

UNIVERSIDAD CENTRAL “MARTA ABREU” DE LAS VILLAS
FACULTAD DE MATEMÁTICA FÍSICA Y COMPUTACIÓN
DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN



Inspección de propiedades estructurales en la arquitectura de un modelador de procesos orientado al valor

TESIS EN OPCIÓN AL GRADO DE MÁSTER EN CIENCIAS DE LA COMPUTACIÓN

Autor: ING. JORGE CARRERA ORTEGA

Tutores: DRA. ANA MARÍA GARCÍA PÉREZ
MSC. LISETT PÉREZ QUINTERO

Consultante: DR. CARLOS PÉREZ RISQUET

Santa Clara, 2016

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1: MARCO TEÓRICO REFERENCIAL DE LA INVESTIGACIÓN	6
1.1 Enfoque a procesos en las organizaciones	6
1.2 Modelado de procesos	8
1.2.1 Propiedades estructurales	11
1.3 Dificultades para la inspección de propiedades durante el modelado.....	13
1.4 Características de las propiedades en el modelado orientado al valor	15
1.5 Evaluación de trabajos en relación a la inspección de propiedades y el factor comprensibilidad en los modelos de proceso.....	16
1.5.1 Modelo base para la inspección de propiedades estructurales.....	16
1.5.2 Herramientas de modelado organizacional y los mecanismos para la inspección de propiedades estructurales.....	18
1.5.3 Estudios que abordan el aspecto: comprensibilidad en los modelos de proceso..	19
1.5.3.1 Análisis dimensional del factor: comprensibilidad	21
1.6 Conclusiones del capítulo	23
CAPÍTULO 2: ARQUITECTURA DE CAPYROX 1.2, HERRAMIENTA PARA LA CAPTURA DE PROCESOS Y REPRESENTACIÓN ORGANIZACIONAL	25
2.1 Breve introducción a CAPYROX 1.2	25
2.2 Entorno de desarrollo.....	25
2.3 Estructura del modelo	27
2.3.1 Control gráfico sobre las relaciones entre procesos.....	27
2.3.2 Manejo de referencias del proceso o los segmentos con los diccionarios.	29
2.3.3 Estructuras de datos utilizadas	31
2.3.3.1 Diccionarios	31
2.3.3.2 Árboles.....	34
2.4 Componentes concebidos.....	35
2.4.1 Vista estática y funcional.....	35
2.4.1.1 Explorador de proyecto	35
2.4.1.2 Área de modelado.....	36
2.4.1.3 Inspector de propiedades	37
2.4.2 Vista dinámica.	40
2.5 Objetos gráficos	41
2.6 Conclusiones del capítulo	42
CAPÍTULO 3: IMPLEMENTACIÓN DEL MECANISMO DE INSPECCIÓN EN LA HERRAMIENTA CAPYROX 1.2	43
3.1 Mecanismo de inspección de propiedades en CAPYROX 1.2.....	43
3.1 Implementación de grupos de propiedades por objetos gráficos del modelo de proceso.	47
3.1.1 Implementación de grupo: general	47
3.1.1.1 Objeto gráfico: proceso/actividad.	47
3.1.1.2 Objeto gráfico: segmento.....	49
3.1.2 Implementación de grupo: costo. Objeto gráfico: proceso/actividad	50
3.1.3 Implementación de grupo: dato. Objeto gráfico: proceso/actividad	51
3.1.4 Implementación de grupo: formato.....	51

3.2 Pruebas de calidad.....	52
3.3 Validación de la herramienta mediante la aplicación práctica en el tejido empresarial.	53
3.3.1 Guía de capacitación y validación según atributos FURPS	54
3.3.2 Casos de aplicación y validación basada en la NC ISO/IEC 9126	56
3.4 Conclusiones del capítulo	60
CONCLUSIONES.....	61
RECOMENDACIONES	62
BIBLIOGRAFÍA.....	63
ANEXOS.....	69
ANEXO 1	69
ANEXO 2	69
ANEXO 3	70
ANEXO 4	71
ANEXO 5	73

ÍNDICE DE FIGURAS

Figura 1: Niveles de procesos de negocio representados en (2):	7
Figura 2: Propiedades propuestas en (12) utilizadas para la definición del mecanismo de inspección de la presente investigación.	15
Figura 3: Modelando con una notación basada en IDEF0	17
Figura 4: “Comprensibilidad” junto a otros atributos de calidad	20
Figura 5: Factores relacionados al atributo de calidad “comprensibilidad” para los modelos de procesos.	21
Figura 6: Competencias relacionadas a la calidad de los modelos (Competencias).....	23
Figura 7: Entorno de desarrollo.....	25
Figura 8: Relación de perspectivas asumidas para el modelo.....	27
Figura 9: Control gráfico de las relaciones (tipo Branch, Boundary, y entre procesos)	28
Figura 10: Inclusiones relacionadas a los diccionarios	29
Figura 11: Ejemplo de relación visual entre un segmento, la propiedad “recurso” y el diccionario de recursos.	29
Figura 12: Diagrama de clases correspondiente a la concepción del modelo	30
Figura 13: Diccionario de recursos.....	31
Figura 14: Diccionario de datos.....	32
Figura 15: Diccionario de indicadores y variables de control	32
Figura 16: Diccionario de competencias.....	33
Figura 17: Explorador de proyectos	35
Figura 18: Concepto Modelo/Vista	36
Figura 19: Área de modelado	36
Figura 20: Patrón MVC	37
Figura 21: Inspector de propiedades.....	37
Figura 22: Estructura de la macro para la definición de una propiedad en Qt (51)	38
Figura 23: Creación de propiedades personalizadas	39
Figura 24: Proceso ordinario de construcción en C++ (51)	39

Figura 25: Proceso de construcción en Qt C++ (51).....	39
Figura 26: Patrón Observador	40
Figura 27: Inspección múltiple, agrupando las propiedades comunes de diferentes objetos	45
Figura 28: Diagrama de clases perteneciente al núcleo del mecanismo de inspección de las propiedades	46
Figura 29: Propiedades personalizadas en grupo general	47
Figura 30: Propiedades del grupo: general. Objeto gráfico: proceso/actividad	48
Figura 31: Propiedades generales del objeto gráfico segmento (caso “recurso”)	49
Figura 32: Propiedades generales del objeto gráfico segmento (caso “entidad de dato”).....	50
Figura 33: Propiedades del grupo: costo.....	50
Figura 34: Propiedades del grupo: dato	51
Figura 35: Propiedades del grupo: formato	51
Figura 36: Etapas de las pruebas de calidad realizadas	52
Figura 37: Componentes del modelo de capacitación	54
Figura 38: Valores de la media por variables	55
Figura 39: Modelo de trabajo con CAPYROX para la definición del proceso GC (38)	57
Figura 40: Diagrama del subproceso “Organización de cursos”, proceso de GC (38)	58
Figura 41: Proceso “Prestación del servicio de alojamiento” (etapa inicial del modelado)	58
Figura 42: Representación del grado de utilización de los contenidos del inspector de propiedades durante el trabajo de modelado del proceso de GC con CAPYROX (38).....	59

ÍNDICE DE TABLAS

Tabla 1: Análisis de herramientas en relación a la implementación del mecanismo de inspección de la Figura 2	19
Tabla 2: Resumen de elementos del entorno de desarrollo.....	26
Tabla 3: Objetos gráficos personalizados	41
Tabla 4: Estadística descriptiva	55
Tabla 5: Validación considerando los aspectos definidos en la NC ISO/IEC 9126	60
Tabla 6: Estudios seleccionados para el FCA (factor comprensibilidad).	69

RESUMEN

La gestión de procesos de negocios integra un amplio conjunto de prácticas valiosas para la identificación y seguimiento de los procesos, dentro de ello el modelado de procesos es de trascendental apoyo. En los enfoques de modelado de procesos orientados al valor, es necesario considerar las adecuadas variantes de representación de información en los diagramas, las dimensiones abarcan consideraciones del costo, del tiempo, de la calidad, del conocimiento, aspectos financieros relacionados a la evaluación de inversiones, entre otros. A veces en los modelos de proceso, buscándose mayor nivel de detalle para un enfoque de sistema y orientación al valor, se afecta la comprensibilidad durante el análisis de mejora en la determinación de estrategias de informatización empresarial. Considerando la propuesta (1), el objetivo de la presente investigación es definir un mecanismo para la inspección de propiedades estructurales en una arquitectura de un modelador de procesos orientado al valor que toma por base la notación IDEF0 para facilitar la comprensibilidad de los modelos. Se pudo elaborar y validar en la práctica el mecanismo mediante su implementación en un modelador de procesos orientado al valor evidenciándose su capacidad para facilitar la comprensibilidad de los modelos según un grupo de dimensiones estudiadas del factor comprensibilidad. En la validación se incluyeron elementos propuestos en los atributos FURPS y en la NC ISO/IEC 9126, los cuales contienen el aspecto: comprensibilidad.

INTRODUCCIÓN

El tejido empresarial a nivel internacional tiene la necesidad de evaluar continuamente su calidad, de acuerdo a procedimientos de mejora continua, en un entorno competitivo cada vez más complejo y acelerado, buscando elevar su productividad con eficiencia y eficacia. Los análisis de mejora propician la necesidad de revisar los procesos y analizar alternativas de mejoras, lo cual incluye la posible evaluación de propuestas de inversión en soluciones informáticas. La gestión de procesos de negocios (BPM por sus siglas del inglés Business Process Management) integra un amplio conjunto de prácticas valiosas para la identificación y seguimiento de los procesos, dentro de ello el modelado de procesos de negocio es un aspecto de trascendental apoyo en la evaluación del funcionamiento actual de una organización y su posible rediseño. Dentro del marco de la ingeniería del software es también asumido el modelado de procesos como una técnica muy prudente dentro del análisis de posibles soluciones informáticas a desarrollar, para poder entender el negocio y llegar a los requisitos del software como acuerdo entre el desarrollador y el experto del negocio.

Entre las ventajas que brinda el modelado de proceso (2) se encuentran la facilitación de resúmenes gráficos de la estructura de los procesos de la empresa, propiciando el entendimiento de su cadena lógica e identificando el intercambio de información requerida; además la comprensibilidad de las dimensiones operativas de la empresa, representando el alcance de su relación con la dimensión técnica. Se ha considerado (3) que en muchas empresas, a nivel mundial, existe una brecha entre aspectos organizacionales del negocio y las tecnologías de la información, por lo que es importante disminuir esa brecha entre la organización y las tecnologías. Otra ventaja es la presentación de los esquemas asociados al funcionamiento actual para realizar sobre esa base los diseños de mejoras pertinentes, posibilitando la evaluación de la factibilidad de hacer determinados cambios en los procesos. También posibilitan estimular la comunicación para el entendimiento de todos los involucrados en los procesos de la empresa, así como el intercambio de ideas durante experiencias en rediseño de procesos.

Se ha planteado (4) que el objetivo de cualquier actividad de modelado es lograr una completa y exacta comprensibilidad del dominio dentro del marco de un problema a resolver y que la calidad de estas representaciones es un factor crítico para lograr comprensibilidad. Se identifica la comprensibilidad como el grado en que la información contenida en un modelo de procesos puede ser fácilmente comprendida por los lectores del modelo (5). Recientemente se han realizado revisiones sistemáticas de la literatura que abordan el tema de la calidad en el modelado de procesos de negocio, una de ellas realiza un análisis formal de conceptos donde constata la concordancia en relación a determinadas directrices de calidad resaltando ampliamente la importancia trascendental de la comprensibilidad (6).

Se han realizado estudios que declaran un grupo de factores que influyen significativamente en la comprensibilidad de los modelos de procesos de negocio. Uno de estos casos es (5), el cual realiza un

análisis crucial al presentar la necesidad de que la comprensibilidad de los modelos de procesos no sea auxiliada exclusivamente para la comunicación entre desarrolladores de soluciones informáticas (que sí son abordados en la literatura principalmente en marcos de trabajo reconocidos, como es el caso de RUP), sino por todas las personas requeridas en el análisis del problema y los diseños de solución en el marco organizacional. Ese estudio declara el escaso conocimiento sobre la relación entre la comprensibilidad de los modelos de procesos y el factor humano. Entre los elementos principales destaca que el tamaño de los modelos impacta en la comprensibilidad, apuntando hacia evidencias de que largos modelos de procesos tienden a tener más fallos formales como la ilegibilidad del estado de terminación y deadlocks (en idioma inglés, traducido al español como puntos muertos) que los modelos de procesos más cortos.

La modularidad en los modelos de proceso también está relacionada a aspectos claves de comprensibilidad abordados en los estudios (7) y (8). Se propone, bajo este principio de diseño la verificación en un sistema complejo sobre una posible composición por subsistemas, (o módulos) que puedan según su funcionamiento ser manejados de manera independiente aunque juntos funcionen como un todo; en los modelos de proceso de negocio se conocen como subprocesos. Existen prerequisites de la modularidad que han sido identificados por (8) siendo provenientes de (9) que refieren que la modularidad puede ser introducida en un modelo con más de 5-7 actividades. La cuestión es, que de acuerdo a un gran número de notaciones de modelado, los diagramas de proceso al tener muchas veces más de 5 actividades se tornan muy difíciles de comprender debido principalmente a las interrelaciones binarias entre ellos que suelen cruzarse y producir dificultades en la identificación de sus orígenes y destinos.

En muchas ocasiones se asocia la comprensibilidad con complejidad, frecuentemente se afirma que es más difícil entender algo muy complejo. A veces los modelos de proceso buscando mayor nivel de detalle, deseando ser más descriptivos, afectan la comprensibilidad. La teoría de la complejidad ha sido ampliamente estudiada en muchas disciplinas, en el contexto de la gestión de procesos de negocio varios aspectos relacionados a la complejidad han sido estudiados (10). En los modelos de proceso de negocio es esperado que esta crezca a medida que aumentan las actividades y sus interrelaciones en la representación (10).

El enfoque de modelado de procesos orientado al valor, expuesto por el autor Jan vom Brocke (11) hace énfasis en la relevancia de considerar la dimensión financiera involucrada en el rediseño de procesos, además de las consideraciones que enfatizan en el comportamiento de las operaciones de negocio. En los enfoques de modelado de procesos orientados al valor, como el presentado en (11), para lograr un análisis que profundice en aspectos financieros relevantes durante el rediseño de procesos, es necesario considerar las adecuadas variantes de representación de información del beneficio y del costo en los diagramas. Las consideraciones del valor son clave dentro de la toma de decisiones respecto a la mejora de procesos (2). Se plantea (12) que el propósito de los procesos de negocio es crear valor y el de la

gestión de procesos de negocio es apoyar esa creación de valor. Existen variables importantes para demostrar la creación del valor en la práctica. Las dimensiones del valor abarcan las consideraciones del costo, del tiempo, de la calidad, del conocimiento, aspectos financieros relacionados a la evaluación de inversiones, entre otros.

En (1) se presenta un procedimiento para llevar a cabo varios análisis desde la ejecución de la fase de planificación de sistemas. Este enfoque abarca la investigación para determinar la factibilidad del cambio del sistema de información existente, con el propósito esencial de entender el problema. Esta propuesta enfatiza un grupo de pasos previos a la etapa de obtención de los requisitos del software. Las acciones del procedimiento para la etapa de planificación incluyen la descripción y modelado de la situación actual utilizando un diagrama de contexto con una notación basada en IDEF0 (del inglés Integration Definition for Function Modeling) para proseguir con una identificación y representación del problema evaluando las propiedades de los procesos. Un factor crítico es la manera de enfrentar las dificultades de comprensibilidad anteriormente mencionadas, que conllevan a una alta utilización de tiempo en la rectificación de los modelos, así como la concreción de la información resumen para los procesos y para todas sus actividades en relación a las dimensiones operativa, técnica y económica.

Se ha planteado (3) que las propiedades estructurales de los procesos pueden ser solamente mostradas utilizando abstracciones de los procesos de negocios del mundo real, dando lugar a observaciones muy útiles en la detección de deficiencias estructurales en esos procesos de negocios. Los mecanismos para la inspección de propiedades estructurales responden a la intencionalidad de las herramientas actuales. Se plantea en (2) que es evidente la pertinencia de un mecanismo para modelar las expectativas del usuario pero se ha demostrado que estos mecanismos deben facilitar la comunicación entre los implicados en la construcción de esas soluciones informáticas y hoy en día se ha encontrado un camino muy positivo para ello mediante la utilización de los entornos gráficos. La afirmación “facilitar la comunicación” tiene variadas dimensiones, que se distinguen de acuerdo al objetivo que tenga la comunicación en determinado contexto. Por ejemplo, siguiendo el enfoque de las actuales herramientas para el modelado de procesos, muchas de las más reconocidas se basan en la notación BPMN (del inglés Business Process Modeling Notation). Sobre esa notación se afirma que sus indicadores gráficos para identificar mensajes, eventos, entre otros ayudan a la comunicación de una manera trascendental. Si se distinguen los casos, muchos hacen alusión a la comunicación funcional entre los desarrolladores de soluciones informáticas, provistos incluso, desde un enfoque funcional, de un mecanismo para la optimización de tiempo y esfuerzo mediante la generación de código a través de los modelos, siempre y cuando estén correctamente referidas las provisiones gráficas de cada propuesta.

Las herramientas de modelado actuales como tendencia responden a la aplicación de la Ingeniería Dirigida por Modelos (conocida como MDE del inglés Model-driven Engineering) (13) donde el desarrollo de software potencia que el software sea el resultado de una cadena de transformaciones semi-automatizadas del modelo, desde donde se genera el código finalmente. En este caso, se aborda una

propuesta (1) que presenta un grupo de propiedades no solo relevantes desde el punto de vista de la generación del código del programador, sino para el análisis de problemas organizacionales previos a la decisión de alternativas de informatización por parte del analista y los especialistas del negocio. Se ha considerado lo referido por (14), que lo que debe primar en estas etapas de planificación es un enfoque hacia la identificación correcta del problema en el sistema actual y no hacia el diseño de la solución (software). Es posteriormente, mediante la fase de análisis, que se concentrarán esfuerzos en la identificación de los requisitos de software, tanto funcionales como no funcionales.

Problema científico: Existen dificultades en la comprensibilidad de los modelos de procesos del negocio para que los mismos logren un enfoque de sistema y orientación al valor durante el análisis de mejora en la determinación de estrategias de informatización empresarial.

Hipótesis: La utilización de un mecanismo para la inspección de propiedades estructurales dentro de la arquitectura en un modelador de procesos orientado al valor que toma por base la notación IDEF0 favorece la comprensibilidad como atributo de calidad en el modelo resultante.

Objetivo general: Definir un mecanismo para la inspección de propiedades estructurales en una arquitectura de un modelador de procesos orientado al valor que toma por base la notación IDEF0 para facilitar la comprensibilidad de los modelos.

Objetivos específicos:

- Elaborar el marco teórico sobre los mecanismos para la inspección de propiedades estructurales durante el modelado de procesos de negocio.
- Diagnosticar el estado actual de la introducción de estos mecanismos en diferentes herramientas (software) que apoyen el modelado de procesos orientado al valor.
- Elaborar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura de manera que logre un modo de interacción adecuado para un modelador de procesos orientado al valor.
- Validar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura mediante su implementación en un modelador de procesos orientado al valor y su capacidad para facilitar la comprensibilidad de los modelos.

En esta tesis se aportan como elementos novedosos varios aspectos, entre los cuales vale destacar:

1. El análisis de las dimensiones de la comprensibilidad en los modelos de proceso, como atributo de calidad de los mismos.
2. La combinación adecuada de patrones de diseño que tributan a lograr una arquitectura robusta en toda la aplicación.

El valor práctico de la misma consiste en haber logrado una herramienta que es pertinente para la organización en procesos de las empresas con enfoque de sistema y que facilita llevar a cabo análisis de factibilidad importantes en la consideración de estrategias de informatización en las organizaciones.

La tesis se estructura en tres capítulos. En el primer capítulo se abordan elementos teóricos relacionados a las propiedades estructurales en el modelado de procesos organizacionales. En el segundo se describen aspectos de la arquitectura de CAPYROX 1.2, y en el tercero se expone la definición del mecanismo de inspección de propiedades, se realiza una presentación de la implementación de los grupos del inspector de acuerdo a los objetos gráficos del modelo y se refieren los resultados de las pruebas de calidad con objetivo de validar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura.

CAPÍTULO 1: MARCO TEÓRICO REFERENCIAL DE LA INVESTIGACIÓN

En el presente capítulo se abordan elementos teóricos relacionados a las propiedades estructurales en el modelado de procesos organizacionales realizando un análisis crítico sobre trabajos relacionados a los mecanismos de inspección de estas propiedades, así como de herramientas que apoyan la técnica de modelado de procesos en relación al enfoque orientado al valor que se toma por base.

1.1 Enfoque a procesos en las organizaciones

El cuerpo del conocimiento del análisis de negocio **(15)**, reconocido como BABOK (por sus siglas del inglés Business Analysis Body of Knowledge), se refiere a los procesos relacionando la idea de cómo las personas en una organización colaboran para cumplir un objetivo, en esa fuente se plantea que existe un número importante de características que los procesos usualmente tienen como son la colaboración entre múltiples personas o grupos; que tienen lugar sobre un periodo de tiempo; pueden ser cumplidos en más de una vía y que son repetibles.

La gestión de procesos de negocio contribuye a la mejora organizacional y es un concepto que se encuentra en muy estrecha relación con la calidad y su gestión. El logro de mejoras en los procesos de las organizaciones tiene un reflejo importante en la calidad de los productos y servicios que estas realizan. Se reconoce que “la calidad de los productos y servicios incluye no sólo su función y desempeño previstos, sino también su valor percibido y el beneficio para el cliente” **(16)**.

La norma NC ISO 9001:2015 **(17)** especifica el “enfoque a procesos” como uno de los principios de gestión de la calidad. Esa norma promueve la adopción de un enfoque a procesos y afirma que “la comprensión y gestión de los procesos interrelacionados como un sistema contribuye a la eficacia y eficiencia de la organización en el logro de sus resultados previstos”.

Sobre el enfoque a procesos se ha declarado **(16)** que “se alcanzan resultados coherentes y previsibles de manera más eficaz y eficiente cuando las actividades se entienden y gestionan como procesos interrelacionados que funcionan como un sistema coherente”. Se ha planteado además que los procesos de negocios son el instrumento clave para organizar las actividades y mejorar el entendimiento de sus interrelaciones **(3)**.

Reconocida literatura del campo de la ingeniería industrial **(18)** define proceso de negocio como un conjunto lógico, relacionado y secuencial (conectado) de actividades que tienen una entrada de un proveedor, le adicionan valor y producen una salida para un cliente; esta consideración está acorde a lo planteado en NC ISO 9000:2015.

La NC ISO 9000:2015 proporciona los conceptos fundamentales, los principios y el vocabulario para los sistemas de gestión de la calidad (SGC) **(16)**, en el epígrafe *términos relativos al proceso* define *proceso*

como un “conjunto de actividades mutuamente relacionadas que utilizan las entradas para proporcionar un resultado previsto”. En **(3)** se define *proceso de negocio* como un conjunto de actividades que están desempeñadas en coordinación en un ambiente técnico y operacional. Esas actividades conjuntamente realizan un objetivo del negocio. Cada proceso de negocio es llevado a cabo por una sola organización, pero puede interactuar con procesos de negocio desempeñados por otras organizaciones. La definición de *proceso* expuesta en esa norma es la tomada como base conceptual para la presente investigación.

En la NC ISO 9001:2015 **(17)** se describen las partes de un proceso proporcionando una representación esquemática de cualquier proceso y muestra la interacción de sus elementos. En esa representación se encuentran: actividades, entradas, salidas, fuentes de entradas y receptores de las salidas. Las entradas y salidas se identifican como materia, energía, información; para el caso de las entradas lo ejemplifican en la forma de materiales, recursos y requisitos; en el caso de las salidas en la forma de producto, servicio y decisión.

Las interacciones entre procesos y los procesos en sí se documentan para apoyar su definición, su análisis e incluso su control. Existen principios de la gestión de la calidad donde se busca la mejora continua y lograr tener correctamente identificados los procesos con todos los niveles de detalles pertinentes resulta imprescindible para pensar en qué se tiene y cómo se puede mejorar esa concepción.

Según **(3)** existen los niveles de procesos que se representan en la Figura 1, desde la estrategia de negocio hasta la implementación de procesos de negocio.

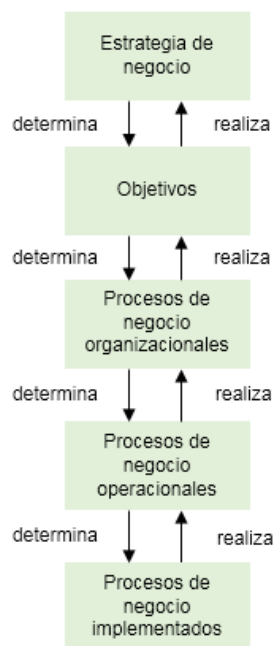


Figura 1: Niveles de procesos de negocio representados en (3): desde la estrategia de negocio hasta la implementación de procesos.

1.2 Modelado de procesos

El modelado de procesos es una técnica que busca apoyar el enfoque a procesos dentro de las organizaciones, propicia mayor comprensibilidad sobre los planteamientos abordados, motivación al intercambio de ideas entre los participantes y un medio de caracterización resultante de los niveles de abstracción que lleva implícito el diseño de procesos.

Los modelos de procesos **(15)** son utilizados para obtener una representación gráfica de los procesos actuales o futuros de una organización; un modelo puede ser utilizado, en su más alto nivel, para obtener un entendimiento general de un proceso o a un nivel más bajo como una base para la simulación de manera que el proceso pueda ser realizado tan eficientemente como sea posible.

La mejora de procesos está centrada en los modelos de proceso de negocio **(19)**. El modelado de procesos se ha convertido en una gran fuerza de ayuda para los enfoques sistémicos y estratégicos de la gestión organizacional. Con este enfoque se logra identificar los contenidos operativos propios de la organización, la definición de cuestiones distintivas que permiten el éxito en la realización de los productos y servicios pero además una proyección hacia los objetivos estratégicos concebidos.

La construcción de modelos de procesos es considerada por algunos estudios como el *proceso de modelado de proceso* **(20)**. En **(15)** se especifica que modelar procesos puede tomar desde unas pocas horas hasta muchos meses, o incluso años, para todos los procesos de una organización.

El modelado de procesos se utiliza para varias perspectivas, como por ejemplo los análisis dirigidos a la optimización de la cadena de valor mediante la introducción de alternativas de informatización. Se considera que la gestión de procesos del negocio tiende a ganar espacio en las organizaciones, siendo posible automatizar y optimizar procesos y servicios para el negocio **(21)**; esa fuente explicita que el modelado de procesos facilita más debates firmes entre los especialistas de Tecnologías de la Información (TI) y las áreas de negocio soportando la representación de los procesos que serán implementados. Se ha referido que el modelado de procesos de negocio ha emergido como una importante práctica para guiar decisiones en el análisis y diseño de sistemas; es además considerada la competencia número uno demandada de los graduados de las Tecnologías de la Información **(22)**. En la definición de estrategias de informatización la comunicación adecuada y la participación conjunta entre los especialistas del negocio y los especialistas TI resulta esencial para lograr producir los cambios operacionales que tributen al desempeño organizacional de manera eficiente y eficaz. La utilización de modelos ayuda a la representación visual de los problemas y la definición de las alternativas de solución para poder evaluarlas y tomar decisiones que consideren tanto las perspectivas operacionales como las estratégicas, sistémicas.

El modelado conceptual crea representaciones, abstracciones; el resultado es una representación que contiene la información crítica que es necesitada para diseñar y aplicar efectivas estrategias

organizacionales, gestión de proceso y del conocimiento y un necesario fundamento para construir los sistemas de información y bases de datos de la organización **(4)**. En esa fuente se plantea que la disciplina del modelado conceptual ha ido ganando más pertinencia con la emergencia de la Arquitectura Dirigida por Modelos (MDA por sus siglas del inglés Model Driven Architecture) que promueve el uso de modelos en todos los pasos de un proyecto de desarrollo de software, desde la planificación inicial. Un enfoque muy reconocido es el de la Ingeniería Dirigida por Modelos (MDE por sus siglas del inglés Model-driven Engineering), que promueve la utilización de modelos como artefacto principal en todas las actividades de ingeniería de software, estudios como **(13)** insisten en la revisión del estado correcto de los modelos alertando que un error en el modelo puede ser propagado hasta el software resultante.

Sin embargo, como se plantea en **(23)**, se ha evidenciado que los procesos de negocio se han convertido en los objetos centrales en muchos esfuerzos de modelado conceptual, ejemplo, para soportar su documentación, mejora y desempeño automatizado. En **(24)** se reafirma la idea de que similar a otras formas de modelado conceptual, los modelos de procesos son primeramente requeridos para ser intuitivos y fáciles de comprender, especialmente en fases de proyectos de sistemas de información que están referidas a la documentación y comunicación de requisitos. Desde ese punto de vista es posible comprender cómo las herramientas de modelado de procesos se han especializado en incluir funcionalidades de apoyo para los análisis en función de las TI, proponiendo facilidades gráficas centradas en ese propósito, esto en ocasiones afecta la comprensibilidad al querer utilizar determinadas notaciones y herramientas más factibles para expertos en TI que para los especialistas del negocio. La cuestión radica también en el hecho de que en etapas de modelado de proceso, para lograr clarificar los requisitos del cliente, es importante identificar correctamente todos los factores del problema a resolver; porque inadecuadas representaciones propician requisitos inexactos que a su vez traerán consigo la necesidad futura de re-programación de código para subsanar dificultades en el entregable. La propagación de errores a etapas más tardías implica el exponencial crecimiento en costo y esfuerzo para rehacer las cuestiones detectadas **(25)**.

Un *modelo de procesos de negocios* consiste en un conjunto de modelos de actividades y restricciones de ejecución entre ellos. Una *instancia de modelo de proceso de negocio* representa un caso concreto en el negocio operacional de una empresa, consistiendo en instancias de actividades **(3)**.

El propósito primario de los modelos de procesos es facilitar la comunicación humana y la solución de problemas **(26)**. Muchos beneficios están asociados con el modelado de procesos de negocio, algunos expertos han identificado la mejora de procesos, la comunicación y el entendimiento compartido entre los beneficios más importantes **(22)**. Los modelos de procesos son utilizados para diferentes propósitos como comunicar un mensaje, compartir conocimiento o visión, como un punto de inicio para el rediseño u optimización de procesos, o como una instrucción precisa para ejecutar tareas de negocio **(27)**.

En **(15)** el modelado de procesos es abordado como una técnica en el análisis de requisitos; y le detallan como propósito el uso para desarrollar y documentar un entendimiento de los flujos de trabajo actuales incluidos en el proceso; identificar oportunidades para la mejora de procesos y recomendar alternativas de solución; comprender y comunicar lógicas de negocio complejas; crear documentación para los manuales de usuario y la formación; crear documentación para propósitos regulatorios; automatizar procesos.

Para la construcción de modelos de negocios se realizan diferentes técnicas de captura de información entre las que se destacan las entrevistas, encuestas, la observación de la realidad, o sea del desempeño de aquellas cuestiones del negocio que se desean representar. En **(22)** se expresa que los modelos de procesos pueden ser obtenidos a través de entrevistas a relevantes especialistas del negocio, o derivadas de documentos organizacionales como las políticas del negocio. Lo fundamental es lograr conectar de manera participativa todas las observaciones pertinentes que brindan los involucrados en las actividades que se modelan. Es importante, sin embargo, mantener siempre el enfoque de procesos y no las consideraciones funcionales aisladas que puedan derivarse de una entrevista particionada en departamentos o responsabilidades específicas.

Estas acciones de modelado pueden tener objetivos claves en los proyectos de desarrollo de sistemas de información, como la de lograr un entendimiento común de cómo el negocio funciona en la actualidad (modelado AS-IS) y cómo se supone que trabaje en el futuro (modelado TO-BE) **(28)**. En este sentido suelen mencionarse las representaciones AS-IS y TO-BE. Las representaciones AS-IS tratan de plasmar la información exacta del funcionamiento actual, principalmente para poder identificar los problemas a resolver y sus dimensiones. Por su parte los diseños TO-BE buscan identificar las soluciones posibles en el contexto del modelo de procesos, teniendo en cuenta las alternativas y variaciones que traen consigo por ejemplo cambios en la tecnología, o identificar medidas relativas a la capacitación de los involucrados.

Para el modelado se utilizan notaciones y herramientas que permiten apoyar a los modeladores para un resultado más legible, rápido y de ser posible estandarizado para propiciar su comprensibilidad. En **(3)** se menciona que existen varias notaciones gráficas para el modelado de procesos de negocio, que en esencia son bastante similares. Esa afirmación permite entender cómo todas esas notaciones en esencia buscan acercarse conceptualmente a lo que un proceso es y cómo representarlo de manera sencilla; pero de acuerdo al objetivo con el que ha surgido cada notación puede tener determinadas variaciones. En **(2)** se expresa que “siguiendo el enfoque de las actuales herramientas para el modelado de procesos, muchas de las más reconocidas se basan en la notación BPMN (del inglés Business Process Modeling Notation). Sobre esa notación se afirma que sus indicadores gráficos para identificar mensajes, eventos, entre otros ayudan a la comunicación de una manera trascendental. Si se distinguen los casos, muchos hacen alusión a la comunicación funcional entre los desarrolladores de soluciones informáticas, provistos incluso, desde un enfoque funcional, de un mecanismo para la optimización de tiempo y esfuerzo

mediante la generación de código a través de los modelos, siempre y cuando estén correctamente referidas las provisiones gráficas de cada propuesta”.

Existen otros lenguajes de modelado ampliamente difundidos para el modelado de procesos, como EPCs (del inglés Event –driven Process Chains), un lenguaje que se utiliza en la herramienta Aris. Otros enfoques como los basados en las Redes de Petri, los diagramas de actividad de UML, YAWL (del inglés Yet Another Workflow Language); BPEL (del inglés Business Process Execution Language for Web Services) e IDEF0 (29). Estos lenguajes presentan fortalezas en su composición que permiten describir los modelos, de acuerdo a diferentes niveles de detalles. La selección de los mismos depende del entorno a utilizar, los propósitos del modelado y las fases dentro de la conceptualización del enfoque a procesos en que se encuentre la organización; también en ocasiones depende de la experiencia que se tenga en uno u otro ya que un lenguaje conocido permite mayor comunicación durante su aplicación. En consecuencia se utilizan herramientas de modelado como Visual Paradigm (30), BizAgi (31), Aris (32), Intalio Designer (33), TIBCO (34), ConceptDraw PRO (35), las opciones de Microsoft Visio (36) y ERwin Process Modeler (37), llegando muchas de ellas a generar código para automatizar funciones.

1.2.1 Propiedades estructurales

El estudio de las propiedades relacionadas a los modelos de procesos de negocio se ha convertido en un aspecto de trascendental importancia para la mejora de procesos. La definición de propiedades permite contextualizar desde el modelo de procesos aquellos parámetros fundamentales que luego podrán ser observados en cada instancia del modelo. Se ha planteado (21) que en el análisis de procesos de negocio y en la fase de modelado es importante tratar no solo propiedades funcionales sino de la calidad del proceso y restricciones operacionales.

En (38) se aborda un enfoque en el que trabajan con propiedades estructurales para procesos de negocio. Esa fuente plantea que la mejora de las actividades de negocio dentro de la organización, basada en nuevos conceptos de las estructuras de los procesos es un reto continuo para las empresas. Según su criterio, expone que los enfoques para la mejora de procesos pueden ser divididos en dos categorías: mejora de las propiedades operacionales de los procesos de negocio (orientada sobre parámetros dinámicos), y mejora de las propiedades estructurales de los procesos de negocio (propiedades estáticas). En la presente investigación se abordan propuestas para estas propiedades estructurales, teniendo en cuenta conceptualizaciones en el diseño de una manera estática para propiciar un positivo impacto en resultados operacionales y económicos durante el desempeño de los procesos modelados.

Sobre las propiedades estructurales en (39) se realiza una observación considerando su relación con los atributos internos del modelo de proceso. Esa fuente enfatiza que los modelos de procesos tienen atributos internos y atributos externos. Los atributos internos son aquellos que pueden ser medidos

puramente en términos del modelo de procesos, separado de su comportamiento; los atributos externos son aquellos que pueden ser medidos solamente con respecto a cómo el modelo de procesos se relaciona con su entorno. Esa fuente presenta un esquema donde explicita la relación previsible que se puede tener desde las métricas de los atributos internos hacia los valores de medida de los atributos externos (como costo, duración, número de errores, entre otros).

En **(3)** se presenta un estudio de las propiedades de los procesos, haciendo un centro de atención en las propiedades estructurales de los modelos de procesos, las cuales enfatiza que no son de aplicaciones específicas ni de dominios específicos. Las propiedades estructurales de los procesos pueden ser solamente observadas utilizando abstracciones de los procesos de negocios del mundo real.

El impacto de propiedades estructurales de los elementos gráficos en relación a la comprensibilidad del modelo ha sido identificado en estudios recientes como **(40)**. También en reciente literatura científica se ha referido que la medición de propiedades estructurales puede ser utilizada para indicar que un modelo es apropiado para ser bien comprendido o si es potencialmente propenso a errores **(25)**.

La **dependencia de datos** entre actividades en los modelos de procesos de negocio es uno de los puntos esenciales que aborda **(3)** en su epígrafe sobre las propiedades estructurales de los procesos. Esa fuente, basada en el hecho de que los datos pueden ser creados, modificados y eliminados durante la ejecución de procesos de negocio, realiza un enfoque hacia dos aspectos claves: la información que las instancias de procesos manipulan y además la dependencia de información entre las actividades del proceso. La dependencia de datos entre las actividades del proceso es reflejada por un flujo de datos. El flujo de datos se mueve desde el parámetro de salida de una actividad y el parámetro de entrada de otra actividad.

Los enfoques semánticos y sintácticos son abordados en estudios sobre las propiedades estructurales, pues esta visión permite identificar patrones para evitar errores en los modelos, que puedan conllevar a modelos fallidos para el desempeño de los procesos. En **(41)** se exponen elementos relativos a las propiedades estructurales desde un enfoque semántico y sintáctico para evitar errores en los modelos de procesos, específicamente para la notación BPMN. Esta fuente, al igual que **(3)** refiere elementos como los llamados “puntos muertos” en los modelos, conocidos en inglés como deadlock y los livelock entre otros problemas en la calidad del modelo. Un deadlock en el modelo de procesos está dado si una cierta instancia del modelo no puede continuar trabajando y no puede alcanzar su final, o sea es una condición utilizada para describir un proceso que no puede ser completado **(41)**. Un livelock puede ser definido como un estado desde el cual es posible proceder (continúa trabajando) pero es imposible que alcance el estado final deseado **(41)**.

Desde el punto de vista organizacional, estos aspectos están determinados en los procesos por un grupo de factores ya identificados como requisitos en estándares como la NC ISO 9001:2015, pensando en el adecuado desempeño organizacional real del cual el modelo es una abstracción, para lograr su diseño y

posibilitar los análisis de mejora. En **(1)** se presenta un enfoque para adoptar estas propiedades como propiedades estructurales desde el pensamiento de la mejora organizacional y los principios de calidad enunciados por estándares como la ISO 9000:2015, ISO 9001:2015 desde la orientación a procesos. En el epígrafe 1.3 se aborda esta consideración, teniendo en cuenta también lo propuesto por el enfoque de modelado de procesos orientado al valor.

1.3 Dificultades para la inspección de propiedades durante el modelado

El ciclo de vida del proceso incluye varias actividades, las cuales pueden incluir por ejemplo las referidas por **(21)**, como modelado de proceso, instanciación del modelo de proceso, administración y realización del proceso, monitoreo del proceso, evaluación del proceso y optimización. Durante la ejecución de estas fases existen un conjunto de mecanismos que las organizaciones implementan de una manera u otra para la inspección de las propiedades que intervienen en cada etapa.

Se ha planteado **(4)** que el objetivo de cualquier actividad de modelado es lograr una completa y exacta comprensibilidad del dominio dentro del marco de un problema a resolver y que la calidad de estas representaciones es un factor crítico para lograr comprensibilidad. A pesar de que se reconoce la importancia de lograr representaciones exactas en los modelos, la noción de calidad en los modelos conceptuales es aún inmadura y es un concepto de evolución rápida.

A inicios de esta década una investigación **(23)** argumentaba que el modelado de procesos es muy aplicado en la práctica, pero que importantes aspectos de calidad no han sido tratados a fondo por las investigaciones. Entre los problemas referidos por esa fuente se encuentra el bajo nivel de competencia de modelado que muchos modeladores eventuales tienen en proyectos de documentación de procesos. Esto refiere para la inspección de propiedades dos problemas fundamentales. Uno de estos problemas es que no se conocen las propiedades específicas más importantes a seguir. El segundo es que las capacidades gráficas de las herramientas de modelado existentes presentan un mecanismo de observación de propiedades básicas para los objetos del diagrama, sin enfatizar en los aspectos de calidad organizacional para que puedan orientarse los modeladores en las propiedades importantes a considerar. Se expone en **(23)** que entender el proceso es crucial en cualquier técnica de análisis de proceso y que por eso el modelo de procesos en sí tiene que ser intuitivo y fácil de entender; esa referencia además afirma que a pesar del soporte que es provisto por las herramientas de modelado los usuarios apenas tienen algo de soporte en la creación de modelos de procesos que los profesionales del negocio puedan fácilmente analizar y entender.

El modelado de procesos tiene una carga cognoscitiva que la relaciona con el factor: comprensibilidad **(42)**. A esto se suma un fenómeno cognoscitivo para los lenguajes de modelado de procesos, caracterizado en **(43)** y es el hecho de lidiar con una sintaxis concreta y una abstracta. La sintaxis concreta trata con aspectos representacionales como los símbolos, colores, la posición de los varios

tipos de nodo. La sintaxis abstracta del lenguaje de modelado de procesos no está referida a los aspectos representacionales, sino captura los varios tipos de elementos de proceso y la relación estructural entre ellos. Por ejemplo la aplicación de modularidad para simplificar un modelo sin cambiar su comportamiento afecta la sintaxis abstracta **(43)**. El mecanismo de inspección de elementos que tributan a la sintaxis abstracta resulta escaso en las herramientas de modelado, existiendo una mayor presencia de variantes para los aspectos concretos de la sintaxis. Se ha planteado **(25)** que no hay disponibles validaciones empíricas para facilitar el proceso de toma de decisiones relacionado a la calidad de los modelos de proceso de negocio.

La abstracción es una generalización que reduce detalles no deseados para retener solamente la información esencial acerca de una entidad o fenómeno **(27)**. En **(27)** se expone que los criterios de abstracción son propiedades de los elementos del modelo de procesos que posibilitan su ordenamiento parcial. La inspección de propiedades en el modelado de procesos muchas veces se ve afectada por una inadecuada consistencia de los niveles de abstracción, esto quiere decir que no siempre se tienen en cuenta la necesidad de acceso rápido a aquellas cuestiones fundamentales que permiten decidir sobre los elementos del modelo que deben permanecer.

Otros de los problemas que afecta la inspección de propiedades en los modelos es la presencia de acciones de introducción de datos para los elementos gráficos del modelo con un nivel de facilidades para su modificación muy limitadas. Esto se refiere fundamentalmente a la separación de responsabilidades de actualización de información en las entidades gráficas sin brindar opciones de reflejar en algunos objetos gráficos los cambios que pueden realizarse en otros, logrando un nivel de modificación más cómoda para el usuario y menos propensa a errores. Por ejemplo, las facilidades de la presencia de diccionarios, la interconexión de propiedades como las que pueden crear las interrelaciones entre procesos entre las cuales se puede citar el caso de las entradas y salidas de los procesos o las actividades. Estudios como **(25)** abordan la importante relación entre las capacidades de modificación y la comprensibilidad como atributos de calidad de los modelos de procesos.

Existen también dificultades relacionadas a la comprensibilidad de los modelos para el seguimiento analítico de la dependencia entre actividades en los modelos de procesos de negocio, estas dependencias son uno de los puntos clave que aborda **(3)** en su epígrafe sobre las propiedades estructurales de los procesos.

En modelos complejos o extensos las interdependencias son muchas dentro de un mismo diagrama y se utiliza como posible solución la modularidad, mediante la incorporación de subprocesos y la descomposición jerárquica de los componentes. De esa manera se producen particiones del modelo para representar componentes de manera más sencilla donde el segmento de la interconexión queda disperso y las relaciones de dependencia se separan visualmente, complejizando su trazabilidad visual por el usuario.

1.4 Características de las propiedades en el modelado orientado al valor

En la gestión de procesos de negocio, lo cual incluye la etapa de diseño de los procesos, la consideración de creación de valor en los procesos tiene un papel fundamental, principalmente para valorar posibles mejoras en los procesos realizando los análisis de factibilidad pertinentes. Se reconoce que “la calidad de los productos y servicios incluye no sólo su función y desempeño previstos, sino también su valor percibido y el beneficio para el cliente” (16).

En esa área investigativa, diferentes estudios se han enfocado para considerar el valor en dimensiones como calidad, tiempo, costo, valores ecológicos, sociales, etc. En (12) se abordan un grupo de fundamentos sobre este interesante enfoque, esa fuente enfatiza que la perspectiva del valor que es más frecuentemente referida en la gestión de procesos de negocios es la operacional, que se enfocan en las medidas de rendimiento claves de tiempo, calidad y costo. En (12) también se puntualiza que bajo el paradigma de este enfoque orientado al valor otras perspectivas deben ser consideradas también y muestran unas perspectivas sobre un sistema multi-dimensional de valor donde aparece la perspectiva de valor financiera, la de calidad, tiempo y costo, la cultural y de conocimiento y entorno. Para la identificación de costos en los modelos de procesos, enfoques como por ejemplo el llamado ABC (del inglés activity-based costing) han tenido un seguimiento destacado dentro de los marcos de trabajo del rediseño de procesos (1).

Pero cada una de estas perspectivas es a su vez compleja, es decir están compuestas por un grupo de factores que permiten su evaluación. En (1) se presenta una propuesta que se enfoca sobre la base del modelado de procesos orientado al valor y presenta un conjunto de propiedades definidas para ello. Las propiedades que aborda se presentan en la Figura 2; en el presente trabajo se utiliza esa consideración buscando un mecanismo para facilitar el hecho de poder inspeccionar esas propiedades en los diagramas de proceso.

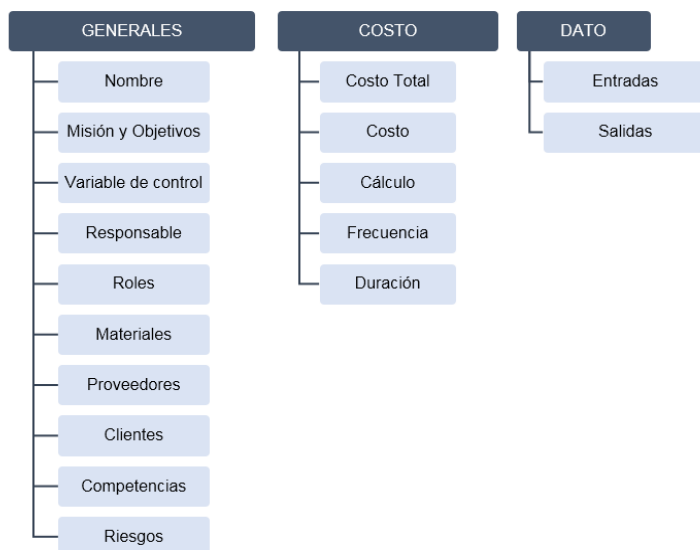


Figura 2: Propiedades propuestas en (1) utilizadas para la definición del mecanismo de inspección de la presente investigación.

Identificar las propiedades con sus interrelaciones y la representación de ellos mediante los modelos de procesos de negocio les permite a los miembros de la organización comunicarse de una manera eficiente y efectiva (3).

1.5 Evaluación de trabajos en relación a la inspección de propiedades y el factor comprensibilidad en los modelos de proceso

Existen notables investigaciones que abordan la calidad de los modelos de procesos, aunque se ha referido (24) que, en particular, estudios que contribuyan al acto de desarrollar modelos de procesos que faciliten la comunicación y la comprensión, son escasos. Sobre esta base, también existen herramientas que apoyan la inspección de propiedades de diferentes formas, lo cual en su medida permite ofrecer alternativas para la comprensibilidad de los modelos, en dependencia de los objetivos del mismo, y la notación de modelado utilizada.

1.5.1 Modelo base para la inspección de propiedades estructurales.

Existen numerosas notaciones para el modelado de procesos. BPMN es una notación de modelado de procesos que se ha convertido en un estándar usado ampliamente. Por medio de los diagramas de proceso de negocio (BPD por sus siglas del inglés Business Process Diagram) objetivos funcionales de la organización pueden ser representados gráficamente utilizando cuatro categorías de elementos: objetos de flujo, objetos de conexión, artefactos y swimlines (21). La fuente anteriormente citada también expone que BPMN es un lenguaje extensible, el cual permite incorporar nuevos objetos de flujo o artefactos dentro del BPD con objetivos específicos haciendo el diagrama más entendible o adaptado a propósitos específicos. Para la presente investigación es importante enfatizar en el planteamiento realizado sobre el carácter entendible de la notación y la adaptación a propósitos específicos.

Hace poco menos de una década, sobre los estados comparativos entre notaciones de modelado, una consideración científica (44) subrayaba las capacidades de IDEF0 para el análisis de procesos de negocio habiendo abordado un entendimiento del modelado de procesos de negocios utilizando el Framework de Zachman (es importante señalar, que en relación a las Arquitecturas Empresariales, recientemente se han realizado estudios (45) que ratifican al Framework de Zachman como el más comúnmente utilizado). En (44) abordaban además las capacidades de BPMN para los procesos de rediseño en el proceso.

BPMN, conocida así por sus siglas en inglés se reconoce como un lenguaje extensible, permitiendo la incorporación de nuevos objetos de flujo con determinados objetivos. De alguna manera esta característica de BPMN hubiese permitido la complementación de las propiedades relacionadas a la calidad que se deseaban incluir pero hubiese complejizado la acción de modelado por parte de los especialistas del negocio al tener que emplear demasiados elementos gráficos.

En **(1)** se presenta un enfoque analítico para la fase de planificación en el ciclo de vida de desarrollo de sistemas informáticos. Este enfoque analiza que “un aspecto erróneo es la transmisión de dificultades organizacionales a las estrategias de informatización. Por ejemplo, las inconsistencias en el desenlace de qué debe hacerse, cómo y cuándo para garantizar el desempeño efectivo de la organización mediante la intercalación del flujo de acciones individuales de los empleados ya que están muchas veces interconectados”. En esa investigación se tiene por base que “para poder llegar a un entendimiento sobre la situación actual no se hace primeramente un modelo As-Is ni To-Be, sino que se parte de una representación elemental de la empresa, para entender su misión. Algunas fuentes explican la diferencia entre este modelado previo y los diseños del modelo As-Is y To-Be, tal es el caso de **(9)**, que ilustra la necesidad de un modelado previo para identificar el “qué”, pues luego de identificarse un proceso determinado mediante un modelado más detallado será posible abordar la descripción del “quién y cómo” siendo estos los modelos As-Is. La conceptualización de una situación futura del sistema sería representada mediante los modelos To-Be”. Para ese modelado inicial se propone la utilización de una notación basada en IDEF0. Esta es una notación sencilla y fácil de utilizar que está básicamente compuesta por rectángulos y flechas. Las flechas mediante su posicionamiento en relación al rectángulo tienen determinado significado. Estas flechas pueden representar mecanismos, entradas, salidas y control, ver Figura 3. Pero para la aplicación de ese enfoque propuesto en **(1)** se requeriría de una herramienta que apoyara modelado, manteniendo principalmente las características de fácil comprensibilidad por la conservación de la sencillez gráfica pero lograr incorporar nuevas propiedades a los elementos del modelo abordados como los procesos y sus relaciones.

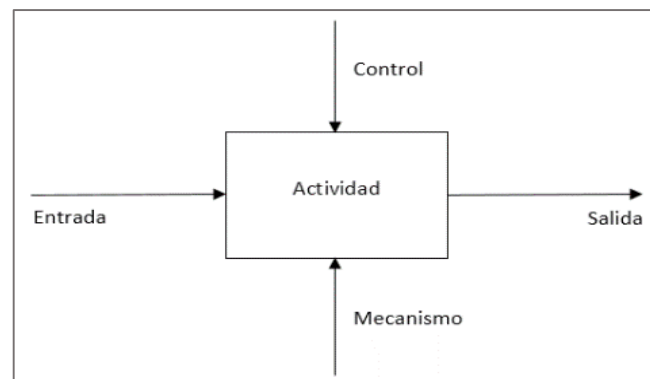


Figura 3: Modelando con una notación basada en IDEF0

La razón de la utilización de la notación IDEF0 es su sencillez gráfica. De acuerdo a la sintaxis de la misma se consideró que era posible lograr la complementación de dicha notación con un mecanismo de inspección de propiedades en CAPYROX sin complejizar los componentes del diagrama **(46)**.

Se ha considerado lo planteado en **(46)** “en el análisis de los enfoques de modelado disponibles en la actualidad se ha tenido en cuenta la coincidencia de propósito con **(47)**; esencialmente porque se orienta hacia el carácter estratégico que pueda lograrse en el modelo resultante. Se seleccionó IDEF0 sobre la

base de requisitos claves para la selección de la metodología de modelado, entre los que destacan: facilidad de aprendizaje y utilización, facilidad de modificación, estandarización, soporte a la modularidad, soporte a la adaptabilidad; y coincidir en características deseadas en los modelos de proceso resultantes como: comprensibilidad, que incluya las funciones elementales, no redundancia, completitud (que incluya todos los procesos importantes). Aunque este lenguaje se ha reconocido desde hace más de veinte años, recientemente se han realizado demostraciones de la positiva aplicabilidad del lenguaje de modelado IDEF0, para el modelado de estrategias y para su automatización (47). Esa fuente alega que los modelos realizados con IDEF0 resultan bien definidos, bien estructurados, fáciles de entender, fáciles de modificar y usar, que pueden ser extendidos a cualquier nivel de profundidad de detalle; y demuestra que el lenguaje IDEF0 tiene suficientes características y capacidades para apoyar el desarrollo de planes estratégicos. Otra de las investigaciones que abordan la compleja relación de los objetivos estratégicos con las métricas de rendimiento operacional opta por la notación IDEF0 (48); los modelos en esta notación consisten en una jerarquía de diagramas interrelacionados, las actividades del diagrama pueden ser descompuestas en diagramas más detallados hasta lograr el detalle deseado para los propósitos específicos. El análisis presentado en (49) propone una utilización de IDEF0 como herramienta de análisis conceptual, definiéndolo como una herramienta conceptual bien estructurada para describir y mejorar procesos, exponiendo entre sus conclusiones que los diagramas IDEF0 son herramientas de análisis muy útiles para describir y comparar procesos de software”. Otros estudios recientes como (38), que consideran la técnica de modelado de la estructura de proceso como la más importante herramienta para la formalización de una estructura organizacional, proponen el método IDEF0. Esa investigación además también trabaja con las propiedades estructurales de los modelos de procesos.

1.5.2 Herramientas de modelado organizacional y los mecanismos para la inspección de propiedades estructurales.

Existen herramientas de modelado que contienen mecanismos para el seguimiento de propiedades estructurales, por ejemplo el caso de los llamados deadlocks, implementando opciones de simulación para evaluar el diseño del flujo. Estos mecanismos de hechos se definen buscando la comprensibilidad del modelo mediante la reducción de errores que se centran en atributos de calidad definidos en la literatura. Por ejemplo, se ha planteado que los modelos de procesos que son difíciles de entender muchas veces contienen errores como deadlocks (25).

Se realizó un estudio con treinta y seis participantes para conocer las herramientas más utilizadas para el modelado de procesos, resultando como las más reconocidas las que se muestran en la Tabla 1. Se realizaron primeramente preguntas abiertas para mencionar cual era la herramienta que conocía (con conocimiento para su utilización). Así fue posible obtener el resultado de la columna “conocimiento”, mediante la suma total de criterios por cada herramienta referida. Luego se le incorporó en análisis para

ver como las propiedades mostradas en la Figura 2 eran abordadas en la herramienta, lo cual se expone en la columna “mecanismo de inspección”. Se estudió la adaptabilidad, derivado del factor: tipo de licencia del producto, si es de código abierto, y un balance del esfuerzo de reprogramación necesario en caso de adaptación (esto incluía la evaluación del aspecto visual, la posible extensión de IDEF0 y la incorporación del mecanismo de inspección). Como factor determinante se encuentra el hecho de conocer si la herramienta soporta IDEF0, que es el lenguaje definido en la propuesta **(1)** que es la investigación base para la presente, constituyendo un requisito fundamental para la opción resultante.

Nombre	Soporta IDEF0	Adaptabilidad (Según licencia, y/o código)	Mecanismo de inspección (Según parámetros de la Figura 2)	Conocimiento (Según encuesta)
Visual Paradigm	-	-	*	***
BizAgi	-	-	*	***
Aris	-	-	*	*
Intalio Designer	-	-	*	*
TIBCO	-	-	*	**
ConceptDraw PRO	Sí	-	*	*
Microsoft Visio	Sí	-	*	***
ERwin Process Modeler	Sí	-	*	*
Bonita	-	-	*	*
Idefine	Sí	-	*	-

Tabla 1: Análisis de herramientas en relación a la implementación del mecanismo de inspección de la Figura 2

En la Tabla 1 se utiliza un señalamiento que significa: (-) nulo, (*) bajo, (**) medio, (***) alto.

1.5.3 Estudios que abordan el aspecto: comprensibilidad en los modelos de proceso.

Teniendo en cuenta lo planteado en **(42)**, una persona comprende un modelo de proceso de negocio cuando es capaz de explicar el modelo, su estructura, su comportamiento, sus efectos sobre su contexto operacional, y sus relaciones con su dominio de aplicación en términos que son cualitativamente diferentes de las muestras utilizadas para construir el modelo de proceso de negocio en un lenguaje de modelado. Esta es la definición del término comprensibilidad optado en la presente investigación.

En la literatura aparecen estudios que abordan en factor comprensibilidad dentro de los modelos de procesos. Un trabajo reciente **(6)**, fundamentado en una revisión de la literatura, realizó un estudio de factores significativos dentro de la calidad del modelado de procesos donde se destacó en primer orden el factor: comprensibilidad. Esa investigación define y utiliza una tabla resumen (matriz) como entrada para proceder a realizar un Análisis Formal de Concepto (FCA por sus siglas en inglés). El Análisis Formal de Conceptos ha sido reconocido **(50)** como un método original para la representación y análisis de datos y constituye un método para la representación del conocimiento. Este método deduce un enrejado del concepto (conocido como concept lattice en inglés). En **(6)** se realiza un FCA considerando como los objetos del dominio de interés a 72 artículos de revistas de valioso prestigio científico y un grupo de factores de calidad como los atributos definidos.

En la presente investigación se utiliza como referencia la matriz creada en **(6)** y se procede a seleccionar los 43 objetos que tienen identificado el atributo *comprensibilidad*. Se ha obtenido así una matriz con solo esos elementos y se ha realizado el lattice de la Figura 4 con ayuda el método FCA.

La Figura 4 indica algunas relaciones de la dimensión *comprensibilidad* con otras dimensiones de calidad en los modelos de proceso. Es posible analizar con la herramienta ConExp algunas relaciones de implicación como: *Comprensibilidad* y *Correctitud* con *Estructuración*; *Comprensibilidad* y *Complejidad estructural* con *Mantenibilidad*; *Comprensibilidad* y *Conectividad* con *Errores*. La *modularidad*, que es uno de los aspectos también observados, se refiere al principio de diseño de tener un sistema complejo

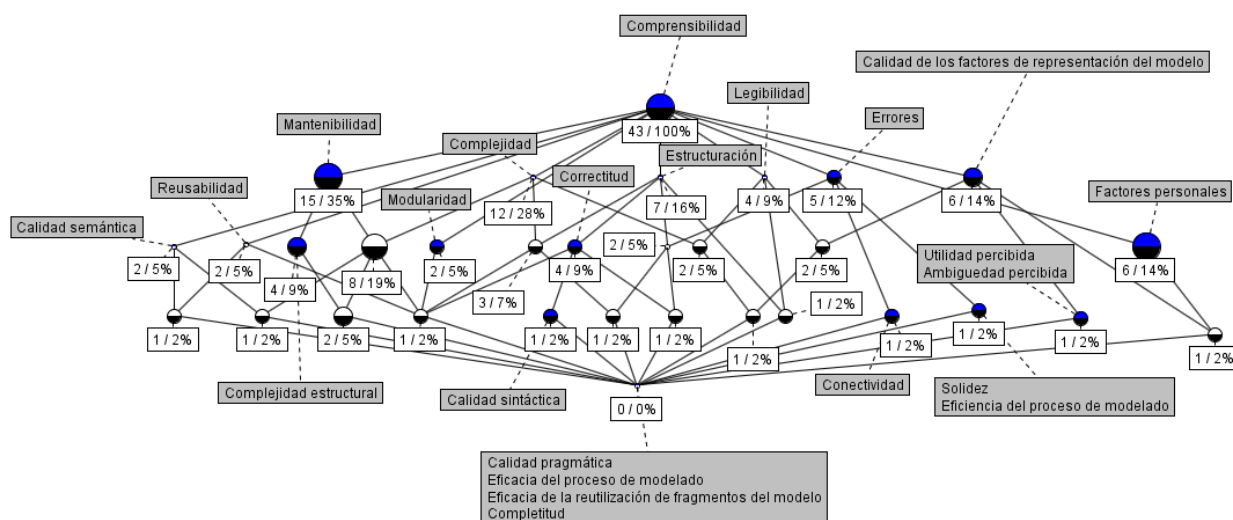


Figura 4: “Comprensibilidad” junto a otros atributos de calidad

compuesto de subsistemas más pequeños que pueden ser manejados independientemente y que funcionan unidos como un todo; usualmente se aplica en modelos de proceso de negocio utilizando subprocesos **(51)**.

El hecho de que la *comprensibilidad* se manifieste como uno de los atributos de calidad mayormente identificado constituye un punto de reflexión importante desde el punto de vista organizacional. La poca *comprensibilidad* de los modelos es un factor perjudicial para el entendimiento mutuo entre especialistas de TI y del Negocio. En campos de investigación amplios y apremiantes como la alineación estratégica entre TI y el Negocio la posibilidad de lograr enfoques comunes y participativos con todas las partes involucradas es fundamental **(52)**; esto ocurre con fuerza en el contexto actual donde es necesario lograr propuestas de mejora sobre la base de adecuados y eficientes encadenamientos productivos para la gestión de las organizaciones **(53)**.

La norma NC ISO 9001: 2015 **(17)** plantea la necesaria identificación de elementos en los procesos como las interdependencias, las responsabilidades, la secuencia, los mecanismos de control e indicadores de desempeño y los riesgos. Esta norma aborda como principios de gestión de la calidad el

enfoque a procesos, el compromiso de las personas y la mejora. Muchas organizaciones utilizan para ello representaciones gráficas de los procesos, para apoyar el entendimiento común de estos elementos, pero es clave lograr la comprensibilidad de estos modelos. Si la comprensibilidad de los modelos que se toman por base para la organización en procesos es incorrecta ocurrirá una interpretación errónea atentando contra la adecuada concepción de los sistemas de gestión de la calidad en las organizaciones, por tanto la comprensibilidad de los modelos logrados es un requisito clave desde el punto de vista organizacional.

1.5.3.1 Análisis dimensional del factor: comprensibilidad

Como se puede observar en el epígrafe anterior, la comprensibilidad es uno de los factores más significativos dentro de los análisis de calidad en el modelado de procesos de negocio, guardando una estrecha relación con otros factores entre los que se puede mencionar la legibilidad, complejidad, reusabilidad, modularidad y estructuración. Pero es importante además analizar cómo se concreta el análisis en base a la comprensibilidad estudiando cuáles son las dimensiones que se refieren con mayor énfasis dentro de su constructo.

Sobre la matriz expuesta en (6) se seleccionaron los elementos que referenciaban la comprensibilidad, obteniéndose 43 elementos. Luego se seleccionaron los que estuvieran en un periodo del 2010 en adelante. Se revisaron esos estudios para seleccionar aquellos que desglosaran del componente comprensibilidad algunas dimensiones, quedando 15 estudios seleccionados (Ver ANEXO 1). Se realizó una matriz para aplicar el FCA resultando el lattice que se muestra en la Figura 5.

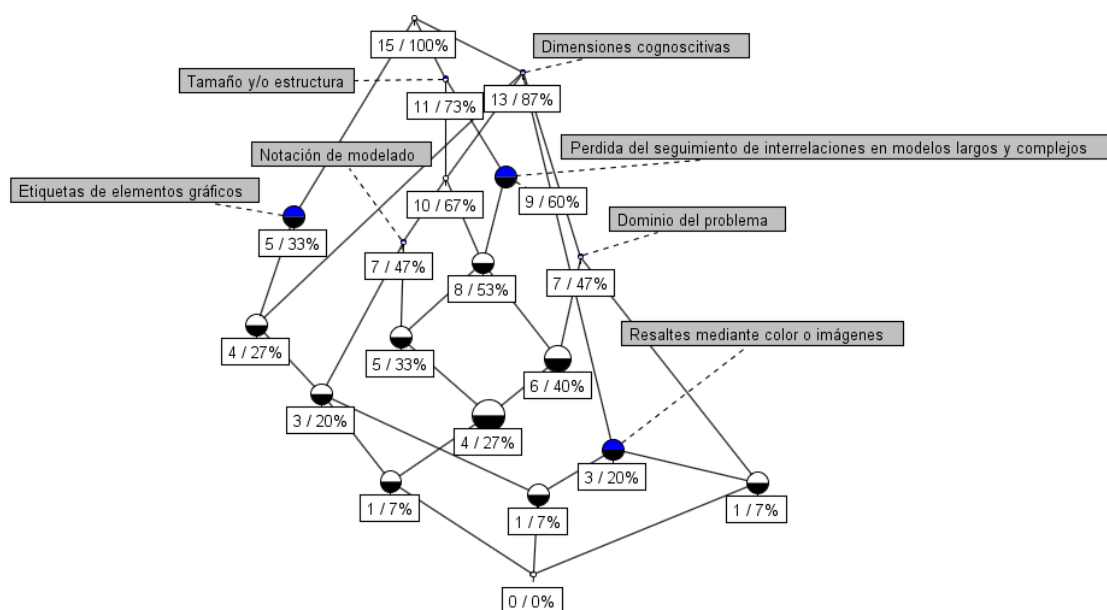


Figura 5: Factores relacionados al atributo de calidad “comprensibilidad” para los modelos de procesos.

Al observar ese estudio es posible notar cómo el tamaño y estructura del proceso se destacan como factores muy relacionados a la comprensibilidad. Cuando un modelo de procesos es grande, o sea presenta muchos diagramas o demasiados subprocesos y actividades relacionados se suele considerar como un modelo grande y complejo.

En muchas ocasiones se asocia la comprensibilidad con complejidad, frecuentemente se afirma que es más difícil entender algo muy complejo. A veces los modelos de proceso buscando mayor nivel de detalle, deseando ser más descriptivos, afectan la comprensibilidad. La teoría de la complejidad ha sido ampliamente estudiada en muchas disciplinas, en el contexto de la gestión de procesos de negocio varios aspectos relacionados a la complejidad han sido estudiados **(10)**. En los modelos de proceso de negocio es esperado que esta crezca a medida que aumentan las actividades y sus interrelaciones en la representación **(10)**.

Al referir la complejidad gráfica de los modelos y la capacidad de comprender esas representaciones complejas es oportuno referir las dimensiones cognoscitivas asociadas a este proceso. En la Figura 5 es posible ver que si bien el tamaño y la comprensibilidad son el segundo factor señalado, le antecede en un primer lugar las dimensiones cognoscitivas.

Las capacidad de manejar el conocimiento y de construir el conocimiento alrededor de la creación e interpretación de los modelos de procesos es principal y distintiva en la calidad de los mismos, a esto además se le unen otros aspectos que constituyen competencias transversales como la capacidad de abstracción, análisis y síntesis; la capacidad de aplicar los conocimientos en la práctica; capacidad de comunicación oral y escrita; habilidades en el uso de las herramientas de apoyo a los procesos de modelado; habilidades para buscar y analizar información; capacidad creativa; capacidad para identificar, plantear y resolver problemas; capacidad para trabajar en equipos; habilidad para trabajar en forma autónoma, compromiso ético y compromiso con la calidad.

Estas competencias definidas como genéricas en **(54)** fueron analizadas en relación con las dimensiones cognitivas para el modelado de procesos, estudiando el proceso de realización de modelos reales llevados a cabo por especialistas de Desoft y especialistas de diferentes empresas. Se analizaron 10 casos de empresas donde se realizó un modelado de procesos, obteniendo los resultados que se muestran en la Figura 6, con un valor máximo y alta concordancia de criterios para los 12 factores. Se utilizó una escala donde las posibles opciones eran 1 (poco importante), 2 (medianamente importante) y 3 (muy importante).



Figura 6: Competencias relacionadas a la calidad de los modelos (Competencias)

En relación a los factores cognoscitivos y su impacto se analizaron las competencias genéricas porque las competencias incluyen elementos que no son solo los conocimientos sino otros aspectos relacionados a estos. Lyle M. Spencer y Signe M. Spencer definen competencia como características subyacentes en el individuo que están causalmente relacionadas con un estándar de efectividad y/o a un rendimiento superior en un trabajo o situación **(55)**. Estas consideraciones abarcan conocimientos, habilidades, aspectos motivacionales y rasgos de la personalidad.

Cuando los modelos son extensos y con muchas interrelaciones es posible que existan dificultades para su entendimiento y seguimiento visual, producto del número de componentes presentes en el diagrama. Esto está muy relacionado a los factores “Tamaño y estructura” y “Dimensiones cognoscitivas” referidos anteriormente. Cuando esto sucede, como alternativa se intenta producir una aplicación de la modularidad para poder clarificar las representaciones utilizando subprocesos y agrupaciones de elementos. Algunas herramientas al aplicar esta variante realizan un cambio de escenario visual para producir esta clarificación y las interrelaciones quedan más dispersas lo cual produce dificultades en el seguimiento de las relaciones desde el elemento de salida al elemento de entrada.

1.6 Conclusiones del capítulo

De acuerdo a los elementos analizados en el presente capítulo fue posible elaborar el marco teórico sobre los mecanismos para la inspección de propiedades estructurales durante el modelado de procesos y diagnosticar el estado actual de la introducción de estos mecanismos en diferentes herramientas que apoyen el modelado de procesos orientado al valor, teniendo en cuenta la propuesta **(1)**. Se pudo arribar a las siguientes conclusiones:

1. La gestión de procesos de negocio contribuye a la mejora organizacional y se encuentra en muy estrecha relación con la calidad y su gestión.

2. Un modelo de procesos puede ser utilizado para obtener un entendimiento general de un proceso y su mejora, cubriendo gran número de detalles relacionados a los enfoques sistémicos, estratégicos y participativos dentro de la organización.
3. La profundización en las propiedades estructurales puede conllevar a facilitaciones de la comprensibilidad más allá de los enfoques sintácticos concretos para posicionarse en una perspectiva de la sintaxis abstracta y enfrentar dificultades cognoscitivas en el proceso de modelado.
4. Dentro de las herramientas de modelado analizadas, que se corresponden con las más utilizadas, no se identificó una que soportase IDEF0 con buena adaptabilidad para asumir las propiedades propuestas en **(1)** en relación al modelado de procesos orientado al valor (que incluye atributos de calidad como las competencias, los riesgos, las variables de control, duración, costo, entre otros).
5. La comprensibilidad es un atributo de calidad transcendental para los modelos de procesos y está relacionada a aspectos claves como: dimensiones cognoscitivas (entre las que se destacan 12 competencias transversales), tamaño y/o estructuras y la pérdida del seguimiento de interrelaciones en modelos largos y complejos.

CAPÍTULO 2: ARQUITECTURA DE CAPYROX 1.2, HERRAMIENTA PARA LA CAPTURA DE PROCESOS Y REPRESENTACIÓN ORGANIZACIONAL.

En este capítulo se describen aspectos de la arquitectura de CAPYROX 1.2. Se enuncian elementos importantes en la concepción del modelo en relación a la notación basada en IDEF0, especificando su estructura, la definición del control gráfico sobre las relaciones entre procesos, el manejo de referencias con los diccionarios y las estructuras de datos utilizadas.

2.1 Breve introducción a CAPYROX 1.2

CAPYROX¹ se desarrolla en Desoft Villa Clara como parte de su cartera de productos, “es una herramienta de utilidad para la captura de procesos en una organización, contribuyendo al análisis necesario para lograr una representación organizacional. La herramienta ayuda en el modelado y análisis respecto a la estructura de los procesos empresariales, logrando la descomposición de actividades involucradas” (56). Esta solución “posibilita especificar criterios sobre costo, tiempo, recursos y otros elementos importantes a tener en cuenta en el rendimiento de un proceso y que son de alta importancia evaluarlos cuando se somete a cambios operacionales” (56).

2.2 Entorno de desarrollo

El entorno de desarrollo utilizado se muestra en la Figura 7; es posible distinguir la utilización de las herramientas para el diseño de modelos durante las fases de ingeniería de software, soluciones para el control de versiones, para la programación mediante el apoyo de una librería gráfica de potentes capacidades y el trabajo para lograr extensiones que posibilitan la gestión de los archivos generados (Ver Tabla 2).

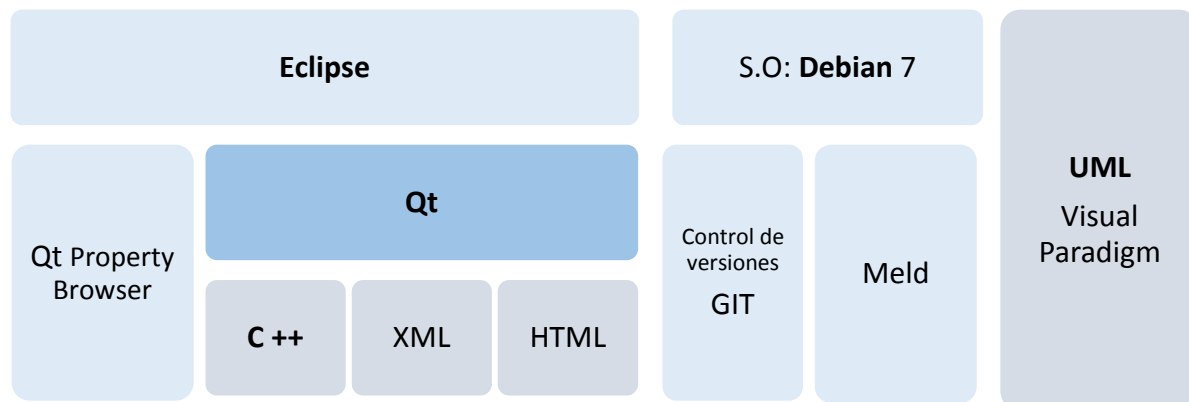


Figura 7: Entorno de desarrollo

¹ Inscrito en el Registro Facultativo de Obras Protegidas y de Actos y Contratos referidos al Derecho de Autor con número 0380-01-2015, en versión 1.0. Tipo de Obra: Software. Autores: Lisett Pérez Quintero y Jorge Carrera Ortega. Titular: Empresa de Aplicaciones Informáticas. Desoft, 2015.

Framework Qt	Qt es un toolkit de plataforma cruzada (cross-platform) desarrollado por Trolltech para construir GUIs en C++. Puede funcionar en varias plataformas: Microsoft Windows, Unix/X11, MAC OS X, e incluye una extensa librería de clases C++ y utilidades para construir las aplicaciones de forma rápida y sencilla.
Qt Property Browser	Qt Property Browser es una librería que le posibilita al usuario editar un conjunto de propiedades. Provee un navegador que visualiza las propiedades dadas con etiquetas y sus correspondientes editores. Por cada tipo de propiedad, provee un controlador el cual puede ser asociado con la fábrica del editor preferido. Además provee un tipo de propiedad basada en variante con su correspondiente controlador y fábrica.
Eclipse (Kepler)	Uno de los IDEs más conocidos con soporte para múltiples plataformas y lenguajes. Es una plataforma de código abierto, que soporta la programación en lenguaje C++. (C/C++ IDE).
C++	Lenguaje de programación (estándar), soporta la abstracción de datos, programación orientada a objetos.
XML	XML por sus siglas en inglés eXtensible Markup Language, utilizado para organizar la información de una manera específica, en la presente solución facilita la creación y utilización de una extensión creada con la identificación .cpx para la salva y carga de archivos de CAPYROX.
HTML	Lenguaje de composición de documentos y especificación de hipervínculos, siendo posible la creación de documentos interactivos mediante el vínculo a hipertexto interconectando documentos.
Plug-in Qt para Eclipse	El plugin Qt para Eclipse permite desarrollar aplicaciones en C++ usando las librerías Qt.
Control de versiones GIT	Se utilizó GIT para el control de versiones, permitiendo el mantenimiento de las versiones y el acceso organizado en su gestión, para lo cual se crearon las ramificaciones pertinentes dentro de la gestión del proyecto de desarrollo.
Meld	Software utilizado para la combinación de contenidos (código) que emite información sobre los cambios realizados mediante la comparación de archivos.
Visual Paradigm Versión 13	Para la realización de los diagramas UML

Tabla 2: Resumen de elementos del entorno de desarrollo

2.3 Estructura del modelo

El modelo propuesto se centra en una clase “*Process*”, que contiene como atributos la información de las propiedades requeridas en el proceso de inspección. Estos atributos permiten trabajar la información del nombre del proceso o la actividad y además un identificador numérico que responde a un índice de la modularidad. *Process* es una clase que permite instanciar los procesos del modelo y también las actividades.

En el modelo ocurren dos distinciones importantes (Ver Figura 8), una hace referencia al control gráfico sobre las relaciones entre los procesos y la otra al manejo de referencias del proceso o los segmentos con los diccionarios.

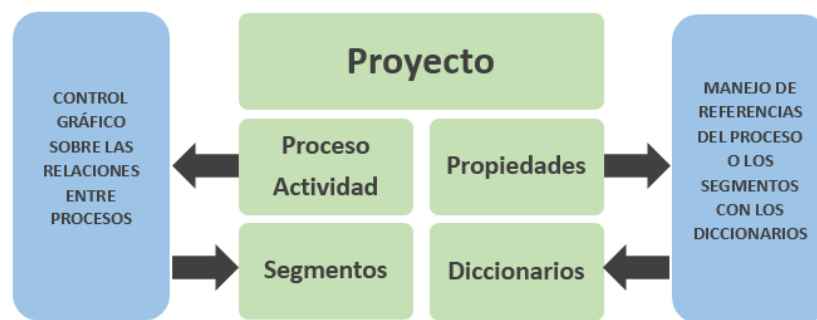


Figura 8: Relación de perspectivas asumidas para el modelo

2.3.1 Control gráfico sobre las relaciones entre procesos

El control gráfico sobre las relaciones entre procesos responde a la necesidad de conocer cuáles son las conexiones extremas del segmento y facilitar su recorrido. Con este enfoque se aborda para el control gráfico la clase “*Element*”, la cual permite trabajar las posibilidades de interrelación controlando los tipos posibles de conexión en los extremos de un segmento. La clase *Element* se define como se muestra a continuación:

```
class Element : public QObject
{
    (...)
public:
    Element();
    Element(const Element &other);
    virtual ~Element();
    virtual bool isBranch() const;
    virtual bool isBoundary() const;
    QString name() const;
public slots:
    virtual void setName(const QString &name);
signals:
    (...)
protected:
    (...)
};
```

Con este enfoque es posible relacionar con *Element*, mediante herencia, las clases “*ArrowSegment*”, “*Boundary*”, “*Process*” y “*Branch*” (Ver Figura 9). El pensamiento en el modelo para esta determinación es que un segmento puede estar conectado en sus extremos a procesos/actividades, bordes de diagrama, puntos de uniones o bifurcaciones de la relación.

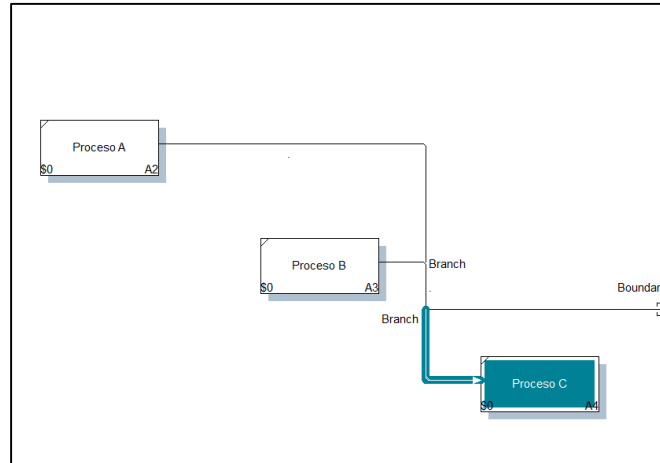


Figura 9: Control gráfico de las relaciones (tipo *Branch*, *Boundary*, y entre procesos)

La propia clase *ArrowSegment* hereda de *Element* para este mismo tratamiento de trazabilidad en el modelo de las interrelaciones.

```

class ArrowSegment : public Element
{
    (...)
public:
    ArrowSegment();
    ArrowSegment(Element *in, Element *out);
    ArrowSegment(const ArrowSegment &other);
    virtual ~ArrowSegment();

    bool isEmpty() const;
    bool isBoundary() const;

    (...)

    virtual bool inputLinked() const;
    virtual bool outputLinked () const;

    (...)

    bool hasInput() const;
    QPointer<Element> input() const;
    void setInput(Element *in);
    bool hasOutput() const;
    QPointer<Element> output() const;
    void setOutput(Element *out);

    (...)
protected:
    (...)
};
  
```

2.3.2 Manejo de referencias del proceso o los segmentos con los diccionarios.

En el modelo se ha definido una vía para la relación con algunas de las propiedades mediante el uso de diccionarios. La fundamentación de su uso se basa en la capacidad que brinda para la actualización de cambios en el proyecto general actualizando la información en todos los diagramas pertinentes respecto a la variación de un elemento contenido en el diccionario. La clase central para esta perspectiva es “*Dictionary*”. Los casos mostrados en la Figura 10 constituyen diccionarios, permitiendo la inclusión general de estos en el proyecto y referenciarlos posteriormente desde algún proceso o relación.

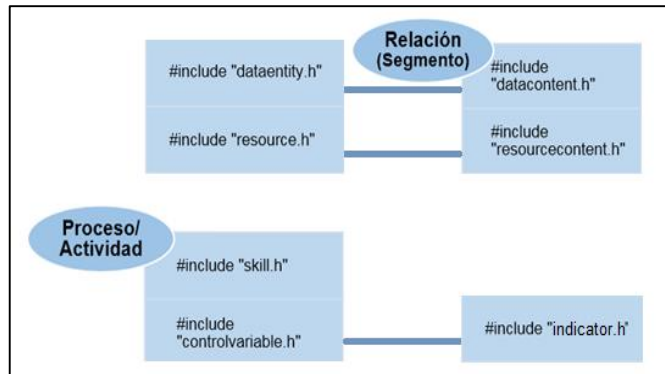


Figura 10: Inclusiones relacionadas a los diccionarios

En la Figura 11 se muestra un ejemplo de relación visual entre un segmento del diagrama, la propiedad “recurso” y el diccionario de recursos. Mayor información al respecto se expone en el epígrafe 2.2.3 *Estructuras de datos utilizadas*. Existen otras clases como “*Risk*” en correspondencia a la propiedad Riesgo que no está definida como Diccionario, aun cuando emplea la estructura de dato tipo lista.

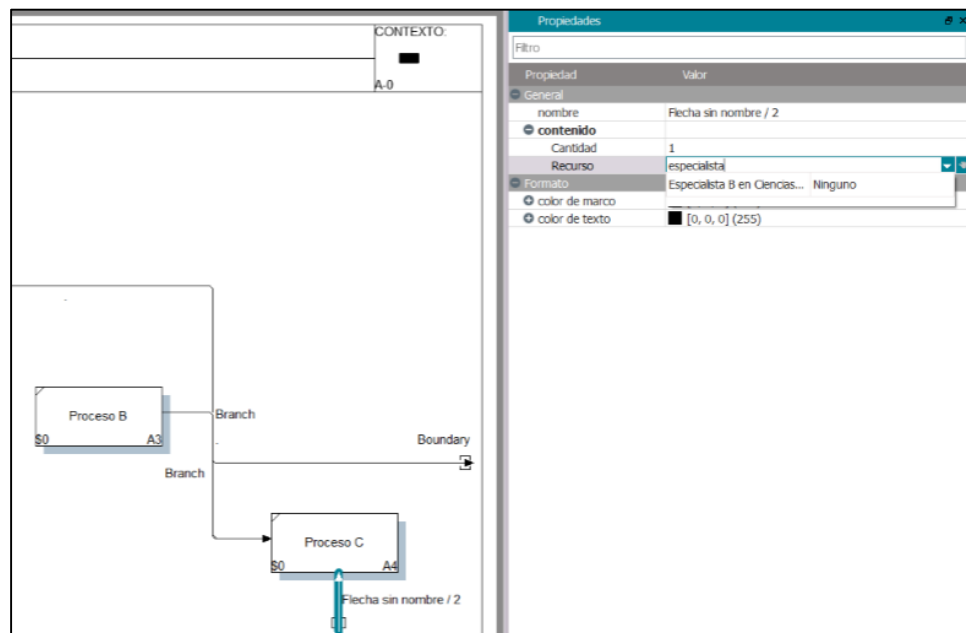


Figura 11: Ejemplo de relación visual entre un segmento, la propiedad “recurso” y el diccionario de recursos.

En la Figura 12 se muestra el diagrama de clases correspondiente a la concepción del modelo.

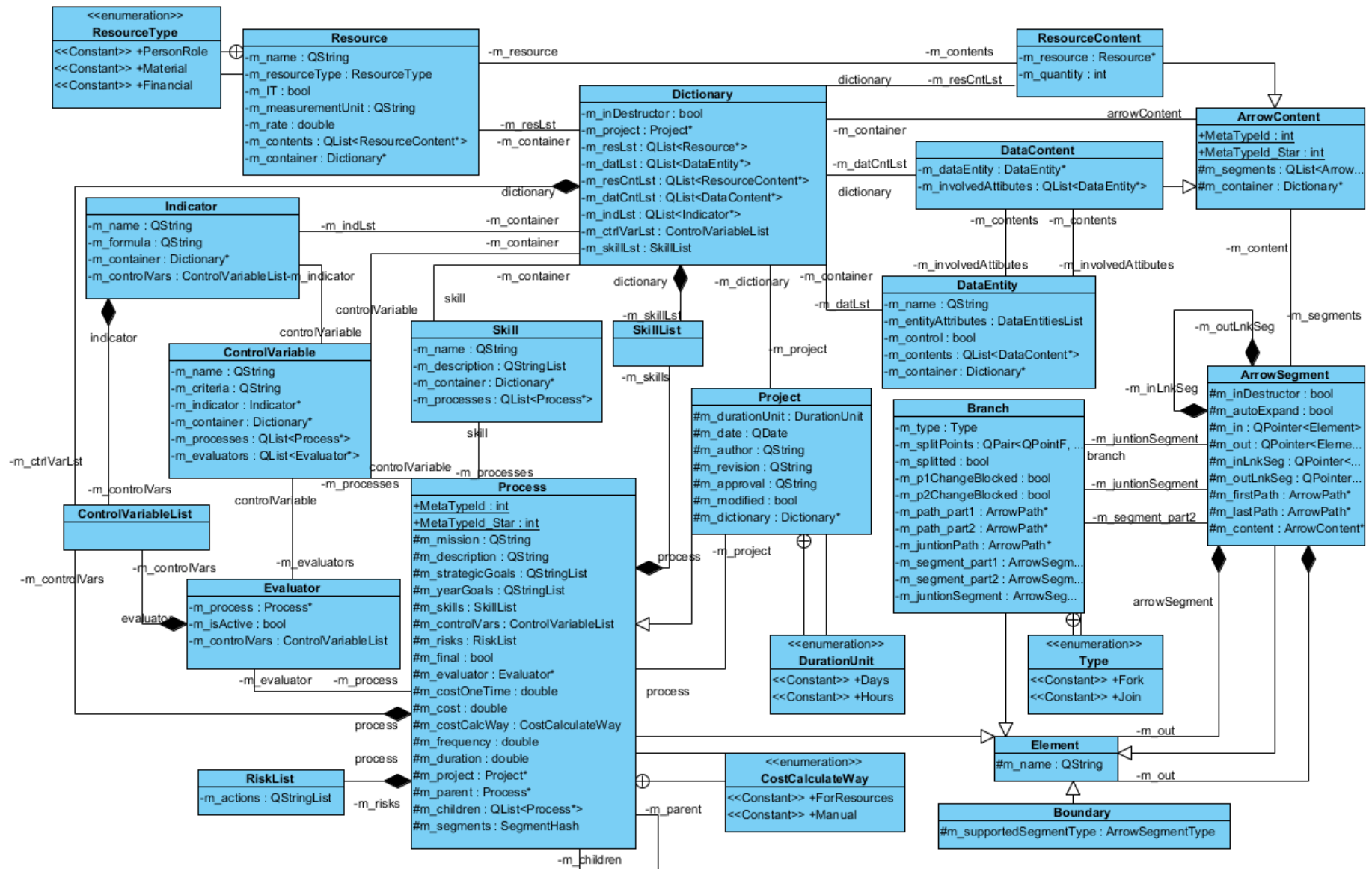


Figura 12: Diagrama de clases correspondiente a la concepción del modelo

2.3.3 Estructuras de datos utilizadas

2.3.3.1 Diccionarios

Para la vinculación entre los componentes un aspecto clave es lograr trabajar con las adecuadas estructuras de dato. Una estructura de dato es un método para realizar un conjunto de operaciones sobre algunos datos (57), esta fuente refiere el clásico ejemplo de seguir la huella de un conjunto de elementos, elementos que están identificados por valores claves de manera que se pueden insertar y eliminar los elementos dentro del conjunto y localizar elementos dentro del conjunto a través de un valor clave. La estructura que soporta esas operaciones es llamada diccionario. En CAPYROX se utilizan las listas, como estructuras fundamentales para crear 4 diccionarios.

La Figura 13 muestra el diccionario de recursos. En este diccionario contendrá la información de recursos de tipo rol, material y financiero. Es posible precisar que algún recurso es TI. También se incluye la información de la etiqueta (unidad de medida) y la tasa que puede precisarse por hora o por día en el caso de recursos de tipo rol y en caso de recursos materiales es el costo unitario.

 Importar		 Exportar			
Nombre		Tipo	TI	Etiqueta	Tasa
1		Rol de persona		Ninguno	0/h

Figura 13: Diccionario de recursos

```
class Resource : public QObject
{
    (...)
public:
    (...)
    QList<ResourceContent *> contents() const;
    void attach(ResourceContent *content);
    void detach(ResourceContent *content);

protected:
    (...)
private:
    (...)
};
```

El diccionario de datos está concebido para registrar los posibles datos a referenciar en las relaciones entre procesos/actividades. Está constituida por el nombre de las entidades y la especificación de atributos para estas entidades de dato (Ver Figura 14).

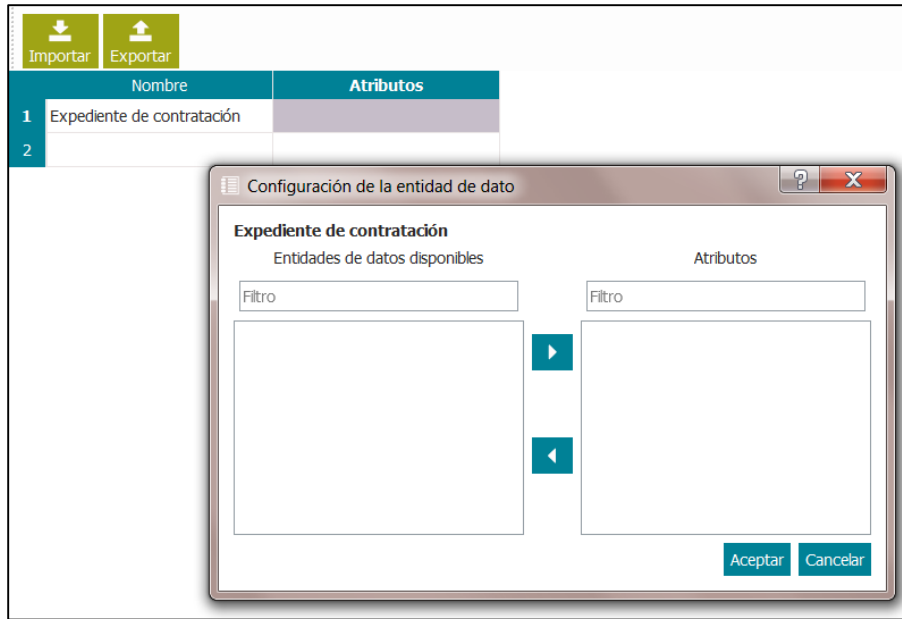


Figura 14: Diccionario de datos

```
class DataEntity : public QObject
{
    (...)
public:
    (...)
    QList<DataContent *> contents() const;
    void attach(DataContent *content);
    void detach(DataContent *content);

protected:
    (...)
private:
    (...)
};
```

La incorporación de un diccionario de indicadores y variables de control (Ver Figura 15) permite obtener esos elementos para poderlos asociarlos a los procesos modelados. Este diccionario permite definir los indicadores con un nombre y su fórmula para luego poder establecer variables de control asociadas a esos indicadores estableciendo además un criterio de medida.

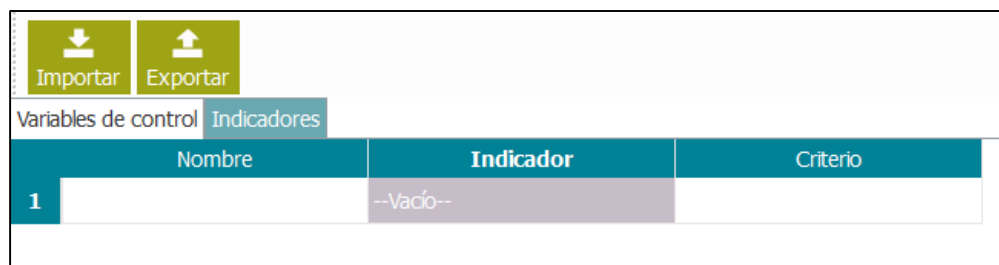


Figura 15: Diccionario de indicadores y variables de control

```

class Indicator
{
    (...)
public:
    (...)
    ControlVariableList controlVariables() const;
    void attach(ControlVariable *controlVar);
    void detach(ControlVariable *controlVar);
protected:
    (...)
private:
    (...)
};
//*****
class ControlVariable
{
    (...)
public:
    (...)
    bool hasIndicator() const;
    Indicator *indicator() const;
    void setIndicator(Indicator *indicator);
    (...)
    QList<Process *> processes() const;
    void attach(Process *process);
    void detach(Process *process);
    (...)
protected:
    (...)
private:
    (...)
};

```

El diccionario de competencias (Ver Figura 16) permite especificar las competencias que luego serán asociadas a un proceso o actividad específica. Contiene un además una lista para especificar las unidades de competencias que constituyen una competencia.

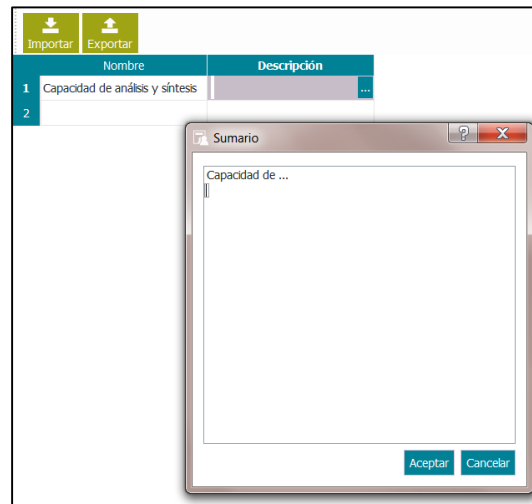


Figura 16: Diccionario de competencias

```

class Skill
{
    (...)
public:
    (...)
    QList<Process *> processes() const;
    void attach(Process *process);
    void detach(Process *process);
protected:
    (...)
private:
    (...)
};

```

2.3.3.2 Árboles

La propia definición recursiva de un árbol fue un factor fundamental en consideración al uso de este tipo de estructura. La clase *Process*, dentro del diagrama de clases (Ver Figura 12) que ilustra la estructura del modelo concebido en esta etapa, representa una abstracción del proceso.

Un proceso/actividad se define como el resultado de la unión de un conjunto de actividades interrelacionadas, presentando una estructura en forma de árbol. Este árbol cuenta con al menos un nodo raíz que según IDEF0 sería el proceso contexto.

Esta es una de las consideraciones que permite la aplicación de modularidad en el ambiente de modelado, permitiendo aplicar la consideración de subprocesos, que agrupan un grupo de actividades dentro del proceso, y además la descomposición de cualquier diagrama en diagramas más detallados ya sea respecto a sus procesos o sus actividades. Esto está en relación a la numeración que reciben los procesos y actividades insertadas, a modo de índice que permite la rápida localización.

```

class Process : public Element,
                public Subject
{
    (...)
public:
    (...)
    bool isDecomposed() const;
    bool hasChildren() const;

    (...)
    Process *parent() const;
    void setParent(Process *parent);

    QList<Process *> children() const;
    (...)
private:
    (...)
protected:
    (...)
};

```

2.4 Componentes concebidos.

2.4.1 Vista estática y funcional.

2.4.1.1 Explorador de proyecto

El explorador de proyectos es un contenedor de proyectos, cada uno con su correspondiente árbol de procesos. El árbol de procesos es una abstracción de la organización de las actividades dentro de los procesos.

La estructura organiza visualmente las composiciones de acuerdo a la contención de actividades por procesos, de manera que las actividades contenidas en el diagrama de un proceso quedan representadas en el árbol como “hijas” de ese proceso/actividad y así sucesivamente **(2)** (Ver Figura 17).

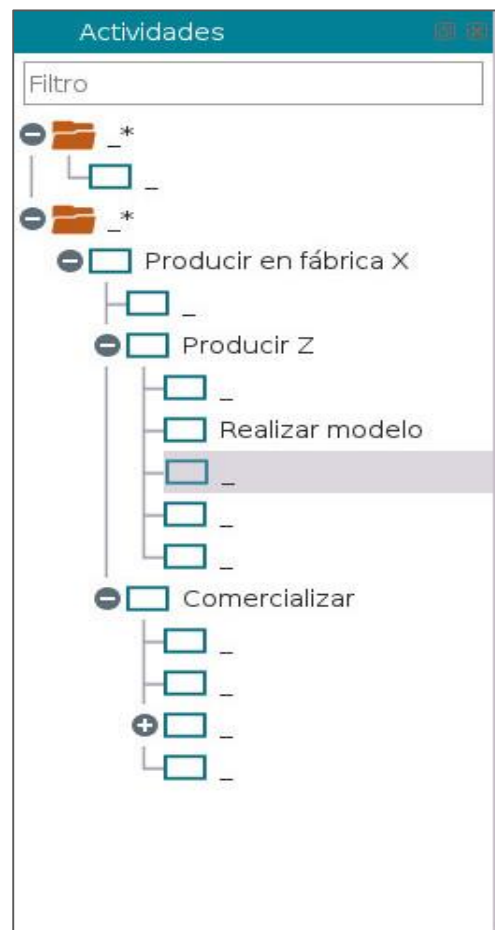


Figura 17: Explorador de proyectos

Este componente se concibe para la visualización de un índice que permite el acceso rápido a un proceso/actividad, además de brindar un esquema para reflejar la estructura.

Su diseño se basa en el concepto modelo/vista (Ver Figura 18), el cual provee una solución que usa una arquitectura más versátil. El concepto modelo/vista facilita el empleo de más de una forma de visualizar el mismo dato, porque un modelo puede ser pasado a muchas vistas. La principal diferencia es que el modelo/vista opera directamente sobre los datos. Desde las clases de las vistas no se conocen las estructuras de los datos, el modelo les sirve de interfaz para encapsular su representación, permitiéndole a la vista a través del delegado leer y escribir sobre su contenido (58).

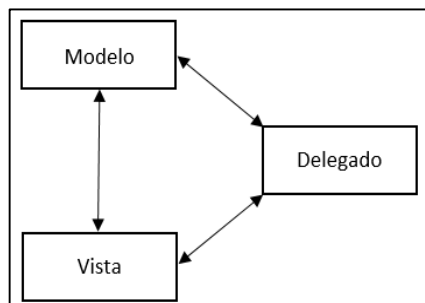


Figura 18: Concepto Modelo/Vista

2.4.1.2 Área de modelado

El área de modelado es un contenedor de diagramas de procesos, está ubicada en la parte central del ambiente de trabajo de CAPYROX, por ser de hecho el espacio más utilizado para el diseño o rediseño del proceso (Ver Figura 19). Permite al usuario la utilización de varios componentes de modelado, los cuales son representados en los distintos diagramas. El diagrama representa una sola unidad de un modelo IDEF0 que detalla un proceso o una actividad.

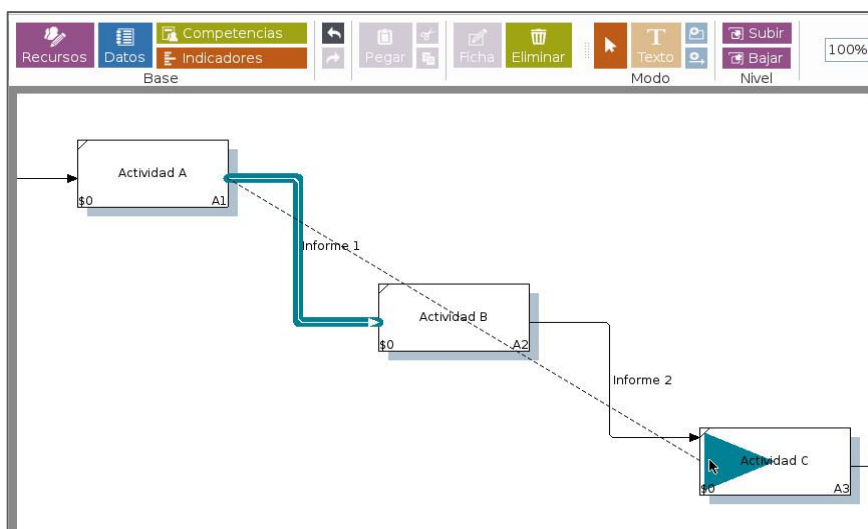


Figura 19: Área de modelado

La clase “QGraphicsScene” provee una superficie para administrar un gran número de elementos gráficos 2D (58) en Qt; “DiagramScene” hereda las funcionalidades que brinda. DiagramScene es la clase que implementa un diagrama IDEF0 de proceso en CAPYROX.

Se basa su diseño en el patrón modelo/vista/controlador (MVC) (Ver Figura 20), el cual provee claras responsabilidades para todas las clases involucradas. El patrón MVC está dirigido a la separación del dato (modelo), la visualización (vista) y la modificación (controlador). Separa el dato de la visualización; evita la duplicación de los datos; puede mostrar el mismo dato en múltiples vistas; así como utilizar la misma vista para múltiples datos. Separa la visualización de la modificación; puede emplear la aplicación de acciones específicas cuando se altera el dato; y la vista tiene solo una interfaz para toda la edición.

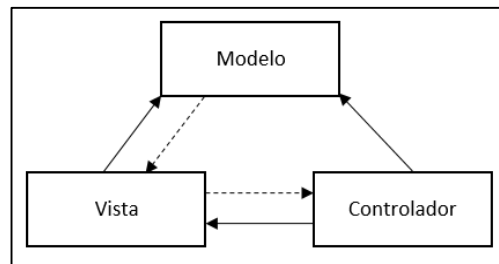


Figura 20: Patrón MVC

2.4.1.3 Inspector de propiedades

El inspector de propiedades es un contenedor de propiedades, cada una con su correspondiente valor. Permite al usuario editar el valor de cada una de sus propiedades. La utilización de la inspección de propiedades ayuda en la ágil ilustración de las características más relevantes respecto a los elementos dentro del diagrama (Ver Figura 21).

Propiedad	Valor
General	
nombre	_
misión y alcance	
descripción	
objetivos estratégicos	
objetivos del año	
competencias	
variables de control	
riesgos	0 Calidad del producto; ...
evaluator	☒ Falso
Variables	
responsable	Ninguno
roles	Ninguno
materiales	Ninguno
financieros	Ninguno
proveedores	Ninguno
clientes	Ninguno
Costo	
Dato	
entradas	Ninguno
salidas	Ninguno
Formato	
color de marco	☒ [0, 0, 0] (255)
color de texto	☒ [0, 0, 0] (255)

Figura 21: Inspector de propiedades

Se basa su diseño en el patrón MVC y en la utilización del patrón meta objeto. Un meta objeto es un objeto que describe la estructura de otro objeto. El “*QMetaObject*” es la implementación del patrón meta objeto de Qt. Proporciona información acerca de las propiedades y métodos de un “*QObject*”. El patrón meta objeto es a veces conocido como el patrón reflejo.

QObject es la clase base de casi todas las clases de Qt, las propiedades es uno de los mecanismos que componen Qt. Qt implementa la introspección en C++: cada *QObject* tiene un meta objeto; y el meta objeto conoce entre otros, sobre las propiedades, e informaciones generales (*QObject::classInfo*) que lo componen.

Un *QObject* tiene propiedades con sus respectivos métodos de acceso y modificación de su valor, esta información es recogida a través de la siguiente macro (Ver Figura 22).

```
Q_PROPERTY ( type name
            READ getFunction
            [WRITE setFunction]
            [RESET resetFunction]
            [NOTIFY notifySignal]
            [DESIGNABLE bool]
            [SCRIPTABLE bool]
            [STORED bool]
            [USER bool]
            [CONSTANT]
            [FINAL] )
```

Figura 22: Estructura de la macro para la definición de una propiedad en Qt (59)

Element es la clase base de los elementos del modelo de CAPYROX (Ver Figura 23). En la figura se muestra cómo se crea la propiedad “*name*” correspondiente al atributo de la clase con el mismo nombre, guardando estrecha vinculación con la utilización de la macro de la Figura 22. Se puede evidenciar en la imagen además, el empleo de las señales (signals) y ranuras (slots), nueva información que comprende también parte del conocimiento de un meta objeto, auxiliarse en **(58)** para más detalles.

```

class Element : public QObject
{
    Q_OBJECT
    Q_PROPERTY(QString name READ name WRITE setName NOTIFY
nameChanged)

public:
    Element();
    Element(const Element &other);
    virtual ~Element();

    virtual bool isBranch() const;
    virtual bool isBoundary() const;

    QString name() const;

public slots:
    virtual void setName(const QString &name);

signals:
    void nameChanged(const QString &name);

protected:
    QString m_name;
};

```

Figura 23: Creación de propiedades personalizadas

Estos metadatos son obtenidos en tiempo de compilación por el compilador meta objeto (moc) (Ver Figura 24 y 25).

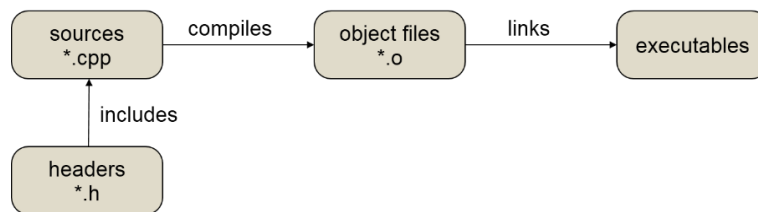


Figura 24: Proceso ordinario de construcción en C++ (59)

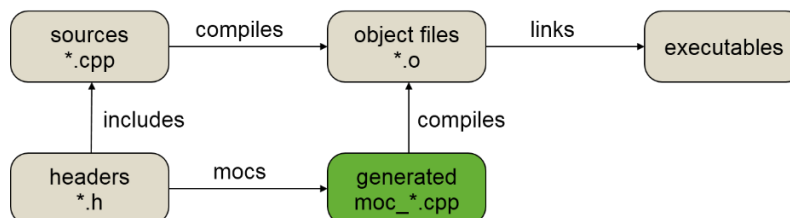


Figura 25: Proceso de construcción en Qt C++ (59)

El moc recolecta los datos de los archivos cabeceras.

2.4.2 Vista dinámica.

CAPYROX presenta una arquitectura diseñada para múltiples vistas (componentes) y su relación entre ellas, cada una coopera en la forma de visualizar la información contenida en el modelo.

Se basa su diseño estructural en el patrón MVC y en el patrón observador (Ver Figura 26), el cual es muy frecuente en aquellos sistemas que se fragmentan en un conjunto de clases que cooperan en la necesidad de mantener la consistencia entre los distintos objetos interrelacionados. Este patrón define una dependencia “uno-a-muchos” entre objetos, para que, cuando uno de ellos cambie su estado, todos los que dependan de él sean avisados y puedan actualizarse convenientemente. El patrón observador describe como establecer estas relaciones. Los objetos claves en este patrón son el sujeto y el observador. *Process* es el elemento del modelo dentro de CAPYROX que implementa el sujeto y tanto el explorador de proyectos como el área de modelado son las distintas vistas que implementan el observador. Un proceso puede tener cualquier número de observadores dependientes. Todos los observadores son notificados cuando el proceso sufre un cambio en su estado. En respuesta, cada observador podrá consultar al proceso para sincronizar su estado con el estado del proceso.

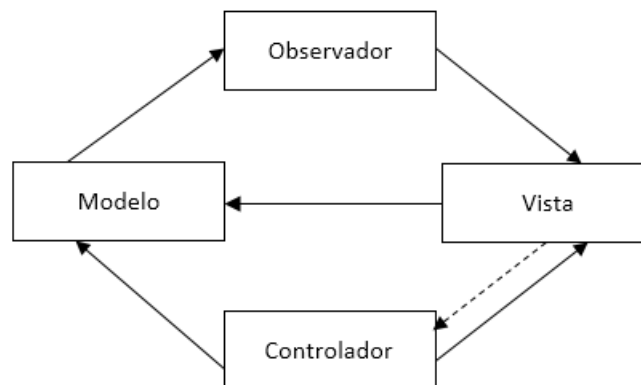


Figura 26: Patrón Observador

El patrón solitario garantiza que una clase sólo tiene una única instancia, proporcionando un acceso global a la misma. “CAPYROXApp” es la implementación del patrón solitario de CAPYROX; proporciona información acerca de su núcleo. “CAPYROXCore” es la implementación del núcleo de CAPYROX, el cual brinda los métodos necesarios para la interacción entre las diferentes vistas. Lográndose así la comunicación con el inspector de propiedades, permitiéndole al usuario poder visualizar y modificar las propiedades de la parte del modelo seleccionado en las demás vistas.

El concepto modelo/vista (Ver Figura 18) en el explorador de proyectos facilita la interacción con el área de modelado a través del delegado. El delegado permite la creación y la sincronización de los diagramas de procesos. Cada diagrama detalla el flujo del proceso seleccionado por el usuario desde el árbol de procesos.

2.5 Objetos gráficos

“*QGraphicsItem*” es la clase base para todos los elementos gráficos en un *QGraphicsScene*, provee un fundamento de poco peso para escribir nuestros propios elementos personalizados. Esto incluye definir la geometría de los elementos, detección de colisiones, su implementación de pintado e interacción de los elementos a través de la manipulación de sus eventos (58).

A continuación se listan un conjunto de elementos gráficos personalizados para detallar un proceso sobre un diagrama IDEF0:

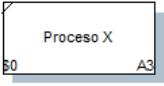
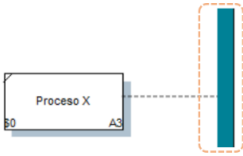
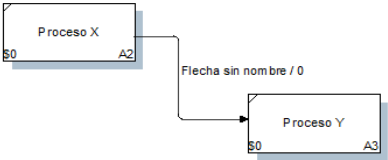
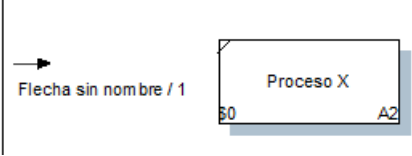
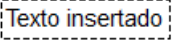
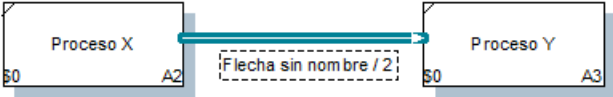
	“<i>GraphicsBox</i>”: Un rectángulo, contiene un nombre y un índice; usado para representar un proceso.
	“<i>GraphicsBoundaryBox</i>”: Un rectángulo; usado para representar los bordes del diagrama.
	“<i>GraphicsArrowSegment</i>”: Un segmento de línea que se origina o termina a los lados de un <i>GraphicsBox</i>, un branch (fork o join), o un <i>GraphicsBoundaryBox</i>.
	“<i>GraphicsBoundarySegment</i>”: Una flecha (con notación especial) que no sigue el requerimiento normal de un segmento en un diagrama; aunque hereda su comportamiento, se emplea para hacer relacionar los segmentos del diagrama padre con el diagrama hijo.
	“<i>GraphicsIndependentText</i>”: Un texto; usado para representar un comentario textual global sobre un diagrama.
	“<i>GraphicsSegmentText</i>”: Un texto; usado para representar el nombre del segmento, asociado a cada <i>GraphicsArrowSegment</i>.

Tabla 3: Objetos gráficos personalizados

2.6 Conclusiones del capítulo

En este capítulo se abordaron aspectos importantes del diseño de la arquitectura que presenta la herramienta CAPYROX 1.2 por componentes y su estrecha cooperación en relación al modelo. Se describieron los diferentes patrones de diseño arquitectónico por componentes. Se abstraieron elementos en relación a la notación basada en IDEF0 para la concepción de un modelo que sirva de soporte a los distintos componentes. Se puede arribar a las siguientes conclusiones:

1. El entorno de desarrollo elegido es capaz de sostener el rendimiento gráfico propicio en correspondencia a las características dinámicas y estáticas concebidas para el modelador, mediante la cooperación simultánea de cada elemento.
2. La estructura del modelo presenta una relación de perspectivas que logra intercalar adecuadamente el control gráfico en procesos y segmentos con el manejo de referencias a los diccionarios desde las propiedades.
3. Los diccionarios concebidos responden a la actualización de cambios de una manera centrada en la organización y no en partes específicas, previendo la consideración de bases en torno a la información, los recursos, las competencias, las variables de control con sus indicadores, lo cual permite un acercamiento crucial de las propiedades al enfoque organizacional abordado en la NC ISO 9001:2015, la propuesta **(1)** e índices de comprensibilidad definidos en 1.5.3.1 como las adecuadas etiquetas de los elementos gráficos y dimensiones cognoscitivas como el compromiso con la calidad y las habilidades para buscar y analizar información.
4. La vista estática y funcional, a través de los patrones empleados, permite percibir una adecuada correspondencia entre las acciones del usuario y la guía que propicia el inspector de propiedades, justificando su apoyo a la comprensibilidad mediante la localización gráfica de elementos ya sea desde el área de modelado o del explorador de proyectos.
5. Los objetos gráficos establecidos permiten responder congruentemente al valor semántico de los componentes de la notación IDEF0.

CAPÍTULO 3: IMPLEMENTACIÓN DEL MECANISMO DE INSPECCIÓN EN LA HERRAMIENTA CAPYROX 1.2

En el presente capítulo se expone la definición del mecanismo de inspección de propiedades para CAPYROX 1.2. Se realiza una presentación de la implementación de los grupos del inspector de acuerdo a los objetos gráficos del modelo, detallados en el capítulo anterior. Se refieren los resultados de las pruebas de calidad con objetivo de validar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura mediante su implementación en un modelador de procesos orientado al valor y su capacidad para facilitar la comprensibilidad de los modelos; para lo cual también se incluye el resultado de un análisis de la aplicación práctica en el tejido empresarial utilizando atributos FURPS y la NC ISO/IEC 9126.

3.1 Mecanismo de inspección de propiedades en CAPYROX 1.2

Para el trabajo con las propiedades se utiliza la librería Qt Property Browser. En el diseño de su arquitectura se define muy bien el empleo del patrón MVC (Ver Figura 20), cada propiedad cuenta con su controlador y editor correspondiente. Este mecanismo solo lo aplica a las propiedades que presentan los tipos de datos básicos conocidos del lenguaje C++ y otros que define el propio Qt.

La librería provee un mecanismo de inspección de propiedades, para ello define la clase “*QtVariantProperty*”, tipo de propiedad basado en variante con su correspondiente controlador y fábrica para la edición. “*QtVariantPropertyManager*” es la clase controlador de variantes, posee un registro de los diferentes controladores por tipos de datos; utiliza el controlador acorde a la consulta de una propiedad. “*QtVariantEditorFactory*” es la clase fábrica de variantes para la edición, posee un registro de las diferentes fábricas de editores por tipos de datos; utiliza la fábrica acorde a la edición de una propiedad. “*QtTreePropertyBrowser*” es la vista empleada para visualizar estas propiedades variantes en forma de árbol, permite la navegación y edición sobre su contenido por el usuario.

En los diagramas derivados del uso IDEF0 existe un conjunto de elementos gráficos propios de la sintaxis de dicho estándar. A estos elementos gráficos se le han definido un conjunto de propiedades en consecuencia al enfoque del modelado de proceso orientado al valor (Ver Epígrafe 3.1). En el mecanismo de inspección de la librería no están definidos los controladores necesarios para gestionar estos tipos de datos. La clase “*QVariant*” actúa como una unión de los más comunes tipos de datos de Qt. C++ prohíbe uniones desde tipos incluidos que tienen constructores o destructores que no son por defecto, por lo que las clases de Qt no pueden ser usadas en uniones. Un objeto *QVariant* mantiene un solo valor de un solo tipo al tiempo. *QVariant* puede ser extendido para soportar otros tipos de datos que no sean los que son mencionados en su enumerador de tipos, auxiliarse en (58) para más detalles.

La inspección de nuevas propiedades requirió extender las funcionalidades de esta librería. Teniendo en cuenta las facilidades que brinda un *QVariant* se definieron las interfaces “*IPropertyManager*” y “*IEditorFactory*” en consecuencia con el controlador y editor necesarios para inspeccionar una propiedad; ambos a personalizar por el nuevo tipo de dato a incorporar dentro de las propiedades válidas para su inspección, así como los cambios necesarios en la implementación del mecanismo provisto por la librería realizados para su soporte.

```
class IPropertyManager : public XPropertyManager<QVariant>
{
    Q_OBJECT
public:
    typedef QMap<int, QtAbstractPropertyManager *> SubPropertyManagerMap;
public:
    IPropertyManager(QObject *parent = 0);
    virtual ~IPropertyManager();
    virtual int getTypeId() const = 0;
    virtual QMap<int, QtAbstractPropertyManager *> subPropertyManagers()
const;
Q_SIGNALS:
    void valueChanged(QtProperty *property, const QVariant &val);
    void attributeChanged(QtProperty *property,
                        const QString &attribute,
                        const QVariant &val);
    void resetProperty(QtProperty *property);
protected:
    virtual void createConnection(VariantPropertyManager *manager);
    virtual void breakConnection(VariantPropertyManager *manager);

    friend class VariantPropertyManager;
};
//*****
class IEditorFactory : public QtAbstractEditorFactory<IPropertyManager>
{
    Q_OBJECT
public:
    IEditorFactory(QObject *parent = 0);
    virtual ~IEditorFactory();
    virtual bool isResettable() const;
protected:
    virtual void connectPropertyManager(IPropertyManager *manager) = 0;
    virtual QWidget *createEditor(IPropertyManager *manager,
                                QtProperty *property,
                                QWidget *parent) = 0;
    virtual void disconnectPropertyManager(IPropertyManager *manager) = 0;
    friend class VariantEditorFactory;
};
```

En los diagramas derivados del uso IDEF0 también coinciden elementos gráficos con iguales niveles en su jerarquía de clases, por lo que presentan propiedades comunes. La inspección múltiple, agrupando las propiedades comunes de diferentes objetos, permiten tomando a uno como referencia, que una vez que se realice un cambio, sea aplicado a los demás por igual. En la Figura 27 se muestra un ejemplo, es importante aclarar que se realizó para el caso de la propiedad “*nombre*” a modo demostrativo, aunque en

los modelos de procesos el usuario no debe especificar procesos/actividades con un mismo nombre. Esto es mayormente aplicable para las demás propiedades.

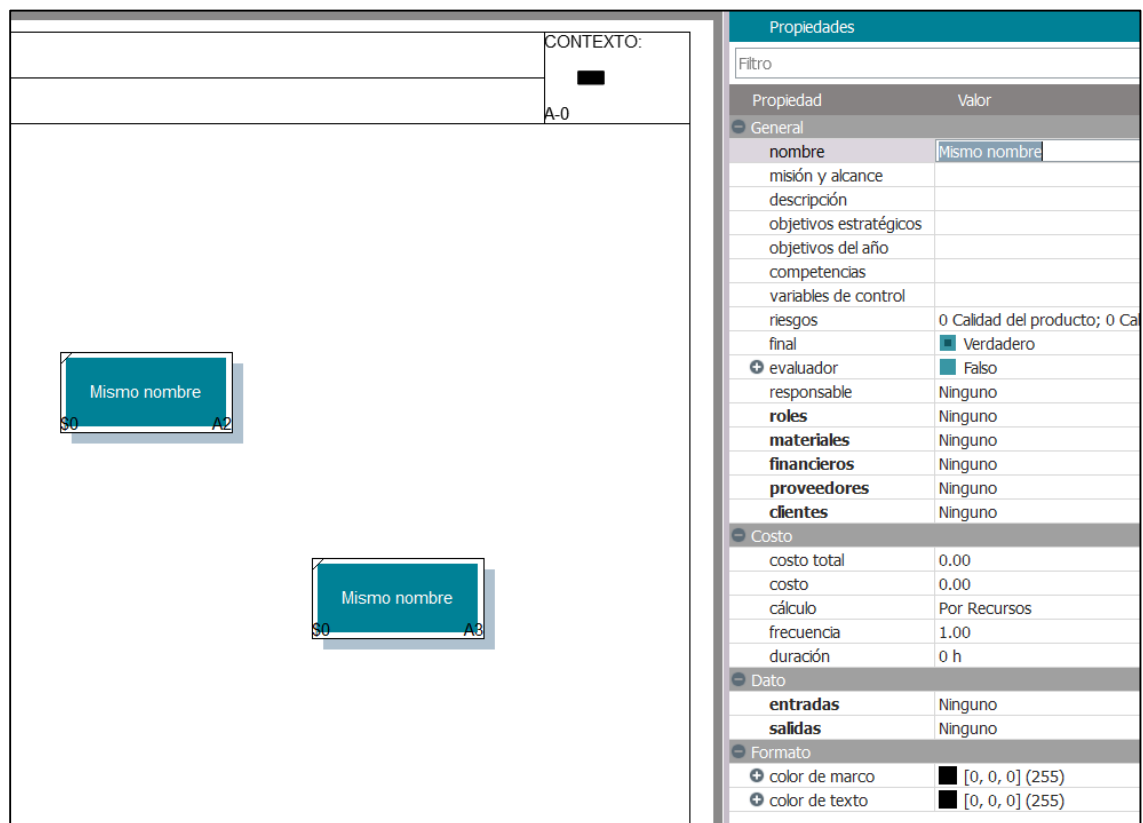


Figura 27: Inspección múltiple, agrupando las propiedades comunes de diferentes objetos

Dentro de la jerarquía de clases de un elemento gráfico se pueden encontrar propiedades que como nuevo tipo de dato no estén incorporadas dentro de las posibles propiedades válidas para su inspección, puesto que no implementan tal mecanismo. Estas propiedades son clases que al heredar de un *QObject* adquieren la capacidad de introspección del mismo y pueden contener dentro sus propias propiedades. La inspección recursiva garantiza no tener que obligatoriamente definir un controlador y una fábrica para la edición de este tipo de propiedad que como "*QObject**" se identifica y se inspecciona su contenido hasta llegar a las propiedades válidas. En el epígrafe 3.1 se aborda con mayor detalle esta funcionalidad.

CAPYROX incorpora como parte de la solución del componente *Inspector de propiedades* estas posibilidades a través del "*ObjectController*", enriqueciéndolo en este sentido. El *ObjectController* es la clase encargada de gestionar todo el mecanismo de inspección de propiedades que brinda la librería y fusionarlo con el empleo del patrón meta objeto de Qt para su correcto funcionamiento, así como almacenar el estado expandido o no de la propiedad en la vista. Cada nuevo tipo de dato que se desee inspeccionar debe implementar las interfaces concebidas y registrarlas a través del controlador de objetos. En la Figura 28 se muestra el diagrama de clases perteneciente al núcleo del mecanismo de inspección de las propiedades.

3.1 Implementación de grupos de propiedades por objetos gráficos del modelo de proceso.

Existe una correspondencia de la definición de los grupos del inspector de propiedades con los objetos gráficos del modelo, de manera tal que para la selección de los objetos se visualizan las propiedades correspondientes. A continuación se detalla esa relación.

3.1.1 Implementación de grupo: general

El grupo general está definido sobre la base de un conjunto de propiedades comunes para todos los objetos gráficos, como lo es el *nombre*. Pero en dependencia del objeto incorpora otras propiedades. Hasta el momento los objetos gráficos que posibilitan la inspección de este grupo general son los procesos/actividades (*GraphicsBox*) y los segmentos (*GraphicsArrowSegment*).

3.1.1.1 Objeto gráfico: proceso/actividad.

Para este caso fue necesario definir nuevas propiedades (Ver Figura 29) en consecuencia con la propuesta presentada para este grupo en **(1)**. Para el caso de las demás propiedades la propia librería ya incorporaba sus tipos de datos al ser básicos, exceptuando el caso de la propiedad “*evaluador*” siendo personalizada. Las actividades evaluadoras se definen dentro de la organización en procesos para poder distinguir una actividad dentro del modelo que tiene la responsabilidad de evaluar un indicador mediante una variable de control.



Figura 29: Propiedades personalizadas en grupo general

Competencias

Este tipo de propiedad requirió de la implementación concreta de las clases “*SkillListPropertyManager*” y “*SkillListEditorFactory*” para el tipo de dato “*SkillList*” del modelo presentado en el epígrafe 2.3.

Riesgos

Este tipo de propiedad requirió de la implementación concreta de las clases “*RiskListPropertyManager*” y “*RiskListEditorFactory*” para el tipo de dato “*RiskList*” del modelo presentado en el epígrafe 2.3.

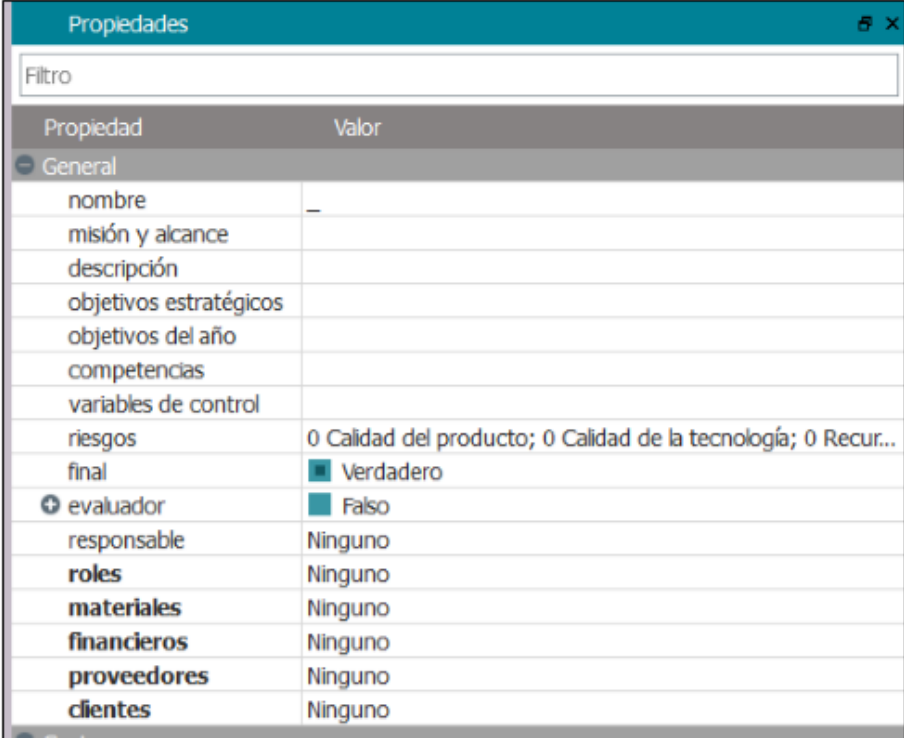
Variables de control

Este tipo de propiedad requirió de la implementación concreta de las clases “*ControlVariableListPropertyManager*” y “*ControlVariableListEditorFactory*” para el tipo de dato “*ControlVariableList*” del modelo presentado en el epígrafe 2.3.

Evaluador

Este tipo de propiedad requirió de la implementación concreta de las clases “*EvaluatorPropertyManager*” y “*EvaluatorEditorFactory*” para el tipo de dato “*Evaluator*” del modelo presentado en el epígrafe 2.3.

En la Figura 30 se presenta una imagen con las propiedades generales para el objeto gráfico: proceso/actividad.



Propiedad	Valor
Filtro	
General	
nombre	—
misión y alcance	
descripción	
objetivos estratégicos	
objetivos del año	
competencias	
variables de control	
riesgos	0 Calidad del producto; 0 Calidad de la tecnología; 0 Recur...
final	☒ Verdadero
evaluador	☒ Falso
responsable	Ninguno
roles	Ninguno
materiales	Ninguno
financieros	Ninguno
proveedores	Ninguno
clientes	Ninguno

Figura 30: Propiedades del grupo: general. Objeto gráfico: proceso/actividad

3.1.1.2 Objeto gráfico: segmento

Para este caso fue necesario definir una nueva propiedad: “*contenido*”. El contenido de un segmento varía en dependencia del tipo de flecha indicada respecto al proceso/actividad. En la Figura 31 se muestra para el caso de una relación de tipo recurso, en la interpretación semántica de IDEF0 indica que se trata de un recurso. Esto permite que en el inspector se visualice la propiedad *contenido* permitiendo especificar el recurso y la cantidad. El recurso es una referencia al diccionario de recursos.

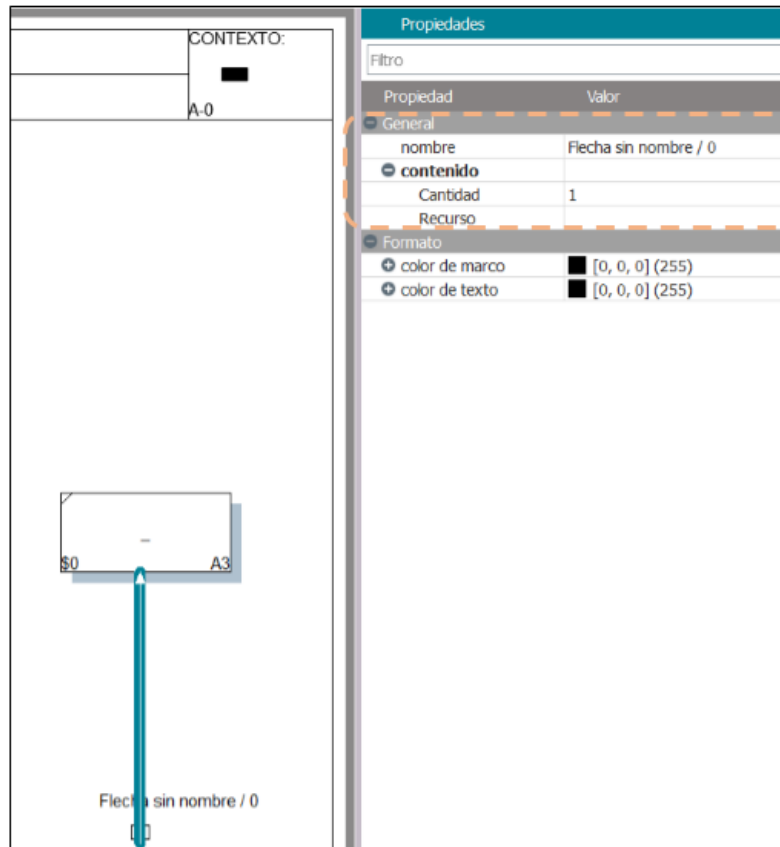


Figura 31: Propiedades generales del objeto gráfico segmento (caso “recurso”)

Otro caso es el de las relaciones que implican dato. Según la semántica de IDEF0 esto puede ocurrir en las entradas, salidas y control. En la Figura 32 se presenta una imagen de una relación de tipo dato. Al seleccionar la flecha en el inspector aparece habilitada la propiedad *contenido* que para este caso constituye una referencia a una entidad que esté en el diccionario de datos.

Este tipo de propiedad requirió de la implementación concreta de las clases “*ArrowContentPropertyManager*” y “*ArrowContentEditorFactory*” para el tipo de dato “*ArrowContent*” del modelo presentado en el epígrafe 2.3.

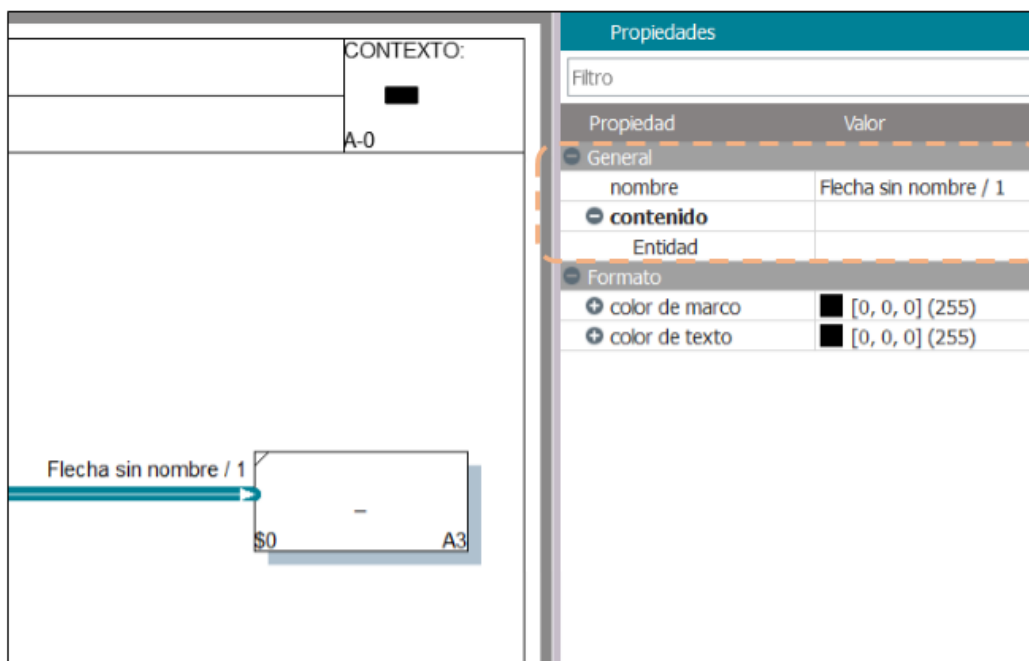


Figura 32: Propiedades generales del objeto gráfico segmento (caso “entidad de dato”)

3.1.2 Implementación de grupo: costo. Objeto gráfico: proceso/actividad

El grupo costo presenta elementos relacionados a la estimación del costo en el proceso/actividad, incluyendo propiedades como el costo total (que es la multiplicación del costo con la frecuencia), el costo, la determinación del tipo de cálculo (si se introduce o se calcula a través de la consideración de los recursos que utiliza el proceso según el modelo). La frecuencia, es un valor relativo, que significa que puede ser un número entero o fraccionario alegando por ejemplo que para una vez de ocurrencia del proceso padre, este ocurre una cierta cantidad de veces (por ejemplo 0,5 indicaría la mitad). La duración de la actividad se especifica con la etiqueta esclareciendo si es por horas (h) o por días (d).

En el modelo se puede observar como la clase proceso permite calcular el costo, teniendo en cuenta la asignación de recursos. Para los recursos materiales, realiza el cálculo teniendo en cuenta la cantidad y el precio unitario (definido para cada recurso material desde el diccionario de recursos). Para los recursos de tipo rol realiza el cálculo teniendo en cuenta la tasa por horas (definido para cada recurso de tipo rol desde el diccionario de recurso) y además la duración de la actividad (Ver Figura 33).

Costo	
costo total	0.00
costo	0.00
cálculo	Por Recursos
frecuencia	1.00
duración	0 h

Figura 33: Propiedades del grupo: costo

3.1.3 Implementación de grupo: dato. Objeto gráfico: proceso/actividad

En esta propiedad se detallan las entradas y salidas como elementos de tipo dato (Ver Figura 34), es decir si mediante las propiedades del segmento se especificó la referencia a un elemento del diccionario de datos, luego en esta propiedad del proceso/actividad es posible identificar ese dato como entrada o salida en correspondencia a lo representado en el modelo (notación basada en IDEF0).

Esta información no es editable, sino que muestra la información de esas entradas y salidas de tipo dato localizadas en el diagrama.

Dato	
entradas	Ninguno
salidas	Ninguno

Figura 34: Propiedades del grupo: dato

3.1.4 Implementación de grupo: formato

Estas son propiedades dentro de los tipos de datos básicos que define la librería, se emplean para la configuración de elementos como color del marco, color del texto.

Estas propiedades permiten la configuración visual del aspecto de los componentes del modelo ya sean procesos/actividades o los segmentos (Ver Figura 35).

Esto permite distinguir por ejemplo algunas relaciones, procesos/actividades.

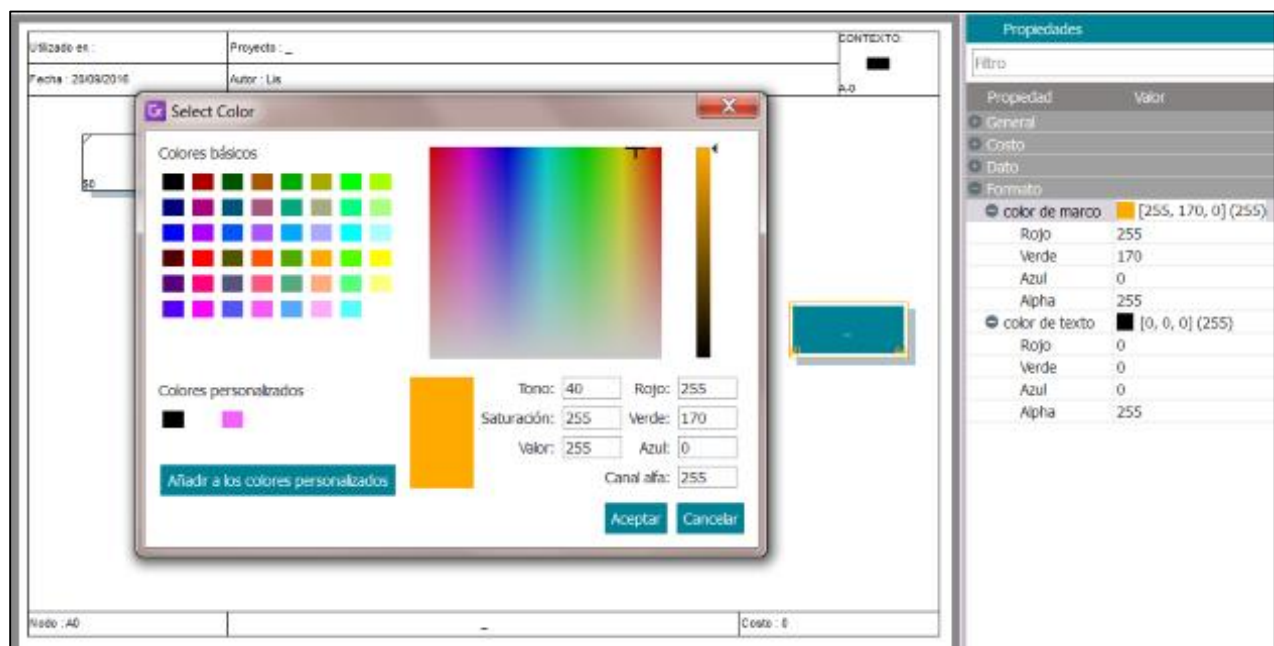


Figura 35: Propiedades del grupo: formato

3.2 Pruebas de calidad.

Las pruebas son actividades realizadas para evaluar y mejorar la calidad del producto, mediante la identificación de defectos y problemas (60). Fundamentalmente se precisaron de casos de pruebas que permitieran la observación de relación entre los componentes.

En la literatura (61) se ha planeado que cuando una aplicación es dividida en un grupo de componentes, se hace necesario pensar en cómo esos componentes comunican los datos y controlan la información.

Las etapas de pruebas llevadas a cabo se representan en la Figura 36, esta consideración está en concordancia a lo planteado en (60).

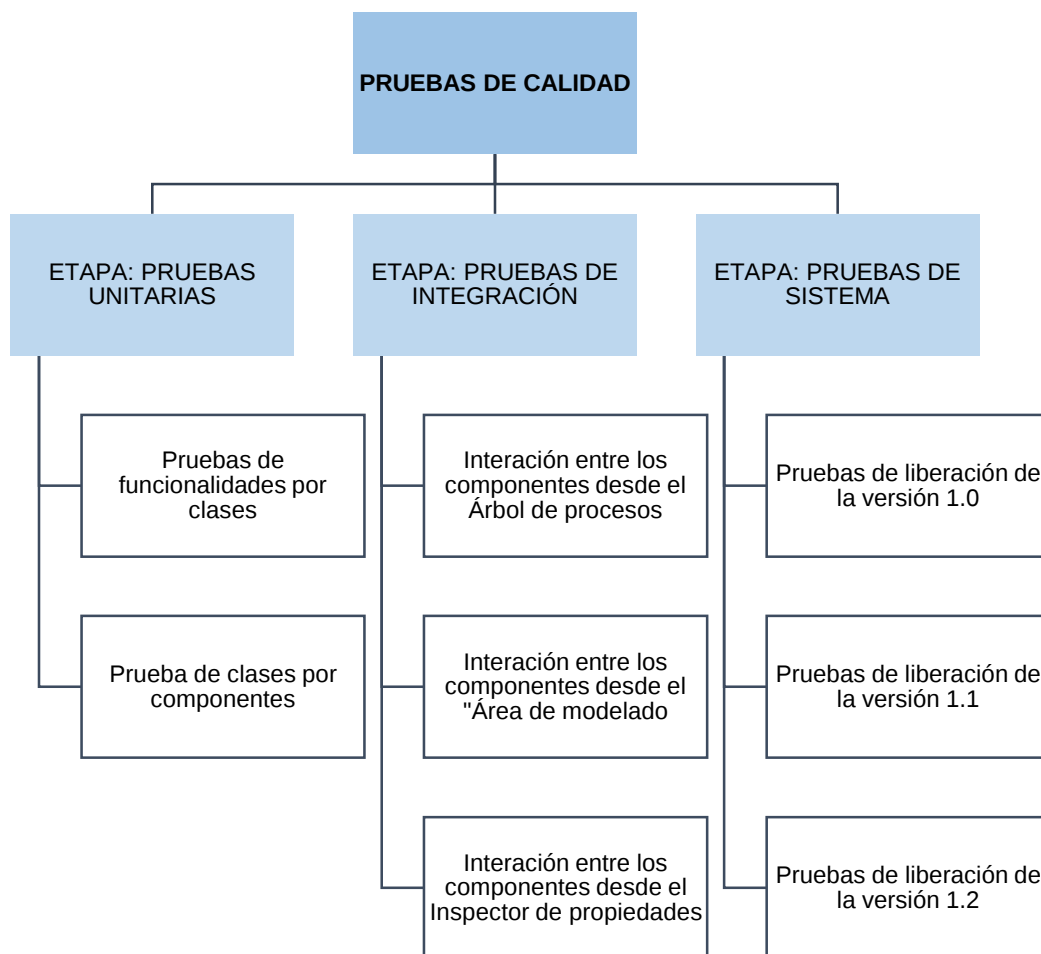


Figura 36: Etapas de las pruebas de calidad realizadas

Al escribirse el código de cada módulo se realizaron pruebas unitarias. En este caso, como se utiliza un lenguaje de programación orientado a objetos, las unidades se determinaron como una clase o una pequeña colección de clases (62).

Los casos de prueba se diseñaron en relación a las funcionalidades a probar. Los casos de prueba se cumplieron, y en caso de algún comportamiento no esperado se realizaba un *debug*, lo cual significa la localización de la causa de error.

Análisis estático: se aplicó la técnica de inspección de código a un nivel unitario, donde de una manera estática se realiza la localización de errores en el código sin llevar a cabo su ejecución. Se realizaron pruebas de caja blanca (estructural) enfocadas a la implementación, se probaron las diferentes estructuras de programación y de datos empleadas. Desde el punto de vista arquitectónico, se analizaron los componentes y las relaciones entre estos.

Análisis dinámico: con esta variante se analizó el software mediante la ejecución del código, o sea el ejecutable sirvió de base a la realización de análisis de casos para determinar defectos. Se tomaron por base casos de pruebas derivados de procesos de modelados reales del ámbito empresarial, para considerar diferentes alternativas. Por lo cual, las pruebas consistieron en la realización de esos casos de modelos a modo de reproducción de diagramas.

Se probó el software de una manera total, teniendo como base el documento de especificación de los requisitos. En Desoft Santi Spíritus se encuentra el laboratorio de pruebas de la región central de Desoft. El software fue sometido a pruebas de caja negra (funcional) de una manera completa. En el ANEXO 2 se muestran las conclusiones del informe de Resultados de Pruebas de CAPYROX, versión 1.

Relacionado a la calidad se aplicó un ciclo para la realización de las versiones, tomando por base el criterio de la aplicación del software en casos reales. Esta experiencia se realizó considerando el enfoque del ciclo de Deming (Planificar- Hacer –Verificar -Actuar). La información de aceptación del cliente, así como control de la satisfacción mediante entrevistas (Ver ANEXO 3) permitió profundizar en los aspectos funcionales y los atributos de calidad como la velocidad, la portabilidad, y la usabilidad.

3.3 Validación de la herramienta mediante la aplicación práctica en el tejido empresarial.

Se realizaron un grupo de aplicaciones en diferentes empresas teniendo en cuenta que existiesen diferencias según el tipo de ministerio y composición de su mapa de proceso en relación a su cantidad de procesos.

Dentro de estos casos se destaca la aplicación de CAPYROX en un proceso de Gestión del conocimiento, el cual demanda un alto grado de abstracción, centrado en el área de la gestión de recursos humanos. Este proceso ha sido definido en Desoft como clave por su relevancia. Para este caso era muy necesario lograr trabajar el diseño de este proceso en un equipo multidisciplinario bajo condiciones de un enfoque estratégico, sistémico, integrador y colaborativo.

3.3.1 Guía de capacitación y validación según atributos FURPS

Se realizó una guía de capacitación para apoyar el proceso de aplicación de la herramienta en las empresas. Esta guía de capacitación está compuesta por dos módulos (Ver Figura 37), un módulo básico de carácter estándar, aplicable a todos los casos y un módulo que se aplica con un carácter más personalizado de acuerdo al resultado de un diagnóstico.



Figura 37: Componentes del modelo de capacitación

Módulo básico de capacitación

Se diseñó un programa de curso para desplegados (**63**) y las guías de clases correspondientes (**64**). La propuesta se sometió a un proceso de revisión y aprobación por parte del Centro de Formación Ramal para la Informática de Desoft y posteriormente, se realizó una primera edición del curso en La Habana. La elaboración del asistente de CAPYROX se sometió a actualización considerando el resultado del curso, dado que fue incluido con mayor nivel de detalle y énfasis un grupo de informaciones según las principales preguntas del contenido del curso.

El manual de usuario forma parte de material de consulta del curso, siendo la principal bibliografía recomendada, para fortalecer el adiestramiento en el uso de esa guía, que además contribuye al auto estudio del desplegado.

Módulo de capacitación personalizado

El módulo de capacitación personalizado parte de un diagnóstico, con el objetivo de conocer el grado de modelado que tienen sus procesos en la actualidad, si existe un modelo conceptual y la verificación de las bases definidas (información de modelado). También se determina la necesidad de capacitación para conformar un plan de adiestramiento sobre el modelado de procesos con CAPYROX, y conciliar un posible apoyo mediante la operación asistida.

Validación según atributos FURPS

Bajo el acrónimo FURPS se consideran un conjunto de atributos de calidad: funcionalidad (*functionality*), usabilidad (*usability*), confiabilidad (*reliability*), desempeño (*performance*) y soporte (*supportability*).

Durante la impartición de la primera edición del curso CAPYROX en Desoft, donde participaron especialistas de diferentes provincias del país, se realizó una encuesta al final del programa para evaluar algunos atributos de calidad de la herramienta de acuerdo al modelo FURPS. La encuesta contenía los criterios mostrados en la Figura 38. Se utilizó una escala tipo Likert (65), en esta escala el valor más bajo era 1 y el más alto 5. Los resultados obtenidos fueron procesados con el paquete estadístico SPSS, en la Tabla 4 se muestra una imagen basada en el procesamiento con SPSS.

Al revisar la media, se puede constatar que los valores están por encima de 4.60. La varianza, como medida cuadrática de variabilidad, se observa en mayores valores para Interfaz de usuario y Rendimiento (Ver Tabla 4).

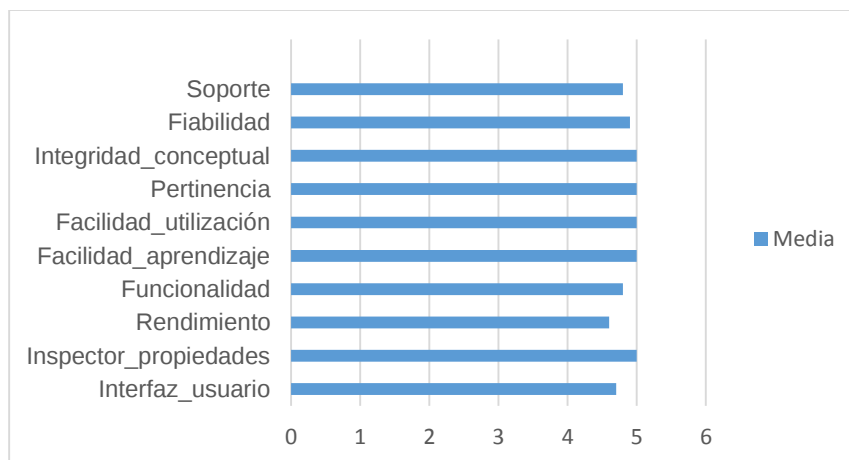


Figura 38: Valores de la media por variables

	N	Mínimo	Máximo	Media	Varianza
Interfaz_usuario	10	3	5	4.70	.456
Inspector_propiedades	10	5	5	5.00	.000
Rendimiento	10	4	5	4.60	.267
Funcionalidad	10	4	5	4.80	.178
Facilidad_aprendizaje	10	5	5	5.00	.000
Facilidad_utilización	10	5	5	5.00	.000
Pertinencia	10	5	5	5.00	.000
Integridad_conceptual	10	5	5	5.00	.000
Fiabilidad	10	4	5	4.90	.100
Soporte	10	4	5	4.80	.178

Tabla 4: Estadística descriptiva

Teniendo en cuenta los criterios de la varianza donde se observan los mayores valores para Interfaz de usuario y Rendimiento, el grupo de análisis logro identificar las siguientes causas. Estas causas se analizaron y tomaron medidas al respecto que se reflejaron como cambios en la versión siguiente de la herramienta. Interfaz de usuario: Debía tener un aspecto con mayor color, según las tendencias actuales de diseños (En la versión siguiente se analizó y realizó una actualización de la interfaz mediante un .css). Rendimiento: La velocidad de procesamiento de cambios gráficos podía cambiar de acuerdo al aumento considerable del tamaño del modelo general (Se hicieron nuevas pruebas de carga, y en la versión siguiente se validó otra vez este aspecto).

3.3.2 Casos de aplicación y validación basada en la NC ISO/IEC 9126

Casos de aplicación

Se realizaron casos de aplicación de la herramienta en la Dirección de Gestión del Conocimiento de la Empresa Desoft, MINCOM; Complejo Hotelero Rancho Luna – Faro de Luna de Gran Caribe, MINTUR; Empresa de Bebidas y Refrescos de Villa Clara, MINAL. Se ha llevado a cabo la realización de dos actividades núcleos fundamentales dentro del enfoque de procesos: el modelado del proceso y su instanciación.

La aplicación en la Dirección de Gestión del Conocimiento de la Empresa Desoft se basó en el modelado del proceso de Gestión del Conocimiento (GC) (Ver ANEXO 4), siendo este un proceso identificado como clave. En el Complejo Hotelero Rancho Luna – Faro de Luna de Gran Caribe (Ver ANEXO 5) y en la Empresa de Bebidas y Refrescos de Villa Clara, se realizó el modelado de todos los procesos de la organización (estratégicos, operáticos y de apoyo),

Durante el modelado, CAPYROX logra estimular los debates de manera concreta, para proponer mejoras en los niveles de diseño y conceptualización de los procesos, lo cual apoya el correcto funcionamiento de los procesos de acuerdo a los objetivos y metas trazadas en la organización y el adecuado nivel de control que deben ejercer las partes involucradas **(46)**. Se realizó un análisis sobre el comportamiento de CAPYROX considerando un conjunto de variables significativas provenientes del modelo NC ISO/IEC 9126 **(66)** como estándar para la calidad del producto software.

La experiencia implicó la realización de un proceso de modelado según el modelo de trabajo visualizado en la Figura 39. Este modelo primeramente se identificó y definió durante la experiencia en el modelado del proceso GC **(46)**, pero se pudo constatar su posible aplicación para otros procesos diferentes a GC. La definición de un proceso de GC está relacionado con muchos componentes tangibles e intangibles y sus necesarios indicadores para poder controlar el desempeño según lo planificado y deseado. La experiencia de modelado de un proceso de este tipo permitió identificar un grupo de fases, componentes y subprocesos de manera integrada **(46)**.

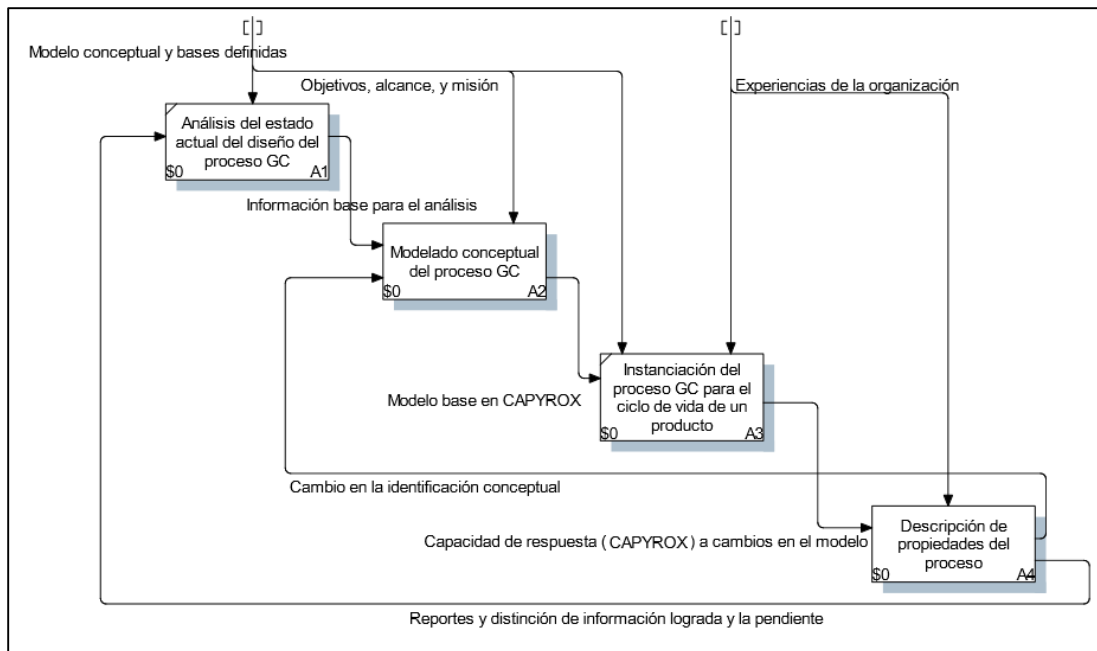


Figura 39: Modelo de trabajo con CAPYROX para la definición del proceso GC (46)

En (46) se expuso que la transición del análisis del estado actual del diseño del proceso GC al modelado conceptual, para la definición de un modelo de procesos utilizando CAPYROX facilitó lo siguiente:

- Concretar los aspectos y las fases del modelo conceptual preliminar en un modelo de procesos con las actividades necesarias según los objetivos previstos.
- Analizar el modelo obtenido en consecuencia con el enfoque estratégico, participativo y sistémico.
- Identificar las responsabilidades por actividades dentro del proceso y del proceso en sí.
- Considerar un grupo de mecanismos de control teniendo en cuenta las actividades necesarias a realizar dentro del proceso.

Sobre estas experiencias se pudo concluir lo siguiente (46):

- CAPYROX permite una fácil comprensibilidad y estimula la comunicación y el intercambio de ideas en equipo lo cual se validó durante las experiencias del modelado del proceso de GC.
- Los elementos de la notación complementados con la información del inspector de propiedades permiten lograr la representación fundamental de base para el enfoque de procesos, incluyendo la representación de las interdependencias, responsabilidades y mecanismos de control entre otros aspectos.
- La herramienta posibilita trabajar desde un enfoque estratégico, sistémico, integrador y colaborativo.

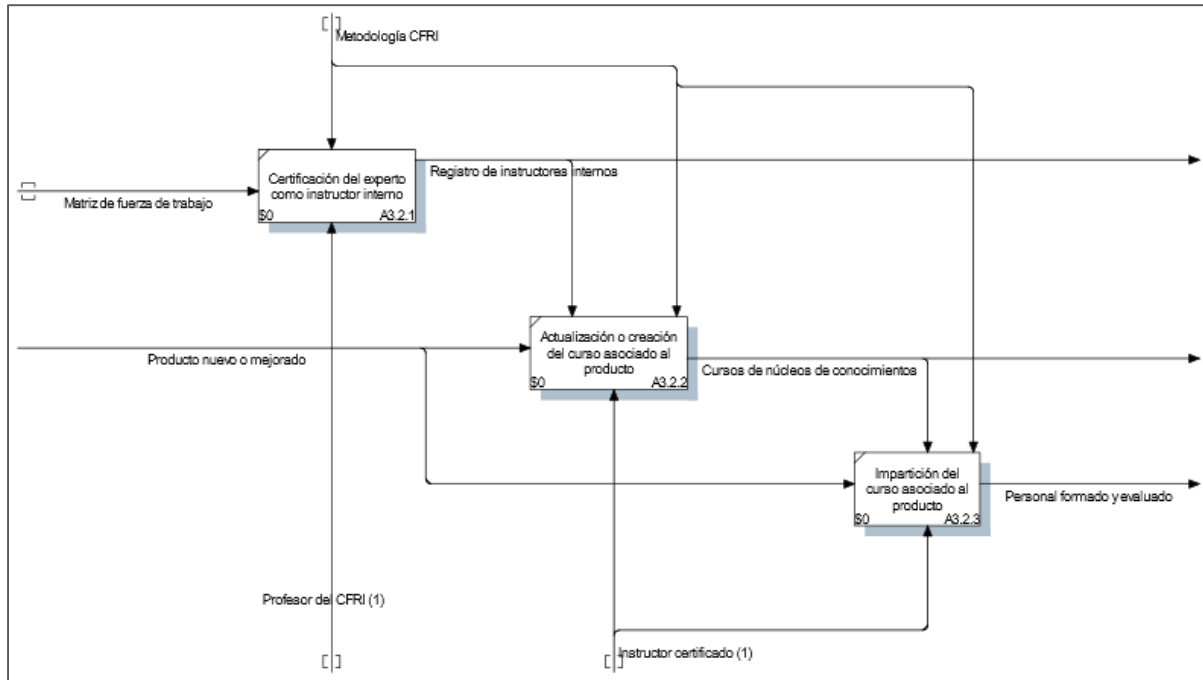


Figura 40: Diagrama del subproceso “Organización de cursos”, proceso de GC (46)

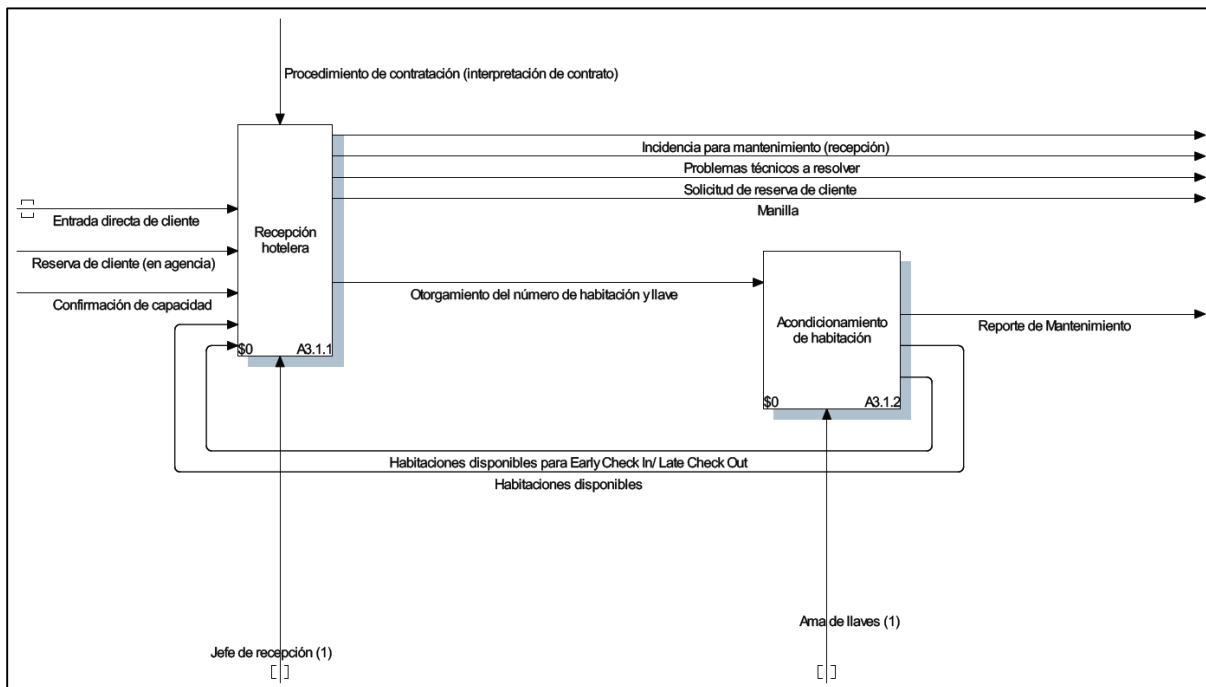


Figura 41: Proceso “Prestación del servicio de alojamiento” (etapa inicial del modelado)

En las Figura 40 y 41 se presentan, a modo de ejemplo, dos de los diagramas generados por CAPYROX. La imagen 40 ejemplifica la identificación de un grupo de actividades para la organización de

cursos dentro del proceso de GC. La imagen 41 muestra el diagrama del proceso Prestación del servicio de alojamiento (imagen en una etapa inicial del modelado).

Sobre la utilización de las propiedades durante la aplicación práctica de la herramienta fue posible diagnosticar lo siguiente. Mediante la aplicación del modelo de trabajo con CAPYROX presentado en la Figura 39 se aplican las propiedades definidas, con un comportamiento de usabilidad presentado en la Figura 42. De esa manera fue posible pasar a un ejercicio definitorio de una validación basada en la NC ISO/IEC 9126, teniendo en cuenta la utilización plena del mecanismo de inspección de propiedades.



Figura 42: Representación del grado de utilización de los contenidos del inspector de propiedades durante el trabajo de modelado del proceso de GC con CAPYROX (46)

Validación basada en la NC ISO/IEC 9126

En los tres casos se llevaron a cabo ejercicios de validación tomando como base de análisis los aspectos definidos en la NC ISO/IEC 9126 (66). En modelos de calidad para la calidad interna y externa, como el abordado en la NC ISO 9126 (66), el nivel de comprensibilidad está definido sub-característica de la usabilidad; y la usabilidad es una de las seis características en las que se categoriza los atributos de calidad del software. Ese estándar define la usabilidad como la capacidad del producto de software de ser comprendido, aprendido, utilizado y de ser atractivo para el usuario, cuando se utilice bajo las condiciones especificadas. La comprensibilidad es identificada como la capacidad del producto de software para permitirle al usuario entender si el software es idóneo, y cómo puede usarse para las tareas y condiciones de uso particulares (Ver Tabla 5).

Atributo de calidad	Resultado
Funcionabilidad	El software cuenta con funciones que permiten una respuesta eficaz durante el diseño de los procesos y apoyo en el análisis de mejora. En las actividades desarrolladas se ha evidenciado la correcta operación en general.
Confiabilidad	El software ha mantenido el nivel de ejecución previsto con un desempeño correcto.
Usabilidad	Los elementos para el trabajo con la herramienta han resultado de fácil comprensibilidad, basada en una notación que proporciona alto nivel de detalle sin complejizar las funciones de modelado para los diagramas. De acuerdo a esto se ha percibido un fácil entendimiento de las acciones de modelado que se han realizado, propiciando la comunicación y el intercambio de ideas con los jefes de procesos y otros miembros involucrados.
Eficiencia	Se ha logrado constatar un adecuado desempeño durante su ejecución en correspondencia a los recursos utilizados, considerando el hardware y materiales que posibilita optimizar ya que brinda visualizaciones y la posibilidad de obtener reportes en formato digital. Los tiempos de respuesta han sido positivos.
Mantenibilidad	Se ha podido constatar una positiva mantenibilidad, considerando la posibilidad de mejora y adaptaciones del software, evidenciándose una buena estabilidad.
Portabilidad	La herramienta responde adecuadamente en diferentes sistemas operativos, pues se han realizado presentaciones sobre sus funcionalidades tomando como base ilustraciones en alternativas como Ubuntu y Debian manteniendo un ambiente visual muy parecido. Los pasos de instalación mostrados resultan de fácil comprensibilidad.

Tabla 5: Validación considerando los aspectos definidos en la NC ISO/IEC 9126

3.4 Conclusiones del capítulo

En el presente capítulo se identificaron elementos relacionados a la elaboración del mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura de manera que pudiese lograr un modo de interacción adecuado para un modelador de procesos orientado al valor. Se logra validar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura mediante su implementación en un modelador de procesos orientado al valor y su capacidad para facilitar la comprensibilidad de los modelos. Es posible arribar a las siguientes conclusiones:

1. Se pudo implementar y validar el mecanismo de inspección de las propiedades asegurando que cumple con atributos de calidad como los incluidos en FURPS y los presentados por NC ISO/IEC 9126, lo cual incluye el aspecto de la comprensibilidad.
2. Se realizaron pruebas de calidad y aplicaciones prácticas que demuestran la consistencia de la arquitectura y la respuesta adecuada a las funcionalidades concebidas para lograr la integración de los componentes gráficos con los grupos del inspector de propiedades.

CONCLUSIONES

De manera general, se pudo definir un mecanismo para la inspección de propiedades estructurales en una arquitectura de un modelador de procesos orientado al valor que toma por base la notación IDEF0 para facilitar la comprensibilidad de los modelos, específicamente en la herramienta CAPYROX.

La elaboración del marco teórico sobre los mecanismos para la inspección de propiedades estructurales durante el modelado de procesos de negocio permitió reconocer cómo la gestión de procesos de negocio contribuye a la mejora organizacional y se encuentra en muy estrecha relación con la calidad y su gestión, para lo cual los modelos de procesos contribuyen brindando comprensibilidad.

Dentro de las herramientas de modelado analizadas, que se corresponden con las más utilizadas, no se identificó una que soportase IDEF0 con buena adaptabilidad para asumir las propiedades propuestas en **(1)** en relación al modelado de procesos orientado al valor, que incluye esenciales atributos de calidad. También fue posible precisar que la comprensibilidad es un atributo de calidad transcendental para los modelos de procesos y está relacionada a aspectos claves como: dimensiones cognoscitivas, y muchas veces se encuentra afectada en modelos largos y complejos.

Se pudo elaborar y validar el mecanismo para la inspección de propiedades estructurales y los componentes necesarios en la arquitectura mediante su implementación en un modelador de procesos orientado al valor evidenciándose su capacidad para facilitar la comprensibilidad de los modelos. Se destaca que la estructura del modelo presenta una relación de perspectivas que logra intercalar adecuadamente el control gráfico en procesos y segmentos con el manejo de referencias a los diccionarios desde las propiedades. La vista estática y funcional, a través de los patrones empleados, permite percibir una adecuada correspondencia entre las acciones del usuario y la guía que propicia el inspector de propiedades, justificando su apoyo a la comprensibilidad mediante la localización gráfica de elementos ya sea desde el área de modelado o del explorador de proyectos. Se ha mantenido un equilibrio con el valor semántico de los componentes de la notación IDEF0. Fue posible validar el mecanismo de inspección de las propiedades asegurando que cumple con atributos de calidad como los incluidos en FURPS y los presentados por NC ISO/IEC 9126, lo cual incluye el aspecto de la comprensibilidad.

Teniendo en cuenta todo lo anteriormente planteado, los ejercicios prácticos llevados a cabo junto a los resultados positivos de las pruebas de calidad unitarias, de integración y de sistema, fue posible evidenciar que la utilización de un mecanismo para la inspección de propiedades estructurales dentro de la arquitectura en un modelador de procesos orientado al valor que toma por base la notación IDEF0 favorece la comprensibilidad como atributo de calidad en el modelo de procesos resultante.

RECOMENDACIONES

Se propone continuar con sucesivas evoluciones del mecanismo de inspección de propiedades desarrollado de modo que se posibilite analizar además otros factores propios de la gestión de los procesos en el área organizacional, además de los aspectos de diseño de procesos que se han concebido dentro del alcance de la presente investigación. Esto se propone teniendo en cuenta que la NC ISO 9001:2015 identifica que el enfoque a procesos implica no solo la definición sino también la gestión sistemática de los procesos y sus interacciones.

Este aspecto podrá verse reflejado en la posible complementación o conexión de la herramienta a otras aplicaciones que aborden la evaluación de las variables de control en el desempeño de los procesos.

BIBLIOGRAFÍA

1. *Enfoque analítico para la fase de planificación en el ciclo de vida de desarrollo de sistemas informáticos*. **Pérez Quintero, Lisett, Carrera Ortega, Jorge y García Pérez, Ana María**. 2016, Informática 2016. "IV Taller Internacional: Las TIC en la Gestión de las Organizaciones".
2. *CAPYROX: Una herramienta útil para el modelado de procesos orientado al valor*. **Carrera Ortega, Jorge y Pérez Quintero, Lisett**. 2016, Informática 2016. "IV Taller Internacional: Las TIC en la Gestión de las Organizaciones".
3. **Weske, Mathias**. *Business Process Management. Concepts, Languages, Architectures*. Berlin : Springer, 2007. 978-3-540-73521-2.
4. *A conceptual modeling quality framework*. **Nelson, H. James , y otros**. s.l. : Springer Science+Business Media, 2011, Software Quality Journal, Vol. 20, págs. 201-228.
5. *A Study into the Factors that Influence the Understandability of Business Process Models*. **Reijers, Hajo A. y Jan, Mendling**. 2011, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, Vol. 41, págs. 449-462.
6. *A systematic literature review of studies on business process modeling quality*. **Moreno-Montes de Oca, Isel, y otros**. 2015, Information and Software Technology, Vol. 58, págs. 187-205.
7. *Which Aesthetic has the Greatest effect on human understanding*. **Purchase, Helen**. Rome, Italy : Springer, 1997, Graph Drawing: 5th International Symposium, Gd'9, págs. 248–261.
8. **Reijers, H.A. y Mendling, J**. Modularity in Process Models: Review and Effects. [ed.] M. Dumas, M. Reichert y M.-C. Shan. *International Conference on Business Process Management*. Milan, Italy : Springer, 2008, Vol. 5240, págs. 20–35.
9. **Sharp, Alec y McDermott, Patrick**. *Workflow Modelling: Tools for Process Improvement and Application Development. Second Edition*. London : ARTECH HOUSE, INC., 2009. 978-1-59693-192-3.
10. *Integrated framework for business process complexity analysis*. **Setiawan, Mukhammad Andri y Sadiq, Shazia**. 2013, Proceedings of the 21st European Conference on Information Systems.
11. *Value-Oriented Process Modelling. Towards a Financial Perspective on Business Process Redesign*. **Brocke, Jan vom , Mendling, Jan y Recker, Jan**. Toronto, ON, Canada : University of Liechtenstein, 14th-17th de August de 2008, Paper presented at the 14th Americas Conference on Information Systems (AMCIS 2008).
12. **Brocke, Jan Vom y Sonnenberg, Christian**. Value-Oriented in Business Process Management. *Handbook on Business Process Management 2. Strategic Alignment, Governance, People and Culture*. Heidelberg : Springer, 2015.
13. *Formal verification of static software models in MDE: A systematic review*. **González, Carlos A. y Cabot, Jordi**. 2014, Information and Software Technology, Vol. 56, págs. 821-838.
14. **Barker, Robert M, y otros**. Tools for Building Information Systems. [ed.] Gavriel Salvendy. *Handbook of Industrial Engineering: Technology and Operations Management*. Third Edition. Toronto : John Wiley & Sons, Inc, 2001, 3.
15. **International Institute of Business Analysis, IIBA**. *The Guide to the Business Analysis Body of Knowledge. Version 2.0 BABOK®*. 2008.

16. **Oficina Nacional de Normalización (NC).** Sistemas de gestión de la calidad-Fundamentos y vocabulario. [ISO 9000:2015 (Traducción certificada), IDT]. Ciudad de La Habana : Oficina Nacional de Normalización (NC), 2015.
17. —. *Sistemas de Gestión de la Calidad - Requisitos [ISO 9001:2015 (Traducción certificada), IDT]*. Ciudad de La Habana : Oficina Nacional de Normalización (NC), 2015.
18. **O. Marrs, Frank y M. Mundt, Barry.** Enterprise Concept: Business Modeling Analysis and Design. [ed.] Gavriel Salvendy. *Handbook of Industrial Engineering: Technology and Operations Management, Third Edition*. s.l. : John Wiley & Sons, 2001.
19. **Sánchez-González, Laura, y otros.** Improving Quality of Business Process Models. *Evaluation of Novel Approaches to Software Engineering, 6th International Conference, ENASE 2011*. s.l. : Springer, 2013, págs. 130-144.
20. *Tying process model quality to the modeling process: the impact of structuring, movement, and speed.* **Claes, Jan, y otros.** 2012, Business Process Management International Conference, págs. 33-48.
21. *A Contribution to Organizational and Operational Strategic Alignment: Incorporating Business Level Agreements into Business Process Modeling.* **Salles, Guilherme , y otros.** 2013, IEEE 10th International Conference on Services Computing.
22. *Factors of process model comprehension-Findings from a series of experiments.* **Mendling, J., Strembeck, M. y Recker, J.** 2012, Decision Support Systems, Vol. 53, págs. 195-206.
23. *Seven Process Modeling Guidelines.* **Mendling, J., Reijers, H.A. y van der Aalst, W.M.P.** 2, s.l. : Elsevier, 2010, Information and Software Technology, Vol. 52, págs. 127-136.
24. *Activity labeling in process modeling: Empirical insights and recommendations.* **Mendling, J., Reijers, H.A. y Recker, J.** 4, 2010, Information Systems, Vol. 35, págs. 467-482.
25. *Quality indicators for business process models from a gateway complexity perspective.* **Sánchez-González, Laura, y otros.** 2012, Information & Software Technology, Vol. 54, págs. 1159-1174.
26. *Syntax highlighting in business process models.* **Reijers, H.A., y otros.** 2011, Decision Support Systems, Vol. 51, págs. 339-349.
27. **Polyvyanyy, Artem, Smirnov, Sergey y Weske, Mathias.** Business Process Model Abstraction. [ed.] Jan vom Brocke y Michael Rosemann. *Handbook on Business Process Management 1. Introduction, Methods, and Information Systems*. s.l. : Springer, 2015.
28. *The Effects of Content Presentation Format and User Characteristics on Novice Developers' Understanding of Process Models.* **Recker, Jan y Dreiling, Alexander.** 2011, Communications of the Association for Information System, Vol. 28.
29. **Decker, Gero y Mendling, Jan.** Instantiation Semantics for Process Models. *Business Process Management. 6th International Conference, BPM 2008*. Milan : Springer-Verlag Berlin Heidelberg, 2008, págs. 164-179.
30. **Visual Paradigm.** [En línea] <https://www.visual-paradigm.com/>.
31. **Bizagi.** [En línea] www.bizagi.com/.
32. **Aris.** [En línea] <http://www.ariscommunity.com/>.
33. **Intalio.** [En línea] www.intalio.com/.

34. **TIBCO**. [En línea] www.tibco.com/.
35. **ConceptDraw**. [En línea] conceptdraw.com/.
36. **Microsoft Office**. [En línea] products.office.com/.
37. **CA**. [En línea] www.ca.com/.
38. *On the Measurement of Business Process Centralization*. **Modrák, Vladimir**. [ed.] Elsevier. 2013, Procedia Technology. CENTERIS 2013 - Conference on ENTERprise Information Systems / PROjMAN 2013 - International Conference on Project MANagement / HCIST 2013 - International Conference on Health and Social Care Information Systems and Technologies, Vol. 9, págs. 547-552.
39. *On Measuring the Understandability of Process Models (Experimental Results)*. **Melcher, Joachim, y otros**. 2009, Business Process Management Workshops, 2009.
40. *On the Usage of Labels and Icons in Business Process Modeling*. **Mendling, Jan, Recker, Jan y A. Reijers, Hajo**. 2010, International Journal of Information System, Modeling and Design, Vol. 1, págs. 40-58.
41. *Detecting structural errors in BPMN process models*. **Mohammed Kherbouche, Oussama, Ahmad, Adeel y Basson, Henri**. 2012, 5th International Multitopic Conference, págs. 425-431.
42. *Cognitive Complexity in Business Process Modeling*. **Figl, Kathrin y Laue, Ralf**. 2011, 23 International Conference on Advanced Information Systems Engineering (CAISE 2011), Vol. 6741, págs. 452-466.
43. *Managing Process Model Complexity via Concrete Syntax Modifications*. **La Rosa, Marcello, y otros**. [ed.] IEEE. 2, 2011, IEEE Transactions on Industrial Informatics, Vol. 7.
44. *Perspectives on Process Modeling*. **Burlton, Roger**. July de 2009, www.bptrends.com.
45. *Arquitectura para diseñar soluciones de inteligencia de negocio*. **Marín-Ortega, Pablo Michel , García-Ávila, Lourdes y Marx Gómez, Jorge**. 1, 2013, Nueva Empresa, Vol. 9, págs. 34-44.
46. *Herramienta CAPYROX: su aplicación en la definición de un proceso de gestión del conocimiento*. **Pérez Quintero, Lisett, y otros**. Santa Clara : s.n., 2016, Congreso Internacional de Información. INFO 2016.
47. *Automation of strategy using IDEF0 - A proof of concept*. **Waissi, Gary R., Demir, Mustafa y Humble, Jane E**. s.l. : Elsevier, 2015, Operations Research Perspectives, Vol. 2, págs. 106-113.
48. *Mapping Strategic Goals and Operational Performance Metrics for Smart Manufacturing Systems*. **Jung, Kiwook, y otros**. 2015, Procedia Computer Science. 2015 Conference on Systems Engineering Research, Vol. 44, págs. 184-193.
49. *A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process*. **Galvan, Sergio, y otros**. 2015, Procedia Computer Science. Conference on ENTERprise Information Systems / International Conference on Project, MANagement / Conference on Health and Social Care Information Systems and Technologies, CENTERIS / ProjMAN / HCist 2015 October 7-9, 2015, Vol. 64, págs. 188-195.
50. *Formal Concept Analysis – Overview and Applications*. **Skopljanac -Macina, Frano y Blaskovic, Bruno**. s.l. : Elsevier, 2014, Procedia Engineering. 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013, Vol. 69, págs. 1258 - 1267.

51. *Human and automatic modularizations of process models to enhance their comprehension.* **Reijers, H.A., Mendling, J. y Dijkman, R.M.** s.l. : Elsevier, 2011, Information Systems, Vols. 881–897, págs. 881–897.
52. *Mapas cognitivos difusos para la selección de proyectos de tecnologías de la información.* **Leyva Vázquez, Maikel Y., y otros.** 4, 2013, Contaduría y Administración, Vol. 58, págs. 95-117.
53. **Gómez, Martha.** No es cuestión de moda. *Economía. Separata de Trabajadores.* No 14. 25 de Mayo de 2015.
54. **Proyecto Tuning- América Latina.** *Reflexiones y perspectivas de la Educación Superior en América Latina. Informe Final- Proyecto Tuning- América Latina 2004-2007.* Bilbao : Publicaciones de la Universidad de Deusto, 2007. 978-84-9830-078-9.
55. **Alles, Martha Alicia.** *Desempeño por Competencias. Evaluación 360 grados.* Buenos Aires : Granica, 2005. pág. 36. 950 641 879 9.
56. **Pérez Quintero, Lisett y Carrera Ortega, Jorge.** *Manual de usuario. CAPYROX 1.2.* Santa Clara : Desoft, 2016.
57. **Brass, Peter.** *Advanced Data Structures.* New York : Cambridge University Press, 2008.
58. **The Qt Company.** *Qt Documentation.* [En línea] 2016. <http://doc.qt.io/qt-4.8/>.
59. *The Qt object model and the signal slot concept.* **Nokia Corporation and its Subsidiary(-ies).** 2010.
60. **A project of the IEEE Computer Society.** *Guide to the Software Engineering Body of Knowledge, SWEBOK®.* California : Institute of Electrical and Electronics Engineers, 2004. ISBN 0-7695-2330-7.
61. **Gorton, Ian.** *Essential Software Architecture.* Berlin : Springer, 2006.
62. **Jalote, Pankaj.** *A Concise Introduction to Software Engineering.* London : Springer, 2008.
63. **Pérez Quintero, Lisett y Carrera Ortega, Jorge.** *Programa de curso CAPYROX.* . Villa Clara. : DESOFT, 2015.
64. —. *Guía de clases del curso: CAPYROX.* Villa Clara. : DESOFT, 2015.
65. **Hernández Sampieri, Roberto, Fernández Collado, Carlos y Baptista Lucio, María del Pilar.** *Metodología de la investigación. Quinta Edición.* México : McGrawHill, 2010.
66. *Ingeniería de software-Calidad del producto-Parte 1: Modelo de la calidad (ISO/IEC 9126-1: 2001, IDT).* **Oficina Nacional de Normalización (NC).** La Habana : Oficina Nacional de Normalización (NC), 2005.
67. *Managing Process Model Complexity via Abstract Syntax Modifications.* **La Rosa, M., y otros.** 2011, Ieee Transactions on Industrial Informatics.
68. *A quality based approach for the analysis and design of business process models.* **Ayad, Sarah.** 2012, International Conference on Research Challenges in Information Science (RCIS 2012).
69. *The Model View Framework.* **Nokia Corporation and its Subsidiary(-ies).** 2010.
70. **Ezust, Alan y Ezust, Paul.** *An Introduction to Design Patterns in C++ with Qt 4.* s.l. : Prentice Hall, 2006.
71. *Refactoring large process model repositories.* **Weber, Barbara , y otros.** s.l. : Elsevier, 2011, Computers in Industry, Vol. 62, págs. 467–486.

72. *What business process modelers can learn from programmers.* **Gruhn, Volker y Laue, Ralf.** 2007, Science of Computer Programmin, Vol. 65, págs. 4-13.
73. *Influence Factors of Understanding Business Process Models.* **Mendling, Jan y Strembeck, Mark.** [ed.] Springer. Berlin : s.n., 2008, International Conference on Business Information Systems, Vol. 7, págs. 142-153.
74. *Recommendation Based Process Modeling Support: Method and User Experience.* **Hornung, Thomas, Koschmider, Agnes y Lausen, Georg.** 2008, Conceptual Modeling Conference (ER 2008).
75. *A discourse on complexity of process models.* **Cardoso, J., y otros.** Berlin : Springer-Verlag Berlin, 2006, Business Process Management Workshops (BPM 2006), págs. 115-126.
76. *Complexity metrics for measuring the understandability and maintainability of business process models using goal-question-metric (GQM).* **Abdul Ghani , Abdul Azim , Wei Geoffrey Muchiri Muketha , Koh Tieng y Pei Wen, Wong.** 5, 2008, IJCSNS International Journal of Computer Science and Network Security, Vol. 8.
77. *Complexity metrics for Workflow nets.* **Bisgaard Lassen, Kristian y van der Aalst, Wil M.P. . s.l. :** Elsevier, 2008, Information and Software Technology, Vol. 51, págs. 610-626.
78. *Applying Software Metrics to evaluate Business Process Models.* **Rolón, Elvira, y otros.** 1, 2006, CLEI-Electronic Journal, Vol. 9.
79. *Prediction models for BPMN usability and maintainability.* **Rolón, Elvira, y otros.** 2009, 2009 IEEE Conference on Commerce and Enterprise Computing.
80. *Optimization of the business process metrics definition according to the bpdms standard and its formal definition in OCL.* **Debnath, N., y otros.** 2010, International Conference on Computer Systems and Applications (AICCSA 2010).
81. *Quality Assessment of Business Process Models Based on Thresholds.* **Sánchez-González, Laura, y otros.** Berlin : Springer, 2010, págs. 78-95.
82. *2 Approaches for business process model complexity metrics.* **Gruhn, Volker y Laue, Ralf.** s.l. : Springer, 2007, Technologies for Business Information Systems, págs. 13-24.
83. *Structuredness and its significance for correctness of process models.* **Laue, Ralf y Mendling, Jan.** s.l. : Springer, 2010, Information Systems and E-Business Management, Vol. 8, págs. 287-307.
84. *On a quest for good process models: The cross-connectivity metric.* **Vanderfeesten, Irene, y otros.** Berlin : Springer, 2008, International Conference on Advanced Information Systems Engineering, págs. 480-494.
85. *What makes process models understandable?* **Mendling, Jan, Reijers, Hajo A. y Cardoso, Jorge.** s.l. : Springer, 2007, International Conference on Business Process Management, págs. 48-63.
86. *Evaluation of BPMN models quality – A family of experiments.* **Rolón, Elvira, y otros.** 2008, International Conference on Evaluation of Novel Approaches to Software Engineering, págs. 56-63.
87. *Understanding Business Process Models: The Costs and Benefits of Structuredness.* **Dumas, Marlon, y otros.** 2012, International Conference on Advanced Information Systems Engineering, págs. 31-46.
88. *Error metrics for business process models.* **Mendling, Jan y Neumann, Gustaf.** 2007, International Conference on Advanced Information Systems Engineering.

89. *A weighted coupling metric for business process models*. **Vanderfeesten, Irene, Cardoso, Jorge y Reijers, Hajo A.** 2007, International Conference on Advanced Information Systems Engineering, Vol. 7, págs. 41-44.
90. *The Impact of Activity Labeling Styles on Process Model Quality*. **Mendling, Jan y Reijers, Hajo.** 2008, AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems, Vol. 129, págs. 117-128.
91. *Analysis and Validation of Control-Flow Complexity Measures with BPMN Process Models*. **Rolón, Elvira, y otros.** s.l.: Springer, 2009, International Workshop on Business Process Modeling, Development and Support, págs. 58-70.
92. *The impact of secondary notation on process model understanding*. **Schrepfer, Matthias, y otros.** 2009, The Practice of Enterprise Modeling Working Conference, Vol. 39, págs. 161-175.
93. *Business process illustration: supporting experience-grounded validation of new business processes by subject matter experts*. **Stolze, Markus.** 2008, IEEE Conference on E-Commerce Technology.
94. *Reducing the Cognitive Complexity of Business Process Models*. **Gruhn, Volker y Laue, Ralf.** 2009, International Conference on Cognitive Informatics, págs. 339-345.
95. *Using Glossaries to Enhance the Label Quality in Business Process Models*. **Peters, Nicolas y Weidlich, Matthias.** 2009, págs. 75-90.
96. *An Exploratory Experiment to Validate Measures for Business Process Models*. **Rolón, Elvira, y otros.** 2007, International Conference on Research Challenges in Information Science (RCIS 2007).
97. *Good and Bad Excuses for Unstructured Business Process Models*. **Gruhn, Volker y Laue, Ralf.** 2007, European conference on pattern languages of programs, págs. 279-292.
98. *Gestión estratégica de tecnologías de información y comunicación y adopción del comercio electrónico en Mipymes de Córdoba, Argentina*. **Jones, Carola, Motta, Jorge y Alderete, María Verónica.** s.l. : Elsevier, 2016, Estudios gerenciales, Vol. 32, págs. 4-13.
99. *Adiestramiento: Las normas de la familia NC ISO 9000. Aplicación práctica*. **Oficina Territorial de Normalización de Villa Clara.** Villa Clara : s.n., 2016.
100. **Manguela Díaz, Gabino.** Apuesta por las cadenas de suministro. *Economía. Separata de Trabajadores.* No 14. 25 de Mayo de 2015.

ANEXOS

ANEXO 1

ID	TÍTULO	REF.
Artículo 1	Seven process modeling guidelines	(23)
Artículo 2	Human and automatic modularizations of process models to enhance their comprehension	(51)
Artículo 4	A Study Into the Factors That Influence the Understandability of Business Process Models	(5)
Artículo 5	Activity labeling in process modeling: Empirical insights and recommendations	(24)
Artículo 6	On the Usage of Labels and Icons in Business Process Modeling	(40)
Artículo 9	Syntax highlighting in business process models	(26)
Artículo 11	Cognitive Complexity in Business Process Modeling	(42)
Artículo 12	Managing Process Model Complexity via Concrete Syntax Modifications	(43)
Artículo 13	Managing Process Model Complexity via Abstract Syntax Modifications	(67)
Artículo 40	Factors of process model comprehension-Findings from a series of experiments	(22)
Artículo 43	Quality indicators for business process models from a gateway complexity perspective	(25)
Artículo 49	The Effects of Content Presentation Format and User Characteristics on Novice Developers' Understanding of Process Models	(28)
Artículo 50	Tying process model quality to the modeling process: the impact of structuring, movement, and speed	(20)
Artículo 60	Improving Quality of Business Process Models	(19)
Artículo 65	A quality based approach for the analysis and design of business process models	(68)

Tabla 6: Estudios seleccionados para el FCA (factor comprensibilidad).

ANEXO 2



Dirección Territorial Sancti Spiritus

Empresa de Aplicaciones Informáticas

4 Conclusiones

Teniendo en cuenta los resultados de las pruebas realizadas al producto verificado, se considera que el mismo debe ser:

Aprobado:

Pues el producto no tiene errores que afectan su normal y correcto funcionamiento.

Se recomienda al Comité de Negocios que declare al producto verificado en correspondencia con los resultados presentados en este informe.

	Nombre y apellidos	Cargo	Firma
Elaborado por	Lian Lisette Hurtado Linares	Jefe de Pruebas	
Revisado por	Lídice Álvarez Pérez	Jefe de Laboratorio	
Aprobado por	Carmen Alonso Brizuela	Director de la unidad organizativa	



ANEXO 3

Cuestionario de guía para entrevistas (control de la satisfacción)

El objetivo de la presente investigación es obtener su criterio sobre algunos aspectos de CAPYROX (herramienta para la captura de procesos y representación organizacional). Resultaría muy útil que usted respondiera las siguientes preguntas. Gracias por su participación.

- a) Conocía previamente la notación de modelado IDEF0 Sí _____ No _____
b) Conocía previamente alguna otra notación de modelado Sí _____ No _____
 Cuáles: _____

Sobre la notación de modelado en que se basa la herramienta:

Facilidad de aprendizaje Buena: _____ : _____ : _____ : _____ : _____ : Mala
Facilidad de utilización Buena: _____ : _____ : _____ : _____ : _____ : Mala
Facilidad de modificación Buena: _____ : _____ : _____ : _____ : _____ : Mala
Soporte a la modularidad Buena: _____ : _____ : _____ : _____ : _____ : Mala

Sobre los modelos de procesos que son posibles de obtener con la herramienta:

Comprensibilidad Buena: _____ : _____ : _____ : _____ : _____ : Mala
Compleitud Buena: _____ : _____ : _____ : _____ : _____ : Mala
Posibilidad de profundizar en detalle Buena: _____ : _____ : _____ : _____ : _____ : Mala

Sobre la herramienta:

Interfaz de usuario Agradable: _____ : _____ : _____ : _____ : _____ : Desagradable
Inspector de propiedades Útil: _____ : _____ : _____ : _____ : _____ : Inútil
Rendimiento Fuerte: _____ : _____ : _____ : _____ : _____ : Débil
Funcionalidad Profunda: _____ : _____ : _____ : _____ : _____ : Superficial
Facilidad de aprendizaje Buena: _____ : _____ : _____ : _____ : _____ : Mala
Facilidad de utilización Buena: _____ : _____ : _____ : _____ : _____ : Mala
Pertinencia Alta: _____ : _____ : _____ : _____ : _____ : Baja
Integridad conceptual Alta: _____ : _____ : _____ : _____ : _____ : Baja
Fiabilidad Buena: _____ : _____ : _____ : _____ : _____ : Mala
Soporte Bueno: _____ : _____ : _____ : _____ : _____ : Malo

Sobre el curso recibido:

Comunicación Buena: _____ : _____ : _____ : _____ : _____ : Mala
Claridad de explicación Buena: _____ : _____ : _____ : _____ : _____ : Mala
Coherencia Buena: _____ : _____ : _____ : _____ : _____ : Mala
Comprensibilidad Buena: _____ : _____ : _____ : _____ : _____ : Mala

ANEXO 4

AVAL

Desde el año 2011, se gestiona en la Empresa de Aplicaciones Informáticas, DESOFT, el Proyecto para la Gestión del Conocimiento.

Este proyecto, a partir de su redimensionamiento en 2015, tiene como objetivos específicos:

- dirigir, organizar, orientar, coordinar, integrar y controlar los procesos de gestión del conocimiento
- planificar, coordinar y controlar el flujo del conocimiento y la información en la empresa
- impulsar, coordinar y estructurar el proceso de captura o adquisición, generación, almacenamiento e indexación, distribución y transferencia y utilización del conocimiento, de la información, de las ideas y las experiencias para mejorar la calidad en el cumplimiento y desarrollo de la misión de la empresa.

La División Territorial Villa clara, de la mencionada empresa, es el piloto para la Gestión del Conocimiento (GC).

Ha resultado una gran experiencia la utilización de la herramienta CAPYROX (Captura de Procesos y Representación Organizacional) que utiliza como base la notación IDEF0, desarrollada por Lisett Pérez Quintero y Jorge Carrera Ortega especialistas de DESOFT Villa Clara. Específicamente, el uso de esta herramienta apoyó el conjunto de decisiones tomadas por un equipo multidisciplinario para la definición de este proceso, GC, lo cual requirió la identificación de las actividades, responsables, entradas, salidas, mecanismos de control, recursos, variables de control e indicadores.

Por la manera de trabajar que aporta Capyrox, que tiene incorporado un variado grupo de elementos gráficos y ventanas de propiedades, que permiten el análisis económico de alternativas de definición de los procesos, la identificación de riesgos y competencias necesarias en las actividades, así como la asociación de los indicadores de eficacia y además es capaz de exportar la ficha del proceso en un formato definido la convierte en bondadosa

ya que la notación gráfica facilita la comprensibilidad, el análisis de situaciones, el trabajo en equipo y es suficientemente simple por la poca cantidad de símbolos que emplea. Cualquier cambio en la representación posibilita la actualización de la ficha del proceso de una manera cómoda y eficaz, para permitir las modificaciones y actualizaciones de los procesos en su mejora continua.

Con Capyrox se ha modelado el proceso macro de GC y una instanciación de este proceso en el desarrollo de un producto, en los momentos de escritura de este aval se trabaja en la modelación de la instanciación del proceso GC para el servicio de Formación.

Con este aval se enfatiza la positiva calidad del producto, teniendo en cuenta que el propósito de empleo de la herramienta en sí se corresponde con la contribución a la gestión de la calidad en la organización.

La Habana, 11 de Octubre de 2016



MSc., Ing. María Elena Maciá Gravier

Lider del proyecto Gestión del Conocimiento

Dirección de Gestión del Conocimiento

DESOFT

Cienfuegos, 14 de Julio, 2016

AVAL
EXPERIENCIA EN LA APLICACIÓN DE LA HERRAMIENTA CAPYROX
COMPLEJO HOTELERO RANCHO LUNA-FARO LUNA

Durante la experiencia que se ha llevado a cabo hasta la actualidad en la aplicación de CAPYROX, de los autores Lisett Pérez Quintero y Jorge Carrera Ortega especialistas de DESOFT Villa Clara, se puede hacer constancia de la positiva calidad del producto, teniendo en cuenta que el propósito de empleo de la herramienta en sí se corresponde con la contribución a la gestión de la calidad en la organización. A continuación se emite un criterio evaluativo sobre relevantes características de calidad para el producto de software CAPYROX tomando como referencia lo planteado en la NC ISO/IEC 9126-1, Ingeniería de Software – Calidad del producto- Parte 1: Modelo de la calidad.

Funcionabilidad: El software cuenta con una correcta capacidad para proporcionar funciones que permiten una respuesta eficaz durante el diseño de los procesos y apoyo en el análisis de mejora. En las actividades desarrolladas se ha evidenciado la correcta operación en general. Mediante el uso adecuado del mismo, se ha evidenciado que proporciona resultados correctos para sus funcionalidades. El mecanismo de los archivos generados permite un manejo de la información ágil y portable.

Confiabilidad: El software ha mantenido el nivel de ejecución previsto con un desempeño correcto. Como otro aspecto importante a destacar en relación a la confiabilidad se han producido reuniones de presentación del alcance, perspectivas y funcionalidades involucradas en la herramienta con especialistas del área de gestión de la calidad y expertos en la rama de la gestión turística emitiéndose un favorable criterio sobre la solución, criterios que han estado en correspondencia a un grupo de normas cubanas de gestión de la calidad que están muy relacionadas a la presente propuesta.

Usabilidad: Los elementos para el trabajo con la herramienta han resultado de fácil comprensibilidad, basada en una notación que proporciona alto nivel de detalle sin complejizar las funciones de modelado para los diagramas. De acuerdo a esto se ha percibido un fácil entendimiento de las acciones de modelado que se han realizado, propiciando la comunicación y el intercambio de ideas con los jefes de procesos y otros miembros involucrados. Además, señalamos la adecuada concepción de acciones de capacitación concebida por los autores de dicha herramienta, esto se ha fundamentado en encuentros presenciales y la existencia de un asistente en la herramienta para la localización de temas contenidos. Ha sido posible la consideración de elementos de formatos en los tipos de letra, tamaño de letra y otros aspectos en correspondencia a la adecuada usabilidad.

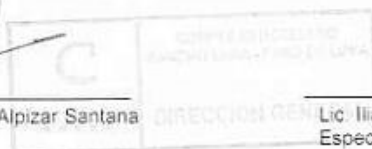
Eficiencia: Se ha logrado constatar un adecuado desempeño durante su ejecución en correspondencia a los recursos utilizados, considerando el hardware y materiales que posibilita optimizar ya que brinda visualizaciones y la posibilidad de obtener reportes en formato digital. Genera diferentes reportes considerando los aspectos modelados, lo cual optimiza tiempo ya que no tienen que ser elaborados complementariamente de forma manual. Los tiempos de respuesta han sido positivos.

Mantenibilidad: Se ha podido constatar una positiva proyección respecto a la mantenibilidad, considerando la posibilidad de mejora y adaptaciones del software, evidenciándose una buena estabilidad.

Portabilidad: Se ha podido observar en encuentros realizados, que la herramienta responde adecuadamente en diferentes sistemas operativos, pues se han realizado presentaciones sobre sus funcionalidades tomando como base ilustraciones en alternativas como Ubuntu y Debian manteniendo un ambiente visual muy parecido. Los pasos de instalación mostrados resultan de fácil comprensibilidad.

Sobre la experiencia desarrollada hasta el momento, emitimos un favorable criterio sobre la calidad durante la utilización, teniendo en cuenta criterios de efectividad, productividad, seguridad y satisfacción en el contexto de su utilización.

MSc. Delfin Giovanni Alpizar Santana
 Director General



Lic. Iliá Dania Curbelo Periaña
 Especialista de Calidad.