

# ANALIZANDO LA VARIABILIDAD DEL PROCESO UNIFICADO CON LA TÉCNICA DE WASHIZAKI

## UNIFIED PROCESS VARIABILITY ANALYSIS USING THE WASHIZAKI'S TECHNIQUE



### AUTOR

PABLO HERNANDO RUIZ MELENJE  
Magister(c)  
\*Institución Universitaria Colegio Mayor del Cauca  
Docente  
Grupo de Investigación I+D en Informática  
phruiz@unimayor.edu.co  
COLOMBIA

### AUTOR

JULIO ARIEL HURTADO ALEGRIA  
Doctor  
\*\*Universidad del Cauca  
Docente Titular  
Grupo de Investigación IDIS  
ahurtado@unicauca.edu.co  
COLOMBIA

### AUTOR

MARTA CECILIA CAMACHO OJEDA  
Magister  
\*Institución Universitaria Colegio Mayor del Cauca  
Docente Asistente  
Grupo de Investigación I+D en Informática  
cecamacho@unimayor.edu.co  
COLOMBIA

### \*INSTITUCIÓN

Institución Universitaria Colegio Mayor del Cauca  
UNIMAYOR  
Educación Superior  
Popayán  
Carrera 5 # 5-40  
admision@unimayor.edu.co  
COLOMBIA

### \*\*INSTITUCIÓN

Universidad del Cauca  
UNICAUCA  
Educación Superior  
Popayán  
Calle 5 No. 4 – 70  
rectoria@unicauca.edu.co  
COLOMBIA

**INFORMACIÓN DE LA INVESTIGACIÓN O DEL PROYECTO:** El presente artículo surge como resultado de la ejecución del proyecto denominado Una Familia de Procesos basada en el Proceso Unificado al interior del grupo de investigación IDIS, de la Universidad del Cauca, y el grupo I+D en informática, de la Institución Universitaria Colegio Mayor del Cauca.

**RECEPCIÓN:** Mayo 8 de 2013

**ACEPTACIÓN:** Mayo 24 de 2013

**TEMÁTICA:** Ingeniería del Software, Ingeniería de Procesos

**TIPO DE ARTÍCULO:** Artículo de Investigación Científica e Innovación

## RESUMEN ANALÍTICO

Adaptar el proceso software al contexto de la organización o a proyectos específicos es una actividad recurrente en la industria de software. Una de las técnicas efectivas para la adaptación del proceso software son las líneas de procesos, debido a que permiten disminuir el esfuerzo en la generación de procesos adaptados mediante la reutilización planificada de los activos de proceso. Una actividad esencial para definir líneas de procesos es determinar adecuadamente su variabilidad para alcanzar flexibilidad, adaptabilidad y reutilización. Este artículo presenta la experiencia de determinar la variabilidad en el Proceso Unificado con respecto a algunas de las adaptaciones y extensiones existentes, usando la técnica de Washizaki para la comparación de modelos de procesos. El principal resultado es un modelo de variabilidad del Proceso Unificado, el cual es un artefacto clave para definirlo como una línea de procesos. Para su obtención el modelo ha sido analizado estáticamente. Además, este trabajo ha permitido evaluar la técnica de comparación de modelos de proceso usando la información empírica de algunos procesos existentes y reportados por distintas comunidades.

**PALABRAS CLAVES:** Variabilidad del proceso Software, Proceso software, Familias de procesos, Proceso Unificado, Adaptación del proceso.

## ANALYTICAL SUMMARY

Adapting the software process to the context of the organization or specific projects is a recurring activity in the software industry. One of the effective techniques for the tailoring software process are the process lines, because they allow to reduce the effort on the generation of adapted processes through planned reuse of process assets. An essential activity to define processes lines is adequately determine their variability to achieve flexibility, adaptability and reuse. In this article presents the experience to determine the variability in the Unified Process with respect to some of the existing adaptations and extensions, using the Washizaki's technique to compare process models. The main result is a model of variability of the Unified Process, which is a key artifact to define it as a process line. In order to obtain it the model has been analyzed statically. This work has allowed to evaluate the comparison technique of models using the empirical information of some existing processes which have been reported by different communities.

**KEYWORDS:** Software process variability, Software Process, families Process, Unified Process, Tailoring Process.

## INTRODUCCIÓN.

El proceso software utilizado en cada organización debe ser adaptado, para que cumpla con las necesidades organizativas o de los proyectos específicos [1] e incluso las circunstancias de un proyecto global en el caso de GSD (Global software Development) [2], de esta forma la adaptación es una tarea recurrente, que

consume tiempo y demanda conocimiento para las organizaciones [3]. El Proceso Unificado (UP) es framework de procesos ampliamente utilizado el cual ha ganado mucho interés en la industria [4] [5]. No existe un cálculo real sobre cuántas organizaciones han tratado y utilizado UP; sin embargo una visión general de reportes de experiencia desde conferencias de ingeniería de software, libros y publicaciones en

revistas indican un considerable interés en el uso de UP en la industria mundial [4]. Particularmente, los estudios realizados en [6] [7] sobre pequeñas empresas de software arrojaron que cerca del 50% de las empresas lo utilizan como marco de referencia. En los trabajos de Hanssen et al. [4] [5], UP es presentado como un marco de procesos adaptable [8] a través de una guía general de adaptación. Sin embargo, ésta guía no proporciona elementos para decidir cómo los elementos del proceso deben ser seleccionados o adaptados de acuerdo a las necesidades específicas de un proyecto [4]. Por esta razón muchas de las adaptaciones del Proceso Unificado no son repetibles y por tanto, cada nueva adaptación requiere de experiencia, por lo cual la adaptación de este proceso en la práctica es propensa al error y requiere de un esfuerzo enorme. Actualmente existen diferentes estrategias para realizar la adaptación de procesos software a proyectos específicos, pero carecen de direccionamiento sobre la consistencia de los procesos derivados o construidos. Además, algunos no permiten de forma coherente y fácil reutilizar partes o fragmentos de los procesos.

En la sección 1 presenta la formulación del problema y la propuesta general. En la sección 2 se describen los trabajos relacionados con el análisis de la variabilidad en líneas de procesos y las adaptaciones del UP y sus derivados. En las secciones 3, 4 y 5 se presenta la variabilidad de la línea de procesos basada en el Proceso Unificado. En la sección 6, se presenta los resultados y discusión, y en la sección 7 se muestran las conclusiones y trabajo futuro.

## 1. FORMULACIÓN DEL PROBLEMA Y DE LA PROPUESTA.

El UP por ser un marco 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 [8]. La adaptación de UP requiere de un trabajo meticuloso y complicado [4], convirtiéndola en uno de sus principales desafíos. UP no direcciona de forma clara su adaptación a unas necesidades específicas [9], ofrece una serie de instrucciones orientadas a la eliminación de elementos no aplicables, adición de elementos requeridos, y modificación de elementos existentes para un proyecto particular. Sin embargo estas instrucciones son orientaciones informales y tienden a ser superficiales [4]. Es claro que UP necesita ser adaptado, pero no

existen las suficientes directrices para efectuar la adaptación, y definitivamente es latente la necesidad de buenas pautas y recomendaciones para realizarla de una forma exitosa [5]. Debido a esto las adaptaciones que las empresas puedan hacer del Proceso Unificado no son costo-efectivas, y se podría estar perdiendo valor en la aplicación de las prácticas que el proceso brinda, agregando los riesgos y costos que podría conducir el usar el proceso inadecuado. La efectividad es baja porque no hay expertos y el costo es alto por el esfuerzo requerido para conducir una tarea de forma manual y sin experiencia. Dentro del estado del arte y de la práctica se encuentran algunos esfuerzos por solucionar el problema [8] [10] [11] [12] y herramientas como EPFC (Eclipse Process Framework Composer) y RMC (Rational Method Composer). Sin embargo, estos enfoques resultan limitados porque la adaptación es manual y depende de la experiencia, haciéndola lenta y propensa al error, además no son acercamientos formales que permiten una adaptación de forma sistemática y metodológica que faciliten la reutilización de conocimiento y experiencia.

Lograr una adaptación sistemática del proceso, requiere identificar formalmente por un lado las posibilidades válidas de variación del Proceso Unificado y por otro qué aspectos del contexto de aplicación del proceso determinan la escogencia adecuada de éstas variaciones. Así, surge la pregunta de investigación ¿Cómo determinar la variabilidad del Proceso Unificado con el fin de ayudar a soportar una adaptación sistemática del proceso? Una de las estrategias que brinda mayor expectativa para facilitar la adaptación (reutilización y evolución) de forma planificada son las líneas de procesos software [13] [14]. Éstas facilitan la adaptación mediante el intercambio de activos de proceso dependiendo de las circunstancias de ejecución [13]. De acuerdo a Simidchieva et al. [13], definir una línea de procesos puede brindar al proceso software los siguientes beneficios: i) Optimizar los esfuerzos en la coordinación, automatización, mejora y entrenamiento. ii) Permitir la reutilización en la definición de nuevos procesos. iii) Facilitar la adaptación de procesos mediante el intercambio de componentes dependiendo de las circunstancias de ejecución. Definir una línea de proceso no es una tarea sencilla, requiere de tiempo, planeación y experiencia. Washizaki [15] presenta una estrategia basada en la comparación de modelos de proceso para definir la arquitectura de la línea de procesos que incorporan elementos comunes y variables en dominios de problemas particulares. En este artículo dicha estrategia es aplicada para determinar la variabilidad de un conjunto de modelos de procesos

basados en el UP y de esta forma definir la arquitectura de una línea de procesos basada en el UP.

## 2. TRABAJOS RELACIONADOS.

Rombach [16] propone el concepto de líneas de proceso software [17] como una forma de gestionar un proceso y sus variantes de manera sistemática. El conjunto de procesos derivables de la línea se denomina una familia de procesos [17]. Las líneas de procesos software son organizadas de acuerdo a las similitudes y diferencias de sus procesos, permitiendo una mejor adaptación a las necesidades de un proyecto específico. Simidchieva et. al. [13] proponen que una familia de procesos, puede ser considerada y definida como el conjunto de diferentes variaciones acordadas de un mismo proceso que pueden ser gestionados como una única línea de proceso. Aunque Simidchieva et. al. [13] y Rombach [16] establecen la necesidad de líneas y familias de procesos respectivamente, son ideas iniciales que aún no indican claramente cómo lograr estos conceptos en forma concreta. Schnieders et al. [18] da soporte a las familias de procesos desde una perspectiva de familias de productos en el dominio de los procesos. Muestra como los aspectos de comportamiento de la arquitectura de una familia de procesos pueden modelarse con diagramas de actividades enriquecidos, y también describe algunas reglas para derivar arquitecturas de procesos. Schnieders [19] motiva la necesidad de la ingeniería de familias de procesos, y presenta enfoque centrado en el mecanismo de variabilidad para modelar arquitecturas de familias de procesos. Hurtado et al. [20] proponen un método, concreto basado en técnicas MDE, para la adaptación sistemática de procesos. Uno de sus fundamentos son las familias de proceso, donde se define un proceso y/o elementos del proceso adecuados para ser aplicados a una situación específica. CASPER [21] es un meta-proceso para definir modelos de procesos software adaptables mediante familias de procesos. Utiliza coherentemente tres enfoques principales: líneas de procesos software, modelamiento del contexto extendiendo modelado del proceso y se basa en transformación de modelos para efectuar la adaptación. Por su parte Araújo et al. [22] definen un meta-modelo para describir el rol y la integración del mecanismo de variabilidad en la arquitectura de líneas de procesos. Araújo propone un acercamiento para la gestión y personalización de las variabilidades en procesos software, y también da soporte a la gestión de las variaciones promoviendo la

derivación automática. En [23] se identifican y analizan los principales aspectos críticos y problemas en la definición de la arquitectura de procesos en un entorno multimodal, además se propone un conjunto de criterios base para diseñar una arquitectura de procesos. Barreto et al. [24] presentan un acercamiento para utilizar líneas de procesos software en empresas de consultoría de procesos. El enfoque describe cuatro características principales: arquitectura de procesos software, línea de procesos software, capacidad de derivación de procesos y uso de componentes de acuerdo al contexto. Para soportar el enfoque se propone una herramienta de apoyo sólo en la definición de los procesos. Ocampo et al. [25] describen SPEARSIM, una herramienta para analizar similitudes y diferencias entre los procesos. La herramienta fue diseñada para apoyar al ingeniero de procesos en la comparación de procesos grandes y complejos. Washizaki [15] define y aplica una estrategia de comparación de modelos de proceso para recuperar una arquitectura de línea de procesos que incorpora las variabilidades.

Para los anteriores enfoques, existe poca evidencia de su aplicación a procesos de software establecidos y ampliamente conocidos y aplicados por la industria, como es el caso del UP. Existe evidencia empírica sobre adaptaciones del UP, es el caso de procesos como: Agile UP (AUP) [26], Basic Unified Process (BUP) [27], Open Unified Process (OpenUP) [28], Open Unified Process Basic (OpenUP/Basic) [27] y Enterprise Rational Unified Processes (EUP) [29]. Estas adaptaciones y extensiones de UP, deben aún ser adaptadas para cubrir las necesidades de las diferentes organizaciones y proyectos, puesto que siguen teniendo un alcance muy amplio. Concretamente el Proceso Unificado en su disciplina de ambiente brinda una guía de ajuste para la adaptación [11], donde se identificaron factores como: contexto del negocio, esfuerzo en el desarrollo, grado de innovación, tipo de aplicación, factores de la organización y la complejidad técnica y de gestión, así como elementos relevantes que pueden influir y guiar la adaptación del proceso. En este mismo enfoque de adaptación se encuentra una guía de adaptación del Proceso Unificado a contextos específicos de proyectos en el ámbito de pequeñas organizaciones presentado en [11]. Pereira et al. [10] proponen un enfoque basado en la definición de un meta-modelo que extiende de RUP incluyendo nuevos elementos y relaciones necesarias para permitir la adaptación del proceso, se orienta a la definición de un proceso adaptable, pero no define los mecanismos para identificar la variabilidad, necesaria

para tomar las decisiones en la adaptación. Un modelo genérico para la adaptación es presentado por Coelho [12], donde utilizando un modelo de comparación de características permite recuperar y reutilizar posibles partes de procesos previamente definidos. No obstante, el proceso es adaptado en forma manual. Esta adaptación fue evaluada en una empresa de desarrollo de software en la que se adaptó la disciplina de planificación y gestión de RUP. Patel et al. [30] proponen un marco conceptual dirigido a las empresas y equipos de desarrollo de software para guiar la adaptación de procesos a través de un método, aunque la adaptación no es asistida, presentan un caso de estudio industrial aplicando su propuesta en una empresa en la cual se ha adoptado RUP como proceso organizacional.

Los mecanismos de adaptación propuestos para el Proceso Unificado no se enmarcan en la definición de una línea de procesos y por tanto la variabilidad de los procesos no es identificada sistemáticamente. Este trabajo propone aprovechar el estado del arte en la definición de procesos adaptables y de líneas de procesos para definir un modelo de variabilidad del Proceso Unificado. Este modelo permitirá definir la arquitectura y elementos del modelo de una línea de procesos basada en el UP, así como la definición de una estrategia de derivación sistemática de los miembros de la familia a partir de la línea de procesos. Para analizar la variabilidad, este trabajo usa la técnica de comparación de Washizaki [15] considerando el conjunto de miembros potenciales: RUP, EUP, OpenUP y BUP.

Aunque en la literatura existen herramientas como la de [25] para hacer un análisis de variabilidad automático, el análisis realizado en este estudio no las ha considerado, debido a que las herramientas disponibles se basan en algoritmos de comparación de procesos, los cuales tienen deficiencias de carácter semántico y pragmático que fueron aspectos claves en la realización de este análisis.

### 3. TÉCNICA SEGUIDA PARA ANALIZAR LA VARIABILIDAD DEL PROCESO.

El lineamiento seguido en este trabajo para determinar la variabilidad del UP es descrito por Washisaki [15]. Se escogió esta técnica debido a que es la más clara encontrada en la literatura. Esta técnica permite crear Arquitecturas de Líneas de Procesos (PLA) que incorporan

elementos comunes y variabilidad en dominios particulares. En esta técnica se define la ingeniería de líneas de procesos como un Sistema de estrategias interrelacionadas y métodos sistemáticos para construir, aplicar y gestionar líneas de procesos. Basados en este concepto se define las siguientes actividades que pertenecen a la ingeniería de dominio y de aplicación para crear una PLA. 1) *Selección de procesos*: tiene como objetivo reunir el conjunto de procesos en un dominio de problema que comparten partes comunes y pueden ser combinados para formar la línea de procesos. 2) *Factor Común y Variantes*: tiene la finalidad de identificar el proceso núcleo que incluye puntos de variación, así como el conjunto de variantes, esta actividad se divide en Definir el Proceso Núcleo y Comparar los elementos entre los procesos. 3) *Requisitos*: tiene el propósito de definir los requisitos previsibles para la línea de proceso de forma que correspondan a lo común y la variabilidad incorporada en la PLA como un modelo de características. 4) *Derivación*: tiene como objetivo utilizar los requisitos para configurar el diagrama de características y la PLA para generar un proceso personalizado. Las actividades de 1) a 3) son de ingeniería de dominio y 4) es de ingeniería de aplicación. La técnica emplea diagramas de actividades descritos en UML para especificar los procesos, y también extensiones de SPEM para detallar lo común y lo variable del proceso. Se utilizan diagramas de características para especificar los requisitos, de la línea de procesos, que corresponden a lo común y a lo variable de la PLA.

Este artículo se enfoca únicamente en determinar la variabilidad del UP en el marco del meta-proceso CASPER [21]. En el enfoque de CASPER el análisis de variabilidad del proceso se ajusta a las actividades 1 y 2 de la técnica de Washizaki [15]. Las actividades 3 y 4 de [15] se asocian al conocimiento contextual y a las reglas para la toma de decisiones de adaptación, lo cual corresponde a una actividad posterior a ésta investigación. Las actividades 3 y 4 serán realizadas utilizando técnicas complementarias brindadas por el enfoque CASPER (Modelo de Características, Modelo de Contexto, Modelo de Alcance y Reglas de Adaptación) tomando como base la variabilidad identificada en este artículo. La técnica descrita fue utilizada para todas las disciplinas del UP, a continuación se muestran los resultados generales. Con el fin de comprender la aplicación se presentan los resultados particulares de analizar la variabilidad en la disciplina de Requisitos.

## 4. APLICANDO LA TÉCNICA PARA DETERMINAR VARIABILIDAD EN UP.

### 4.1 SELECCIÓN DE PROCESOS.

En esta actividad se hizo un análisis a los posibles procesos, extensiones y adaptaciones de UP, que conformarían la línea de procesos. Algunas versiones encontradas de UP fueron: Open Unified Process (OpenUP), Rational Unified Process (RUP), Enterprise Unified Process (EUP), BasicUP (BUP), CMMI Unified Process (CUP), Essential Unified Process (EssUP), Grown UP (GUP), Java Unified Process (JUP), Xerox Unified Process (XUP) entre otras. Se hizo un análisis de las descripciones de los elementos de los procesos como fases y disciplinas, que nos permitiera asegurarnos que mantuvieran lo esencial del UP, que su objetivo principal fuera desarrollo de software y su disponibilidad para el análisis. En la tabla 1 se muestra de forma resumida el análisis de los procesos de acuerdo a los criterios de selección.

**TABLA 1.** Análisis de Procesos.

Criterios Procesos	Criterios		
	Conforme al UP	Orientado al Desarrollo de Software	Disponibilidad del Proceso
Open UP	Parcialmente	Si	Librería Libre en EPF
RUP	Completamente	Si	Librería Comercial RMC
EUP	Completamente	Parcialmente	Textual
BUP	Completamente	Si	Textual
CUP	Parcialmente	Parcialmente	Textual limitada
EssUP	Parcialmente	SI	Textual limitada
GUP	Parcialmente	SI	Textual limitada
JUP	Parcialmente	SI	Textual limitada
XUP	Parcialmente	No	Textual limitada

Luego de analizar las diferentes versiones del UP bajo los criterios anteriores, se seleccionaron como miembros de la línea de procesos a UP, RUP, EUP, OpenUP y BUP, debido a que fueron los procesos que cumplieron de mejor forma los criterios de selección.

RUP [31] es una variante comercial de UP. Proporciona un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es garantizar la producción de

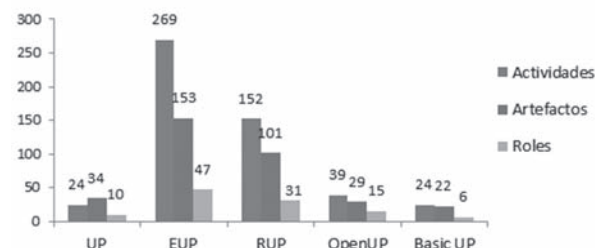
software de alta calidad que satisfaga las necesidades los usuarios finales, dentro de una planificación y presupuesto predecibles. Por su parte EUP [29], es una extensión de RUP, que adiciona fases y disciplinas que tratan de cubrir algunas falencias de RUP. OpenUP [28] es una versión liviana mínima, suficiente y abierta de UP, no provee lineamientos para todos los elementos que se manejan en un proyecto, pero sí cuenta con los componentes básicos que pueden servir de base a procesos específicos. BUP [27] es una versión simplificada de OpenUP. Es un proceso sencillo que sigue siendo fiel a los principios de RUP, pero es dirigido a proyectos y equipos pequeños.

### 4.2 FACTOR COMÚN Y VARIANTES.

#### 4.2.1 Definir el proceso núcleo inicial.

En esta actividad se define el proceso núcleo inicial el cual será comparado con los demás miembros que harán parte de la familia. La elección inicial del proceso núcleo se puede hacer por medio de selección del proceso más pequeño entre los miembros de la familia [15]. Para hacer la selección fue necesario determinar, para cada uno de los procesos, la cantidad de elementos de proceso, actividades, artefacto y roles, para posteriormente realizar una comparación en términos de la cantidad y así determinar el proceso núcleo inicial. Como resultado de esta comparación Basic UP, UP y OpenUP fueron los procesos núcleo candidatos, debido a que poseen un número menor de elementos que los demás como se muestra en la figura 1. Esta cuantificación se realizó manualmente a partir de las especificaciones de los cinco procesos disponibles en [8] [27] [28] [29] [31]. Esto fue debido a que la mayoría de especificaciones de estos procesos son textuales y no están formalmente especificadas.

**FIGURA 1.** Comparación de Procesos mediante actividades, artefactos y roles.





De los procesos candidatos, el proceso seleccionado fue el UP, debido a que a pesar de ser el segundo en tamaño, es el proceso originalmente formulado y además en la literatura se encuentra mayor información de su descripción y conjunto de decisiones de diseño detrás del proceso.

Es importante aclarar que al iniciar el análisis las actividades del proceso núcleo son todas las del UP, pero esto cambiará en la medida que el análisis de todos los procesos de la familia avance.

#### 4.2.2 Estrategia de identificación de variabilidad.

En este trabajo se hizo el análisis completo de los flujos de trabajo de Requisitos, Análisis, Diseño, Implementación y Pruebas de UP, con los correspondientes flujos de los procesos que son parte de la familia. Sin embargo, en esta sección solo se detalla la variabilidad encontrada en la disciplina de requisitos. La clasificación de los elementos de proceso se estableció en: punto de variación, variante, opcional y núcleo. El punto de variación es una actividad vacía del proceso que tiene relaciones con actividades variantes, es decir la relación entre un punto de variación y sus variantes es que durante la configuración de un proceso es obligatorio que cada punto de variación sea remplazado por una de sus variantes. La opcionalidad de una actividad se refiere a que puede ser elegida sin ninguna obligación en la configuración de un proceso y el núcleo del proceso son aquellas actividades permanentes que definen la estructura general del proceso. Debido a que las definiciones de los elementos del proceso en la comparación no siempre coinciden fue necesario fusionarlos e incluso darles una identidad. Para el caso de las variantes, fue necesario en algunos casos utilizar estrategias de abstracción para generalizar los elementos del proceso. La estrategia de identificación de variabilidad seguida, consiste en hacer la comparación actividad por actividad de los procesos. Particularmente en este trabajo la comparación se hizo teniendo en cuenta el nombre de las actividades, su definición y su estructura de desglose. A continuación se explica brevemente el proceso de comparación seguido en este trabajo. Sea A, el conjunto de actividades que hacen

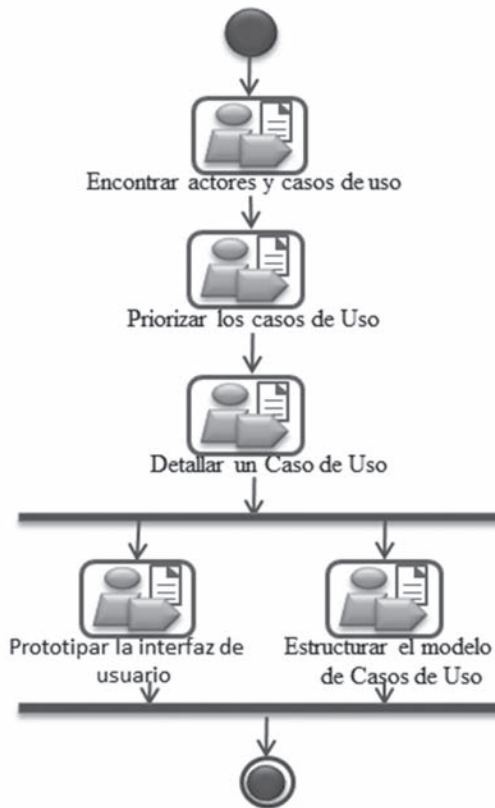
parte del proceso núcleo, y B el conjunto de actividades que hacen parte del proceso a comparar. Entonces, *i)* Si una de las actividades de B tiene similitud con una de A, las actividades son fusionadas en una sola y pasaran a ser parte del núcleo. *ii)* Si una actividad de B es una especialización de una de A, entonces la actividad de B pasa a ser una variante y la actividad de A pasa a ser un punto de variación. *iii)* Si una actividad B es una generalización de una de A, la actividad de B pasa a ser un punto de variación mientras que la actividad de A pasa a ser una variante. *iv)* Si la comparación de una actividad de B no arroja algún tipo de similitud, entonces es necesario verificar que su actividad antecesora posee algún tipo de variabilidad, si es así entre la actividad comparada y su antecesora se establece una transición de requerida. *v)* Si la comparación de una actividad de B no arroja algún tipo de similitud y se considera que es una actividad importante puede pasar a ser parte de la línea de procesos como una actividad opcional. *vi)* Para los casos *ii)*, *iii)* y *v)* es necesario agregar las diferentes transiciones para que los procesos generados sean consistentes. *vii)* Si existen actividades que tienen más de dos transiciones de salida es necesario restringirlas mediante exclusividad, de forma que en la generación de un proceso en particular solo se seleccione una transición.

##### 4.2.2.1 Análisis particular de la variabilidad en la disciplina de requisitos.

Siguiendo la técnica de Washizaki [15] se hizo la comparación de los flujos de trabajo de requisitos de los procesos miembros con respecto al flujo de requisitos de UP mostrado en la figura 2.

El análisis se inicia haciendo la comparación de los flujos de requisitos UP y Open UP. La primera actividad comparada fue "Identificar y Esbozar Requisitos" de Open UP, ver Figura 3, con las actividades del flujo de trabajo de UP. Se encontró que la actividad "Identificar y Esbozar requisitos" de Open UP, es una generalización de "Encontrar actores y Casos de uso" de UP, en este caso según [15] se hizo un intercambio entre las dos actividades, donde la actividad de UP pasa a ser la variante, una especialización, denotada por <<V>> y la actividad de Open UP queda como punto de variación denotado por <<PV>>.

FIGURA 2. Flujo de trabajo de Requisitos UP.

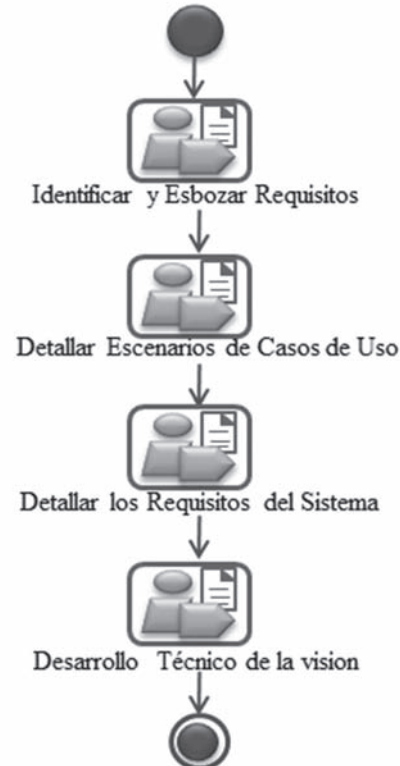


Fuente: [8]

La comparación de la actividad “Detallar Escenario de caso de Uso” de Open UP arrojó que era una especialización, es decir una variante, de la actividad “Detallar un Caso de Uso” de UP, por lo tanto se adicionó al flujo como una variante y “Detallar un Caso de uso” pasó a ser un punto de variación. Siguiendo con la comparación, la actividad del flujo de Open UP “Detallar los requisitos del Sistema” se identificó que no tiene algún tipo de variabilidad con las actividades del núcleo. Cuando sucede esto, de acuerdo a [15], es necesario devolverse a la actividad antecesora en el flujo de trabajo que es comparado, en este caso la actividad “Detallar Escenario de caso de Uso”, y verificar si provee algún tipo de variabilidad. Como “Detallar Escenario de caso de Uso” es una variante, se debe establecer una transición, enlace de requerido denotado por <<R>>, entre “Detallar Escenario de caso de Uso” y la actividad “Detallar los requisitos del Sistema”, donde esta última actividad pasa hacer parte de la variante al igual que su antecesora, con la restricción de dependencia entre ambas (Se escogen las dos o ninguna). La última actividad del flujo de Open UP “Desarrollo Técnico de la Visión” no tiene algún tipo de variabilidad con las actividades del núcleo, y como su antecesora “Detallar los requisitos del Sistema” es una variante, se establece

una transición, enlace de requerido, entre “Detallar los Requisitos del Sistema” y la actividad “Desarrollo técnico de la visión”, donde esta última actividad pasa hacer parte de la variante al igual que su antecesora, con la restricción de dependencia entre ambas.

FIGURA 3. Flujo de trabajo de Requisitos de Open UP.



Fuente: [28]

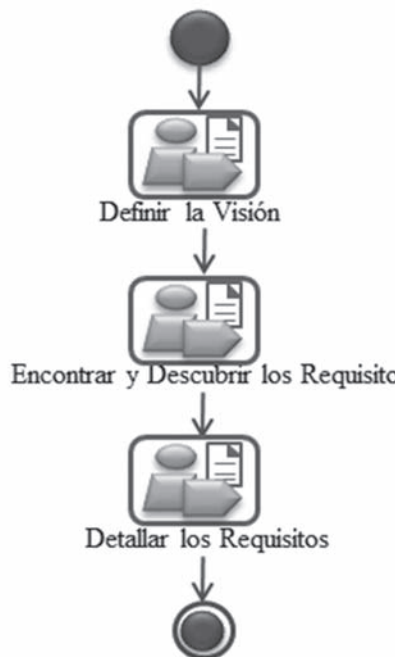
La siguiente comparación fue entre el flujo de trabajo de Basic UP presentado en la Figura 4 y el flujo del proceso núcleo actualizado. Entonces, haciendo un análisis de comparación se encontró que la actividad “Definir la Visión” de Basic UP no poseía algún tipo de variabilidad, pero debido a que la visión es importante cuando el producto es nuevo o para las primeras iteraciones del proceso surgió la necesidad de adicionarla al flujo como una actividad opcional denotada por <<O>> al inicio de flujo de trabajo. Continuando con la comparación, las actividades “Encontrar y Describir los requisitos” y “Detallar los requisitos” de Basic UP estaban inmersas en “Identificar y Esbozar Requisitos” y en “Detallar un Casos de Uso” respectivamente, por lo tanto estas actividades se fusionaron y pasaron a ser parte del núcleo de forma que el flujo queda igual.

El siguiente flujo comparado fue el de RUP, ver figura 5, la primera actividad es “Analizar el problema”, al comprar ésta actividad se identificó que era una especialización de



“Identificar y Esbozar Requisitos”, de tal forma que pasó a hacer una variante. La comparación de las actividades “Entender las necesidades de los Stakeholders” y “Definir el sistema” de RUP no arrojó algún tipo de variabilidad. Entonces, fue necesario verificar si las actividades antecesoras a estas actividades en el flujo de RUP tenían variabilidad. La actividad antecesora de “Entender las necesidades de los Stakeholders” es “Analizar el Problema” la cual es una variante, por lo tanto por se estableció una transición requerida entre las dos actividades. La actividad “Definir el Sistema” tiene como antecesora a “Entender las necesidades de los Stakeholders” que es a su vez parte de la variante formada con “Analizar el Problema”, de tal forma que se procedió a establecer una transición requerida entre “Entender la necesidades de las Stakeholders” y “Definir el sistema”. Las tres actividades enlazadas por medio de las dos transiciones requeridas definidas anteriormente hacen parte de una sola variante.

FIGURA 4. Flujo de trabajo de Requisitos de BUP.



Fuente: [27]

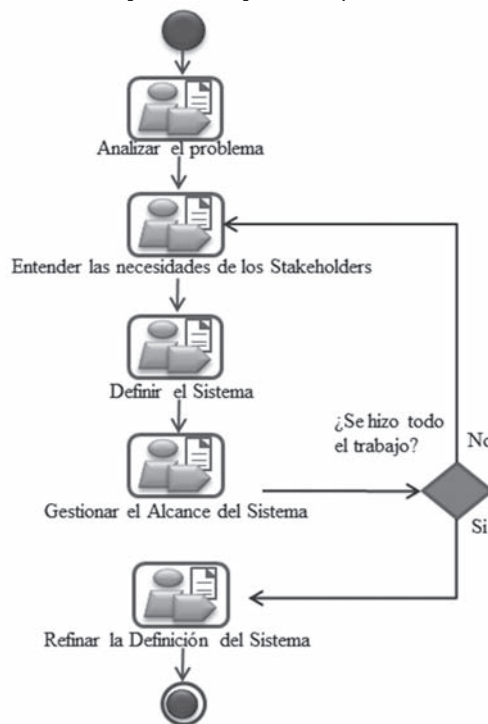
Por su parte la actividad “Gestionar el Alcance del Sistema” es una generalización de “Priorizar casos de Uso”, de esta forma pasará a ser punto de variación y “Priorizar casos de Uso” una variante, de la misma forma “Refinar la Definición del Sistema” con “Detallar un Caso de Uso”.

Para que no se pierda la estructura de los procesos que hacen parte de la familia, es necesario agregar las

transiciones necesarias entre las diferentes actividades. Además, para que no exista ambigüedad en los procesos generados, en el caso que una actividad tenga más de dos transiciones de salida, es necesario etiquetarlas con << XOR >>, de forma que una solo transición debe ser seleccionada para un proceso concreto.

El flujo de trabajo resultado del análisis de variabilidad se muestra en la Figura 6. Donde se puede observar que la actividad “Definir la Visión” es opcional. Por su parte la actividad “Identificar y Esbozar Requisitos” tiene dos variantes descritas por “Encontrar Actores y casos de Uso” y la variante compuesta por “Analizar el problema”, “Entender las necesidades de los Stakeholders” y “Definir el Sistema”. La siguiente actividad “Gestionar el Alcance del sistema” es un punto de variación donde “Priorizar Casos de Uso” es su variante. La siguiente actividad “Refinar la definición del Sistema” es un punto de variación que tiene como variante a “Detallar un Caso de Uso” que a su vez es punto de variación y tiene como variante al conjunto de actividades “Detallar Escenarios de casos de Uso” y “Detallar Requisitos del Sistema”. Las dos últimas actividades “Prototipar de interfaz de usuario” y “Estructurar el modelo de casos de uso” son del núcleo del proceso las cuales no tienen ningún tipo de variabilidad.

FIGURA 5. Flujo de trabajo de Requisitos de RUP.



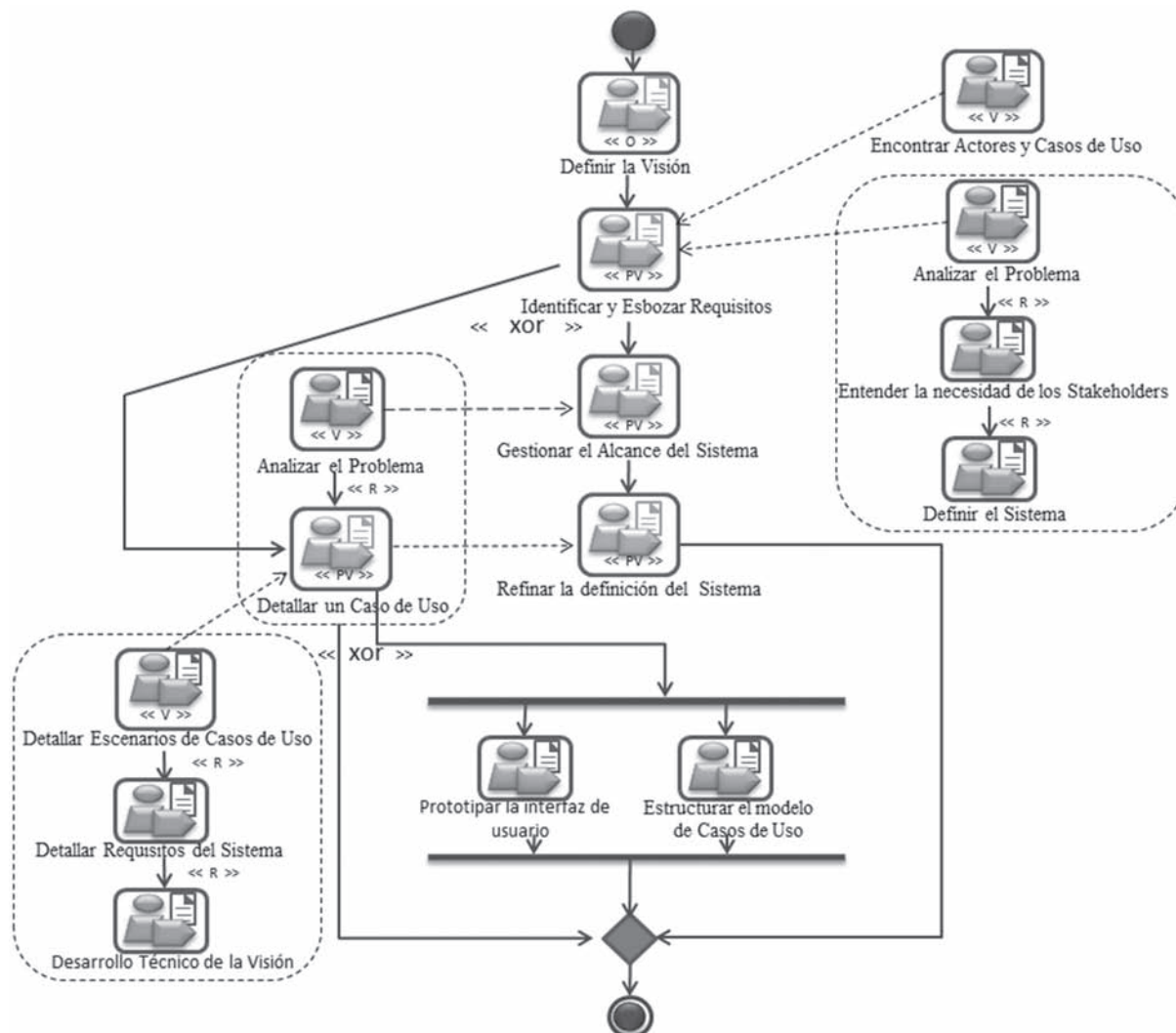
Fuente: [31]

#### 4.2.2.2 Encapsular actividades y resolver puntos de variación.

Aunque esta actividad y la actividad Definir Restricciones no son parte de la metodología de Washizaki [15] fue necesario incorporarlas durante el diseño de la familia de proceso. El encapsulamiento de las actividades se hace con el objetivo de agrupar un conjunto de actividades que son parte de una variante y que tienen transiciones

definidas entre ellas como requeridas. Es el caso de las actividades "Analizar el Problema", "Entender las Necesidades de los Stakeholders" y "Definir el sistema" las cuáles serán encapsuladas en una actividad general llamada "Requisitos". También las actividades "Detallar escenarios de Casos de uso" y "Detallar Requisitos del Sistema" y "Desarrollo Técnico de la visión" serán encapsuladas en la actividad "Detallar Escenarios de Casos".

FIGURA 6. Flujo de trabajo de Requisitos de la línea de procesos.



Los puntos de variación en las líneas de procesos son actividades del flujo de trabajo que son vacías y por lo tanto siempre que se configure un proceso es necesario seleccionar una actividad que tomara el lugar del punto de variación. En el caso de Washizaki [15] la actividad que es punto de variación es una generalización y por lo tanto no es vacía, debido a que tiene algo en común

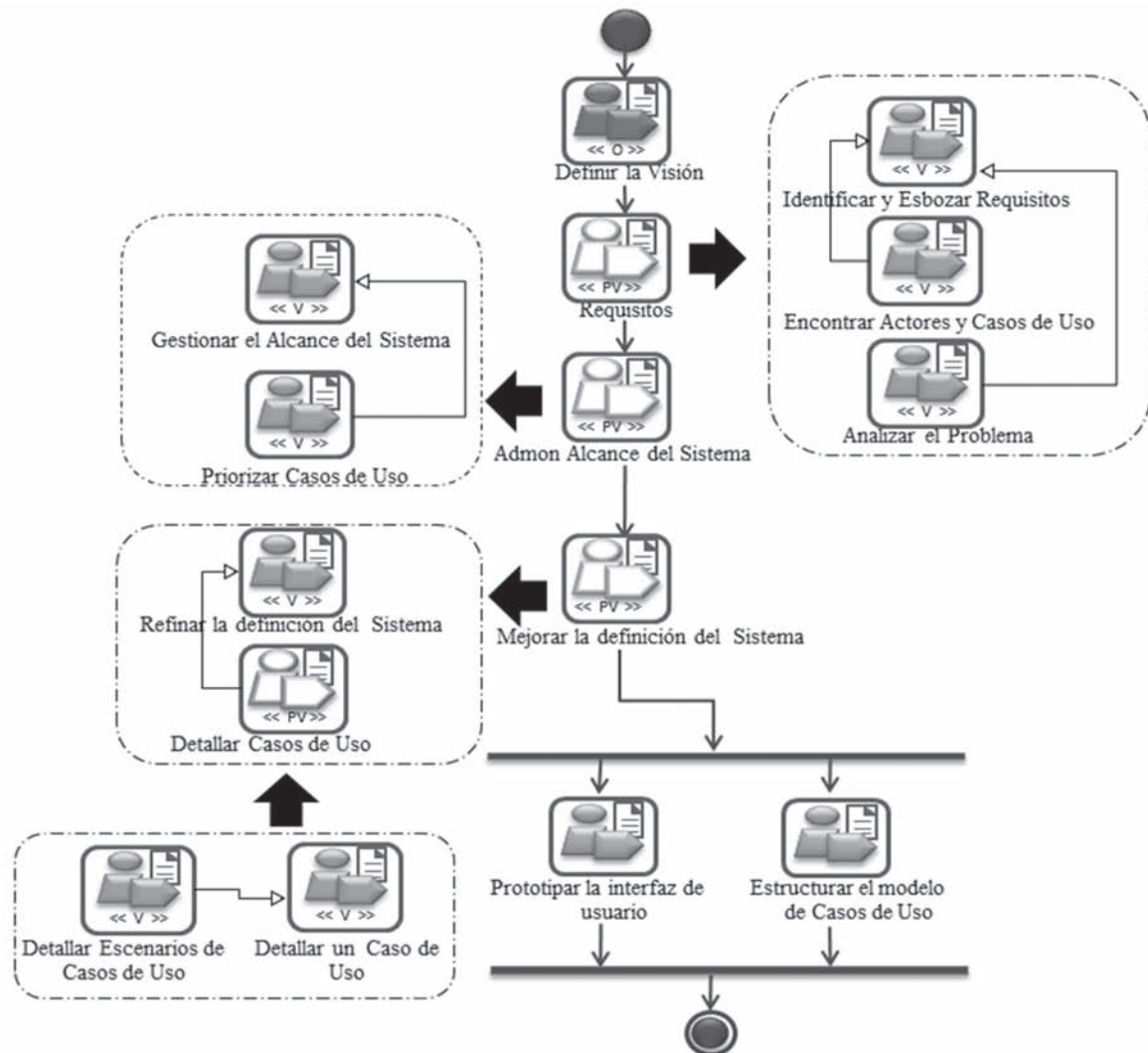
con sus especializaciones, entonces para ser conformes al modelamiento de líneas de procesos fue necesario que el punto de variación encontrado con la técnica de Washizaki pasara a ser también una alternativa de un punto de variación más general. La idea es seguir la nomenclatura definida para los puntos de variación en las líneas de proceso y así también poder utilizar

herramientas para modelar la variabilidad del proceso como lo es [32]. De esta forma el primer punto de variación resuelto fue el definido inicialmente por la actividad "Identificar y Esbozar Requisitos", la cual pasa a ser una variante de una actividad más general llamada "Requisitos", de tal forma que ahora el punto de variación está definido por "Requisitos" donde sus tres variantes son "Identificar y Esbozar Requisitos", "Encontrar actores y Casos de uso" y "Analizar el Problema". Todos los puntos de variación identificados fueron resueltos de la misma manera, donde el representado por "Gestionar el Alcance del Sistema" queda redefinido por la actividad general "Gestión del Alcance del Sistema" y sus variantes "Priorizar Casos de uso" y "Gestionar el alcance del sistema". El punto de variación "Refinar la definición del Sistema" fue redefinido mediante "Mejorar la Definición del Sistema" con sus variantes "Refinar la

Definición del Sistema" y "Detallar casos de Uso".

A su vez, el punto de variación "Detallar un caso de uso" fue redefinido mediante la actividad general "Detallar Casos de Uso" y sus variantes "Detallar Escenarios de Casos de Uso" y "Detallar un Caso de Uso". Como en la solución de los puntos de variación, las variantes tienen especificación en común, entonces fue necesario aplicar un patrón de generalización entre las variantes, de tal forma que se pueda seguir representando la parte común a ser reutilizada. Por ejemplo para el punto de variación resuelto "Requisitos" fue necesario denotar que "Identificar y Esbozar Requisitos" es una generalización de "Encontrar actores y Casos de Uso" y de "Analizar el Problema". De la misma forma se aplicó la generalización a las variantes de los demás puntos de variación identificados.

FIGURA 7. Flujo de trabajo de Requisitos de la línea de procesos conforme a la nomenclatura de SPL.



#### 4.2.2.3 Definir restricciones.

La generación de un nuevo proceso es la combinación de las diferentes opcionalidades y variantes del proceso donde el resultado puede ser incoherente. Las restricciones tienen como objetivo permitir que los procesos generados sean planificados y por lo tanto bien elaborados. Las restricciones en este trabajo fueron diseñadas teniendo en cuenta los productos de trabajo de salida de una actividad que son entradas obligatorias a otras actividades del mismo proceso y también la inclusión de actividades que son necesarias para la generación de los procesos miembros de la familia. Para el caso particular de la disciplina de requisitos se identificaron las restricciones de la figura 8. El tipo de restricciones utilizadas son del tipo: *i)* Para la característica A que requiere la característica B, se denota de la siguiente forma  $\neg A \vee B$ , lo que nos permite definir las restricciones entre que actividades se deben desarrollar debido a que una actividad produzca un artefacto que es entrada obligatoria a otra actividad. *ii)* Para la característica A que excluye a la característica B, se denota de la forma  $\neg A \vee \neg B$ , que permite definir la restricción de selección exclusiva de una de las variantes que hacen parte de un punto de variación.

La definición de las restricciones también sirve para disminuir la complejidad de los diagramas generados con la técnica de Washizaki, por ejemplo la transición que existe entre las actividades Identificar y Esbozar Requisitos y Detallar un caso de uso, figura 6, se puede representar mediante la restricción Identificar y Esbozar Requisitos Detallar un caso de uso, donde se establece que la segunda actividad requiere que se ejecute la primera, entonces de esta forma se pueden borrar del diagrama transiciones donde el diagrama resultado es menos complejo. El resultado de aplicar Washizaki en éste estudio se presenta en la figura 6, y el modelo final del flujo de requisitos con la aplicación de las actividades "Encapsular Actividades y Resolver puntos de variación" y "Definir Restricciones" se muestra en la figura 7.

FIGURA 8. Restricciones del flujo de trabajo de la disciplina de requisitos.

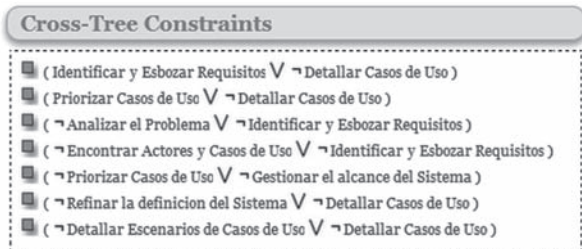
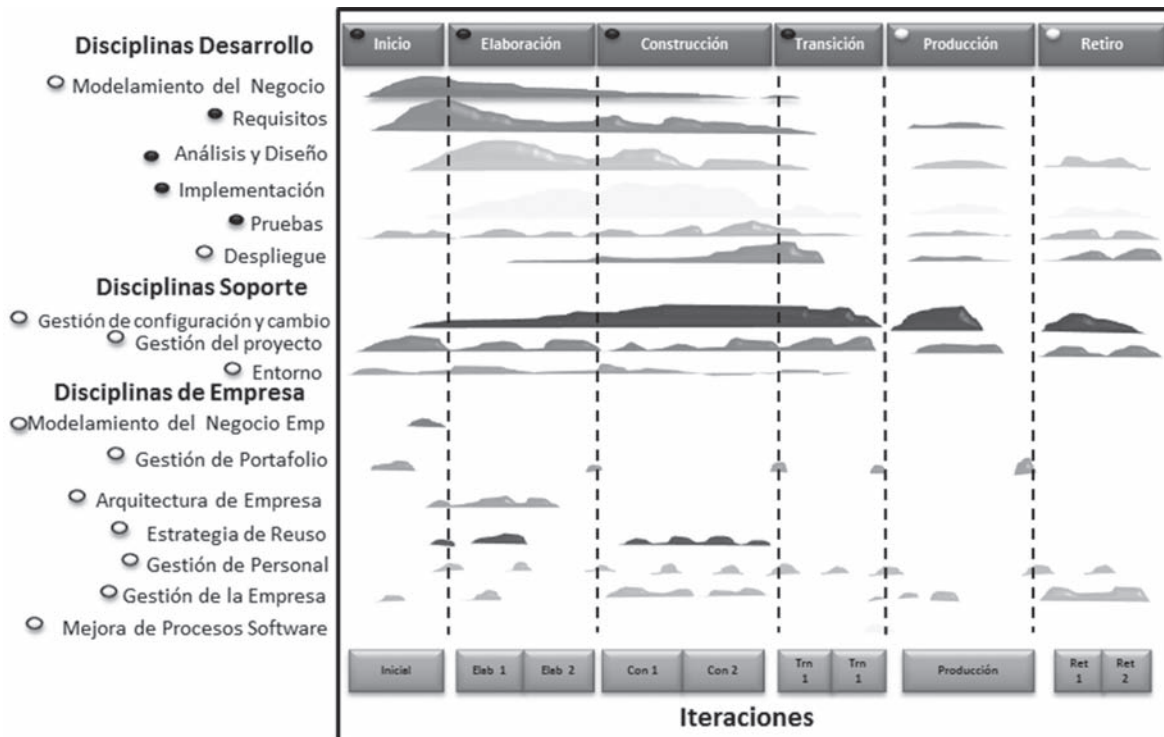


FIGURA 9. Variabilidad Estructural de la Familia de Procesos UP.



Fuente: [29]



#### 4.3 DEFINIENDO EL NÚCLEO ESTRUCTURAL DEL PROCESO Y SU VARIABILIDAD.

En este trabajo se hizo también la abstracción estructural del proceso que corresponde a las fases y disciplinas del proceso visualizándolas como actividades atómicas. El núcleo estructural del proceso se logró consolidar haciendo comparaciones entre las diferentes fases y disciplinas de los procesos. En este caso para identificar el núcleo estructural se tuvo en cuenta como núcleo inicial a UP de acuerdo a [15]. El resultado de la comparación entre las diferentes fases de los procesos arrojó como resultado que el proceso núcleo de la familia cuenta con cuatro fases comunes Inicio, Elaboración, Construcción y Transición. Las fases Producción y Retiro de EUP son fases adicionales que serán parte de la variabilidad estructural de la familia como fases opcionales. En la comparación de las disciplinas se analizaron los cinco procesos considerando la iteración básica de UP como el core inicial. Las disciplinas identificadas como comunes luego de la comparación y que serán parte de la familia de procesos son: Requisitos, Análisis y Diseño, Implementación y Pruebas. El resultado de la variabilidad estructural de la familia se presenta en la figura 9. Muestra que en el eje horizontal del proceso, sus fases, existen dos que son opcionales denotadas por las circunferencia sin relleno en el extremo izquierdo del rectángulo, Producción y Retiro, y cuatro que son comunes, inicio, elaboración, construcción y transición, denotadas por la circunferencia rellenas de color negro. Al igual en el eje vertical, las disciplinas del proceso, se tienen nueve disciplinas opcionales y cuatro comunes.

#### 5. MODELOS DE CARACTERÍSTICAS.

Los diagramas de características (Feature Diagrams) de la variabilidad estructural y de las diferentes disciplinas del proceso fueron implementados con las herramientas del proyecto SPLOT [32].

La variabilidad de la disciplina de requisitos se muestra en la figura 10. Los diagrama de características nos permiten, calcular las posibles configuraciones de un proceso y además plasmar e identificar restricciones para la generación de procesos.

#### 6. RESULTADOS Y DISCUSIÓN.

El principal resultado es la identificación de variabilidad del Proceso Unificado entre diferentes extensiones y adaptaciones del mismo. Para la determinación de la variabilidad del proceso se utilizaron diferentes descripciones del UP disponibles. El análisis y revisión de la variabilidad se realizó de forma manual lo que implicó un esfuerzo considerable (110 horas-Persona).

La determinación de la variabilidad entre los procesos que hacen parte de la línea de procesos deja como resultado un modelo de variabilidad del Proceso Unificado. La variabilidad total de la Línea de procesos se resume en las Tabla 2 y Tabla 3. No obstante los resultados mostrados revelan que existe una buena cantidad de posibilidades de generación de procesos, combinatoria de toda la variabilidad de la familia, y que pueden ser no útiles, por lo tanto fue necesario establecer las restricciones descritas previamente con el fin de generar modelos de proceso coherentes.

FIGURA 10. Diagrama de Rasgos de la disciplina de requisitos.

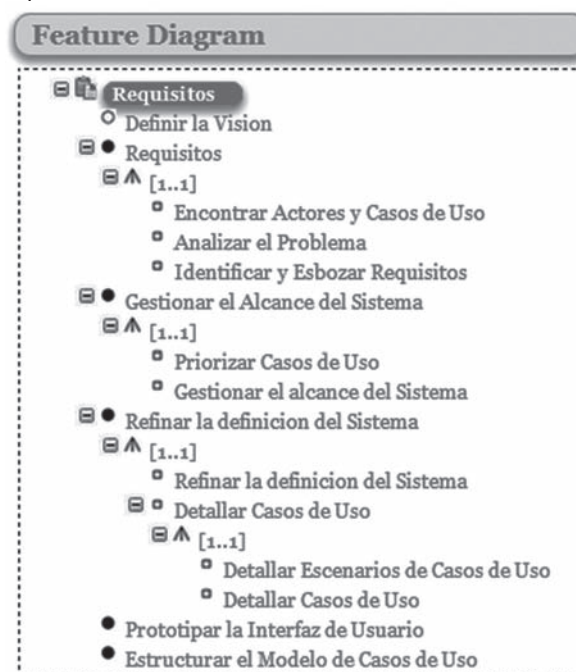


TABLA 2. Variabilidad Estructural.

Variabilidad Estructural	Puntos de Variación	Opcionales	Alternativas
Fases	2	2	0
Disciplinas	10	10	8
Total	12	12	8

TABLA 3. Variabilidad de las Disciplinas.

Variabilidad en flujos de trabajo	Puntos de Variación	Opcionales	Alternativas
Requisitos	4	1	8
Análisis	1	1	3
Diseño	1	1	2
Implementación	1	0	3
Pruebas	4	1	9
Total	11	4	25



La tabla 4 muestra el número de las configuraciones de procesos posibles, antes definir restricciones, el número de restricciones y las configuraciones válidas para cada disciplina después de aplicar las restricciones.

El resultado obtenido en este trabajo es el pilar para ir refinando la familia, la cual también debe definir las situaciones en que el proceso será aplicado y las reglas que ajustarán el proceso a dichas situaciones.

**TABLA 4.** Configuraciones de las Disciplinas.

Flujos de trabajo	Posibles Configuraciones	Restricciones	Configuraciones validas
Requisitos	36	7	16
Análisis	6	0	6
Diseño	6	2	3
Implementación	3	0	3
Pruebas	51	9	28

En algunas ocasiones fue necesario particularizar y tener en cuenta nuevos elementos para que la técnica Washizaki [15] se ajustara a las necesidades. Es el caso de las actividades Encapsular Actividades y Resolver puntos de variación y Definir Restricciones, que no hacen parte de la técnica, pero son necesarias con el fin de seguir la nomenclatura definida para los puntos de variación en líneas de procesos y proveer procesos coherentes en la generación de procesos derivados de la línea de procesos.

## 7. CONCLUSIONES Y TRABAJO FUTURO.

En este artículo se sigue la estrategia descrita por Washizaki [15] para determinar empíricamente la variabilidad del Proceso Unificado con respecto a adaptaciones y extensiones de este proceso. El principal resultado es un modelo de variabilidad útil para facilitar la adaptación del UP que es un proceso utilizado mundialmente por la industria. La identificación de la variabilidad del proceso es una actividad compleja que requiere tiempo y experiencia, donde la determinación adecuada de esta variabilidad, permite a las líneas de procesos dotarlas de flexibilidad, adaptabilidad y reutilización. Aunque el resultado obtenido hasta el momento es un avance significativo y base inicial de la variabilidad de la familia, aún debe ser refinada mediante evidencia empírica y reportes de investigación. Debido al nivel de granularidad expresado, por cada uno de los procesos, la comparación arrojó diferencias sustanciales debido a que los mismos procesos pueden contener implícitamente en su definición familias de procesos, por lo que se puede concluir que el alcance de la línea fue muy amplio.

Como trabajo futuro se está realizando la versión final del núcleo de la familia usando una estrategia de reducción del alcance tanto para la variabilidad estructural, así como para la variabilidad interna en las disciplinas de requisitos y pruebas, las cuales exhibieron demasiada variabilidad. Para ello se utilizará UP como proceso base por presentar una estructura completa y definida y sobre él diferenciar los otros modelos e introducir moderadamente la variabilidad, es decir un subconjunto manejable de las variabilidades identificadas en éste trabajo. Además, como parte del modelado de la familia de procesos se espera especificar los requisitos de la familia mediante modelos de contexto, de acuerdo a CASPER [21] con los cuales se definirán las situaciones donde el proceso será aplicado, y que serán insumos para la definición y ejecución de reglas de adaptación. Se espera mediante evidencia empírica y reportes de investigación, lograr la especificación de los requisitos, mediante modelos de contexto que nos permitan establecer la relación entre los requisitos y la variabilidad de la familia.

## 8. REFERENCIAS.

- [1] Kalus, Georg., & Kuhrmann, Marco. Criteria for software process tailoring: a systematic review. International Conference on Software and System Process. ICSSP. 2013.
- [2] Martínez, Tomás., García, Félix., Piattini, Mario., & De Lucas, Francisco. Process variability management in global software development: a case study. International Conference on Software and System Process. 2013.
- [3] Ruiz, Pablo., & Hurtado, Julio. A software process line based on the Unified Process. Computing Congress (CCC), 2012 7th Colombian. 2012.
- [4] Hanssen, Geir, Bjornson, Finn., & Westerheim, Hans. Tailoring and Introduction of the Rational Unified Process. P. Abrahamsson et al. (Eds.): EuroSPI 2007, LNCS 4764, pp. 7–18, Springer-Verlag Berlin Heidelberg. 2007.
- [5] Hanssen, Geir, Westerheim, Hans., & Bjornson, Finn. Tailoring RUP to a defined project type: A case study. In: Bomarius, F., Komi-Sirviö, S. (eds.) PROFES 2005. LNCS, vol. 3547, Springer, Heidelberg. 2005.
- [6] Hurtado, Julio., Pino, Francisco., & Vidal, Juan. Sistema Integral para el Mejoramiento de los Procesos de desarrollo de Software en Colombia SIMEP-SW. 2006.

- [7] Camacho, Marta., & Hurtado, Julio. Analyzing the Viability for Adopting the Software Process Line Approach in Small Entities. Computing Congress (CCC), 2012 7th Colombian. 2012.
- [8] Jacobson, Ivar., Booch, Grady., & Rumbaugh, Jim. Unified Software Development Process. Addison-Wesley. 1999.
- [9] Hartmann, Julio., Fontoura, Ra., & Price, Roberto. Using Risk Analysis and Patterns to Tailor Software Processes. Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS).2005.
- [10] Pereira, E., Bastos, R., & Oliveira, T. A Systematic Approach to Process Tailoring. International Conference on Systems Engineering and Modeling. ICSEM. 2007.
- [11] Ruiz, Pablo., Maya, Carlos., Pantoja, Libardo., & Hurtado, Julio. Guía de Adaptación del Proceso Unificado a Proyectos Específicos para las Pymes Desarrolladoras de Software. V Congreso Colombiano de Computación. 2010.
- [12] Coelho, C. Maps: Modelo de Adaptación de proceso software. Master' Thesis. Federal University of Pernambuco – UFPE. 2003.
- [13] Simidchieva, Borislava., Clarke, Lori., & Osterweil, Leon. Representing Process Variation with a Process Family. In qing Wang, Dietmar Pfahl, and David M. Raffo, editors, ICSP '2007, volumen 4470 of LNCS, pages 109-120. 2007.
- [14] Proyecto Adaptación de Modelos de Procesos de Software "ADAPTE". 2011. Recuperado ( 2013 Abril 19) de <http://www.adapte.cl>.
- [15] Washizaki, Hironori. Building Software Process Line Architectures from Bottom Up. In J. Münch & M. Vierimaa, eds., Product-Focused Software Process Improvement, LNCS, 415-421, Springer. 2006.
- [16] Rombach, Dieter. Integrated Software Process and Product Lines, Post-Proceedings of the Software Process Workshop 2005, LNCS Vol.3840, 2005.
- [17] Matsumoto, Yoshihiro. Japanese Software Factory, in Encyclopedia of Software Engineering, (ed.) J.J. Marciniak, John Wiley & Sons, 1994.
- [18] Schnieders, Arnd., and Weske, Matias. Activity Diagram Based Process Family Architectures for Enterprise Application Families in Enterprise Interoperability. G. Doumeingts, et al., Editors. Springer London. p. 67-76. 2007.
- [19] Schnieders, Arnd. Variability Mechanism Centric Process Family Architectures. ECBS '06 Proceedings of the 13th Annual IEEE International Symposium and Workshop on Engineering of Computer Based Systems. Pages 289 – 298. 2006.
- [20] Hurtado, Julio., Quispe, Alcides., Bastarrica, Cecilia., & Ochoa, Sergio. A MDE Production Strategy for Software Process Lines. International Conference on Software and Systems Processes. 2011.
- [21] Hurtado, Julio., & Bastarrica, Cecilia. Building Software Process Lines with CASPER. International Conference on Software and Systems Processes. ICSSP. 2012.
- [22] Araujo, Fellipe., Freire, Marilia., Dos Santos, Wanderson., & Kulesza, Uira. An Approach to Manage and Customize Variability in Software Processes. Brazilian Symposium on Software Engineering. 2010.
- [23] Pesantes, M., Lemus, C., Mitre, H.A., & Mejia, J. Identifying Criteria For Designing a Process Architecture in a Multimodel Environment, International Conference on Software and System Process (ICSSP). 2012.
- [24] Barreto, Ahilton., Duarte, Elaine., Rocha, Ana., M., Lemus, C., Mitre, H.A., & Murta, Leonardo. Supporting the Definition of Software Processes at Consulting Organizations via Software Process Lines, in Proceedings of the 2010 Seventh International Conference on the Quality of Information and Communications Technology. 2010, IEEE Computer Society. p. 15-24.25.
- [25] Ocampo, Alexis., Bella, Fabio., M., Lemus, C., Mitre, H.A., & Münch, Jürgen. Software process commonality analysis. Softw. Process: Improve. Pract., 10: 273–285. doi: 10.1002/spip.229. 2005.

- [26] Agile UP (AUP). Recuperado (2013, Abril 24) de <http://www.ambyssoft.com/unifiedprocess/agileUP>.
- [27] R. Balduino, Basic Unified Process: A Process for Small and Agile
- [28] Open Unified Process (OpenUP): Recuperado (2012, Junio 16) de <http://epf.eclipse.org/wikis/openup/>
- [29] Enterprise Rational Unified Proceses (EUP). Recuperado (2012, Junio 12) de <http://www.enterpriseunifiedprocess.com/>
- [30] Patel, Chaitali., Cesare, Sergio., Iacovelli, Nicola., & Merico, Antonio. A Framework for Method Tailoring: A Case Study. 2nd OOPSLA Workshop on Method Engineering for Object-Oriented and Component-Based Development. 2004.
- [31] Rational Unified Process Best Practices for Software Development Teams. Rational Software White Paper. TP026B, Rev 11/01. Recuperado (2012, Junio 23) de [http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251\\_bestpractices\\_TP026B.pdf](http://www.ibm.com/developerworks/rational/library/content/03July/1000/1251/1251_bestpractices_TP026B.pdf)
- [32] Software Products Lines Online Tools - SPLOT. Recuperado (2013, Abril 24) de <http://www.splot-research.org/>