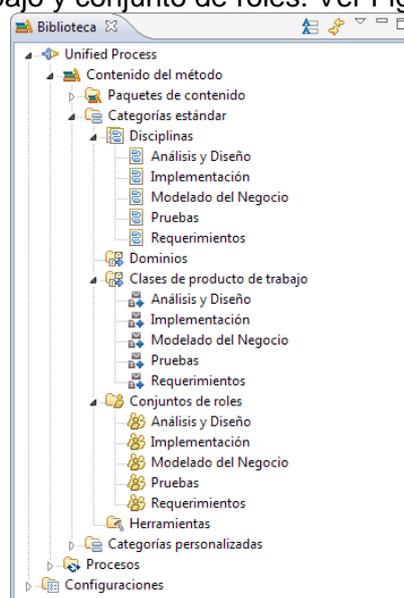


Figura 18 Categorías Estándar utilizadas en UP

#### 2.3.2.1.4 Categoría personalizada

Los elementos del contenido de método (y también los elementos en uso en procesos) se pueden organizar alternativamente mediante categorías personalizadas. Estas categorías personalizadas pueden contener cualquier tipo de elemento, proceso, categoría, etc. Son útiles para organizar de forma lógica elementos que no se puedan organizar mediante las categorías estándar. Las categorías personalizadas permiten organizar el contenido de acuerdo al esquema que queramos, sirviendo como medio eficaz para organizar el contenido para su publicación. La Figura 19 muestra la estructura de las categorías personalizadas definidas en el plugin y en la Figura 20 se muestra su publicación en HTML.

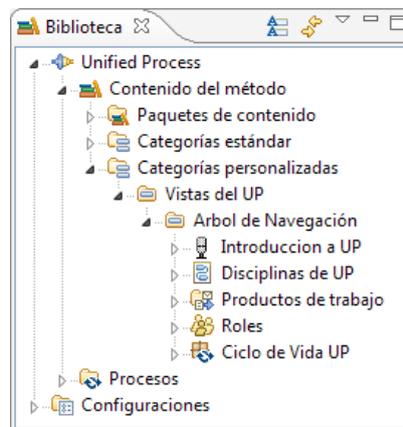


Figura 19 Estructura de las Categorías Personalizadas

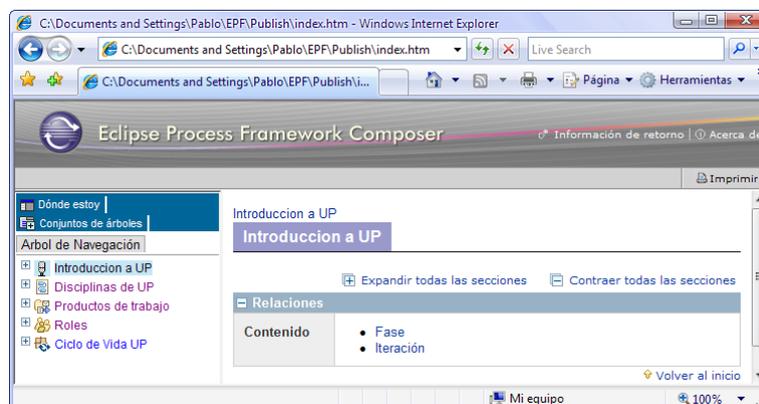


Figura 20 Publicación de las Categorías Personalizadas

#### 2.3.2.2 Procesos

Un proceso es un tipo de actividad que describe una estructura para tipos particulares de proyectos o partes de ellos [12][11]. Los procesos software definen cómo serán ejecutados los proyectos software.

En SPEM 2 existen dos clases principales de procesos: patrón de proceso y proceso para despliegue.

### 2.3.2.2.1 Patrón de Proceso

Un patrón de proceso es un proceso especial “fragmento de proceso” que describe un grupo de actividades reusables como solución a algún tipo de problema o situación habitual [11]. Se definen para poder ser empleados más de una vez en uno o varios procesos o con fines de organización.

En el plugin se definió una jerarquía de cinco paquetes de proceso (process package) por cada una de las disciplinas de UP, ver Figura 21 donde cada paquete de proceso tiene varios patrones de proceso con el fin de servir como bloques para construir el proceso de despliegue, ver Figura 22.

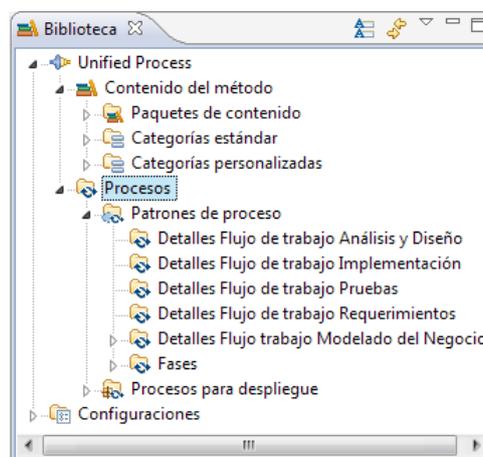


Figura 21 Paquetes de Proceso definidos para el Plugin.

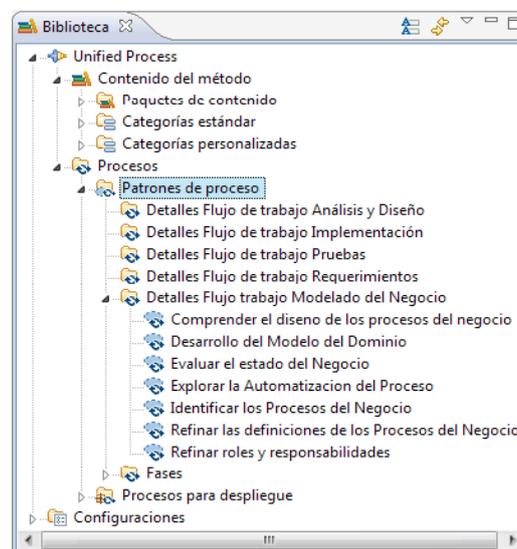


Figura 22 Patrones de proceso

### 2.3.2.2.2 Proceso para despliegue

Describe una aproximación completa e integrada para realizar un tipo específico de proyecto, abarcando un ciclo de vida completo de desarrollo o mantenimiento [12][11]. Sirven como plantillas para planificar y ejecutar los proyectos. En un proceso de despliegue se ensamblan patrones de proceso y elementos en uso (tareas, roles y productos de trabajo en uso). En el proceso de despliegue para el plugin se llama “CicloVidaUP”, está dividido en las fases de UP, inicio, elaboración, construcción, y transición. En cada fase se reutilizaron los patrones de proceso definidos en los paquetes de proceso mencionados en la sección anterior, ver Figura 23.

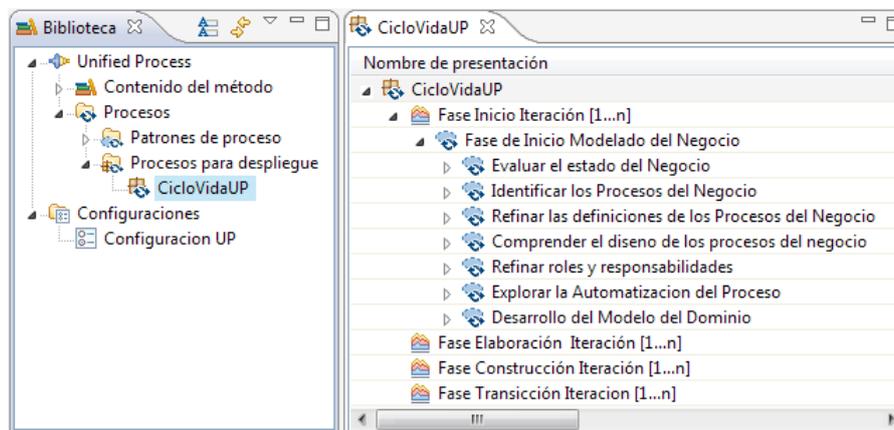


Figura 23 Proceso para despliegue.

## CAPITULO 3. DESCRIPCIÓN DEL PROYECTO.

En este capítulo, se muestra cómo se realizó la evaluación de los estudios, y se exponen los aspectos más relevantes que se tuvieron en cuenta en la definición de la Guía de Tailoring.

### 3.1 Evaluación de estudios

En esta sección, se evalúan estudios e investigaciones referentes a la adaptación de procesos de desarrollo software, que para el objetivo de éste trabajo fueron los que más aportaron en la realización de la guía de tailoring de UP.

Es importante destacar que la investigación no solo se basó en los estudios mencionados en el capítulo 1, sino también se revisaron otros documentos que permitieron identificar algunos pasos genéricos, características y elementos necesarios en la adaptación del proceso software a proyectos específicos. La evaluación de los estudios se hizo de acuerdo a la valoración de los pasos genéricos que debe tener el proceso de adaptación, los cuales son:

- **Realizar la caracterización del proyecto**, en este paso se pretende identificar algunas particularidades del proyecto que pueden afectar la adaptación del patrón de procesos software.
- **Seleccionar las entidades del proceso**, en este punto, se eligen las entidades que harán parte del proceso adaptado.

- **Realizar la configuración del proceso**, se efectúan las posibles modificaciones al patrón de procesos de acuerdo a las entidades seleccionadas que harán parte del proceso adaptado.

Además, de los pasos genéricos en la guía de tailoring se consideró necesario adicionar el paso “desplegar el proceso”, aunque este paso no se vislumbra en la mayoría de los estudios, para la guía es importante incluirlo, ya que por medio de éste se puede mostrar de una forma real el resultado de la aplicación de la guía, es decir, con la ejecución de este paso se pretende dar a conocer el proceso adaptado para su futura utilización. Entonces, de esta manera los pasos contemplados para la guía de tailoring y la evaluación de los estudios son: *realizar la caracterización del proyecto, seleccionar los elementos del proceso, realizar la configuración del proceso y desplegar el proceso.*

### 3.1.1 Criterios de evaluación

Para realizar la evaluación de los estudios e investigaciones relacionados con la adaptación de procesos de desarrollo software, se tomaron los criterios fundamentación, simplicidad y completitud de [30][31], que permitieron seleccionar las características claves de los estudios para cada uno de los pasos genéricos de la adaptación y de esta forma introducirlas a la guía de tailoring de UP.

A continuación se presentan cada uno de los criterios con su correspondiente definición y el rango de valores que pueden tomar de acuerdo al grado de cumplimiento con cada criterio.

- **Criterio de fundamentación:** Hace referencia a las bases conceptuales, bibliografía y casos de estudio disponibles o ejemplos, que sustentan o que originaron el paso en la adaptación del proceso software.
  1. No se encontró suficiente información de este paso en el estudio o investigación.
  2. Se encontró información de este paso en el estudio o investigación, pero no casos de estudio o ejemplos.
  3. Se encontró suficiente información de este paso en el estudio o investigación con casos de estudio y ejemplos.
- **Criterio de simplicidad:** Este hace referencia a la reducción de trabajo que se debe realizar para llevar a cabo cada paso, es decir el estudio o investigación contiene lo necesario para desarrollar el paso en la adaptación del proceso. Este es, tal vez, uno de los criterios más importantes, debido a los pocos recursos con los que cuentan las PyMEs\_DS y que de una u otra manera significarían una barrera para la incorporación de la guía.
  1. El paso no contiene los elementos necesarios (sub-pasos, tareas) para ser realizado.
  2. El paso contiene demasiados elementos (sub-pasos, tareas), volviendo complejo y pesado.
  3. El paso contiene los elementos necesarios y suficientes para ser ejecutado.

- **Criterio de completitud:** Especifica detalladamente todas las prácticas de tal forma que se entienda lo que se debe hacer y cómo se debe hacer, es decir, cada una de los sub-pasos o tareas deben ser lo suficientemente explicativas para facilitar su ejecución. A continuación se muestran los criterios de completitud:
  1. El paso no detalla las prácticas o actividades.
  2. El paso detalla que se debe hacer.
  3. El paso especifica detalladamente qué se debe hacer y cómo se debe hacer.

Para hacer referencia de forma abreviada a los criterios de evaluación, en la Tabla 1 se define la nomenclatura para cada uno de los criterios que serán utilizados en la Tabla 2.

<b>NOMENCLATURA DE CRITERIOS</b>	
<b>Criterio</b>	<b>Abreviatura</b>
Fundamentación	Fun.
Simplicidad	Sim.
Completitud	Com.

**Tabla 1 Nomenclatura de criterios**

Con el fin de seleccionar las características más adecuadas para dar cumplimiento a los objetivos de esta investigación, en la Tabla 2, se presenta la valoración de los estudios e investigaciones mediante los criterios definidos anteriormente y aplicados a cada uno de los pasos genéricos de la adaptación de procesos identificados en esta investigación.

<b>Evaluación de estudios e investigaciones de adaptación</b>																
<b>Estudios y Modelos</b>	<b>Pasos genéricos en el la adaptación de procesos</b>															
	<b>Caracteriza el proyecto</b>				<b>Selecciona los elementos del proceso</b>				<b>Realiza la configuración del proceso</b>				<b>Despliega el proceso</b>			
	<b>Criterios</b>				<b>Criterios</b>				<b>Criterios</b>				<b>Criterios</b>			
	Fun	Sim	Com	total	Fun	Sim	Com	total	Fun	Sim	Com	total	Fun	Sim	Com	total
<b>Tailoring and Verifying software Process</b>				0				0	2	2	1	5				0
<b>Tailoring Software Evolution Process</b>				0				0	2	1	1	4				0
<b>MAPS</b>	2	3	2	7	1	1	1	3	1	1	2	4				0
<b>A Matrix Approach to Software Process Definition</b>				0	2	3	2	7	1	1	1	3				0
<b>Tailoring a Software Process for Software Project Plans</b>	2	3	2	7	2	1	2	5	1	1	1	3	1	1	1	3

Evaluación de estudios e investigaciones de adaptación																
Estudios y Modelos	Pasos genéricos en el la adaptación de procesos															
	Caracteriza el proyecto				Selecciona los elementos del proceso				Realiza la configuración del proceso				Despliega el proceso			
	Criterios				Criterios				Criterios				Criterios			
	Fun	Sim	Com	total	Fun	Sim	Com	total	Fun	Sim	Com	total	Fun	Sim	Com	total
SPEM				0				0	2	2	2	6	2	2	2	6
RUP (Disciplina de entorno)	1	1	1	3	1	2	2	5	1	1	1	3	1	1	1	3

**Tabla 2. Evaluación de estudios e investigaciones de adaptación.**

Cuando los estudios no especifican alguno de los pasos genéricos de la adaptación del proceso software se les asigno 0 (cero) en el total de cada uno de los criterios, donde el puntaje máximo por paso es de 9 (nueve) puntos. Es importante aclarar que la elección de los estudios valorados fue realizada teniendo en cuenta su aporte a los pasos genéricos en la adaptación, por este motivo hay estudios que no aportan a algunos de los pasos genéricos, pero que son importantes en algún otro paso en especial.

Como resultado de la evaluación de los estudios se obtuvo que:

- “MAPS”, “Tailoring a Software Process for Software Project Plans” y “RUP” (disciplina de entorno) aportaron en la definición del paso **“realizar la caracterización del proyecto”**.
- “RUP” (disciplina de entorno), “A Matrix Approach to Software Process Definition”, “Tailoring a Software Process for Software Project Plans” aportaron a la definición del paso de **“selección de las entidades del proceso”**.
- “Tailoring and Verifying software Process”, “Tailoring Software Evolution Process” y SPEM contribuyeron principalmente en la definición del paso **“realizar la configuración del proceso”** y para la definición del paso **“desplegar el proceso”** se tuvo en cuenta a SPEM y EPFC.

### 3.2 Enfoque de la Guía de Tailoring

El proceso unificado está dividido en un eje horizontal y uno vertical [8]. El primer eje representa el tiempo, e ilustra aspectos dinámicos del proceso descritos en fases, iteraciones, e hitos. El segundo eje representa las disciplinas que agrupan actividades; este eje describe un aspecto estático del proceso representado en términos de disciplinas, actividades, flujos de trabajo, artefactos y roles. Debido a la complejidad del Proceso Unificado y a la división, en los ejes mencionados, en este trabajo se restringió la adaptación hacia el eje vertical o disciplinas, dado a que la adaptación de disciplinas es lo más complejo y el mismo enfoque puede ser aplicado para adaptar el eje horizontal o dinámico, pues así como se adaptan patrones es posible adaptar

procesos. Realizar la adaptación del proceso por medio de las disciplinas permite la reducción de la complejidad del proceso, puesto que es más fácil analizar y adaptar disciplina a disciplina que todo el proceso en su conjunto.

La adaptación de las disciplinas en este trabajo está orientada a la elección del conjunto de artefactos que harán parte del proceso adaptado a proyectos específicos.

### **3.3 Caracterización del proyecto**

Para realizar la caracterización de los proyectos en la guía de tailoring se tuvieron en cuenta algunos aspectos de [25] y [26], que en la evaluación de los estudios fueron los que más aportaron para realizar el primer paso genérico que debe realizarse en la adaptación del proceso software a proyectos específicos.

En la siguiente sección se presentarán las variables de caracterización y los niveles de caracterización que hacen parte del primer paso de la guía de Tailoring de UP.

#### **3.3.1 Variables para realizar la caracterización de los proyectos.**

Las variables de caracterización tienen como fin identificar algunas particularidades del proyecto que pueden afectar la adaptación del proceso; por su parte los niveles de caracterización tienen el propósito de brindar un conjunto de valores (niveles) determinados que pueden ser asignados a las variables en el momento de efectuar la caracterización del proyecto.

Debido a la gran cantidad de características que pueden influir en un proyecto en el momento de realizar la caracterización y con el fin de disminuir la complejidad de ésta, se eligió un número limitado de características, y se optó por utilizar algunas de las definidas en [10][25] y [26], las cuales se dividen en tres tipos:

- Características de desarrollo
- Características restrictivas
- Características que priorizan el proyecto

**3.3.1.1 Características de desarrollo:** Caracterizan el proyecto teniendo en cuenta sólo los aspectos de desarrollo de software, sin considerar factores de la organización. Las variables de desarrollo que se deben tener en cuenta son:

**3.3.1.1.1 Variable de tamaño del equipo:** La variable de tamaño del equipo hace referencia a la cantidad de personas que serán utilizadas para efectuar el proyecto. Esta variable puede indicar el número de personas que recibirán asistencia, seguimiento y, si es necesario, capacitación.

Para que un proceso de desarrollo de software sea efectivo éste debe cumplir con la función de coordinar las personas implicadas en el desarrollo del proyecto, teniendo en cuenta que para equipos de diferentes tamaños se necesitan diferentes maneras de realizar la coordinación; de igual manera la asignación de responsabilidades de las actividades del proceso también se ve afectada por el tamaño del equipo, puesto que

en equipos pequeños ésta se puede realizar de manera informal lo cual no es posible en equipos grandes.

El tamaño del equipo dentro del desarrollo impacta directamente la comunicación entre los miembros del equipo. Cuando se trata de equipos pequeños las formas de comunicarse son informales y de cierta manera se puede llegar a concluir que son suficientes para lograr coordinación del equipo; este tipo de comunicaciones son menos costosas; mientras que con equipos de un número mayor de miembros la eficacia de la comunicación es menor y se requieren comunicaciones más oficiales que implican un aumento en el costo de la coordinación de los proyectos [32] [34].

En [8], se describe la necesidad de aumentar la sobrecarga de gestión (planificación, comunicación, coordinación, evaluación de los progresos, de evaluación, administración) a medida que el tamaño del equipo aumenta. En pequeños equipos, la necesidad de documentar los artefactos intermedios es baja, el enfoque está en los artefactos y las actividades de planificación y gestión las cuales son realizadas de manera informal. Aumentar el tamaño del equipo causa mayor necesidad de documentar los artefactos intermedios, teniendo un mayor énfasis en los artefactos de gestión y una mayor formalidad en la planificación y gestión

El tamaño del equipo afecta todo proceso de desarrollo puesto que la asignación de responsabilidades, la comunicación y la coordinación de los miembros del equipo son necesarias durante toda la ejecución del proceso.

**Niveles de caracterización de la variable tamaño del equipo:** Con el fin de brindar los posibles niveles que puede tomar la variable de tamaño del equipo y teniendo en cuenta que este trabajo está orientado hacia las PyMEs\_DS, se han definido los niveles de caracterización de acuerdo con [37]; de esta manera los niveles en los cuales se puede ubicar la variable del tamaño del equipo son:

- Nivel 1: Muy pequeño: 1-10 personas
- Nivel 2: Pequeño: 11-50 personas
- Nivel 3: Mediano: 51-200 personas

**3.3.1.1.2 Variable distribución geográfica del equipo:** En estos tiempos cada vez es más común que un proyecto sea desarrollado por más de una organización e incluso existen organizaciones que pueden ubicar satélites (sucursales de desarrollo) en diferentes puntos geográficos, debido al ritmo acelerado de la globalización de las empresas y a la gran diferencia en el valor pagado por mano de obra en las diferentes regiones. Por este motivo la existencia de equipos de desarrollo distribuidos tienden a ser más común, lo que provoca un impacto directo en el proceso de desarrollo de software afectando la productividad del equipo [33].

Las formas más efectivas de comunicación son las que enfatizan el contacto personal, cara a cara, entre los miembros del equipo [34]. En los equipos distribuidos geográficamente el contacto personal es menos frecuente, por esta razón se pueden tener más problemas de coordinación; en esta situación se puede requerir de más tiempo para resolver y expresar dudas debido a la falta de comprensión o interpretación [38]. Por lo tanto, en los equipos de desarrollo distribuidos y en los

equipos grandes, se necesitan formas de comunicación más formales, consecuentemente la documentación producida en este tipo de equipos debe ser clara, exacta y detallada para evitar ambigüedades y generación de dudas que pueden ser difíciles de resolver.

La variable de distribución geográfica del equipo, como su nombre lo dice, ayuda a definir cómo están distribuidos geográficamente los miembros del equipo involucrados en la ejecución del proyecto.

**Niveles de caracterización de la variable de distribución geográfica del equipo:**

Los niveles de caracterización que puede tomar la variable de la distribución geográfica del equipo fueron tomados de [35], donde los niveles en los cuales se puede ubicar esta variable son:

- Nivel 1: Misma Oficina
- Nivel 2: Mismo edificio, diferentes oficinas
- Nivel 3: Misma ciudad, misma empresa, diferente edificio
- Nivel 4: Misma ciudad, diferente empresa
- Nivel 5: Diferente ciudad

**3.3.1.1.3 Variable de experiencia del equipo:** La variable de experiencia del equipo tiene como fin indagar a cerca de previos conocimientos y experiencias que ha tenido el equipo de desarrollo. La experiencia del equipo es un factor importante en la definición de un proceso de desarrollo de software [10]. Por ser una característica compleja, la experiencia del equipo según [25] se divide en tres características:

- Experiencia en proceso.
- Experiencia técnica.
- Experiencia en aplicación.

Previas experiencias en UP afectan el proceso de desarrollo, puesto que un equipo sin experiencia en el proceso debe ser más monitoreado y controlado; los equipos que tienen previas experiencias se pueden tornar un poco más eficientes gracias al conocimiento que tienen del proceso.

Equipos sin experiencia técnica y de aplicación pueden asignar periodos de tiempo inadecuados para el desarrollo de las actividades y para las iteraciones del proceso [8].

La experiencia en la aplicación tiene un impacto sobre todo el proceso, pero es más significativo en la disciplina de requerimientos, pues se espera que, para dominios conocidos, los requerimientos sean elegidos de una forma más completa y precisa, permitiendo una supervisión más informal de los requerimientos.

Por su parte la experiencia técnica influencia las disciplinas de análisis y diseño e implementación, porque para dominios desconocidos se necesita un mayor número de modelos para garantizar la comprensión del problema y la implementación de la solución.

La experiencia técnica hace referencia al conocimiento previo por parte de los integrantes del equipo de desarrollo con la tecnología, herramientas y paradigmas adoptados para el desarrollo del proyecto. Este tipo de experiencia afecta a la disciplina de implementación, puesto que gran parte del éxito de ésta depende de las facilidades encontradas con la tecnología, además una buena experiencia técnica puede demandar menos esfuerzos cuando se efectuó la disciplina de implementación. La experiencia técnica, también puede afectar a la disciplina de análisis, ya que el uso de nuevas tecnologías ó nuevos paradigmas pueden cambiar la manera de modelar el sistema.

**Niveles de caracterización de la variable experiencia del equipo:** Debido a la división de la característica de experiencia del equipo en: experiencia en el proceso, experiencia técnica y experiencia en el dominio de la aplicación, los niveles de clasificación también se dividen de acuerdo a cada tipo de experiencia [25].

La experiencia en el proceso es definida como el número medio de proyectos que utilizaron UP en los que participaron los miembros del equipo. Los niveles que puede tomar este tipo de experiencia son:

- Nivel 1: No hay ningún proyecto
- Nivel 2: 1 proyecto
- Nivel 3: 2 a 3 proyectos
- Nivel 4: 4 a 5 proyectos
- Nivel 5: Más de 5 proyectos

La experiencia en el dominio de la aplicación es definida como el número medio en proyectos del mismo dominio de la aplicación en que los miembros del equipo participaron. La experiencia en el dominio de la aplicación puede tomar los siguientes niveles:

- Nivel 1: No hay ningún proyecto
- Nivel 2: 1 proyecto
- Nivel 3: 2 a 3 proyectos
- Nivel 4: 4 a 5 proyectos
- Nivel 5: Más de 5 proyectos

La experiencia técnica es definida como el tiempo de experiencia de los miembros del equipo con las principales tecnologías que se utilizarán en el proyecto:

- Nivel 1: No hay ningún proyecto
- Nivel 2: 1 proyecto
- Nivel 3: 2 a 3 proyectos
- Nivel 4: 4 a 5 proyectos
- Nivel 5: Más de 5 proyectos

**3.3.1.1.4 Variable de criticidad del sistema:** La variable de criticidad del sistema intenta indagar las consecuencias en el caso de que ocurra alguna falla en el sistema.

Sistemas críticos son aquellos que no pueden fallar, en virtud de que pueden causar daños irreparables.

La criticidad del sistema es una variable que puede afectar todo el proceso puesto que está relacionada con la calidad del producto. Cuanto más crítico, mayor debe ser la calidad del sistema; para tener un producto de calidad el proceso debe ser más riguroso [39].

La documentación generada durante la ejecución de un sistema crítico debe ser de buena calidad, puesto que debe generar confianza y una buena comprensión a los integrantes del equipo en el momento en que ellos la utilicen. Par obtener una buena documentación el proceso debe contar con mecanismos de control y verificación de la producción de dicha documentación.

Dado el caso del desarrollo de un sistema crítico la variable de criticidad del proyecto puede influir de forma más significativa en las disciplinas de requerimientos y pruebas, debido a que la disciplina de requerimientos tendrá que identificar muy bien los requerimientos no funcionales de la aplicación y por su parte la disciplina de pruebas debe ser más rigurosa para poder prevenir o identificar las posibles fallas del sistema antes de que salga al mercado.

**Niveles de caracterización de la variable criticidad del sistema:** Los niveles de caracterización propuestos para los proyectos en cuanto a su criticidad del proyecto fueron tomados de [35] los cuales son:

- Nivel 1: Bajo desempeño
- Nivel 2: Pérdidas bajas (pérdidas fácilmente recuperables)
- Nivel 3: Daños moderados
- Nivel 4: Pérdidas grandes (pérdidas irrecuperables)
- Nivel 5: Riesgo de perder vidas

**3.3.1.1.5 Variable del tamaño del proyecto:** El tamaño del proyecto puede medirse de varias formas como: el tamaño del producto, puntos de función o líneas de código. Para este trabajo el tamaño del proyecto será medido a través del costo del proyecto. El costo del proyecto es la inversión realizada por la organización en el proyecto. Para una inversión mayor en los proyectos, es probable que las empresas prefieran adoptar un proceso más predecible.

Según las estadísticas los proyectos grandes, tienen probablemente menos éxito que los proyectos pequeños [36][40]. En general, cuanto más grande sea el proyecto, hay más cantidad de recursos utilizados, actividades realizadas y artefactos producidos. Esto implica que el proceso debe ser riguroso y además debe proporcionar más control sobre las disciplinas del proceso cuando el tamaño del proyecto es mayor.

**Niveles de caracterización de la variable tamaño del proyecto:** Para realizar la clasificación de los proyectos en cuanto al costo del proyecto se tomó como base [36], cuyos valores fueron adaptados a las necesidades de la industria colombiana. Los niveles de clasificación de los proyectos en cuanto a su tamaño son:

- Nivel 1: Hasta \$ 5'000.000
- Nivel 2: Entre \$ 5'000.000 y \$ 15'000.000
- Nivel 3: Entre \$ 15'000.000 \$ 25'000.000
- Nivel 4: Entre \$ 25'000.000 y \$ 35'000.000
- Nivel 5: Más de \$ 35'000.000

### 3.3.1.2 Características restrictivas

Algunas características tienen la propiedad de limitar la libre definición del proceso software en la medida en que imponen restricciones a éste. Por este motivo, estas características se denominan restrictivas, a continuación se presentan las variables restrictivas que hacen parte de la guía de tailoring de UP.

**Herramientas:** El uso de herramientas de apoyo que automatizan parte de desarrollo puede facilitar la puesta en práctica de algunas actividades, que incluso no son estrictamente necesarias para el proceso de desarrollo, debido al bajo costo que esta actividad tendrá.

Por otro lado, la falta de herramientas de apoyo puede hacer que actividades que podrían ser ejecutadas en menor tiempo con la implementación de herramientas generen un mayor costo o tiempo necesario para llevarlas a cabo manualmente.

Cabe notar que el uso de herramientas de apoyo también puede extender la definición del proceso en la medida en que el proceso tenga que ajustarse a la forma de trabajo de las herramientas, esto puede implicar un desfase en el tiempo asociado para el desarrollo del proyecto.

**Normas aprobadas:** La utilización de normas de calidad como CMMI, ISO y modelos de procesos como Moprosoft<sup>7</sup>, requieren que ciertas actividades se lleven a cabo para mantener la conformidad del proceso con estas normas; el hecho de tener que cumplir con ellas, hace que el desarrollo del proyecto sea más formal.

**Presupuesto:** proyectos que trabajan con un presupuesto limitado puede sufrir graves limitaciones con respecto a su proceso de desarrollo. Las limitaciones presupuestarias pueden impedir que las principales actividades del proceso puedan ser llevadas a cabo por falta de recursos para realizar contrataciones, compra de hardware y software necesarios, tornando difícil la ejecución del proyecto.

### 3.3.1.3 Características que priorizan el proyecto

Las características analizadas anteriormente (de desarrollo y restrictivas) pueden permitir determinar las adaptaciones necesarias del proceso software para adecuarse al proyecto y las restricciones de estas adaptaciones. Sin embargo para efectuar la adaptación, no solo se puede tener en cuenta las características de desarrollo y restricciones, también se debe considerar la expectativa de la organización desarrolladora en relación con el proyecto; en otras palabras, es preciso analizar cuáles son las prioridades del proyecto.

---

<sup>7</sup> MOPROSOFT modelo de procesos para la Industria de Software.

En este trabajo, las prioridades del proyecto serán analizadas, sobre la base de la calidad del producto deseado y el tiempo necesario para que este producto llegue al cliente, ya sea un contratista, la propia organización desarrolladora o cualquier otro mercado de consumo.

Aunque la relación entre la calidad y el tiempo de desarrollo es, en parte, determinada por las variables de criticidad del sistema, el calendario y exigencias del contrato, también es posible el escenario en el cual se puede solicitar que un software, incluso no crítico, tenga la mayor calidad posible.

Según [25] lo anterior puede deberse a muchos factores, tales como:

- La organización puede querer reforzar su imagen, garantizando productos de alta calidad.
- El mercado del software puede ser muy competitivo, que exige productos de alta calidad para mantener la competitividad.
- El desarrollo de un producto de calidad puede generar futuros proyectos con el contratista.

Por otro lado, se puede esperar que un proyecto se termine lo más pronto posible, sacrificando la calidad, en este caso podría deberse a factores, tales como:

- La organización de desarrollo puede querer ganar una tajada de un mercado pionero en expansión o en un nuevo nicho de mercado.
- El producto puede necesitar ajustes urgentes debido a los cambios en medio ambiente en el que se inserta.
- La organización puede necesitar urgente los recursos financieros que se generarán con la realización del proyecto.
- Los recursos asignados al proyecto se debe ser reorganizados para proyectos más importantes.

Las prioridades del proyecto requieren cambios en el proceso de desarrollo. En los proyectos cuya prioridad es la entrega del producto, los procesos pueden ser ágiles, dado que se centran más en el desarrollo y menos en actividades de control y aseguramiento de la calidad. En los proyectos donde la calidad es la prioridad, es importante disponer de mecanismos de control, documentación más completa y procesos más predictivos.

Una característica peculiar de las prioridades del proyecto es que, a diferencia de las características del desarrollo, no sólo afectan a las disciplinas específicas del proceso, también al proceso en su conjunto. Tanto la calidad del producto como el tiempo de desarrollo afectan la exactitud de las pruebas, la gestión de la configuración, la calidad de los modelos de análisis, la definición de los requerimientos, las decisiones de implementación, el nivel de control en términos de gestión de proyectos, etc.

Por estar vinculado al proceso de desarrollo y dar una visión general del proyecto, las prioridades del proyecto se puede utilizar como directrices de adaptación del proceso.

**Niveles de caracterización de la variable que prioriza el proyecto:** Los niveles de clasificación propuesta para la prioridad del proyecto fueron desarrollados sobre la base de una relación entre la calidad y el tiempo de desarrollo. Según [25] se realiza una distribución de 100 puntos entre la calidad y el tiempo de desarrollo, de acuerdo con lo que es más relevante para el proyecto. Los extremos no se tienen en cuenta porque, en la práctica, es extremadamente raro para un proyecto que el tiempo o la calidad no tenga relevancia. El proyecto debe ser clasificado de acuerdo a la situación que más se aproxime a su realidad. Los niveles de clasificación para esta variable fueron tomados de [25], cuyos niveles de clasificación son los siguientes:

- Nivel 1: Calidad = 90, tiempo = 10
- Nivel 2: Calidad = 70, tiempo = 30
- Nivel 3: Calidad = 50, tiempo = 50
- Nivel 4: Calidad = 30, tiempo = 70
- Nivel 5: Calidad = 10, tiempo = 90

### **3.4 Seleccionar las entidades del proceso**

Para efectuar el paso de la selección de las entidades que conformaran el proceso adaptado, en la guía de tailoring, se utilizaron elementos de [10][23] y [26], es importante resaltar que estos estudios contribuyeron en la realización de este paso, debido a que obtuvieron las mejores puntuaciones en las valoraciones de los criterios tenidos en cuenta para realizar la evaluación de los estudios e investigaciones.

La adaptación de UP a proyectos específicos según el enfoque de la guía, luego de haber caracterizado el proyecto, es necesario escoger los artefactos de las disciplinas que serán parte del proceso adaptado. La adaptación enfocada en la elección de los artefactos, áreas claves de procesos, o bloques de procesos no es una idea nueva, por su parte [10][18][23][25][26] son trabajos que sugieren esta forma para realizar la adaptación. Efectuar la adaptación de esta manera tiene como ventaja que solo se definirán las entidades necesarias para el proceso de desarrollo.

Según [26] elegir y adaptar los bloques de procesos<sup>8</sup> para un proyecto concreto es un proceso que consiste en seleccionar los bloques de procesos adecuados de un catálogo de procesos, esto implica cumplir algunas opciones básicas para la utilización de los distintos bloques de procesos, las cuales pueden ser:

1. Algunos bloques de procesos son ejecutados siempre, tal como están definidos en el proceso.
2. Algunos bloques de procesos puede ser adaptados en función de un conjunto de directrices (en general, las directrices afirmarán si los bloques de procesos podrán ser modificados o si sólo algunas partes de los bloques de construcción se pueden usar).
3. Algunos bloques de procesos se pueden descartar con la debida aprobación.
4. Algunos bloques de procesos podrán ser sustituidos con la debida aprobación.

---

<sup>8</sup> BLOQUES DE PROCESOS hace referencia a fragmentos del proceso que se definen para poder ser empleados más de una vez en uno o varios procesos o con fines de organización.

5. Nuevos bloques de procesos se pueden añadir con la debida aprobación.

Para este caso, las necesidades del cliente y la filosofía de gestión proporcionan las principales entradas para las decisiones de adaptación. Las características del proyecto (tamaño, complejidad, formalidad y control), son entradas necesarias para elegir y adaptar el conjunto correcto de bloques de procesos para el proyecto.

En el paso de la selección de los artefactos en la guía de tailoring del proceso unificado, se tienen en cuenta las opciones básicas de la utilización de los bloques de procesos de [26]. Es importante aclarar que el enfoque de este trabajo no está orientado hacia bloques de proceso sino hacia los artefactos del proceso.

En [10] se realiza la selección de las entidades del proceso de acuerdo a una clasificación de cada entidad; esta clasificación consiste en determinar si una entidad del proceso se debe realizar, se debería realizar, se podría realizar y no se haría; entonces, dependiendo de cómo se categorice las entidades se definirá el proceso. Esta es una forma general de cómo se debe realizar la selección de las entidades del proceso software, pero no es una descripción clara, puesto que la forma de realizar la adaptación del proceso no se tiene en cuenta las características de los proyectos de forma explícita y solo brinda información general que puede ser útil, pero no es suficiente en el momento de realizar la adaptación. Por su parte [23] en la selección de las entidades del proceso tiene en cuenta cuatro tipos de procesos y tres criterios de adaptación, entonces, para seleccionar las actividades del proceso se realiza una matriz, donde la primera columna especifica condiciones en pseudocódigo que identifican las situaciones en las cuales determinadas actividades se deben llevar a cabo y las otras columnas representan los criterios para realizar la adaptación. En esta matriz se especifica una relación entre actividades y criterios de adaptación por medio de una clasificación de las actividades del proceso, cada actividad se identifica como obligatoria u opcional y sus respectivas revisiones se clasifican como formal o informal. Entonces, para establecer la relación entre los artefactos del proceso y las variables de caracterización, en la guía de tailoring del Proceso Unificado se optó por escoger y adaptar la clasificación brindada por [10], ver Tabla 3; además, se escogió la forma de representar la relación entre artefactos y variables de caracterización de [23], ver Tabla 4, donde se muestra la relación, por medio de la clasificación de los artefactos, entre los artefactos de una determinada disciplina y las variables de caracterización.

<b>CLASIFICACIÓN DE ARTEFACTOS</b>	
<b>Clasificación</b>	<b>Explicación</b>
SP	El artefacto se producirá, es un artefacto clave y puede causar problemas en el desarrollo si no es producido
NP	El artefacto no se producirá.
SR	El artefacto se producirá con restricciones (informal, sólo algunas partes del documento)
I	Es indiferente. La variable no afecta al artefacto

**Tabla 3. Clasificación de Artefactos. Tomado y adaptado de [10].**

En la Tabla 4, se plasma un ejemplo de la clasificación de artefactos vs. variables de caracterización, en ésta se puede verificar que la relación de las variables de caracterización (caracterización 1 y caracterización 2), se realiza a través de los niveles de caracterización (nivel de caracterización 1 y nivel de caracterización 2), y de la clasificación (SP, NP, SR, I) de cada uno de los artefactos (artefacto 1, artefacto 2 y artefacto 3) de una determinada disciplina (disciplina ejemplo). Debido a que el mismo artefacto de una disciplina puede tener diferentes clasificaciones en cada una de las variables de caracterización, ver la clasificación del artefacto 1 en la Tabla 4, surge la necesidad de unificar las clasificaciones de cada artefacto por medio de la comparación de las clasificaciones de las variables.

<b>MATRIZ DE CLASIFICACIÓN DE ARTEFACTOS POR DISCIPLINAS VS. VARIABLES DE CARACTERIZACIÓN</b>		
<b>Disciplina: Disciplina ejemplo</b>	<b>Variable de Caracterización 1</b>	<b>Variable de Caracterización 2</b>
<b>Artefactos</b>	<b>Nivel de Caracterización1</b>	<b>Nivel de Caracterización 2</b>
Artefacto 1	SP	NP
Artefacto 2	SR	SR
Artefacto 3	I	SP

**Tabla 4. Ejemplo de Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización. Tomado y adaptado de [23].**

Según [41], los artefactos candidatos a ser parte del proceso deben cumplir por lo menos los siguientes criterios:

- El artefacto comunica un acuerdo de conocimiento importante en situaciones donde el tipo de comunicación más simple, como la verbal, no es práctica, como en el caso de equipos grandes o distribuidos, o crea un gran riesgo para proyecto.
- El artefacto permite adquirir el conocimiento necesario, a las personas que entran al equipo durante la ejecución del proyecto, de una forma más fácil y rápida.
- La inversión en un artefacto traerá un retorno a largo plazo, en el mantenimiento del sistema, o el artefacto será utilizado varias veces durante el desarrollo.
- El artefacto es impuesto por la organización, por el cliente o por cualquier norma adoptada.

Realizar la selección de los artefactos no es suficiente en un proceso, pero son el soporte para identificar otras entidades que serán parte del proceso, por ejemplo conociendo los artefactos, se consigue identificar las actividades que no producen

artefactos, de esta forma se puede considerar si se realiza o no dicha actividad, pues no tendría sentido efectuar una actividad que no produzca algún artefacto o también pueden resultar actividades que produzcan artefactos que no serán tenidos en cuenta en el proceso; entonces de esta manera la selección de los artefactos incide en la selección de las actividades. Un proceso deficiente puede involucrar actividades innecesarias que inducen a desperdiciar tiempo y dinero, o la omisión de actividades que son necesarias [4].

Para aumentar la eficiencia del proceso a una situación específica [8] propone los siguientes enfoques:

1. Considerar el proceso con  $n$  pasos y mejorar la eficiencia de cada uno de estos pasos.
2. Considerar el proceso con  $n$  pasos y eliminar algunos pasos innecesarios, transformando el proceso en un proceso con  $m$  pasos, donde  $m < n$ .
3. Considerar el proceso con  $n$  pasos y utilizar mayor competencia en las actividades realizadas y recursos aplicados en cada paso.

La selección de los artefactos en la guía de tailoring, tiene en cuenta los dos primeros enfoques mencionados anteriormente, el primer enfoque se utiliza cuando las entidades del proceso se realizan de una forma simplificada o con un menor grado de formalidad, es decir se mejora la eficiencia de la ejecución de estas entidades del proceso. El segundo enfoque se utiliza después de realizar la selección de los artefactos que harán parte del proceso, donde es posible considerar la no realización de actividades, o sea, se considera la eliminación de algunos pasos del proceso. Es importante mencionar que el tercer enfoque no se tiene en cuenta en la guía de tailoring, puesto que utilizar una mayor competencia en las actividades realizadas y recursos aplicados en el proceso, puede ser lujo que las PyMEs\_DS no pueden soportar, debido a que cuentan con características que pueden limitar el empleo de más competencia en el proceso.

Por último, es necesario decir que el ejecutor de la adaptación juega un papel importante en la selección de las entidades del proceso, puesto que su conocimiento, experiencia y talento son un aporte trascendental para realizar la adaptación.

En resumen, la guía de tailoring de UP para realizar la selección de los artefactos se tiene en cuenta los estudios, conceptos y enfoques mencionados en esta sección. La descripción de cómo realizar la selección de los artefactos se detalla en la sección 3.3.2 *Segundo paso – Selección de las entidades del proceso*.

### **3.5 Configuración del proceso**

El Proceso Unificado en su intento de ser aplicado a múltiples proyectos posee una descripción general del proceso de desarrollo, la cual no se ajusta a las necesidades de los proyectos específicos de las PyMEs\_DS, y por lo tanto es imperioso efectuarle algún tipo de configuración (modificación). Efectuar la configuración al proceso es necesaria, pues por medio de ésta se materializa la adaptación del proceso, es decir,

la configuración es una actividad inevitable en la cual se efectúan los cambios necesarios que debe sufrir el proceso para cumplir con las características del proyecto específico.

En [19][17], para realizar la configuración del proceso se utilizan las operaciones de adición, borrado, división y fusión, de las cuales en la guía de tailoring, para realizar la configuración del proceso, se tuvieron en cuenta solo las operaciones de borrado o eliminación y adición de entidades, y se definió la operación de reemplazo cuyo propósito es realizar la sustitución de una entidad en el proceso de desarrollo. Por su parte en [12] se definen una serie de mecanismo los cuales pueden ayudar a realizar la configuración del proceso, estos mecanismos se pueden aprovechar gracias a que en este trabajo se realizo una implementación del Proceso Unificado a través de SPEM, el resultado de la implementación es un plugin de UP llamado Unified Process, para más detalles ver sección 2.3 *El Proceso Unificado descrito a través de SPEM 2.0*. Esta implementación permite que se puedan reutilizar elementos del contenido del plugin con ciertas modificaciones por medio de los mecanismos de variabilidad, que permiten cambiar elementos de método o de proceso sin modificar directamente el original. La variabilidad de un elemento de método permite definir diferencias (adiciones, cambios, omisiones) con el elemento original.

Para realizar la configuración del proceso en la guía de tailoring se han definido una serie consideraciones que pretenden orientar algunos aspectos que se deben apreciar cuando se efectúan las operaciones de adaptación de entidades del proceso. Por lo tanto la empresa tiene la posibilidad de realizar cambios al proceso que pueden clasificarse como: adición, eliminación y reemplazo.

### **3.5.1 Operación de Eliminación**

Desde que los estándares de procesos son escritos para proyectos generales de software, la operación de eliminación puede ser la más común en la adaptación de procesos [17]. La operación de eliminación es definida como la supresión de una entidad (tarea, artefacto, rol) en el proceso de desarrollo[19]. Para eliminar una entidad se debe considerar qué entidades del proceso son afectadas por dicha eliminación, por ejemplo, cuando una actividad es eliminada, los artefactos generados desde la actividad también tienen que ser eliminados.

#### **3.5.1.1 Consideraciones al realizar la eliminación de entidades.**

Debido al enfoque de la guía de tailoring centrado en la elección de los artefactos que harán parte de las disciplinas, la operación de eliminación tiene el mismo enfoque, puesto que primero se considerará la eliminación de artefactos, para posteriormente verificar, si es posible, la eliminación de otras entidades del proceso como tareas y roles.

##### **3.5.1.1.1 Eliminación de artefactos.**

Más que eliminar el artefacto, es necesario también identificar las relaciones que tiene con otras entidades del proceso, para saber cómo la eliminación del artefacto afectará a otras entidades del proceso; entonces, en la eliminación de un artefacto es preciso tener en cuenta las siguientes relaciones:

- 1) Relaciones con tareas
- 2) Relaciones con roles

Las relaciones entre artefactos y tareas se pueden observar en los anexos (A.1, A.2, A.3, A.4, A.5), donde es fácil identificar las relaciones para poder efectuar la eliminación, conociendo las tareas que serán modificadas, entonces, es necesario eliminar el artefacto de las tareas afectadas ya sea de entrada o salida, y de esta forma eliminar las relaciones entre estas dos entidades del proceso.

La eliminación de las relaciones entre artefactos y roles es similar al caso anterior, en los anexos (A.6, A.7, A.8, A.9, A.10) se plasman las relaciones que tienen los artefactos con los roles en las disciplinas. Entonces, conociendo las relaciones entre estas dos entidades se puede realizar la eliminación del artefacto sin que se generen ambigüedades en el proceso, ya que los roles no tendrán ningún tipo de relación con los artefactos eliminados, lo que puede significar una disminución en las responsabilidades del rol debido a que tendrá menos artefactos bajo su responsabilidad.

Es preciso enfatizar, que todas las relaciones del artefacto a eliminar deben desaparecer, puesto que, si existen entidades del proceso que tienen relaciones con los artefactos que no serán producidos, podría generar entorpecimiento en la realización del proceso y causar confusión entre los ejecutores del mismo.

#### **3.5.1.1.2 Eliminación de tareas y roles**

Es necesario aclarar que la eliminación de tareas y roles dependen de la eliminación de los artefactos, pues la supresión de estas dos entidades es una consecuencia de efectuar la eliminación de artefactos debido a que, para eliminar tareas y roles es necesario que por su parte las tareas no produzcan ningún tipo de artefacto requerido por otras tareas y que los roles no tengan responsabilidad por artefactos o tareas, es decir, que tanto tareas como roles no posean relaciones con otras entidades del proceso.

#### **3.5.2 Operación de adición**

Es definida como la adición de una entidad al proceso de desarrollo de software [17]. La operación de adición está clasificada en: adición de entidades del estándar y adición de entidades definidas por el usuario. Las entidades del estándar son aquellas que ya están definidas en el estándar de procesos, y las entidades definidas por el usuario son aquellas que son definidas por un diseñador de procesos.

##### **3.5.2.1 Consideraciones al realizar la adición de entidades**

La adición de entidades del proceso es una actividad que brinda la opción de añadir entidades del estándar y adición de entidades definidas por el usuario. Cuando se realiza la adición de entidades en el proceso de desarrollo, se deben tener en cuenta que las entidades del proceso pueden ser roles, tareas y artefactos, en cada caso es

importante considerar algunos aspectos que pueden ser cruciales en el momento de realizar dicha adición.

Antes de mencionar los aspectos que se pueden considerar en la adición de las entidades del proceso es imprescindible recordar la relación intrínseca que tienen estas entidades. Un artefacto no puede ser adicionado por sí solo, siempre debe haber una tarea que lo produzca o lo utilice y una tarea no puede ser adicionada sin que produzca o modifique algún tipo de artefacto. Por su parte un rol no puede ser adicionado sin que sea responsable de alguna tarea o artefacto.

Algunos aspectos que pueden considerarse en la adición de entidades del proceso son descritos a continuación.

#### **3.5.2.1.1 Adición de roles.**

Un rol define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos trabajando juntos como un equipo. Las responsabilidades de un rol son tanto el llevar a cabo un conjunto de actividades como el ser el responsable de un conjunto de artefactos [8].

Es imprescindible que en la adición de los roles se cumpla con los aspectos mencionados en la definición de este elemento del proceso, entonces se debe definir como mínimo lo siguiente:

- 1) Especificar las habilidades necesarias.
- 2) Especificar las tareas que realiza.
- 3) Especificar los artefactos que son responsabilidad del rol.

En el primer caso, es necesario especificar las destrezas que debe tener la persona o el grupo de personas que van a desempeñar este rol, puesto que para la ejecución exitosa del proceso se debe contar con el personal idóneo que pueda cumplir con las expectativas plasmadas en las habilidades del rol.

En la especificación de las tareas que realiza y artefactos que son responsabilidad del rol, es donde se asignan los compromisos y relaciones del rol, se deben especificar las tareas y los artefactos que están bajo su responsabilidad. Es importante señalar que la descripción de las responsabilidades del rol es necesaria, puesto que carecería de valor adicionar un nuevo rol si éste no tiene ningún tipo de relación (responsable o ejecutor) con algunas de las entidades del proceso, es decir el rol debe adquirir responsabilidades con el proceso de desarrollo, ver Figura 24, en la parte de relaciones.

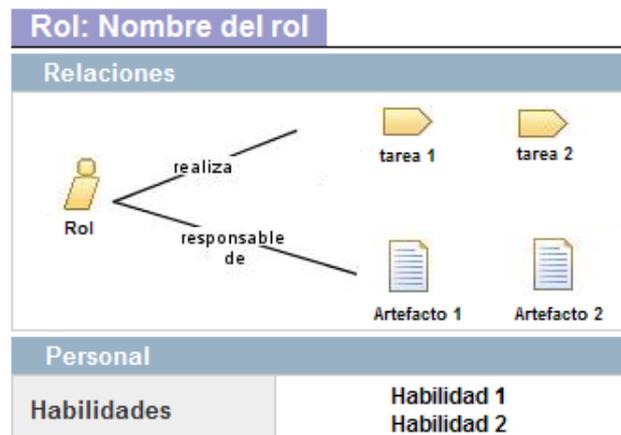


Figura 24 Elementos necesarios en la definición del Rol

### 3.5.2.1.2 Adición de tareas.

Una tarea en concreto es una unidad de trabajo que una persona que desempeñe un rol puede realizar. Las tareas tienen un objetivo concreto, normalmente expresado en términos de crear o actualizar algún producto. Cada tarea es asignada a un rol específico, y afecta a uno o un número pequeño de artefactos.

En la adición de una tarea es necesario que establezca como mínimo lo siguiente:

- Especificar el objetivo de la tarea.
- Especificar los artefactos de entrada y salida.
- Especificar el rol responsable.

La especificación del objetivo de la tarea es necesaria, ya que permite identificar los productos de trabajo que serán creados o actualizados en la ejecución, y además, orienta la ejecución del proceso, puesto que permite la valoración de la ejecución de la tarea en términos del cumplimiento de sus objetivos.

Establecer las relaciones de la tarea es uno de los aspectos importantes que hay que tener en cuenta cuando se hace la adición, es decir, especificar las entradas, salidas y el rol responsable de dicha tarea, ver Figura 25.

Tarea: Nombre tarea	
Objetivo	
El objetivo de esta tarea es...	
Relaciones	
Roles	Realizador principal: <ul style="list-style-type: none"> <li>• Rol</li> </ul>
Entradas	Obligatoria: <ul style="list-style-type: none"> <li>• Artefacto 1</li> <li>• Artefacto 2</li> </ul>
Salidas	<ul style="list-style-type: none"> <li>• Artefacto 1</li> <li>• Artefacto 2</li> <li>• Artefacto 3</li> </ul>

**Figura 25 Elementos necesarios para la definición de una tarea.**

Cuando se desea adicionar una entidad dentro del proceso de desarrollo, se tiene que considerar dónde la entidad es adicionada y qué tipo de relación hay entre la nueva entidad y la entidad adyacente [19]. Entonces es determinante conocer como la adición va alterar o modificar el proceso, puesto que en la adición de la tarea previamente se debe definir en qué parte del proceso será adicionada; para realizar la adición en este trabajo sólo se tendrán en cuenta las tareas posteriores de la tarea a adicionar, donde se pueden presentar los siguientes casos:

- 1) La tarea adicionada produce artefactos que son entrada a tareas posteriores.
- 2) La tarea adicionada no afecta tareas posteriores (caso en el que ésta sea adicionada al final de un flujo de tareas).

En la siguiente sección se ilustran los casos mencionados anteriormente y se dan posibles soluciones a cada uno de ellos

- **Caso 1: La tarea adicionada produce artefactos que son entrada a tareas posteriores**

En este caso entonces, es necesario que en la definición de la tarea se precisen los artefactos que serán producidos o modificados y que afectaran las tareas posteriores, es decir, los artefactos que son el resultado de la nueva tarea y que serán consumidos por las tareas posteriores, esto implica modificar las tareas posteriores en sus entradas para poder establecer las relaciones de las tareas por medio de los artefactos producidos por la tarea adicionada y consumidos por las tareas posteriores.

- **Caso 2: La tarea adicionada no afecta tareas posteriores.**

La relación entre actividades es necesaria porque una actividad suelta, sin relaciones, no aportaría a la ejecución exitosa del proceso. Es importante mencionar que pueden haber tareas que no tengan relaciones, no afectan tareas posteriores, este tipo de tareas se pueden encontrar al final de un flujo de trabajo, cuyo resultado será cumplir a cabalidad o parcialmente con el objetivo del flujo de tareas. En este caso es importante evaluar si la tarea

adicionada está generando valor al flujo de trabajo en términos de conseguir los resultados esperados.

Los artefactos de entrada de la tarea adicionada pueden ser completamente nuevos o pueden estar definidos por el proceso, si es el caso de que sean nuevos hay que definirlos según la adición de artefactos que se muestra en el siguiente apartado.

### 3.5.2.1.3 Adición de Artefactos.

Un producto o artefacto es un trozo de información que es producido, modificado o usado durante el proceso de desarrollo de software. Los productos son los resultados tangibles del proyecto, las cosas que va creando y usando hasta obtener el producto final [8]. Los roles utilizan los artefactos como entrada para desarrollar una actividad, y también son el resultado o salida de dichas actividades. En la adición de artefactos se debe especificar lo siguiente:

- Especificar el objetivo del artefacto.
- Especificar las tareas donde el artefacto es entrada y salida.
- Especificar el rol responsable.

La especificación del objetivo del artefacto es inevitable, puesto que permite identificar como el artefacto ayuda parcialmente a alcanzar los objetivos del proceso.

Indicar las relaciones del artefacto es un aspecto trascendental en el momento de efectuar la adición, puesto que un artefacto no aportaría valor si no tiene relaciones, es decir, si no es entrada a alguna tarea que lo utilice o lo modifique. Por lo tanto, es necesario definir las tareas a las cuales el artefacto será entrada, salida y también definir el rol a cargo de dicho artefacto, para de esta manera establecer las relaciones entre las entidades del proceso, ver Figura 26.

Artefacto: Nombre del Artefacto		
Objetivo		
El objetivo de este artefacto es...		
Relaciones		
Roles	Responsable:	
	<ul style="list-style-type: none"> <li>• Rol 1</li> </ul>	
Tareas	Entrada a:	Salida de:
	<ul style="list-style-type: none"> <li>• Actividad 1</li> <li>• Actividad 2</li> </ul>	<ul style="list-style-type: none"> <li>• Actividad 3</li> <li>• Actividad 4</li> </ul>

**Figura 26 Elementos necesarios para adicionar un artefacto.**

En la adición de un artefacto se puede tener incidencias parecidas a las de la adición de una tarea, donde se pueden presentar los siguientes casos:

- El artefacto está relacionado con tareas posteriores
- El artefacto no es entrada a tareas posteriores.

Como la adición de tareas y artefactos tienen semejanza entonces las posibles soluciones serán parecidas (en la siguiente sección se describen soluciones a cada uno de los casos mencionados anteriormente).

- **Caso1: El artefacto está relacionado con tareas posteriores**  
En este caso, es necesario precisar cuáles tareas posteriores consumen el artefacto para modificar sus entradas, y de esta manera establecer las relaciones entre tareas y el artefacto adicionado. Similar a lo que sucede con las tareas, un artefacto suelto sin relaciones no aporta nada a la ejecución del proceso.
- **Caso 2: El artefacto no es entrada a tareas posteriores.**  
De la misma forma que en las tareas, este tipo de artefactos se pueden encontrar al final de un flujo de trabajo, cuyo resultado será cumplir a cabalidad o parcialmente con objetivo del flujo de tareas, entonces para este caso es importante evaluar si el artefacto adicionado está generando valor al flujo de trabajo en términos de conseguir los resultados esperados.

### 3.5.2.2 Operación de adición vista como el mecanismo de “variabilidad de contribución de SPEM”.

Desde otra perspectiva, la adición de entidades “nuevas” al proceso se puede ver como una contribución de estas entidades a las entidades existentes en la definición del proceso. Las reglas definidas para la adición como contribución están derivadas sintáctica y semánticamente de la variabilidad de contribución en SPEM 2.0 [12].

La idea de la adición derivada de la contribución de SPEM 2.0, además de utilizar sus reglas para realizar la adición, es brindar una solución que sirva para evitar las complicaciones que pueden surgir entre las relaciones de las entidades del proceso en el momento de adicionar nuevas entidades, ya que por medio de este mecanismo sólo añade atributos y asociaciones a la base del proceso y por lo tanto nunca invalida o sustituye alguna información de la base del proceso.

Una entidad E que se adiciona a una entidad denotada EB de la base del proceso, añade sus valores de atributos e instancias de asociación a EB, sin modificar directamente las propiedades que ya tiene EB (es una adición). Como resultado de realizar la adición la entidad EB se muestra combinada con los atributos y relaciones de E. Las reglas que rigen dicha combinación son:

- El valor de los atributos de E se concatena al final del contenido de los correspondientes atributos de EB.
- Las instancias de asociaciones de entrada o de salida con cardinalidad “muchos” de E se añaden a EB.
- Las instancias de asociaciones de entrada y de salida con cardinalidad “1” (como la relación “realizador principal” entre una tarea y un rol) definidas en E se ignoran si ya existen en EB. En caso contrario, se añaden.
- Una entidad de la base del proceso puede recibir varias adicciones.
- La adición es transitiva.

Algunos ejemplos de cómo se realiza la interpretación de las reglas que rigen la variabilidad de contribución se encuentran en el anexo C.18.

### 3.5.3 Operación de Reemplazo.

Esta operación es definida como la sustitución de una entidad en el proceso de desarrollo. La operación de reemplazo tiene como objetivo intercambiar entidades nativas<sup>9</sup> de la empresa por las definidas en el proceso, esta operación intenta rescatar y reutilizar entidades de la empresa para no someterla a cambios drásticos en la forma como desarrollan o ejecutan este tipo de entidades.

En el reemplazo de entidades del proceso, el principal problema es cómo realizar el ensamble de las entidades del proceso<sup>10</sup>, puesto que el ensamble es una actividad clave en la integración de varios activos de proceso. En realidad el ensamble, corresponde a la operación de reemplazo de una actividad de un proceso por la de otro (posiblemente emergente de la misma organización). Debido a lo anterior en la siguiente sección se hará mención al reemplazo como el ensamble de actividades y homologación de artefactos.

#### 3.5.3.1 Consideraciones en el ensamble de actividades y homologación de Artefactos

Es importante mencionar que el enfoque del ensamble de actividades descrito a continuación, se concentra en la actividad previa y posterior de la actividad a reemplazar.

Antes de realizar el ensamble de las actividades, es necesario verificar la compatibilidad entre dichas actividades, para luego centrarse solo en el ensamble de éstas y en la homologación de los artefactos. Para el ensamble de actividades del proceso se utilizan como base algunas incidencias de los mecanismos de ensamble de componentes de procesos<sup>11</sup> definidos en [SPEM] debido a la similitud entre estos dos elementos, ver Tabla 5.

ACTIVIDAD VS. COMPONENTE		
	Actividad	Componente
<b>Descripción</b>	Una actividad representa una unidad de trabajo general en un proceso	Un componente de proceso contiene un proceso representado por una actividad

<sup>9</sup> ENTIDADES NATIVAS se definen como tareas, artefactos y roles propios del proceso de la empresa.

<sup>10</sup> ENSAMBLE DE ENTIDADES DEL PROCESO hace referencia a la conexión de entidades del proceso por medio de la homologación de las relaciones entre entidades.

<sup>11</sup> COMPONENTES DE PROCESOS son paquetes especiales de procesos que por lo general son utilizados cuando la empresa realiza sub-contratación de cierta parte del proceso de desarrollo.

ACTIVIDAD VS. COMPONENTE		
	Actividad	Componente
<b>Entradas y Salidas</b>	Cada actividad tiene artefactos de entrada y salida, y un objetivo claro expresado comúnmente en términos de crear o actualizar artefactos.	Un componente define un conjunto de puertos de productos de trabajo requeridos o suministrados, los cuales definen los tipos de productos de trabajo usados, producidos, o modificados por el proceso.
<b>Estructura</b>	Una actividad puede tener una estructura interna formada por agregación de elementos, que pueden ser de varios tipos, no solo de trabajo: actividades más simples (anidamiento) y elementos de método.	Un componente puede asumir una estructura interna compuesta por componentes de procesos más simples.

**Tabla 5. Actividad vs. Componente.**

Para realizar el ensamble de actividades primero se debe encapsular conceptualmente el método o el proceso interno de las actividades, es decir, no importarán los pasos o técnicas de cómo se realizan las actividades involucradas en el reemplazo. De esta forma se utiliza el mecanismo de ensamble basado en el principio de caja negra, el cual nos permite centrarnos en el ensamble de las actividades por medio de la homologación de entradas y salidas.

La Figura 27 muestra una actividad del proceso llamada desarrollo (Development) que define dos productos de entrada ‘Modelo de Análisis’ y ‘Modelo de Diseño’ así como una salida ‘Aplicación’.



**Figura 27 Representación gráfica de una actividad**

El ensamble de las actividades del proceso inicia simplemente por conectar las actividades, a través de sus entradas y salidas. Esto parece trivial, pero hay varios problemas que se deben resolver.

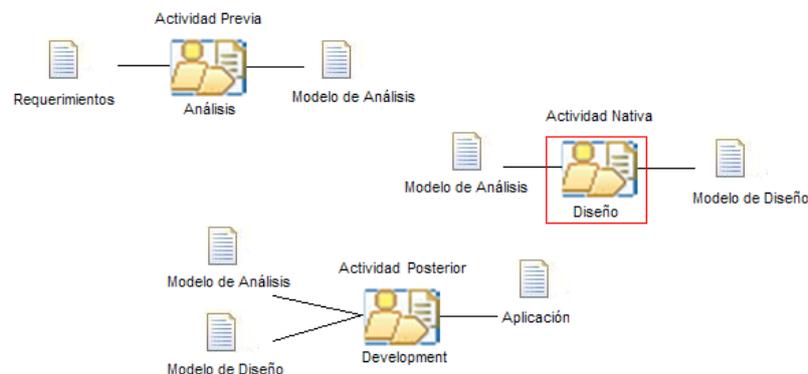
- 1) El número y el tipo de productos de trabajo de entrada y salida podría no corresponder entre las actividades.
- 2) Los nombres y la constitución de los artefactos podría variar entre actividades.
- 3) El nombre y la definición de los roles podría variar entre las actividades.

No existe la garantía que las actividades correspondan, en particular si las actividades son complejas y originadas desde fuentes que tienen diferentes métodos de

desarrollo, la concordancia puede ser imposible; sin embargo, para ensamblar y mapear diferentes actividades del proceso, se utilizará un mecanismo de ensamble similar a la notación de puertos UML 2.0<sup>12</sup> con el fin de ajustar las variaciones que puedan existir entre las actividades del proceso. En los componentes de proceso se definen puertos de entrada y salida, los cuales corresponden a los artefactos de entrada y salida de las actividades en SPEM.

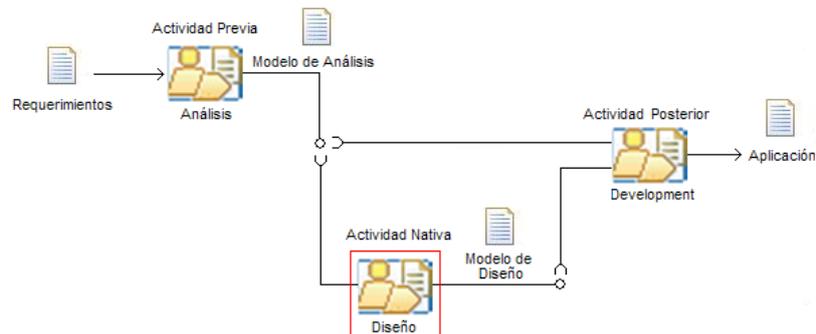
Con el fin de expresar las relaciones que se necesitan establecer entre las actividades del proceso, se utilizarán líneas como conectores entre actividades; es importante recordar que una actividad puede tener entradas ó productos de trabajo opcionales, que no necesitan ser ensamblados, pero si es necesario ensamblar las entradas ó producto de trabajo obligatorios.

El esfuerzo necesario para realizar el ensamble de las actividades de la Figura 28 puede ser insignificante, puesto que la actividad nativa que será ensamblada tiene correspondencia en nombres, entradas y salidas con la actividad previa y posterior, de esta forma efectuar el ensamble de las actividades es trivial, pues solo bastaría con efectuar el ensamble de entradas y salidas entre las actividades como se muestra en la Figura 29, pero pueden existir problemas que pueden interferir en el montaje de las actividades, como la incompatibilidad de nombres y correspondencias en las entradas o productos de trabajo de dichas actividades. En esta situación, un mayor esfuerzo es necesario para ensamblar los elementos existentes en la configuración de las actividades.



**Figura 28 Actividades a ser ensambladas**

<sup>12</sup> NOTACIÓN DE PUERTOS UML 2.0 se refiere a la representación de entradas y salidas utilizada en el ensamble de componentes de proceso.



**Figura 29 Ensamble de las Actividades Previa, Posterior y Nativa.**

Para realizar el ensamblaje de las actividades previa, posterior y nativa, es reiterativo mencionar que se debe encapsular el método ó el proceso interno de las actividades, para de esta manera utilizar el mecanismo de ensamblaje basado en el principio de caja negra, el cual nos permite concentrarnos en las entradas ó productos de trabajo de las actividades.

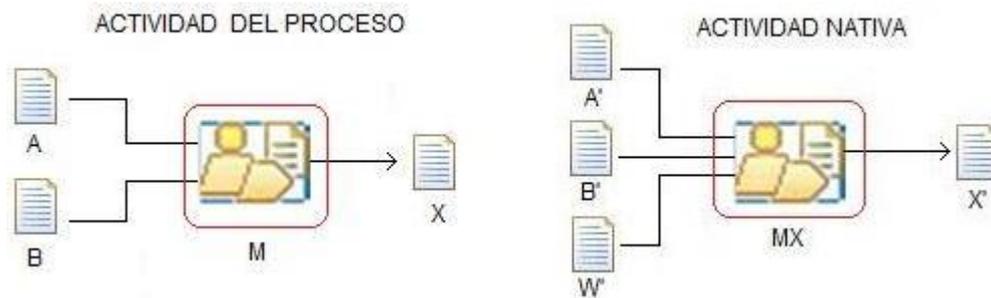
Cuando se realiza el ensamblaje de actividades podemos enfrentar varias situaciones y problemas (que por motivos de alcance de este proyecto no serán abordados), pero si se explicaran y ofrecerán soluciones a algunos casos básicos que pueden servir de ayuda en momento de reemplazar actividades en el proceso. Algunos casos típicos que pueden pasar en el ensamblaje de actividades son descritos a continuación:

- La actividad reemplazante<sup>13</sup> o nativa requiere de una entrada no suministrada por la actividad previa del proceso.
- La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa.
- La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso.
- La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior.

En la siguiente sección se ilustran los casos mencionados anteriormente y se dan posibles soluciones a cada uno de ellos. Se muestra una gráfica por cada caso, donde 'M' denota la actividad del proceso (actividad a ser reemplazada), y 'MX' denota la actividad nativa o reemplazante (actividad que sustituirá a otra en el proceso). Es importante destacar que tanto la actividad 'M' y 'MX' tienen sus artefactos de entrada, provenientes de la actividad previa, y de salida, suministrados por estas actividades a las actividades posteriores.

- **Caso 1: La actividad reemplazante o nativa, ver Figura 30, requiere de una entrada no suministrada por la actividad previa del proceso.**

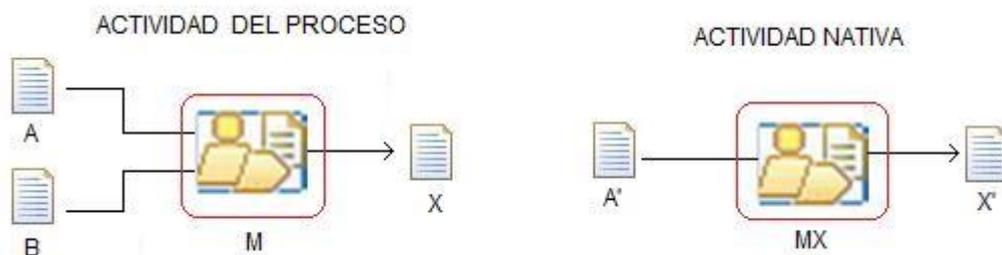
<sup>13</sup> ACTIVIDAD REEMPLAZANTE. Se define como una actividad propia del proceso de la empresa la cual sustituirá a una actividad del patrón de proceso.



**Figura 30 Actividad reemplazante o nativa, requiere de una entrada no suministrada por la actividad previa del proceso.**

Una posible solución para este caso puede ser ensamblar una nueva actividad que suministre el elemento perdido ó que la actividad previa se adecue para desarrollar la entrada para la actividad nativa.

- **Caso 2: La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa. Ver Figura 31.**

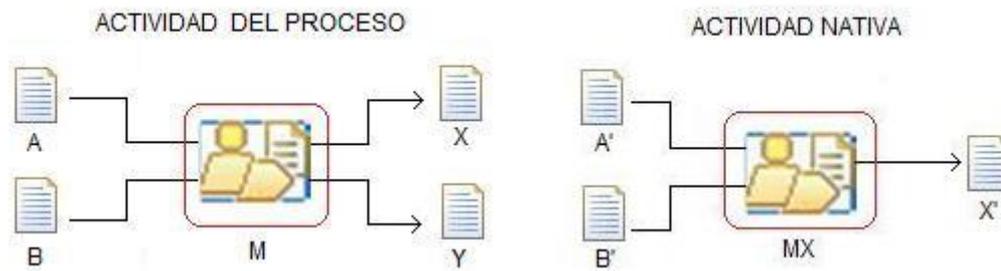


**Figura 31 La actividad reemplazante o nativa no requiere de una entrada que es suministrada por la actividad previa.**

En este caso pueden surgir dos acontecimientos, que la actividad previa tenga una sola salida y por lo tanto quede huérfana, o que la actividad tenga otras relaciones con la actividad nativa y solo se pierda una relación.

En el primer acontecimiento es posible borrar la actividad previa, pues esta va a carecer de relaciones con la actividad nativa (y hacia atrás todas las demás actividades huérfanas). Esto queda bajo responsabilidad de la persona encargada de hacer la adaptación, y en el segundo acontecimiento se puede evaluar la realización de dicho artefacto por el cual están relacionadas las actividades, ya que carecería de sentido realizar un artefacto que no se utilice en el proceso.

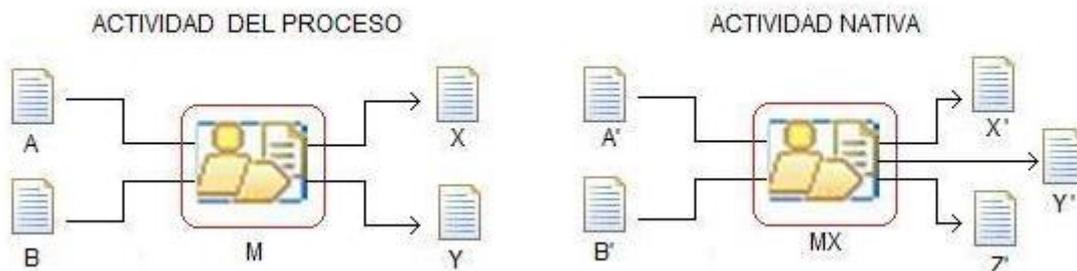
- **Caso 3: La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso. Ver Figura 32.**



**Figura 32** *La actividad reemplazante o nativa no suministra una salida para la actividad posterior del proceso.*

También para este caso, una posible solución puede ser ensamblar una nueva actividad que suministre el elemento perdido ó que la actividad previa, en este caso la nativa, se adecue para desarrollar la entrada para la actividad posterior.

- **Caso 4:** *La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior, ver Figura 33.*



**Figura 33** *La actividad reemplazante o nativa suministra una salida que no es requerida por la actividad posterior*

En este caso se puede borrar el producto de salida huérfano, previa evaluación de lo que implicaría la no ejecución de este artefacto, puesto que carecería de sentido realizar un artefacto que no se utilice en el proceso.

Las soluciones planteadas para los casos básicos pueden utilizarse o pueden tenerse en cuenta en casos de reemplazo de actividades más complejos, por ejemplo cuando hay una actividad nativa que tiene un producto de entrada y suministra una salida, mientras que la actividad del proceso tiene dos productos de trabajo de entrada y salida. En particular para dar solución a este ejemplo se pueden utilizar paso a paso los casos básicos 2 y 3, de esta manera se pretende mostrar que los casos base pueden servir de solución a casos de reemplazo más complejos.

### 3.5.3.2 Homologación de Artefactos

En la homologación de artefactos entre la actividad del proceso y la actividad nativa pueden ocurrir los siguientes casos:

- El nombre de los artefactos puede variar entre la actividad del proceso y la nativa.
- Los artefactos pueden integrarse o separarse para permitir el ensamble entre actividades.
- La estructura de los artefactos es distinta.

En la siguiente sección se dan posibles soluciones a los casos mencionados anteriormente.

- **Caso 1: *El nombre de los artefactos puede variar entre la actividad del proceso y la nativa.***

Inicialmente hay que evaluar si corresponde al mismo artefacto, puesto que puede variar el nombre pero su propósito sea el mismo, lo cual sería el caso más trivial donde la homologación sería directa siempre y cuando se establezca una relación entre los nombres, que puede ser concatenarlos (para evitar ambigüedades en el proceso). Pero también se puede dar el caso en el que los nombres correspondan pero su propósito sea diferente, entonces para este caso es necesario llegar a un consenso entre el artefacto nativo y el del proceso, es decir, tratar de unificar los artefactos para que el cambio no sea tan traumático dentro del proceso de la organización.

- **Caso 2: *Los artefactos pueden integrarse o separarse para permitir el ensamble entre actividades.***

Puesto que varios artefactos nativos pueden conformar un solo artefacto del proceso o viceversa, se puede tratar de integrar o dividir los artefactos en función de cumplir el propósito del artefacto a ser reemplazado.

- **Caso 3: *La estructura de los artefactos es distinta.***

Cuando la estructura del artefacto es distinta, es importante enfocarse en el objetivo del artefacto puesto que, así tengan estructuras diferentes, pueden cumplir con el mismo propósito, lo que facilitaría la homologación. Dado el caso en el que las estructuras sean distintas, se puede recurrir a la misma solución del caso 1 de unificar los artefactos.

### 3.5.3.3 Operación de reemplazo vista como el mecanismo de “variabilidad reemplaza de SPEM”

En el reemplazo de una entidad de un proceso por la de otro se pueden considerar las reglas definidas para el reemplazo de la variabilidad reemplaza<sup>14</sup> de SPEM 2.0 [12].

---

<sup>14</sup> VARIABILIDAD REEMPLAZA Mecanismo de SPEM 2.0 que permite realizar el reemplazo de entidades del proceso.

Una entidad E puede reemplazar (sustituir) los atributos y las asociaciones de salida de una entidad denotada EB de la base del proceso, sin modificar directamente ninguna propiedad de EB. Las reglas que regulan el reemplazo son:

- Los valores de los atributos de EB se reemplazan por los respectivos valores de los atributos no vacíos de E.
- Las instancias de asociación de salida de EB se reemplazan por las de E.
- Las instancias de asociación de entrada con cardinalidad “muchos” de EB se amplían con las instancias de dichas asociaciones existentes para E.
- Las instancias de asociación de entrada con cardinalidad “uno” de EB se dejan intactas. Las instancias incluidas en E que no están en EB se añaden a EB.
- Una entidad base sólo puede ser reemplazado por otra única entidad.
- El reemplazo es transitivo.

Algunos ejemplos de cómo se realiza la interpretación de las reglas que rigen la variabilidad de reemplaza se encuentran en el anexo C.19

### **3.6 Desplegar el proceso adaptado**

Debido a la utilización de SPEM en este trabajo como lenguaje de modelado del Proceso Unificado, se puede disponer de capacidades para facilitar la comprensión y comunicación humana, facilitar la reutilización, dar soporte a la mejora de procesos, dar soporte a la gestión del proceso y guiar la automatización del proceso; además de brindar la posibilidad de documentar, presentar, administrar, intercambiar, mantener, publicar e implementar procesos y generar automáticamente documentación en formato para la web.

En la guía de tailoring se utilizó SPEM y Eclipse Process Framework Composer (EPFC) como facilitadores para ejecutar la adaptación y el despliegue del proceso<sup>15</sup>. Es importante mencionar que el EPFC tiene dos propósitos principales:

- Proveer a los desarrolladores con una base de conocimiento de activos que les permita buscar, administrar y desplegar contenido. Esta base de conocimiento puede ser usada como referencia y material educativo, y forma la base para el proceso de desarrollo (el segundo propósito). El EPFC está diseñado para ser un administrador de contenido que provee una estructura de administración y un aspecto común para todo el contenido, en vez de ser un sistema administrador de documentos en el cual se almacenan y se acceden documentos difíciles de mantener, que tienen cada uno su propia forma y formato.
- Proveer capacidades de ingeniería de procesos para la selección, adecuación y rápido ensamblado de procesos para proyectos de desarrollo concretos.

En el despliegue se muestra el resultado de aplicar la guía de tailoring al patrón de procesos UP (Plugin Unified Process), es decir se hace disponible el proceso para su

---

<sup>15</sup> DESPLIEGUE DEL PROCESO hace referencia a la publicación del proceso.

futura utilización donde EPFC sirve para editar<sup>16</sup> el plugin *Unified process* y generar automáticamente el proceso adaptado.

El despliegue del proceso es importante para los equipos de desarrollo, puesto que estos necesitan definir la forma de aplicar sus métodos de desarrollo y mejores prácticas a lo largo del ciclo de vida del proyecto. En otras palabras, es necesario definir y conocer el proceso de desarrollo. Además los equipos de desarrollo requieren de conocimiento actualizado para aprender sobre el proceso y necesitan soporte para ejecutarlo correctamente. Algunos de los problemas que pueden enfrentar los equipos de desarrollo si el proceso no es publicado o no se tiene un fácil acceso a él pueden ser: interpretación errónea del proceso, re-trabajo, desconocimiento general del proceso, y de responsabilidades, etc. Entonces, para la empresa es trascendental desplegar el proceso que regirá la ejecución del proyecto, pues es imperioso, para los miembros del equipo conocer el proceso y de esta forma poder enfrentarlo de un mejor modo. Desplegar el proceso también es importante para la guía de tailoring, porque es una de las formas verídicas de mostrar el resultado de la ejecución de la misma.

### **3.7 Resumen de la guía de Tailoring**

La guía de tailoring del proceso unificado a proyectos específicos (GTUP) tiene como objetivos:

- Guiar la planeación y ejecución del tailoring del proceso software UP a proyectos específicos de las PyMEs\_DS.
- Suministrar elementos que faciliten la adaptación de UP a proyectos específicos.

Es importante mencionar que la guía de tailoring no pretende reemplazar a la(s) persona(s) encargada(s) de efectuar la adaptación, puesto que el conocimiento, la experiencia y el talento de esta(s) persona(s) es un aporte trascendental para realizar la adaptación de forma exitosa.

Como se muestra en la Figura 34, GTUP está conformada por cuatro pasos: primero realizar la caracterización del proyecto, segundo seleccionar las entidades del proceso, tercero realizar la configuración del proceso y cuarto efectuar el despliegue del proceso. La adaptación del proceso bajo la elección del conjunto de artefactos que harán parte del proceso adaptado de cada disciplina hace que GTUP sea iterativa en el segundo y tercer paso, pues se ejecutan repetitivamente para cada disciplina del proceso. Es importante enfatizar que los pasos de la guía fueron definidos de acuerdo a los pasos genéricos de la adaptación de procesos a proyectos específicos los cuales fueron identificados en la ejecución de la investigación para realizar la guía.

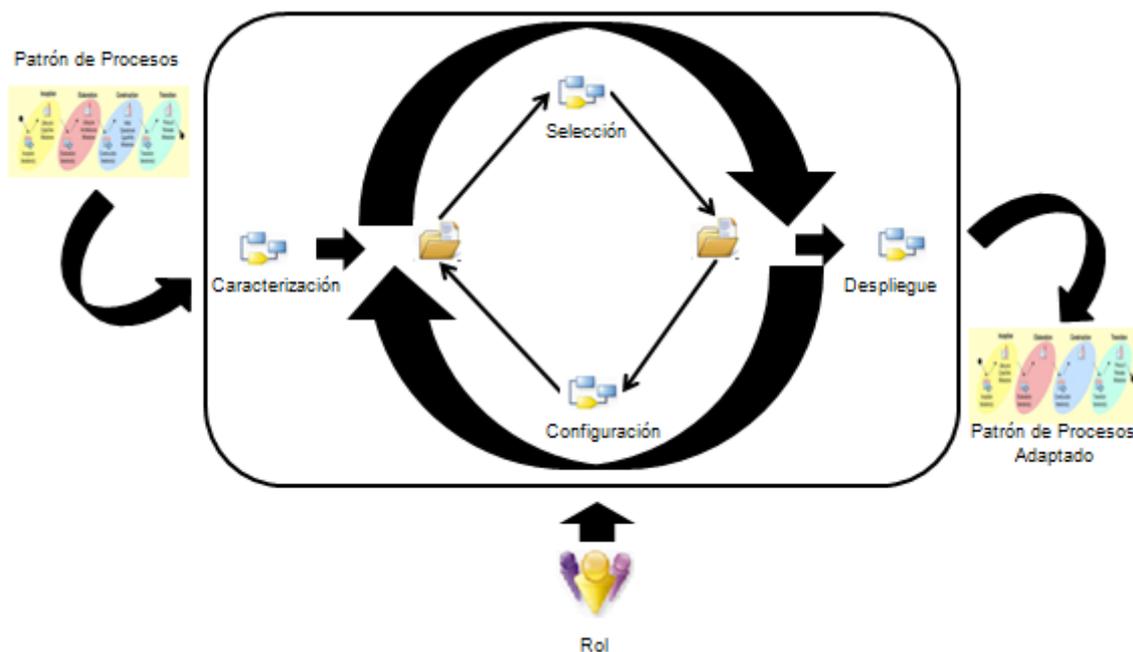
GTUP es una guía liviana de adaptación a proyectos específicos para las PYMES\_DS, pues este tipo de empresas cuentan con características como: bajos recursos, poco recurso humano, disponibilidad económica limitada, etc. y necesitan de soporte para

---

<sup>16</sup> EDITAR hace referencia a documentar y publicar fragmentos de método, procesos o metodologías

realizar la adaptación que tenga en cuenta las características existentes en la industria.

Para mostrar, facilitar y ejecutar la adaptación de UP, fue necesario implementar UP a través de SPEM por medio de EPFC.



**Figura 34 Estructura de la Guía de tailoring del Proceso Unificado - GTUP.**

A continuación se mencionaran los elementos y pasos que hacen parte de la guía y que son necesarios para realizar la adaptación del proceso software UP.

### 3.7.1 Elementos de la guía

**3.7.1.1 Patrón de procesos:** El patrón de procesos o proceso de desarrollo software UP, es el elemento de entrada a la guía para realizar la adaptación del proceso. Para facilitar la ejecución de la adaptación del proceso, UP se ha implementado a través de SPEM obteniendo como resultado un Plugin del proceso unificado<sup>17</sup> que cuenta con los elementos básicos para hacer posible que las PyMEs\_DS puedan desarrollar software de acuerdo a sus necesidades.

**3.7.1.2 Proceso adaptado:** El proceso adaptado es el resultado de aplicar la guía de tailoring al patrón de procesos. Este proceso se caracteriza por ser adecuado a las necesidades de los proyectos de las PyMEs\_DS puesto que en su confección se tuvieron en cuenta las características del proyecto que afectan el proceso de desarrollo.

<sup>17</sup> PLUGIN DEL PROCESO UNIFICADO es un paquete que representa un contenedor físico para las entidades propias del Proceso Unificado.

**3.7.1.3 Rol encargado de realizar la adaptación:** Es el rol o la(s) persona(s) que tiene como responsabilidad realizar la adaptación del proceso, este rol puede ser desempeñado por un gestor de proyectos, ingeniero de procesos o por una persona que tenga buenos conocimientos en UP. Este rol debe poseer las siguientes habilidades:

- Habilidades para una buena comunicación.
- Buena toma de decisiones en la adaptación del proceso.
- Tener un conocimiento en la definición de procesos.
- Tener conocimiento de la estructura de UP.
- Tener previas experiencias trabajando en proyectos software que han utilizado UP como proceso de desarrollo.

### **3.7.2 Pasos para realizar la adaptación**

**3.7.2.1 Caracterizar el proyecto:** La caracterización del proyecto es uno de los pasos más importantes en la adaptación del proceso, puesto que permite relacionar las características del proyecto que afectarán el proceso. Para realizar la caracterización del proyecto se tienen en cuenta los posibles niveles que pueden tomar las variables de caracterización.

**3.7.2.2 Seleccionar las entidades del proceso:** En la selección de las entidades del proceso es donde se identifican las posibles entidades que serán parte del proceso adaptado teniendo en cuenta la caracterización del proyecto.

**3.7.2.3 Configuración del proceso:** La configuración del proceso contempla los cambios que hay que realizar al patrón de procesos, en este paso, es donde la guía de tailoring brinda una serie de consideraciones que se deben tener cuando se adicionan, eliminan y reemplazan entidades del proceso.

**3.7.2.4 Desplegar el proceso adaptado:** En general el despliegue del proceso adaptado se realiza con el fin de hacer disponible el proceso para su utilización; para el caso concreto de la guía de tailoring, el resultado de este paso es un plugin del proceso que cuenta con las entidades necesarias para ejecutar los proyectos, entonces desplegar el proceso es publicar el UP adaptado en formato para la web.

La siguiente sección de este capítulo, contiene información detallada acerca de las actividades, pasos, guías (material de soporte) y artefactos necesarios para aplicar la guía de tailoring.

## **3.8 Pasos de la guía de Tailoring del Proceso Unificado.**

### **3.8.1 Primer paso - Caracterización del Proyecto.**

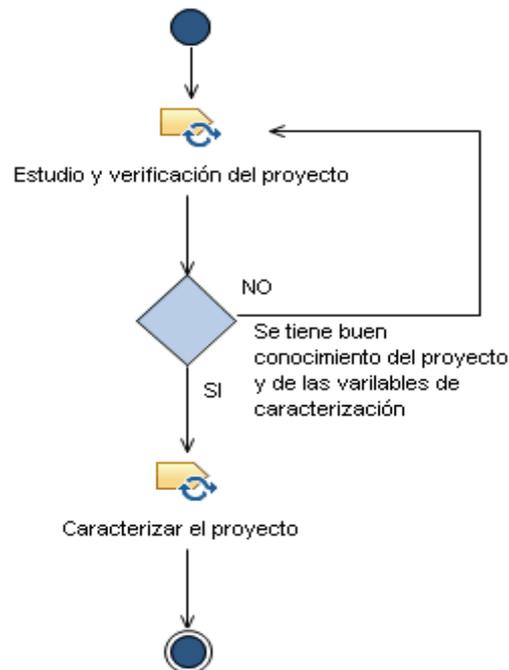
Este paso, es el inicio de la guía de tailoring del Proceso Unificado, en este se caracteriza el proyecto con base en las variables de caracterización y se establecen los niveles de cada una de las variables, los cuales guiarán la adaptación del proceso.

Para realizar la caracterización de los proyectos software, se consideraron las principales características de los proyectos que tienen impacto para adaptar el proceso, ver sección 3.1.3 Variables para realizar la caracterización de los proyectos.

Antes de realizar la caracterización, se debe estudiar el proyecto, con el fin de conocerlo mejor y verificar que por lo menos contiene los elementos necesarios para poder realizar la caracterización del proyecto. Es importante que en esta verificación se tenga en cuenta que la descripción del proyecto debe dar respuesta a las variables de caracterización.

Es posible dependiendo de la necesidad de la PyMEs\_DS adicionar o eliminar alguna de las variables de caracterización mencionadas en la sección 3.1.3 *Variables para realizar la caracterización de los proyectos.*, puesto que las variables de esa sección no son necesariamente obligatorias cuando se efectúa la caracterización, pues en algún momento no pueden expresar las necesidades del proyecto o la empresa; por esto es necesario que la(s) persona(s) encargada(s) de realizar la adaptación verifiquen cada una de las variables y por ende seleccionar aquellas que bajo sus criterios sean las más adecuadas para el proyecto y la organización.

A continuación se presenta el flujo de trabajo para el primer paso, ver Figura 35.



**Figura 35 Flujo de trabajo para el primer paso - Caracterización del Proyecto**

### 3.8.1.1 Actividades

#### 3.8.1.1.1 Estudio y verificación del proyecto

Esta actividad tiene como objetivo estudiar y conocer mejor el proyecto, para verificar que la descripción del proyecto contiene los elementos necesarios para poder realizar la caracterización, esto implica ilustrarse y conocer las variables de caracterización y

la descripción del proyecto, las cuales si es necesario, se pueden ajustar dependiendo de las necesidades del proyecto o de la organización.

### **Entrada**

- **Conocimiento del proyecto.**

Con este artefacto se pretende hacer referencia al conocimiento previo del proyecto que debe poseer la(s) persona(s) encargada(s) de efectuar la adaptación.

### **Salida**

- **Conocimiento acerca del proyecto y las variables de caracterización.**

Esté producto de trabajo tiene como objetivo hacer referencia al conocimiento que debe adquirir la(s) persona(s) encargada(s) de realizar la adaptación de las variables de caracterización y de la descripción del proyecto. Este producto de trabajo materializa el objetivo de la tarea "*Estudio y verificación del proyecto*".

### **Guía:**

- **Variables de caracterización de los proyectos.**

Está guía tiene el propósito de brindar la definición de las variables para realizar la caracterización de los proyectos en la guía de tailoring (ver anexo C.1).

### **Pasos**

- **Estudiar el proyecto.**

En este paso se pretende estudiar el proyecto, para verificar que la descripción del proyecto contiene los elementos necesarios para poder realizar la caracterización del proyecto.

- **Estudiar las variables de caracterización.**

Esté paso tiene el propósito de estudiar la definición de las variables para realizar la caracterización de los proyectos, para lo cual se debe utilizar la guía (Variables de caracterización de los proyectos).

- **Ajustar las variables de caracterización**

Es posible dependiendo de la necesidad de la PyMEs\_DS adicionar o eliminar alguna de las variables de caracterización, puesto que éstas variables no son necesariamente obligatorias, pues en algún momento no pueden expresar las necesidades del proyecto o la empresa; por esto es necesario que la(s) persona(s) encargada(s) de realizar la adaptación verifiquen cada una de las variables y por ende seleccionen aquellas que bajo sus criterios sean las más adecuadas para el proyecto y la organización.

#### **3.8.1.1.2 Caracterizar el proyecto**

Esta actividad tiene como objetivo asignar a las variables de caracterización los correspondientes niveles de caracterización de acuerdo a las particularidades del proyecto plasmadas en la descripción del proyecto.

#### **Entrada**

- **Conocimiento acerca del proyecto y las variables de caracterización.**

#### **Salida**

- **Niveles de las variables de caracterización de los proyectos [Diligenciado].**

Este artefacto hace referencia a la plantilla de los niveles de las variables de caracterización de los proyectos diligenciada, la cual tiene el propósito de facilitar la ejecución de la caracterización del proyecto, en su contenido están los niveles de clasificación de las variables de caracterización, los cuales se deben asignar de acuerdo a las particularidades del proyecto y la organización (ver anexo B.1).

#### **Pasos**

- **Leer y entender los niveles de las variables de caracterización de los proyectos.**

En este paso se pretende familiarizar a la persona(s) encargada(s) de realizar la adaptación con la plantilla niveles de las variables de caracterización de los proyectos.

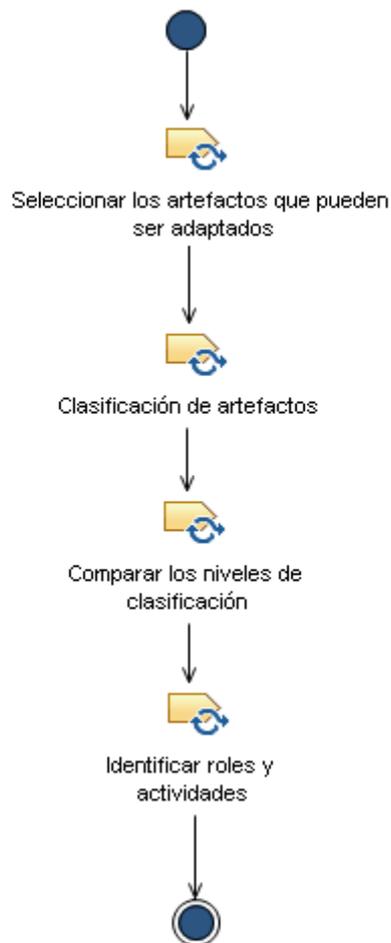
- **Realizar la caracterización del proyecto**

En este paso se efectúa la caracterización del proyecto, esto implica seleccionar el nivel de clasificación de cada variable, lo cual consiste en emparejar las particularidades del proyecto (variables) con los niveles de caracterización definidos en la guía.

### **3.8.2 Segundo paso – Selección de las entidades del proceso.**

En este paso, es donde se seleccionan las entidades del proceso que harán parte del proceso adaptado. El producto de trabajo más importante resultante de este paso es una matriz con las entidades del proceso que harán parte del proceso para un proyecto específico. Para realizar la selección de las entidades del proceso se establece una relación entre los artefactos y los niveles de caracterización por medio de la clasificación de cada artefacto.

A continuación se presenta el flujo de trabajo para el primer paso, ver Figura 36.



**Figura 36 Flujo de trabajo para el segundo paso - Selección de las entidades del proceso.**

### 3.8.2.1 Actividades

#### 3.8.2.1.1 Seleccionar los artefactos que pueden ser adaptados.

Pueden existir artefactos que son ejecutados siempre, tal como están definidos en el proceso, pues no pueden ser adaptados debido a que la organización debe realizarlos a cabalidad, ya sea por el tipo de contratación hecha con el cliente, ó por cumplir con algunas políticas de la organización ó por proteger la conformidad del proceso con normas de calidad y/o modelos de procesos que han sido adoptados por la organización. Entonces el objetivo de esta actividad es seleccionar aquellos artefactos que pueden sufrir modificaciones o que se puede prescindir de ellos

#### Entrada

- **Documento con los artefactos de las disciplinas.**

En esta entrada tiene como fin dar a conocer los objetivos de los artefactos de las disciplinas del proceso, con el fin de ayudar en el entendimiento y en la selección de aquellos artefactos que la empresa considere necesarios para el desarrollo de los proyectos de la empresa (ver anexos C.9, C.10, C.11, C.12, C.13).

### Salida

- **Plantilla artefactos que serán adaptados.**

Esta plantilla permite realizar la selección de los artefactos, de la correspondiente disciplina, que pueden ser adaptados (ver anexo B.2).

### Pasos

- **Verificar Normas adoptadas**

Esté paso tiene como fin, verificar si la empresa debe cumplir con normas de calidad por ejemplo (CMMI, ISO) y/o modelos de procesos, las cuales pueden restringir la selección de algunos artefactos.

- **Identificar artefactos de las disciplinas que se deben mantener**

Este paso tiene como fin, identificar los artefactos de las disciplinas que se deben realizar o mantener en la disciplina. Esta identificación se puede realizar de dos formas:

1. Verificando las normas de calidad y/o modelos de proceso.
2. Revisando el tipo de contratación hecha con el cliente.

- **Seleccionar Artefactos**

En este paso se realiza la selección de los artefactos de la disciplina que pueden ser adaptados <sup>18</sup> y por lo tanto se materializa el objetivo de la actividad “*Seleccionar los artefactos que pueden ser adaptados*”.

#### 3.8.2.1.2 Clasificación de artefactos

El producto resultado de ejecutar esta actividad es una matriz donde se plasma el nivel de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina, identificando para cada nivel, los artefactos que se producirán, no se producirán, se producirán con restricciones y los que son indiferentes. Ver Tabla 3. Clasificación de Artefactos.

### Entrada

- **Plantilla artefactos que serán adaptados [Diligenciado].**
- **Niveles de las variables de caracterización de los proyectos.**

---

<sup>18</sup> ARTEFACTOS QUE PUEDEN SER ADAPTADOS hace referencia a los artefactos del proceso que pueden sufrir algún cambio.

## Salida

- **Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización.**

En esta matriz se recopilan los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina y se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de la disciplina (ver anexo B.3).

## Guías

- **Tabla de clasificación de artefactos.**

Este documento tiene el objetivo de brindar las posibles clasificaciones que puede tomar un artefacto en el momento de realizar su selección. (Ver anexo C.2).

- **Ejemplo de clasificación de artefactos**

En este se orienta como se debe realizar la clasificación de artefactos vs. variables de caracterización en la “*Matriz de clasificación de artefactos por disciplinas vs. Variables de caracterización*” (ver Anexo C.14).

## Pasos

- **Identificar los niveles**

En este paso se identifican los niveles de las variables de caracterización del proyecto.

- **Análisis de artefactos**

En este paso se realiza el análisis de la información proporcionada por cada artefacto de las disciplinas, teniendo en cuenta los anexos C.9, C.10, C.11, C.12, C.13. Es de anotar que la relación entre las variables de caracterización y los artefactos del proceso se pueden obtener a través del análisis de la información proporcionada por UP de cada uno de los artefactos.

- **Establecer los niveles de clasificación**

En este paso se establecen los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de cada disciplina en la matriz “*Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización*”, identificando para cada nivel los artefactos que se producirán (SP), no se producirán (NP), se producirán con restricciones (SR) y los que son indiferentes (I).

### 3.8.2.1.3 Comparar los niveles de clasificación.

Puesto que el mismo artefacto de una disciplina puede tener diferentes clasificaciones en cada una de las variables de caracterización, entonces es necesario comparar la clasificación de las variables con el fin de unificar ésta para cada artefacto. En la

comparación pueden surgir problemas puesto que la clasificación de cada una de las variables no es uniforme, entonces para tomar una decisión de la clasificación final del artefacto se puede optar por realizar una solución intermedia o decidir con base en las variables restrictivas como normas, presupuesto, calendario, herramientas y la prioridad del proyecto.

#### **Entrada**

- **Matriz de clasificación de artefactos vs. variables por disciplina [Diligenciada].**

#### **Salida**

- **Matriz final de clasificación de artefactos por disciplina.**  
En esta matriz se recopilan la clasificación final de cada artefacto con respecto a las variables de caracterización (ver anexo B.4 ) y se adicionan los artefactos que fueron seleccionados como no adaptables (los que se deben producir) en la plantilla *artefactos que serán adaptados*, diligenciada en la sección 3.8.2.1.1.

#### **Pasos**

- **Identificar artefactos con distintas clasificaciones**  
En este paso se identifican los artefactos que tienen diferentes clasificaciones en cada una de las variables de caracterización utilizando la "*Matriz de clasificación de artefactos por disciplinas vs. variables de caracterización*" previamente diligenciada.
- **Comparar la clasificación de las variables.**  
En este paso se realiza la comparación de los artefactos que tienen diferentes clasificaciones en las variables de caracterización, con el fin de unificar la clasificación de cada artefacto. Como resultado en este paso se obtiene la "*Matriz final de clasificación de artefactos por disciplina*".

En la comparación pueden surgir problemas puesto que la clasificación de cada una de las variables no es uniforme, entonces para tomar una decisión de la clasificación final del artefacto se pueden considerar algunos criterios como por ejemplo presupuesto y/o calendario del proyecto u optar por realizar el artefacto de una forma intermedia.

#### **3.8.2.1.4 Identificar roles y actividades**

Conociendo los artefactos del proceso que se deben producir se puede deducir los roles y actividades necesarios para dichos artefactos, esta actividad tiene como fin identificar los roles y actividades indispensables para producir los artefactos de las disciplinas que harán parte del proceso adaptado.

#### **Entrada**

- **Matriz final de clasificación de artefactos por disciplina.**

## Salida

- **Roles y actividades necesarios para la disciplina.**

Este artefacto tiene el propósito de recopilar los roles y actividades indispensables para producir los artefactos de la disciplina del proceso adaptado.

## Guías

- **Relaciones entre actividades, roles y artefactos por disciplina.**

Esta guía tiene el propósito de brindar las relaciones entre las entidades de las disciplinas.

## Pasos

Es importante resaltar que para ejecutar los siguientes pasos es necesario tener en cuenta la guía (*Relaciones entre actividades, roles y artefactos por disciplina.*)

- **Deducir los roles**

Este paso tiene como fin identificar los roles indispensables para cada disciplina. Conociendo los artefactos que se van a desarrollar, “Matriz final de clasificación de artefactos por disciplina”, se puede deducir cuales roles harán parte de las disciplinas del proceso. Los roles identificados en este paso se plasman en el artefacto (*Roles y actividades necesarios para la disciplina*) **él cual es** salida de esta actividad.

- **Deducir actividades**

Este paso tiene como fin identificar las actividades indispensables para cada disciplina. Conociendo los artefactos que se van a desarrollar, “*Matriz final de clasificación de artefactos por disciplina*”, se puede deducir las actividades necesarias para producir dichos artefactos. Las actividades identificadas en este paso se plasman en el artefacto de salida de esta actividad.

### 3.8.3 Tercer paso – Realizar la configuración del proceso.

En este paso, es donde la guía de tailoring del Proceso Unificado brinda una serie de consideraciones que se deben tener cuando se realizan las operaciones de adaptación; puesto que luego de conocer los artefactos, identificar las actividades y roles necesarios que serán parte del proceso adaptado, es necesario eliminar, reemplazar ó adicionar entidades de la disciplina de acuerdo a su clasificación o de acuerdo a los criterios de la persona encargada de realizar la adaptación.

A continuación se presenta el flujo de trabajo para el primer paso, ver Figura 37.

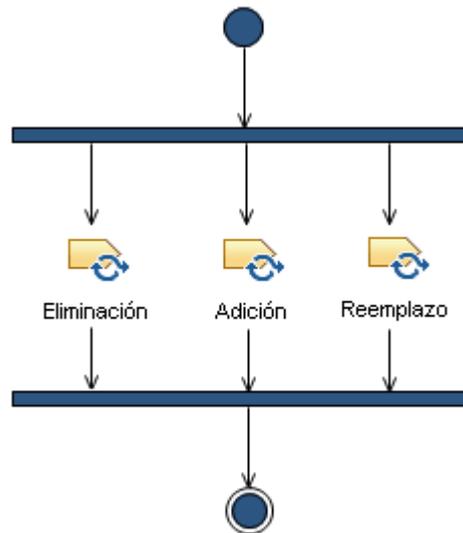


Figura 37 Flujo de trabajo para el tercer paso - Configuración del proceso.

### 3.8.3.1 Actividades

#### 3.8.3.1.1 Eliminación

La actividad de eliminación puede ser la más común en la adaptación de procesos [17], esta actividad está definida como la supresión de una entidad (tarea, artefacto, rol) en el proceso de desarrollo [19], para eliminar una entidad se debe considerar qué entidades del proceso son afectadas por la eliminación. Por ejemplo, cuando una actividad es borrada, los artefactos generados desde la actividad también tienen que ser borrados. Para información detallada acerca de cómo realizar esta actividad ir a la sección 3.1.5.1 Operación de Eliminación.

#### Entrada

- **Disciplina del Proceso (EPFC).**  
Este artefacto hace alusión a los patrones de proceso implementados en SPEM en el Plugin “*Unified process*” que describen las disciplinas del proceso.
- **Roles y actividades necesarios para la disciplina.**
- **Matriz final de clasificación de artefactos por disciplina.**

#### Salida

- **Disciplinas del Proceso [Actualizadas en EPFC].**

## Guías

- **Consideraciones al realizar la eliminación de entidades.**

En esta guía se plasman algunas consideraciones que se deben tener en cuenta cuando se realiza la eliminación de entidades en las disciplinas del proceso (ver anexo C.15 ).

## Pasos

- **Comparar artefactos por disciplina**

Este paso tiene el propósito, de realizar la comparación entre la “*Matriz final de clasificación de artefactos por disciplina*” y “*Disciplina del proceso*”, para obtener los artefactos que deben ser eliminados de la “*Disciplina del proceso*”.

- **Comparar roles y actividades por disciplina**

Este paso tiene como fin, efectuar una comparación entre los artefactos “*Disciplina del proceso*” y “*Roles y actividades necesarios para la disciplina*”, para deducir cuales roles y actividades de la “*Disciplina del proceso*” deben ser eliminados.

### 3.8.3.1.2 Reemplazo

Esta actividad es definida como la sustitución de una entidad en el proceso de desarrollo. Esta actividad tiene como objetivo intercambiar entidades nativas de la empresa por las definidas en el proceso, esta operación intenta rescatar y reutilizar entidades de la empresa para no someterla a cambios drásticos en la forma como desarrollan o ejecutan este tipo de entidades.

## Entrada

- **Disciplinas del Proceso (EPFC).**

## Salida

- **Disciplinas del proceso (EPFC) [Actualizadas].**

## Guía

- **Consideraciones en el ensamble de actividades y homologación de Artefactos.**

En esta guía se plasman algunas consideraciones que se deben tener en cuenta cuando se realiza el ensamble de actividades y la homologación de artefactos del proceso (ver anexo C.16).

## Pasos

- **Identificar las entidades nativas.**

En este paso, se identifican las entidades nativas del proceso de la empresa que se pueden intercambiar por entidades definidas en el proceso.

- **Intercambiar las entidades nativas.**

En este paso, se ejecuta el intercambio, entre las entidades nativas y las entidades propias del proceso en el artefacto “*Disciplinas del Proceso (EPFC)*”.

### 3.8.3.1.3 Adición

Esta actividad es definida como la operación de adición de una entidad al proceso de desarrollo de software [17]. La operación de adición está clasificada en adición de entidades del estándar y adición de entidades definidas por el usuario. Las entidades del estándar son aquellas que ya están definidas en el estándar de procesos, y las entidades definidas por el usuario son aquellas que son definidas por un diseñador de procesos. Cuando se desea adicionar una entidad dentro del proceso de desarrollo, se tiene que considerar dónde la entidad es adicionada y qué tipo de relación hay entre la nueva entidad y la entidad adyacente [19].

#### Entrada

- **Disciplinas del Proceso (EPFC).**

#### Salida

- **Disciplinas del proceso (EPFC)[Actualizadas].**

#### Guías

- **Consideraciones al realizar la adición de entidades.**

En esta guía se plasman algunas consideraciones que se deben tener en cuenta cuando se realiza la adición de entidades a las disciplinas del proceso (ver anexo C.17).

#### Pasos

- **Identificar las entidades que se pueden adicionar.**

En este paso, se identifican las entidades extras a las disciplinas del patrón de procesos que se necesitan adicionar.

- **Adición de las entidades nativas.**

En este paso, se efectúa la adición de las entidades extras a las disciplinas del proceso en el artefacto “*Disciplinas del Proceso (EPFC)*”.

### 3.8.4 Cuarto paso- Desplegar el proceso

En este paso es donde muestra el resultado de aplicar la guía de tailoring al patrón de procesos UP, es decir donde se realiza el despliegue del proceso adaptado con el fin de hacerlo disponible para su utilización. Este paso puede estar un poco ligado a la

manipulación y utilización de conceptos definidos en SPEM y EPFC, debido a que fueron utilizados como facilitadores para poder efectuar la adaptación en la guía tailoring.

A continuación se presenta el flujo de trabajo para el cuarto paso, ver Figura 38.



**Figura 38 Flujo de trabajo para el cuarto paso - Desplegar el proceso.**

### 3.8.4.1 Actividad

#### 3.8.4.1.1 Publicar el proceso

Esta actividad tiene como objetivo publicar el proceso adaptado para su futura divulgación.

#### Entrada

- **Recopilación de las Disciplinas del Proceso (EPFC).**  
Este artefacto hace referencia a todas las disciplinas del proceso que previamente han sido adaptadas.

#### Salida

- **Plugin Unified Process [Adaptado].**  
Este artefacto hace referencia al plugin del “*Unified Process*” adaptado.

#### Guías: Publicación del contenido del Plugin en EPFC.

- Esta guía pretende ayudar a realizar el paso del despliegue del proceso por medio de la publicación del contenido del Plugin.

#### Pasos

- **Recopilar las disciplinas del proceso.**  
Este paso pretende reunir todas las disciplinas adaptadas del proceso para su posterior despliegue.

- **Desplegar el proceso**

En este paso se realiza el despliegue del Plugin del proceso unificado adaptado en formato web.

---

## **CAPITULO 4. APLICACIÓN DE GTUP – AMBIENTE ACADEMICO Y EMPRESARIAL**

En este capítulo se describe a grandes rasgos como se efectuó la aplicación de la Guía de Tailoring en un ambiente de tipo académico y otro de tipo empresarial, teniendo en cuenta que el objetivo de la aplicación de la guía era determinar, mediante opiniones de las personas encargadas de ejecutar la adaptación, si ésta, cumplía con el objetivo de guiar la planeación y ejecución de la adaptación, y si contenía y suministraba los elementos necesarios para facilitar la ejecución de la adaptación de UP; es importante resaltar que la aplicación de la guía se enfocó en el proceso de adaptación y no en el proceso adaptado o proceso resultado. No se tuvo en cuenta y no estaba dentro del alcance del proyecto analizar dicho proceso, debido a que realizar éste análisis requiere demasiado tiempo, pues sería necesario poner a prueba el proceso adaptado en el desarrollo de un proyecto, para luego realizar algún tipo análisis.

Para facilitar la aplicación de la guía se utilizó el plugin de UP (Unified Process) y se implementó la descripción de la guía de tailoring en Eclipse Process Framework Composer, obteniendo como resultado un plugin llamado GTUP. Estos dos plugins fueron utilizados en todas las aplicaciones de la guía, tanto para que los participantes tuvieran acceso fácil a la descripción del proceso UP como a la guía de tailoring, ya sea por medio de formato web o por medio de la definición de los plugins mediante Eclipse Process Framework Composer (EPFC). El acceso mediante EPFC brindó en las aplicaciones la ventaja de poder editar <sup>19</sup> los repositorios de elementos básicos del método (actividades, guías, plantillas, pasos) y patrones de proceso que fueron utilizados en la definición de los dos plugins; además, la publicación de la Guía (formato web), agilizó el seguimiento de la misma y facilitó el acceso a su definición a los integrantes del proyecto; cabe resaltar que para aplicar GTUP no es estrictamente obligatoria la publicación de la guía, sino que en cuestión de tiempo y accesibilidad, ésta facilita su aplicación.

En las aplicaciones de la guía no se tuvieron en cuenta todas las disciplinas del proceso, se seleccionaron las disciplinas básicas o de control de proceso, éstas son: Modelado del Negocio, Requerimientos, Análisis y Diseño, Implementación y Pruebas; con esto no se restringe la aplicación de la guía ni su definición, ya que la adaptación de las disciplinas está orientada de forma general a la elección del conjunto de artefactos que harán parte del proceso adaptado a proyectos específicos, éste enfoque permite que la guía sea aplicada a cualquier disciplina de UP y posiblemente a cualquier proceso software que en su definición contenga disciplinas.

A continuación se detalla cómo se realizó la aplicación de la Guía de Tailoring en el ambiente académico y empresarial.

---

<sup>19</sup> Editar hace referencia a las posibles modificaciones y configuraciones que se pueden realizar a la definición de un Plugin en EPFC

## **4.1 Ambiente Académico**

### **4.1.1 Prueba de Concepto**

La prueba de concepto en este ambiente sólo estaba enfocada en determinar la claridad y accesibilidad del contenido de la definición de la guía para verificar si cumplía con la planificación y ejecución de la adaptación del proceso. Se llevo a cabo con siete estudiantes del curso de ingeniería del software III del programa de ingeniería de sistemas, los cuales, mediante la utilización de la publicación de la guía hicieron recomendaciones y/o sugerencias que ayudaron a efectuar ajustes en la guía permitiendo que ésta sea más clara, comprensible e intuitiva para ser ejecutada.

A continuación se detallan los pasos con los cuales se ejecutó la prueba de concepto:

#### **4.1.1.1 Contextualización de la guía.**

Para realizar la contextualización de la guía fue necesario realizar una capacitación con el objetivo de que los estudiantes involucrados tuvieran bases conceptuales en SPEM, EPFC, GTUP y UP. Para que los estudiantes se familiarizaran con algunos de los conceptos involucrados en el proyecto, fue necesario antes de acudir a la capacitación enviar, por medio de correo electrónico las siguientes lecturas (ver anexo D 1.1 y D 1.2 respectivamente):

1. Documento Taller SPEM y EPF.
2. Descripción del UP y UP a través de SPEM.

La capacitación se realizó en dos secciones de cuatro horas, donde se profundizaron conceptos y aclararon dudas sobre las lecturas. En la capacitación se trataron temas de ingeniería de software, se enfatizó en el proceso unificado y se dio a conocer SPEM como lenguaje de modelado de procesos y Eclipse Process Framework Composer como editor de procesos, éste último utilizado en el desarrollo del plugin que contiene la descripción de la guía y que posteriormente se iba a utilizar para efectuar la prueba de concepto.

#### **4.1.1.2 Ejecución prueba de concepto**

Paran efectuar la prueba de concepto los estudiantes tuvieron acceso tanto al plugin de la Guía como a la publicación web de la misma, con el propósito de revisar aspectos tales como: claridad, orden lógico de los pasos de la adaptación, estructura general y nivel de detalle del contenido (actividades, flujos de trabajo, pasos, guías y plantillas). Lo que se pretendía era capturar obstáculos, restricciones y/o conflictos que la(s) persona(s) encargada(s) de ejecutar la guía podrían tener al tratar de aplicarla.

Como resultado de la prueba de concepto se obtuvo recomendaciones y sugerencias como: incluir un glosario de términos como parte de la guía, profundizar en el nivel de detalle de la descripción de algunos conceptos, pasos y plantillas entre otros.

Luego de la prueba de concepto, la guía se ajustó teniendo en cuenta algunas de las sugerencias hechas por los participantes de la prueba.

#### **4.1.2 Ambiente académico en un proyecto**

En este escenario se aplicó la guía de forma controlada a las disciplinas de Modelado del Negocio, Implementación y Pruebas de UP a un proyecto específico de comercio electrónico. La aplicación de la guía en este escenario se realizó con la colaboración de tres estudiantes de último semestre de ingeniería de sistemas quienes en conjunto desempeñaron el rol encargado de efectuar la adaptación.

El proyecto de comercio electrónico tenía como objetivo, desarrollar una aplicación web para compra y venta de productos que apoye a las nuevas empresas que se están creando y que necesitan ampliar su mercado a nivel nacional por medio de las tecnologías de la información y la comunicación (TICs). En resumen, básicamente la aplicación software permitía mostrar información corporativa de la empresa, sus productos y también permitir la compra y venta de éstos a través de Internet; para obtener más información del proyecto (ver anexo D 2.1). Es importante destacar que en este tipo de proyectos cuando se está construyendo un sistema o familia de aplicaciones donde el modelo del negocio puede servir de entrada a varios proyectos (varias empresas), es necesario realizar un buen modelado del negocio [10].

Previo al inicio de la aplicación de la guía, se realizó una reunión entre los estudiantes y autores de la guía, en la que se contextualizó la guía tratando los siguientes temas: Proceso unificado, EPFC, GTUP y SPEM. Luego de tener claro por parte de los estudiantes los conceptos involucrados en el proyecto, se procedió a su aplicación.

A continuación se detallan los pasos de la aplicación.

##### **4.1.2.1 Primer paso - Caracterización del Proyecto**

En este paso se hizo la lectura del documento que contiene la información del proyecto (ver anexo D 2.1); se aclararon dudas y se centralizaron ideas importantes sobre el proyecto; también, se realizó la lectura de las variables de caracterización y se dejó claro la importancia de éstas en la caracterización del proyecto. En el momento que se culminó la actividad de *Estudio y Verificación del proyecto*, se observó que la descripción del proyecto no tenía información acerca del presupuesto, entonces el equipo encargado de realizar la adaptación se vio obligado a realizar algunas averiguaciones para obtener información acerca del presupuesto del proyecto; luego de conocer el presupuesto y en consecuencia el tamaño del proyecto se dio por terminada la actividad.

Luego de tener claro tanto la descripción del proyecto y las variables de caracterización, se realizó la actividad de *Caracterizar el Proyecto* donde se procedió a diligenciar el artefacto “*Niveles de las variables de caracterización de los proyectos*” (ver anexo D 2.2).

#### 4.1.2.2 Segundo paso - Selección de las entidades del proceso

Debido a que en este ambiente se iba a realizar la adaptación de las disciplinas de Modelado del Negocio, Implementación y Pruebas, fue necesario que el segundo y tercer paso de GTUP se efectuara en tres iteraciones, una para cada disciplina.

En cada una de las iteraciones realizadas en este paso, se procedió a seleccionar los artefactos que iban a ser adaptados, teniendo en cuenta que algunos artefactos, por el tipo de proyecto se debían realizar de manera obligatoria. Para avanzar correctamente en la ejecución de este paso, se hizo la lectura del documento, “*Artefactos de las disciplinas*”, donde se dan a conocer cada uno de los objetivos de los artefactos de las disciplinas involucradas, con el fin de ayudar en el entendimiento y en la selección de los artefactos que se consideren necesarios para cada disciplina (ver anexo C.9, C.12, C.13) y posteriormente se procedió a diligenciar la “*Plantilla artefactos que serán adaptados*” por cada disciplina (ver anexo D. 2.3, D.2.4, D.2.5).

Luego de diligenciar las anteriores plantillas se procedió a completar la “*matriz de clasificación de artefactos por disciplinas vs. variables de caracterización*”, identificando los artefactos que se producirán (SP), no se producirán (NP), se producirán con restricciones (SR) y los que son indiferentes (I) por cada nivel de caracterización (ver anexos D.2.6, D.2.7, D.2.8), además en la “*Matriz final de clasificación de artefactos por disciplina*” (ver anexo D.2.9, D.2.10, D.2.11) se consolidó la clasificación final de cada uno de los artefactos de las disciplinas.

#### 4.1.2.3 Tercer Paso – Configurar el Proceso

Teniendo en cuenta las consideraciones descritas en la sección 3.1 *Configuración del proceso*, se efectuaron las posibles modificaciones al patrón de procesos UP, para lo cual se utilizó EPFC como editor del plugin “*Unified Process*” para realizar las eliminaciones de las entidades de las disciplinas que no iban a ser parte del proceso adaptado. Estas entidades a eliminar se obtuvieron comparando individualmente los artefactos de “*Matriz final de clasificación de artefactos por disciplina*” (ver anexo D.2.9, D.2.10, D.2.11) y “*Roles y actividades necesarios para la disciplina*” (ver anexo D.2.12, D.2.13, D.2.14) con las disciplinas del patrón de procesos.

#### 4.1.2.4 Cuarto Paso - Desplegar el proceso.

En esta actividad mediante la utilización de EPFC y con el Plugin de UP configurado, se procedió a utilizar la funcionalidad que ofrece EPFC para publicar el proceso adaptado y permitir fácil acceso a los miembros del proyecto mediante el formato web.

La aplicación de la guía en el ambiente académico termina con la actividad de desplegar el proceso donde fue necesario leer con profundidad el documento guía “*Publicación del Plugin*”, puesto que para efectuar esta actividad se debían manejar y conocer términos y características del entorno de EPFC.

## **4.2 Ambiente Empresarial**

Con el fin de aplicar la guía en un ambiente real, se pidió colaboración a la empresa INPUT Technologies perteneciente a Parquesoft Popayán, para que aplicara la guía en la adaptación de las disciplinas de Requerimientos y Análisis y Diseño para uno de sus proyectos. Esta empresa se dedica a brindar soluciones software integrales que contribuyen al mejoramiento continuo de los procesos y facilitan la obtención de la calidad de las empresas en general. Para conocer más sobre la empresa (ver anexo D.3.1).

La aplicación de la guía se llevó a cabo en dos pasos generales:

1. Contextualización de la guía a los integrantes de la empresa.
2. Aplicación de la guía a un proyecto específico.

### **4.2.1 Contextualización de la guía**

Para realizar la contextualización de la guía, se entregó la misma documentación que en la prueba de concepto (ver anexos D 1.1 y D 1.2), con la ventaja de que en este escenario se contaba con cierto nivel de experiencia en desarrollo de proyectos y fue más fácil la asimilación de conceptos y temáticas necesarias utilizadas en la guía de tailoring.

La empresa mediante una reunión, comunicó a los autores de la guía que el proyecto en el cual se iba a aplicar la guía era el modulo de actas y reuniones de calidad que haría parte del producto software llamado COMPROMISO.

COMPROMISO es una herramienta software basada en las normas ISO 9000 con el objetivo de apoyar el sistema de gestión de la calidad, brindando la posibilidad de hacer un seguimiento completo de los procesos y requisitos necesarios para cumplir con dicho estándar. Este producto va dirigido a las empresas certificadas, empresas en proceso de certificación o empresas que busquen organizar sus procesos internos como la comunicación y divulgación del sistema de gestión de calidad, administración de la documentación, seguimiento de no conformidades y gestión de equipos (equipos de computo, maquinaria, instrumentos de medición y demás). Para más información del producto software compromiso (ver anexo D 3.2).

El proyecto consistía en desarrollar un módulo que permitiera registrar y mantener actualizadas y disponibles las actas del comité de calidad y reuniones en general, además llevar control de la reunión durante el desarrollo de la misma siguiendo cada uno de los temas a tratar en el orden del día y las opiniones, aportes, actividades y tareas que surjan en cada uno de ellos. También que permitiera almacenar las tareas generadas en cada una de las reuniones y realizar seguimiento de aquellas que están pendientes. Para más información (ver anexo D.3.3).

Luego de haber ultimado detalles del contexto de la guía y conociendo el proyecto en el cual se iba a ejecutar, se procede a realizar la aplicación de la guía.

A continuación se detallan las actividades específicas llevadas a cabo en la aplicación de la guía al Proyecto Actas y Reuniones de calidad.

## **4.2.2 Primer paso - Caracterización del Proyecto**

### **4.2.2.1 Estudio y verificación del proyecto**

En esta actividad se hizo la lectura del documento que contenía la información del proyecto en el cual se iba a aplicar la guía. En esta parte se resaltó que como el proyecto iba a ser desarrollado tratando de cumplir con normas de calidad, puesto que esta empresa había participado en varios proyectos de mejora de proceso (Software Process Improvement), entonces se debía tener en cuenta que algunos artefactos serían de carácter obligatorio, es decir, éstos no podían ser adaptados. Al final de la lectura se concluyó que en la descripción del proyecto se encontraba la información necesaria y suficiente para realizar la caracterización del proyecto.

Luego se hizo la lectura del documento en donde, por parte de la guía, se da a conocer la definición de cada una de las variables de caracterización. Al final de esta actividad se concluyó que se tenía la información suficiente para continuar con la siguiente actividad.

### **4.2.2.2 Caracterización del proyecto**

En esta actividad, se procedió a diligenciar el artefacto “*Niveles de las variables de caracterización de los proyectos*” (ver anexo D 3.4). Durante la ejecución de esta actividad se observó que el equipo encargado del proyecto tenía nociones de UP, es decir, conocía y utilizaba UP, pero no existía una aplicación bien formalizada del proceso; además por ser una empresa que ya había participado en proyectos de mejora de procesos software, se enfatizó que la prioridad del proyecto sería la calidad; lo que arrojó como resultado que la variable que prioriza el proyecto (calidad vs. tiempo) se iba a dar así: 90% calidad y 10% tiempo. Este rango se asignó precisamente porque esta empresa orienta sus desarrollos teniendo en cuenta la la calidad y en este caso la prioridad no era el tiempo puesto que este proyecto no era desarrollado con base en la petición de un cliente específico sino orientado a satisfacer una necesidad general de mercado.

## **4.2.3 Segundo paso - Selección de las entidades del proceso**

Debido a que en este ambiente se iba a realizar la adaptación de las disciplinas de Requerimientos y Análisis y Diseño, como sucedió en el anterior escenario de aplicación de la guía de tailoring, fue necesario que el segundo y tercer paso de GTUP se efectuara en dos iteraciones, una para cada disciplina.

### **4.2.3.1 Seleccionar los artefactos que pueden ser adaptados**

Al inicio de esta actividad se hizo la lectura del documento “*Artefactos de las disciplinas*” (ver anexos C.10, C.11), donde se da a conocer la descripción de los artefactos de las disciplinas. Es importante mencionar que para seleccionar los

artefactos que pueden ser adaptados se tuvo en cuenta que éstos no afectarían las mejoras realizadas al proceso de desarrollo mediante previos proyectos SPI.

Luego se procedió a diligenciar la plantilla “*Artefactos que serán adaptados*” para las dos disciplinas (ver anexos D.3.5, D.3.6).

#### **4.2.3.2 Clasificación de artefactos**

En esta actividad se diligenció, para cada disciplina, la plantilla “*matriz de clasificación de artefactos por disciplinas vs. variables de caracterización*” (ver anexo D.3.7, D.3.8) en la cual se definieron los niveles de clasificación de cada una de las variables de caracterización con respecto a los artefactos de las disciplinas.

La clasificación se realizó a criterio de la persona encargada de ejecutar la adaptación y con base en la descripción de los artefactos de las disciplinas (ver anexo C.10, C.11)

#### **4.2.3.3 Comparar los niveles de clasificación.**

En esta actividad se decidió tener en cuenta el calendario y la prioridad del proyecto para unificar algunos de los artefactos que no obtuvieron una clasificación homogénea en cada una de las variables de caracterización. Es importante resaltar que algunos artefactos no tuvieron diferencias en los niveles de clasificación, es decir, su clasificación fue uniforme, por lo tanto al diligenciar la “*matriz final de clasificación de artefactos por disciplina*” (ver anexos D.3.9, D.3.10) estas clasificaciones no sufrieron alteraciones.

#### **4.2.3.4 Identificar roles y actividades**

Con base en el resultado de la anterior actividad, es decir, con la “*Matriz final de clasificación de artefactos por disciplina*” ya diligenciadas; Se procedió a identificar, utilizando como apoyo el material (Anexos A.2, A.3), roles y actividades necesarias para poder realizar los artefactos del proceso adaptado clasificados como SP, SR u Obligatorio en la “*Matriz final de clasificación de artefactos por disciplina*” (ver anexos D.3.9, D.3.10). Al terminar esta actividad se diligenció el artefacto “*Roles y Actividades necesarios*” para cada una de las disciplinas (ver anexo D.3.10, D.3.11).

### **4.2.4 Tercer Paso – Configurar el Proceso.**

#### **4.2.4.1 Realizar la configuración del proceso.**

Para realizar este paso se utilizó las consideraciones al realizar las modificaciones que pueden sufrir las disciplinas del patrón de procesos descritas en la sección 3.1. *Configuración del proceso.*

En la configuración del proceso, se utilizó EPFC como editor del plugin “*Unified Process*” para realizar la eliminación de las entidades del proceso que no iban a ser parte de las disciplinas adaptadas, esto se hizo comparando individualmente los artefactos de “*Matriz final de clasificación de artefactos por disciplina*” (ver anexo

D.3.9, D.3.10) y “Roles y actividades necesarios” para las disciplinas (ver anexo D.3.10, D.3.11) con las disciplinas del patrón de procesos respectivamente.

#### **4.2.5 Cuarto Paso - Desplegar el proceso.**

##### **4.2.5.1 Desplegar el proceso**

En esta actividad mediante la utilización de EPFC y con el plugin de UP editado, se procedió a utilizar la funcionalidad que ofrece EPFC para publicar el proceso adaptado y de esta forma permitir fácil acceso a los miembros del proyecto mediante el formato web.

Cabe notar que dentro de los objetivos del trabajo de grado no estaba analizar el proceso adaptado sino realizar la aplicación de la guía de adaptación de UP, por lo tanto, la aplicación en el ambiente empresarial se da por terminado con la actividad de desplegar el proceso, no sin antes obtener algunas conclusiones de la aplicación de la guía de tailoring.

#### **4.3 CONCLUSIONES DE LA APLICACIÓN DE GTUP**

Luego de realizar la aplicación de GTUP en los dos ambientes se concluyó:

- Desde el punto de vista de la definición y estructura de la guía, ésta no necesitó ajustes considerables, pues las personas involucradas en las aplicaciones de la guía manifestaron, que ésta era clara, coherente y brindaba la secuencia de pasos suficientes y detallados para ser aplicada.
- Se concluyó que la guía de tailoring orienta la adaptación del proceso UP, lo cual se sustenta en lo siguiente: a) En las reuniones llevadas a cabo, al terminar las aplicaciones, se manifestó que la guía era determinante para realizar de manera exitosa la adaptación. b) Los participantes de las aplicaciones sí pudieron seguir y realizar la adaptación de las disciplinas sin problemas considerables. c) Se obtuvieron como resultado cinco disciplinas adaptadas las cuales están disponibles en el Plugin *Unified Process*.
- La eliminación de algunos elementos innecesarios del patrón de procesos software, permitió ajustar el proceso a las características del proyecto, lo que puede implicar reducción en tiempo y costo para el desarrollo del proyecto.
- Las MyPymes\_DS en primera medida pueden utilizar las disciplinas adaptadas del proceso, y replicar el proceso de adaptación a las disciplinas faltantes utilizando la guía tailoring.
- El haber Utilizado EPFC en las aplicaciones de la guía facilitó la adaptación mediante la edición de los repositorios de elementos básicos del método y patrones de proceso definidos en el Plugin del Proceso Unificado.

- Las MyPymes\_DS mediante estas aplicaciones de la guía y otras futuras podrán contar con una familia de procesos y disciplinas adaptadas en EPFC, las cuales podrán ser reutilizadas en otros proyectos específicos.
- De todo lo anterior se puede concluir que la guía de tailoring tuvo gran aceptación y las personas involucradas en la aplicación de la guía, manifestaron su facilidad en el proceso de aplicación y claridad en la definición.

**Nota:** Como soporte se tienen fotos de las aplicaciones de la guía (ver anexo E)

## **CAPITULO 5. LECCIONES APRENDIDAS, CONCLUSIONES Y TRABAJOS FUTUROS.**

Este capítulo tiene como objetivo realizar una colección de lecciones aprendidas y conclusiones relacionadas con el desarrollo del proyecto, así como también posibles trabajos que se salen del alcance del proyecto y que se podrían realizar en esta misma área de trabajo.

### **5.1 Lecciones Aprendidas**

En esta sección se realiza una recopilación de lecciones aprendidas obtenidas a partir del desarrollo del proyecto, como también de la aplicación de GTUP en el ambiente académico y empresarial.

- En este tipo de proyectos, donde se necesita la colaboración de personas y entidades externas para su desarrollo, es imprescindible tener en cuenta la disposición en tiempo y recursos asignados para llevar a cabo las fases del proyecto, evitando así, retrasos en el desarrollo. En este sentido, en los ambientes académico y empresarial recibimos colaboración ya que las actividades que aportaban en el desarrollo del proyecto se realizaban en horarios no laborales y no académicos. La parte empresarial dedicaba el horario de 6 a 8 pm. tres días a la semana de manera aleatoria y la parte académica los días sábados en horas de la mañana y cuando fue necesario en horas de la tarde.
- Con el fin de evitar malas interpretaciones conceptuales, se debe realizar una capacitación adecuada de GTUP a los miembros de las empresas, de forma que se garantice que ha sido entendida y se ha apropiado completamente; puesto que la temática de adaptación de procesos es un área poco conocida y esto se evidencio en las capacitaciones realizadas tanto a nivel académico como empresarial.
- Para que la ejecución de GTUP en las PyMEs\_DS sea exitosa, es necesario que las empresas tengan claros los conceptos referentes a UP, con el fin de que los procesos sugeridos por GTUP sean de fácil asimilación para los integrantes del equipo de desarrollo. Para verificar la claridad en los conceptos, de manera dinámica (charlas informales), se realizaban preguntas donde se establecía si el concepto era claro, o si por el contrario existían ambigüedades.

En caso de que esto último sucediera, se procedía a realizar la explicación del concepto a través de ejemplos o de manera más detallada.

## **5.2 Conclusiones Generales**

En esta sección, se detallan cada una de las conclusiones generadas durante el transcurso de este proyecto, tanto las generadas en el proceso de investigación, como de la aplicación de GTUP.

- Los estudios y modelos de tailoring que se encuentran en la literatura enfatizan en la necesidad de efectuar la adaptación del proceso software, pero no explican detalladamente cómo efectuar dicha adaptación, es decir, no especifican un camino claro y completo que oriente a las empresas cuando se enfrentan a la adaptación del proceso software.
- A pesar de la poca importancia que le ha dado la industria de desarrollo de software y las comunidades de investigación a la adaptación de procesos software, GTUP puede ser una buena iniciativa para dar a conocer esta temática y para encaminar a las PyMEs\_DS en el fortalecimiento del proceso software.
- GTUP puede servir para adaptar disciplinas de cualquier proceso software, ya que sus pasos de adaptación son genéricos y enfocados en la elección de las entidades necesarias para proceso.
- Utilizar SPEM como lenguaje de modelado del Proceso Unificado y de la guía de tailoring, brinda las capacidades para facilitar la comprensión y comunicación humana, facilitar la reutilización, dar soporte a la mejora de procesos, dar soporte a la gestión del proceso y guiar la automatización del proceso; además de brindar la posibilidad de documentar, presentar, administrar, intercambiar, mantener, publicar e implementar procesos.
- El contar con una descripción del proceso unificado mediante un Plugin en una herramienta gratuita es una ventaja para las PyMEs\_DS, pues facilita la definición, seguimiento, conocimiento del proceso software dentro de la organización ayudando en el fortalecimiento del proceso software.
- El resultado de este trabajo es la *Guía de Tailoring del Proceso Unificado a proyectos específicos para las PyMEs\_DS*, con la que se pretende orientar y ser un punto de apoyo para efectuar la adaptación del proceso software UP en las PyMEs\_DS.
- Las PyMEs\_DS son un poco apáticas a alterar sus tareas habituales, por lo tanto es difícil convencer a las personas involucradas para que apliquen conceptos o adopten actividades que alteren en cierta medida el curso normal de estas ya sea en tiempo o recursos.

- Sin la existencia de un plugin de UP, puede ser muy complicado manejar y gestionar la adaptación, debido a que la descripción de UP bajo SPEM permite contar con una visión global del proceso la cual puede facilitar su comprensión y por ende la adaptación del mismo.
- El éxito del resultado de aplicar la guía (proceso adaptado), depende de la(s) persona(s) encargada(s) de efectuar la adaptación, puesto que su conocimiento, habilidad y experiencia son de vital importancia en la toma de decisiones.

### **5.3 Trabajos futuros**

- Realizar nuevas aplicaciones de la guía en otras empresas y con las disciplinas faltantes.
- El tailoring de procesos software es un área que está madurando; GTUP abre la posibilidad de seguir investigando y trabajando en esta área. Los futuros proyectos de investigación en este contexto, cuentan desde ya con una base teórica y metodológica para seguir madurando y aportando en esta línea de investigación.

## Referencias

- [1] Grady Booch, Ivar Jacobson, and James Rumbaugh, Unified Modeling Language 1.3, White paper, Rational Software Corp.
- [2] Ivar Jacobson, Magnus Christerson, Patrik Jonsson, and Gunnar Övergaard, Object-Oriented Software Engineering—A Use Case Driven Approach, Wokingham, England, Addison-Wesley, 582p.
- [3] Grady Booch, Jim Rumbaugh, and Ivar Jacobson, Unified Modeling Language - User's Guide, Addison-Wesley, 1999.
- [4] Pedreira Oscar, Piattini Mario, Luaces R. Miguel and Brisaboa Nieves R. A Systematic Review of Software Process Tailoring. ACM SIGSOFT Software Engineering.
- [5] Rational Software Corporation, Product: Rational Software Corporation, 2002
- [6] Philippe Kruchten, The 4+1 View Model of Architecture, IEEE Software, 12 (6), November 1999, IEEE, pp.42-50.
- [7] Philippe Kruchten, A Rational Development Process, CrossTalk, 9 (7), STSC, Hill AFB, UT, pp.11-16.
- [8] Walker Royce, Software Project Management—A Unified Framework, Addison-Wesley, 1998.
- [9] Rational Software Corporation, Rational Unified Process. Best Practices for Software Development Teams, 1998
- [10] Ivar Jacobson, Grady Booch, and Jim Rumbaugh, Unified Software Development Process, Addison- Wesley, 1999.
- [11] MONTILVA, C. Jonás A., Ph.D. Mejoramiento de los procesos de desarrollo de software. Universidad de Los Andes Facultad de Ingeniería Postgrado en Computación Mérida, Venezuela.
- [12] Software & Systems Process Engineering Meta-Model Specification *Version 2.0*,OMG Document Number: formal/2008-04-01 Standard document URL: <http://www.omg.org/spec/SPEM/2.0/PDF>.
- [13] Fitzgerald, B., Russo, N., O'Kane, T. (2003): Software development method Tailoring at Motorola. Communications of the ACM 46(4), pp. 65-70.
- [14] Singh, R., International Standard ISO/IEC 12207 Software Life Cycle Processes. 1998.
- [15] Peng Xu. Knowledge Support in Software Process Tailoring. Management Proceedings of the 38th Hawaii International Conference on System Sciences, 03-06 Jan. 2005 p.87c.
- [16] IEEE/EIA, Industry Implementation of International Standard ISO/IEC 12207.2: 1995. (ISO/IEC 12207 standard for information technology - software life cycle processes - implementation considerations. IEEE/EIA, 1998.

- [17] Il-Chul Yoon, Sang-Yoon, Doo-Hwan Bae. Tailoring and Verifying software Process.IEEE. apsec, p. 202, Eighth Asia-Pacific Software Engineering Conference (APSEC'01), 2001
- [18] Ginsberg, M., Quinn, L. Process Tailoring and the software Capability Maturity Model. Technical report, Software Engineering Institute (SEI).
- [19] Tailoring Software Evolution Process. Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing.
- [20] Klingler Carol Diane, A CASE STUDY IN PROCESS DEFINITION, ACM 0-89791-621-2 /93/0009-0065 1.50.
- [21] Medición del Software Ontología y Metamodelo universidad castilla de la mancha.
- [22] Standish Group, T23E-T10E STANDISH GROUP REPORT [http://www.spinroot.com/spin/Doc/course/Standish\\_Survey.htm](http://www.spinroot.com/spin/Doc/course/Standish_Survey.htm).
- [23] SCHULTZ, D. et al., A Matrix Approach to Software Process Definitions, 25th Annual Software Engineering Workshop, Goddard Space Flight Center, 2000.
- [24] Saiedian Hossein, Carr Natsu. Characterizing a Software Process Maturity Model for Small Organizations. ACM SIGICE Bulletin.
- [25] C. C. Coelho, "MAPS: Modelo de Adaptación de proceso software".
- [26] Budlong Faye C, Szulewski Paul A. Tailoring a Software Process for Software Project Plans.
- [27] Haumer Peter, Eclipse Process Framework Composer Part 1: Key Concepts Second Revision, April 2007
- [28] Geir K. Hanssen, Hans Westerheim, Finn Olav Bjørnson, Tailoring RUP to a defined project type: A case study, Norway.
- [29] Watts S. Humphrey. Managing the Software Process. Addison-Wesley.
- [30] Rodríguez, D. Modelos para la creación y gestión del conocimiento: aproximación teórica. Educar, 37. 2006. p.p. 25-39.
- [31] Llantén. Carlos, Capote. Joanna. Marco Conceptual para la Implantación de Gestión del Conocimiento en un Programa de Mejora de Procesos Software en PyMEs\_DS.
- [32] TURK, D., FRANCE, R., RUMPE, B., Limitations of Agile Software Processes, XP2002: Extreme Programming Conference.
- [33] JONES, C., Positive and Negative Factors that Influence Software Productivity, Software Productivity Research, 1998.
- [34] COCKBURN, A., Selecting a Project's Methodology, IEEE Software, July/August 2000.
- [35] BOEHM, B., et al., Software Cost Estimation with COCOMO II, Prentice Hall, 2000.
- [36] THE STANDISH GROUP INTERNATIONAL, CHAOS: A Recipe for Success, The Standish Group International, Inc., 1999.

- [37] Caracterización de las MiPyMEs en Colombia. Disponible en:  
[http://www.camarasogamoso.org/index.php?option=com\\_remository&Itemid=278&func=startdown&id=17](http://www.camarasogamoso.org/index.php?option=com_remository&Itemid=278&func=startdown&id=17)
- [38] HERBSLEB, J.D., et al., An Empirical Study of Global Software. Development: Distance and Speed, International Conference on Software Engineering (ICSE 2201), 2001.
- [39] Zahran Sami, Software Process Improvement: Practical Guidelines for Business Success Addison-Wesley. USA 1998.
- [40] JONES, C., Project Management Tools and Software Failures and Successes, Software Productivity Research, CrossTalk, Julho 1998.
- [41] LEFFINGWELL, D., Agile Requirements Methods, The Rational Edge, July, 2002.