

JERARQUÍA Y GRANULARIDAD DE COMPONENTES DE SOFTWARE PARA
PYMES EN BOGOTÁ

MARIBEL ARIZA ROJAS
JUAN CARLOS MOLINA GARCÍA

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA DE SISTEMAS
JULIO, BOGOTÁ D.C.
2005

**JERARQUÍA Y GRANULARIDAD DE COMPONENTES DE SOFTWARE PARA
PYMES EN BOGOTÁ**

**MARIBEL ARIZA ROJAS
JUAN CARLOS MOLINA GARCÍA**

**Proyecto De Grado Presentado Para Optar Al Título De Ingeniero De
Sistemas**

**Ingeniero Rafael Andrés González Rivera MSc.
Profesor Investigador Área de Ingeniería de Software
Director de la Investigación**

**PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA DE SISTEMAS
JULIO, BOGOTÁ D.C.
2005**

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
CARRERA DE INGENIERIA DE SISTEMAS

Rector Magnífico: Padre Gerardo Remolina Vargas S.J.

Decano Académico Facultad de Ingeniería: Ingeniero Francisco Javier
Rebolledo Muñoz

Decano del Medio Universitario Facultad de Ingeniería: Padre Antonio José
Sarmiento Novoa S.J.

Director Carrera de Ingeniería de Sistemas: Ingeniera Hilda Cristina Chaparro
López

Director Departamento de Ingeniería de Sistemas: Ingeniero Germán Alberto
Chavarro Flórez

Nota de Aceptación

Firma del Director del Proyecto

Firma del Jurado

Firma del Jurado

BOGOTÁ D.C., JULIO DE 2005

Artículo 23 de la Resolución No. 1 de Junio de 1946:

“La Universidad no se hace responsable de los conceptos emitidos por sus alumnos en sus proyectos de grado.

Sólo velará porque no se publique nada contrario al dogma y la moral católica y porque no contengan ataques o polémicas puramente personales. Antes bien, que se vean en ellos el anhelo de buscar la verdad y la Justicia”

DEDICATORIA:

A mi Padre, Jorge Alirio, por su apoyo y enseñanzas, y a mi Madre, Carmen Alicia, por el amor incondicional a sus hijos, por todos los sacrificios que no han sido ni serán en vano. Los admiro y respeto, gracias por ayudarme a ser quien soy.

A mi hermano César Jair, por ser un amigo y un ejemplo a seguir. A mi hermana y mejor amiga Nathalia, por escucharme, aconsejarme y apoyarme siempre. A ambos gracias, no podría tener mejores hermanos que ustedes.

A mis amigos, ustedes saben quienes son, gracias por caminar todo este tiempo a mi lado, nada sería lo mismo sin ustedes.

A Nicolás, por su apoyo, paciencia y apoyo incondicional. Su amor me hace ser una persona mejor.

Y finalmente, a Dios, quien me permitió llegar a este punto y colocó en mi camino a todas las personas maravillosas que hoy me acompañan e hicieron posible la culminación de esta meta.

Maribel

“La educación no debería ser una preparación para la vida; debería ser vida”

Anthony de Mello

...y lo es.

A todos los que me enseñaron a caminar en vez de mostrarme el camino....Gracias.

Padres, hermanos, maestros, amigos y a ti..... lo logré.

Juan Carlos

AGRADECIMIENTOS:

Quisiéramos agradecer a todas aquellas personas que hicieron posible la realización de este trabajo, especialmente a nuestros contactos al interior de las Pequeñas y Medianas empresas que nos ayudaron en la realización del estudio de campo. Estamos muy agradecidos con Francisco Aguirre y Alejandro Carrillo por sus importantes aportes en los resultados de dicho estudio.

A los todos los estudiantes que constituyeron el grupo PICS (Programa de Investigación de Componentes de Software) Miguel Ángel Ávila, Rodrigo Fonseca, Dawid Junnco y Daniel Reyes. El trabajo realizado en equipo fue de vital importancia para este proyecto de investigación.

Estamos en deuda también con Miguel Eduardo Torres, por su colaboración y su tiempo. Y un especial agradecimiento a nuestro director, Rafael González, por sus comentarios, enseñanzas y su apoyo a través de todo este proceso.

TABLA DE CONTENIDO

RESUMEN EJECUTIVO.....	13
INTRODUCCIÓN.....	15
1. JUSTIFICACIÓN	17
2. OBJETIVOS	23
2.1. Objetivo General	23
2.2. Objetivos Específicos.....	23
3. MARCO TEÓRICO.....	24
3.1. Componentes de Software.....	24
3.1.1. Definición	24
3.1.2. Características	25
3.2. Diseño de Componentes.....	27
4. TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN Y LAS PYMES COLOMBIANAS	31
4.1. Definición y Antecedentes.....	31
4.2. Estado Actual del las Pymes Colombianas	33
4.3. Estructura Administrativa	35
4.4. Sistemas de Información Organizacionales	39
4.5. Oportunidades y Amenazas del Mercado	44
4.6. Resultados Obtenidos del Estudio de Campo.....	47
5. CLASIFICACIONES DE COMPONENTES DE SOFTWARE	50
5.1. Infraestructura común de componentes en capas	50
5.2. Arquitectura de Sistemas de SW Basados en Componentes	52
5.3. Infraestructura de componentes orientada a la reutilización	54
5.4. Clasificación por Facetas	55
5.5. Repositorios de Componentes y sus Clasificaciones.....	58
6. JERARQUÍA DE DOMINIO DE COMPONENTES DE SOFTWARE PARA PYMES.....	61
6.1. Componentes de Negocio.....	62

6.2. Componentes de Servicio	63
6.3. Componentes Operacionales de Persistencia	64
6.4. Componentes Operacionales.....	65
6.5. Componentes de Integración	65
6.6. Componentes GUI	66
7. SELECCIÓN DEL NIVEL PRINCIPAL.....	70
7.1. Criterios de Selección	70
7.1.1. El compromiso del Gobierno Nacional con el desarrollo tecnológico de la PyMe.....	70
7.1.2. Inventario Tecnológico	72
7.1.3. Evaluación de la relación actual entre las TIC's y PyMes	72
7.2. Selección del Nivel Principal de la Jerarquía de Componentes	74
8. GRANULARIDAD DE COMPONENTES DE SOFTWARE	75
8.1. Consideraciones de la Granularidad en el DSBC	75
8.2. Definición de Granularidad.....	76
8.2.1. Niveles y Unidades de Descripción.....	79
8.2.1.1. Descripción de Mercadeo	79
8.2.1.2. Descripción Técnica:.....	79
8.2.1.3. Descripción de Interfaz.....	82
8.2.1.4. Descripción de la Funcionalidad.....	83
8.2.1.5. Descripción de Extensión.....	86
8. RESULTADOS OBTENIDOS	90
9. CONCLUSIONES.....	92
10. RECOMENDACIONES Y TRABAJO FUTURO.....	95
11. BIBLIOGRAFÍA.....	99
ANEXOS.....	105
1. Formato Encuesta	105
2. Listado de Empresas Encuestadas	113
3. Tabulación de Resultados	114

LISTA DE FIGURAS

Figura 0: Programa de Investigación de Componentes de Software (PICS) 1	21
Tabla 1: Clasificación Ley 509 de 2000, Mipyme 2	31
Gráfica 1: Distribución de la PyMe por sector 3	32
Gráfica 2: Distribución Geográfica de la PyMe 4	32
Tabla 2: Referencia para antigüedad 5	33
Gráfica 3: Antigüedad de la PyMe Colombiana 6	33
Figura 1: Componentes Básicos Organizacionales 7	36
Figura 2: Niveles de administración 8	37
Figura 3: Niveles administrativos 9	38
Figura 4: Sistemas de Información dentro de las organizaciones 10	40
Figura 5: Portafolio Potencial de TIC's y Niveles de Admón 11	43
Figura 6: Infraestructura común de componentes en capas 12	50
Figura 7: Clasificación de Componentes definida por medio de capas 13	52
Figura 8: Capas de las aplicaciones basadas en la reutilización dirigida a la Ingeniería de Software de Negocio 14	53
Figura 13: Elementos del proceso de reutilización 15	75
Tabla 3: Elementos de representación de Unidades de Descripción 16	78
Figura 13: Descripción de la Granularidad de un Componente de Software 17	89

GLOSARIO

-A-

Agenda de Conectividad: Estrategia del Ministerio de Comunicaciones, con la cual se busca la integración, articulación y desarrollo de la política de estado, que busca la masificación del uso de las TIC's en sector económico de la PyMe.

Asunciones: Asumir como verdadero.

API: Application Programming Interface

-B-

Biblioteconomía: Es la disciplina encargada de la conservación, organización y administración de las bibliotecas.

-D-

DSBC: Desarrollo de Software Basado en componentes

-G-

GUI: Guide User Interface

-O-

Ofimática: Son todas aquellas técnicas, servicios o procedimientos que aportan las tecnologías de la información y cuya ejecución se realiza en el entorno de oficinas o ámbitos relacionados. En definitiva, la ofimática abarca los tratamientos de textos y gráficos, la gestión documental, la agenda, la planificación del tiempo y las actividades, la mensajería electrónica, etc.

-P-

PyMe: Pequeña y Mediana Empresa

-T-

TIC's: Tecnologías de Información y Comunicación

RESUMEN EJECUTIVO

Este proyecto de investigación lleva como título “Jerarquía y Granularidad de Componentes de Software para PyMes en Bogotá” y hace parte de un Programa de Investigación titulado “Componentes de Software para PyMes”, cuyo investigador principal, director de este proyecto, es el Ing. Rafael González Rivera. Este programa de investigación busca construir un entorno que apoye el desarrollo de software basado en componentes dentro de las PyMes, para que estas puedan estar en capacidad de atender sus propias necesidades informáticas. Para esto fueron planteadas tres grandes líneas de investigación: “Definición Ontológica de Componentes”, “Metodología para Desarrollar Componentes” y “Metodología para desarrollar Sistemas de Información con Componentes”, y es de esta primera línea de donde se desprende este proyecto.

El objetivo principal es definir una Jerarquía que incluya los grupos de componentes más importantes para el desarrollo de software al interior de las PyMes y la definición de un nivel de descripción (Granularidad) para los componentes que forman la jerarquía y especialmente para el nivel que represente el grupo de componentes capaces de ofrecer el mayor valor agregado posible para los desarrollos de software de las PyMes.

Para alcanzar este objetivo se llevó a cabo en primer lugar un estudio de campo dentro de quince (15) PyMes bogotanas, diseñado y realizado por tres grupos de investigación, que pertenecen a PICS (Proyecto de Investigación de Componentes de Software), siendo esta la primera vez que varios grupos de investigación trabajaban en conjunto para alcanzar diferentes objetivos. Los resultados obtenidos por este estudio de campo representaron uno de los primeros puntos de análisis para la definición de la jerarquía y la granularidad. En este proceso de análisis también fueron tenidos en cuenta diferentes

fuentes de información bibliográfica, proyectos de grado de universidades nacionales e internacionales, entre otros.

Se definió entonces una Jerarquía de Componentes de Software constituida por seis (6) diferentes niveles, los cuales abarcan desde los Componentes GUI hasta los Componentes de Negocio. Se seleccionó este último como el más importante para las PyMes y fue definido un nivel de descripción integrado por cinco (5) niveles y ocho (8) unidades de descripción, estas últimas permiten detallar con más precisión los niveles de la granularidad y además facilitan la comprensión y clasificación de cada componente.

Algunas de las conclusiones más relevantes obtenidas en este proyecto fueron en primer lugar la necesidad de orientar el software empresarial hacia la generación de ventajas competitivas, buscando proveer soluciones orientadas al cliente y al manejo estratégico de los recursos tanto humanos como empresariales y en segundo lugar la inminente necesidad de incrementar el nivel de competitividad de las PyMes, tanto a nivel nacional como internacional.

INTRODUCCIÓN

El sector de las PyMes es uno de los sectores de crecimiento económico más importante para Colombia, y la tendencia mundial hacia la globalización y la erradicación de las fronteras geográficas, ha empezado a obligar al sector a crear estrategias de competencia en un entorno cada vez más cambiante y exigente. En este orden, el gobierno nacional ha dado inicio a una serie de estrategias, comúnmente orientadas a la apertura de las PyMes en el mercado internacional y a la evolución de las Tecnologías de Información que se utilizan al interior de estas organizaciones.

Por otra parte, el desarrollo de software basado en componentes (DSBC) se ha convertido en una de las más importantes metodologías de desarrollo, la cual trae consigo una serie de ventajas representadas en reducción de tiempos de desarrollo y costos, dando a las PyMes una nueva opción para hacer uso de últimas herramientas tecnológicas, a bajo costo y de rápido desarrollo.

Este proyecto de investigación busca definir dos conceptos muy importantes relacionados con el DSBC y las PyMes como lo son: la Jerarquía de Componentes de Software, y su Granularidad.

El documento se encuentra dividido en cuatro secciones, a continuación se señalan algunas de las características más importantes de cada una de ellas:

La primera sección destaca las características del DSBC, incluyendo temas como definición y diseño de componentes de software. La segunda sección ofrece una evaluación acerca del estado actual de las PyMes colombianas, su estructura administrativa, el portafolio potencial de Tecnologías de Información y Telecomunicaciones, y el compromiso del gobierno nacional con la evolución de la PyMe. Esta sección también incluye los resultados más relevantes del

estudio de campo realizado por este proyecto de investigación dentro de quince PyMes Bogotanas.

La tercera sección expone en primer lugar las diferentes clasificaciones de componentes existentes, para luego definir la jerarquía propia de la investigación con base en dichas clasificaciones. Posteriormente, se exponen los criterios de selección del nivel perteneciente a esta jerarquía, el cual elige los componentes capaces de ofrecer a la PyMe una mayor ventaja competitiva. Finalmente, esta sección incluye la definición del concepto de granularidad para componentes y el nivel de detalle con el cuál serán descritos dentro de este proyecto.

La cuarta y última sección describe la aplicabilidad de este proyecto en trabajos futuros y las principales conclusiones y resultados obtenidos del trabajo realizado.

1. JUSTIFICACIÓN

El desarrollo basado en componentes se ha denominado como la evolución natural del desarrollo de software orientado a objetos. Esta metodología de desarrollo, por medio del modelamiento y la abstracción de los objetos que representan un problema, logran unificar criterios en la forma como se desarrollaban programas, así como crear un lenguaje estandarizado que permite acortar la brecha entre usuario y programador. Esta también introdujo la reutilización de software, lo cual logra que los desarrollos a posteriori utilicen recursos existentes sin incurrir en gastos de desarrollo adicionales. La reutilización permite avanzar más rápido en desarrollos con características similares a partir de desarrollos existentes¹.

El Desarrollo de Software basado en componentes busca, dentro de otros objetivos, reducir el tiempo del proyecto, el esfuerzo que requiere implementar una aplicación y los costos en general. Si se logran reducir estos tres aspectos se lograría incrementar el nivel de productividad en los equipos de desarrollo y minimizar el riesgo global del proyecto, así como los costos de este. De esta manera las pequeñas empresas pueden tener una mayor confiabilidad si quieren realizar una inversión tecnológica. Otra ventaja es poder integrar lo mejor de varias tecnologías (desarrollos multiplataforma), para desarrollar una aplicación de manera personalizada y a la medida de las necesidades de los clientes. Esto permite a los desarrolladores y a la empresa adquirir las tecnologías que más se adapten a sus necesidades, y de esta forma no incurrir en gastos de licenciamiento o soporte, porque además de la actualización de las grandes soluciones, muchas de estas tecnologías son gratis y existen bajo las premisas de software libre y de código abierto, lo cual añade otra gran ventaja al DSBC.

¹ **BERTRAN** Meyer. "Significado de componentes". En **SCOUT** Amber y **CONSTANTINE** Larry . "The Unified Process Construction Phase". CMP Books. 2000. 212-221 p.

Esta clase de desarrollos ya han probado su confiabilidad y buenos resultados, organizaciones como Linux Open Source, y la creciente comunidad de desarrolladores JAVA entre otros, muestran que el desarrollo por medio de código abierto y licenciamiento gratis es viable y produce resultados concretos y factibles. El software realizado con código abierto enriquece el desarrollo basado en componentes, por medio de una comunidad mundial de desarrolladores que se encarga de aportar mejoras. El software libre o de libre licenciamiento ofrece de manera gratuita o con costos muy bajos desarrollos valiosos y de gran utilidad, los cuales contienen las características necesarias para ser utilizadas en las PyMes.

Las PyMes en nuestro país representan un importante sector de la economía. Sin embargo, Colombia se encuentra en un retraso tecnológico que hace que las PyMes pierdan competitividad ante el mercado mundial. Muy pocas enfrentan el desafío de buscar tecnologías que las ayuden, y otras, no saben como enfrentar este problema. Para estas empresas la tecnología se ha convertido en una herramienta que puede ser un factor decisivo respecto a su competitividad en el mercado.

El software es el elemento tecnológico de mas alcance y de mayor aceptación por parte de los sectores de servicios, micro establecimientos e industria según el DANE² (Departamento Administrativo Nacional de Estadística), esto demuestra que estas herramientas son necesarias y pueden ofrecer una importante ventaja competitiva. La investigación realizada por el DANE buscó medir los resultados de la Agenda de Conectividad. De acuerdo a esta investigación: el 54.4% en el sector público, el 41.1% en micro establecimientos y 46.6% en la industria utilizan software nacional.

² "Medición de las tecnologías de la información y las comunicaciones TIC's" - Resumen ejecutivo (Diciembre 2003),[On Line] <http://www.dane.gov.co>

Ahora bien, el 23.6% del software utilizado por los micro establecimientos es para uso administrativo y no se utiliza en gran medida software orientado a la productividad, automatización de procesos o a estrategias comerciales. Estas son sólo algunas de las cifras que permiten concluir que las PyMes creen en el software producido en el país, pero aún existen varias barreras que impiden emprender planes estratégicos donde se considere a la tecnología como uno de los factores determinantes en el éxito de la pequeña y mediana empresa nacional.

Esta investigación hace parte del Programa de Investigación Componentes de Software para PyMes, cuyo propósito es: “Proveer a las PyMes de herramientas informáticas atractivas y viables que les permitan modernizarse y hacerse competitivas en un contexto cada vez más dinámico, exigente e integrado”³, y este proyecto de investigación, al ser parte fundamental de este programa, no puede ser ajeno al cumplimiento de este propósito. Es así, como la definición de la Jerarquía y Granularidad de Componentes de Software, fines últimos de este proyecto, dan apoyo al proyecto desarrollado en paralelo: “Repositorio de Metadatos de Componentes de Software para PyMEs Colombianas” y posteriores proyectos pertenecientes al programa de investigación que buscarán definir una metodología tanto para desarrollar componentes como para desarrollar sistemas de información basados en componentes. En conclusión, la definición de la Jerarquía y Granularidad pretende facilitar el proceso de desarrollo e integración de componentes, dando inicio al proceso de adopción de TIC’s dentro del sector económico de las PyMes, y orientando su utilización a perspectivas de negocio estratégicas y competitivas.

³ **GONZALEZ**, Rafael. Programa de Investigación: Componentes de Software para PyMes. Pontificia Universidad Javeriana. Facultad de Ingeniería. 2004. 3 p.

Por otra parte es muy importante explicar cuál es el **ORIGEN** de este Proyecto de Investigación. En la actualidad se está desarrollando un Programa de Investigación de Componentes de Software (PICS), el cual está siendo dirigido por el Ing. Rafael González; este proyecto cuenta con el apoyo del Departamento de Ingeniería de Sistemas y en él trabajan tanto estudiantes como profesores del Departamento.

El propósito principal de PICS es: “Proveer a las PyMes de herramientas informáticas atractivas y viables que les permitan modernizarse y hacerse competitivas en un contexto cada vez más dinámico, exigente e integrado”⁴. La pregunta fundamental del programa es: ¿Cómo construir componentes de software para las PyMEs colombianas?, y la forma de darle solución y de atacar el problema fue resuelto a través del planteamiento de tres líneas de investigación como se muestra en la figura 1. Existen entonces tres grandes proyectos de investigación:

1. Definición Ontológica de Componentes de Software
2. Metodología para desarrollar Componentes de Software
3. Metodología para desarrollar sistemas de información basados en componentes.

⁴ **GONZALEZ**, Rafael. Programa de Investigación: Componentes de Software para PyMes. Pontificia Universidad Javeriana. Facultad de Ingeniería. 2004. 3 p.

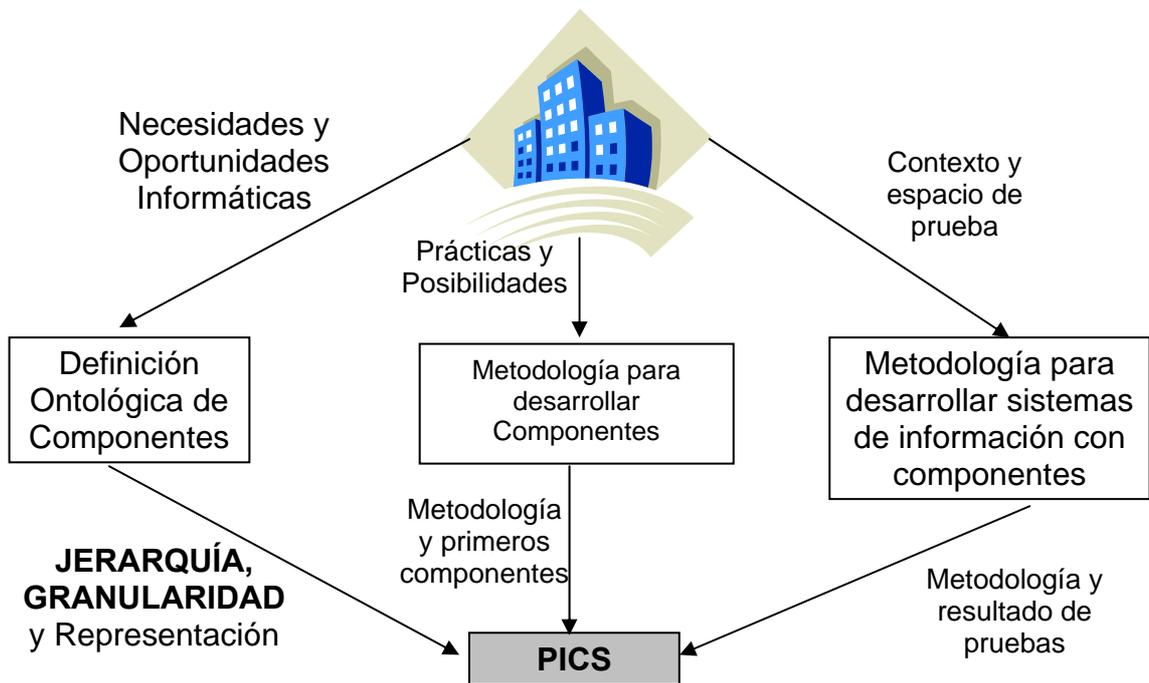


Figura 0: Programa de Investigación de Componentes de Software (PICS) 1

El primero de estos contiene el presente Proyecto, su tiempo estimado de desarrollo es de un año. Todos los elementos que se obtienen de este proyecto como son: Jerarquía, Granularidad y la Representación de Componentes se basan en las necesidades y oportunidades informáticas de las PyMes colombianas,

El segundo proyecto, de duración estimada un año, hará uso de estas definiciones, y de esta forma se propondrá una metodología para la construcción de componentes de software. Hasta este punto de la investigación es de vital importancia el reconocimiento de la situación actual del desarrollo de software dentro de las PyMes, identificar fortalezas, debilidades, oportunidades, etc., esta fue una de las razones principales por las cuáles se llevo a cabo un estudio de campo que permitiera un primer acercamiento a estos procesos de desarrollo de software. La finalización de la metodología dará paso a la creación de componentes de prueba, definidos formalmente, catalogados y archivados en un repositorio de componentes, este repositorio es también un

resultado de un proyecto de investigación realizado por estudiantes de pregrado del departamento de sistemas, el cual tiene por título: “Repositorio de Metadatos de Componentes de Software para PyMEs Colombianas”

Luego se dará inicio al tercer proyecto, durante el año 3 de investigación “donde dado el contexto de las PyMes del país, se propondrá una metodología para construir un sistema de información (o un conjunto de sistemas de información) basándose en los componentes propuestos. Esta última parte es fundamental, pues articula los resultados técnicos anteriores en la realidad colombiana y pone a prueba las posibilidades del desarrollo por componentes como un foco de desarrollo para las PyMEs. Esto no solo implica la construcción de una metodología adecuada totalmente a la realidad de las PyMEs (y su respectiva prueba de campo en las mismas), sino que puede implicar la formulación de posteriores proyectos en materia de difusión y promoción de la metodología. Esto último se anota dándole cabida a la posibilidad de que nos encontremos con PyMEs no interesadas en avanzar en materia de tecnología de información y sin un reconocimiento y capacidad clara de apalancarse en esta tecnología para su supervivencia o, más aún, de hacerse líderes en su desarrollo para volverse competitivos a nivel mundial”⁵.

De esta forma vemos que el origen y la fundamentación tanto de PICS como del Proyecto esta en brindar a las PyMes oportunidades reales para su innovación y progreso tecnológico. Es muy gratificante poder aportar una parte fundamental a un programa de investigación tan ambicioso como este, y además trabajar con la convicción de estar construyendo oportunidades reales de progreso que orienten la utilización del software hacia una perspectiva estratégica y competitiva más allá de la administrativa.

⁵ **GONZÁLEZ**, Rafael. “Programa de Investigación de Componentes de Software para PyMes”. On line: “<http://ainsuca.javeriana.edu.co/%7Ergonzal/pics/>”

2. OBJETIVOS

2.1. Objetivo General

Definición Jerárquica y Granularidad de Componentes de Software orientadas al Desarrollo de Software basado en Componentes dentro de las PyMEs en Bogotá.

2.2. Objetivos Específicos

- Ampliar nuestro conocimiento en el área de desarrollo de software basado en Componentes.
- Conocer el estado actual de los procesos de desarrollo de software dentro de las PyMEs en Bogotá.
- Identificar cuáles son los tipos de componentes de software que representen una ventaja competitiva en el proceso de desarrollo de software.
- Definir una Jerarquía de Componentes de Software propia de la Investigación, con base en el análisis y la comparación de los diferentes tipos de Jerarquías existentes.
- Identificar el nivel, dentro de la Jerarquía de la Investigación, más apropiado para las PyMEs en Bogotá.
- Definir la descripción de componentes (Granularidad) con la que se debe trabajar el nivel seleccionado con base en las necesidades que deben suplir los desarrolladores de Software dentro de las PyMEs en Bogotá.

3. MARCO TEÓRICO

3.1. Componentes de Software

La historia de la utilización de componentes de software proviene desde los logros de la revolución industrial, surgiendo a partir de la necesidad de estandarizar los elementos de los productos realizados en línea, como los automóviles⁶.

Los componentes de software surgen como una evolución derivada de la programación orientada a objetos, estos ofrecen un desarrollo mucho más rápido y con mucho menos esfuerzo, reduciendo costos y tiempos de desarrollo, aprovechando las ventajas de la reutilización del Software. Estas y otras cualidades, han logrado su reconocimiento en el desarrollo de software y durante varios años se han estado desarrollando sus principales conceptos e ideas que permiten mejoras en los sistemas basados en componentes. Todo esto abarca elementos como algoritmos, diseños, arquitecturas de software, planes, documentación, requerimientos entre otros. Para entender lo que es un componente se deben aclarar conceptos tan importantes como lo son su definición y características más representativas.

3.1.1. Definición

Actualmente existen varias definiciones de Componentes de Software complementarias entre sí, sin embargo, se toman dos de las más representativas y abarcan los temas más generales en lo que a componentes se refiere.

⁶ **GONZÁLEZ**, Rafael. "Ontología de Software para PyMes". 2004. [On Line]. http://ainsuca.javeriana.edu.co/~rgonzal/pics/ontologia.html#Planteamiento_del_problema_

Componente de Software:

- Es un paquete coherente de artefactos de software que puede ser desarrollado independientemente y entregado como una unidad, este puede ser compuesto e intercambiado con otro componente para construir algo mucho más grande⁷.
- Un componente es una parte no trivial, casi independiente, y reemplazable de un sistema que llena claramente una funcionalidad dentro de un contexto en una arquitectura bien definida. Un componente se conforma y provee la realización física por medio de un conjunto de interfaces⁸.

Estas definiciones permiten un primer acercamiento y un reconocimiento de una amplia gama de componentes, para filtrar y refinar más estas definiciones, así como de acortar el conjunto de posibles componentes de software, hace falta la definición de ciertos atributos y características que estos deben tener, que complementen la definición de lo que es un componente.

3.1.2. Características

Para identificar un componente de software que se adapte a las definiciones anteriores, se deben especificar algunas de sus características más importantes, esto garantiza la exclusión de algunos artefactos que pueden ser incluidos dentro de la definición de componente de software, pero que pueden generar problemas a la hora de contextualizarlos y ubicarlos en un desarrollo de software basado en

⁷ **SZYPERSKI**, Clemens. "Component Software: Beyond Object Oriented Programming". 2ª Edición. Addison – Wesley Edition. 2002. 589 p .

⁸ **KRUTCHEN**, Philippe. **KROLL**, Per. "Rational Unified Process". 2ª Edición. Addison – Wesley Edition. 2002. 400 p.

componentes.

1. Identificable: Debe tener una identificación que permita acceder fácilmente a sus servicios y que permita su clasificación.
2. Auto contenido: Un componente no debe requerir de la utilización de otros para cumplir la función para la cual fue diseñado.
3. Puede ser reemplazado por otro componente: Se puede reemplazar por nuevas versiones u otro componente que lo reemplace y mejore.
4. Con acceso solamente a través de su interfaz: Debe asegurar que estas no cambiaran a lo largo de su implementación.
5. Sus servicios no varían: Las funcionalidades ofrecidas en su interfaz no deben variar, pero su implementación puede hacerlo.
6. Documentado: Un componente debe estar correctamente documentado para facilitar su búsqueda si se quiere actualizar, integrar con otros, adaptarlo, etc.
7. Genérico: Sus servicios deben servir para varias aplicaciones.
8. Reutilizado dinámicamente: Puede ser cargado en tiempo de ejecución en una aplicación.
9. Independiente de la plataforma: Hardware, Software, Sistema Operativo, etc.⁹

Otra forma de identificar un componente es a través de los patrones, bien sean de diseño, arquitectura o análisis, un patrón describe un problema, una solución y un contexto en el que funciona. Los patrones más conocidos son los aplicados a la programación orientada a objetos o patrones de diseño. El diseño por componentes de software ha planteado sus propios patrones de análisis para la especificación de Componentes, los cuales proveen soluciones en el ciclo del desarrollo de un sistema basado en componentes, ayudando al ciclo de vida de

⁹ **Montilva C.** Jonás A, **Arapé Nelson Colmenares** Juan Andrés. "Desarrollo Basado en Componentes", 2003. 3 p.

este, por medio de la toma de decisiones tempranas y específicas¹⁰. Aunque los patrones hacen parte de las consideraciones de un diseño basado en componentes no son las únicas consideraciones a tener en cuenta.

Teniendo en cuenta la definición y las características de lo que es un componente, es más sencillo reconocerlo e identificarlo, ya que este puede ser un modulo, una clase, un procedimiento o función, un subsistema, o una aplicación.

3.2. Diseño de Componentes

Para definir el desarrollo basado en componentes, es importante precisar el concepto de diseño basado en componentes, lo cual es una metodología especializada de abordar una solución de software basada en componentes.

Existen reglas de diseño y parámetros para abordar no solo la parte inicial de la fabricación de un componente, sino también una ingeniería de software basada en componentes, estos temas son lo bastante amplios para tratar aquí, sin embargo es necesario resaltar ciertas generalidades de estos temas, que ayudarán al entendimiento de la construcción de la jerarquía.

El DSBC, confía en la existencia de un inventario de componentes existentes de software, así como la aparición de las tecnologías de integración de componentes tales como CORBA y COM¹¹. Este

¹⁰ **D'SOUZA**, Desmond Francis, **WILLS**, Alan Cameron. "Object components and Frameworks. The Catalysis Approach".1998 p

¹¹ **KUNDA**, Douglas, **BROOKS** Laurence. "Component-Based Software Engineering for Developing Countries: Promises and Possibilities". Department of Computer Science University of York, Heslington.

inventario se puede ver como el repositorio de componentes, pero estos componentes deben basarse en tecnologías que permitan su integración, de modo que puedan interactuar con otros independientemente de su proveedor.

Desde el punto de vista del diseño se necesita que los componentes sean rápidos de ubicar e identificar y así realizar rápidas adaptaciones a modelos existentes o nuevos,¹² un desarrollo por componentes puede realizarse a través de la adaptación de componentes preexistentes o por medio de la creación de repositorios aplicando el proceso de ingeniería de dominio¹³, el objetivo de la ingeniería de dominio consiste en identificar, construir, catalogar y diseminar un conjunto de artefactos de software en un determinado dominio de aplicación. Una vez se tiene el inventario de componentes es necesaria su clasificación para lograr esta rápida ubicación.

También se debe mencionar, que se debe partir de componentes que se puedan adaptar, bien sea a modelos monolíticos; que son diseños ya establecidos y sus partes son fragmentadas y así construir componentes; o no monolíticos que surgen como una unión de componentes que pueden funcionar a través de varios contextos¹⁴. La manera de buscar componentes no debe ser a partir de un diseño, arquitectura o modelo preexistente, se trata de acoplar componentes a un sistema, los componentes deben trabajar dentro del contexto de una arquitectura bien definida¹⁵; esta arquitectura de software será la encargada de la conducta de estos componentes y la forma en como

¹² **D'SOUZA**, Desmond Francis, **WILLS**, Alan Cameron. Op. Cit.. 345 p.

¹³ **MONTILVA**, Jonás. "Desarrollo de Software Basado en Componentes". Facultad de Ingeniería. Universidad de los Andes Venezuela .[On-Line] http://www.istec.org/events/ga2003/results/presentations/TDSBC_V2_Unidad_1.pdf

¹⁴ **SZYPERSKI**, Clemens. Op. Cit. 356 p.

¹⁵ **BROWN** A. W, **WALLNAU** K. C. "The current state of CBSE", IEEE Software 15 (1998) 37-46. [on-line] <http://www.idt.mdh.se/kurser/phd/CBSE/literature/The-current-state-of-CBSE.pdf>

estos interactúan y se coordinan para alcanzar la misión del sistema¹⁶.

La aplicación de una buena arquitectura da como resultado una correcta utilización de los componentes lo cual se ve reflejado en un producto óptimo, a bajo costo en cuanto a su mantenimiento y extensibilidad. Estas necesidades de arquitectura influyen en una clasificación y en la construcción de una jerarquía de componentes, esta última no debe basarse en ninguna de las arquitecturas definidas por las diferentes tecnologías, pero si necesita una arquitectura genérica que soporte su definición.

Para que un componente sea reutilizado en diversos contextos, debe ser suficientemente genérico para ser utilizado a través ellos, debe también ofrecer mecanismos que le permitan especializarse según las necesidades del sistema al que se adapte. Esto significa, que se debe entender qué partes del componente son comunes a través de esos contextos. Un componente también, debe ser diseñado para separar las piezas que varían a través de contextos y seleccionar cuidadosamente plug - points donde esa variación puede ser encapsulada y localizada, ya que estos plug - points serán los lugares donde los componentes especializados se pueden conectar para adaptar el comportamiento total que le dará forma al sistema¹⁷. El éxito de estos plug - points depende exclusivamente de las interfaces del componente, ya que estas son las responsables de dar el acceso a los servicios que el cliente (u otros componentes) necesita.

Los Modelos de Componentes son la implementación de un conjunto de software ejecutable dedicado, con los elementos necesarios para soportar la ejecución entre componentes. Los modelos de componentes

¹⁶REYNOSO, Carlos. "Introducción a la Arquitectura de Software". Universidad de Buenos Aires. [On-Line] [Http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.asp](http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.asp)

¹⁷ SZYPERSKI, Clemens. Op. Cit. 485 p.

ayudan al desarrollo de aplicaciones tales como frameworks y módulos, estos definen la construcción de un componente, por lo general adaptado a una tecnología. Definiendo como este se puede comunicar e interactuar con un sistema de componentes y como otros componentes lo harán con este¹⁸.

¹⁸ Ibid. pág. 487

4. TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN Y LAS PYMES COLOMBIANAS

4.1. Definición y Antecedentes

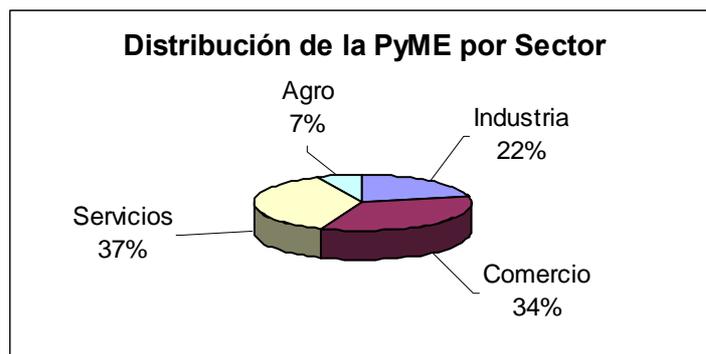
En Colombia, de acuerdo con las Ley 509 de 2000 Mipyme, clasifica a las empresas de la siguiente forma:

TIPO DE EMPRESA	EMPLEADOS
Micro	Hasta 10
Pequeña	Entre 11 y 50
Mediana	Entre 51 y 200
Grande	Mas de 200

Tabla 1: Clasificación Ley 509 de 2000, Mipyme 2

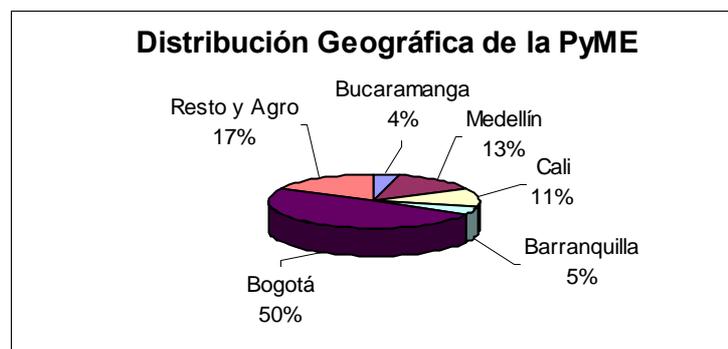
La falta de información oficial acerca de la cantidad de Pymes existentes en Colombia, no permite especificar el número de empresas que se encuentran en funcionamiento en la actualidad, esto se debe a que muchas de ellas trabajan de manera informal y no existe un mecanismo de encuestas que permita obtener mediciones verídicas en este sector. Sin embargo, de las 800 empresas que se estiman, se encuentran registradas en el país, en el Centro de Apoyo de Tecnología Informática (CATI), clasifica el 96.25% como Pymes y mipymes, y el 3.75% como medianas.

Según un estudio realizado por la fundación FUNDES¹⁹, basado en nueve (9) bases de datos entre los años de 1998 y 2001, dentro de las ciudades de Bogotá, Cali, Medellín, Barranquilla y Bucaramanga, el número de PyMEs en Colombia es de 47.750, la distribución por sectores, se encuentra definida como sigue²⁰:



Gráfica 1: Distribución de la PyMe por sector 3

La distribución geográfica de las PyMEs en Colombia se representa en la siguiente Grafica 2:



Gráfica 2: Distribución Geográfica de la PyMe 4

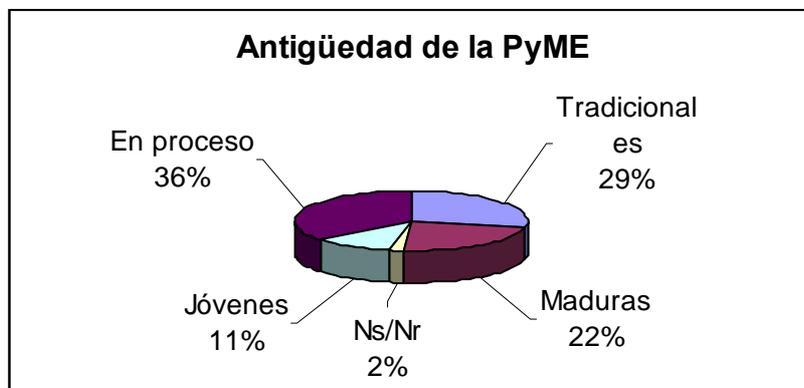
¹⁹ La red FUNDES es una organización que se encuentra ubicada en 10 países latinoamericanos y promueve la creación y el desarrollo sostenible y eficiente de las PyMEs en América Latina.

²⁰ **RODRÍGUEZ**, Astrid Genoveva. "La Realidad de la PyMe Colombiana: Desafío para el Desarrollo" Bogotá: FUNDES, 2003. 11 p.

En cuanto a antigüedad se puede establecer que el 87% tiene 6 años o más de haber sido creada, esto se explica ya que la mayoría de las PyMes comienzan siendo micro empresas y necesitan un tiempo largo para poder aumentar de tamaño y fortalecerse en el mercado. El siguiente gráfico muestra la antigüedad de las PyMes en Colombia ²¹ :

Antigüedad	Descripción (años)
Tradicionales	Más de 22
Maduras	Entre 15 – 22
Jóvenes	Menor a 6
En proceso	Entre 6 -15
Ns / Nr	0

Tabla 2: Referencia para antigüedad 5



Gráfica 3: Antigüedad de la PyMe Colombiana 6

4.2. Estado Actual del las Pymes Colombianas

Las PyMes representan un importante sector para la economía nacional, durante el año 2000, las PyMes aportaron el 35% de la producción

²¹ Ibid. 13 p.

industrial, 10% las pequeñas y 25% las medianas. Durante ese mismo año las pequeñas empresas generaron el 17% del empleo industrial y las medianas el 29%²². Durante el año 2002²³, representaban el 96% de los establecimientos comerciales, generaron el 40% de la producción bruta, el 34% del valor agregado, el 60% del empleo industrial y el 25 % de las exportaciones no tradicionales. En los países en vía de desarrollo, como Colombia, los sectores económicos que generen empleo, que brinden equilibrio social y que además distribuyen la riqueza de forma eficaz, son sectores de especial atención, y en este caso las PyMes representan uno de esos sectores, y por otro lado son un área llamativa para la inversión tanto económica como intelectual.

De acuerdo con CATI ²⁴ el índice de Creatividad Económica, el cuál une los aspectos relacionados con Innovación Tecnológica y Difusión de Tecnología, ubica a Colombia por debajo del 88% de los países. En innovación, Colombia se encuentra en el puesto 53 a nivel mundial, y sexto en América Latina. En el área de difusión tecnológica esta registrado en el puesto 53 y 9 en Latinoamérica. Los indicadores de innovación y Desarrollo tecnológico evidenciaron el mayor crecimiento en el área de innovación y un deficiente crecimiento en el área de desarrollo tecnológico. Según CATI, dentro del sector privado, la importancia de las actividades de investigación y desarrollo brindado por los empresarios, no cuenta con el valor que estas merecen, y esto se debe al costo que este tipo de actividades involucra.

Por otro lado, los porcentajes de producción de software han demostrado tener un importante crecimiento²⁵, llegando a satisfacer la

²² **RODRÍGUEZ**, Astrid Genoveva. Op. Cit. 15 p.

²³ Fuente ACOPI y Semana 2003

²⁴ **VALDES**, Luis Eduardo. "Situación Actual de la Informática en Colombia". CATI (Centro de Apoyo de la Tecnología Informática) [on-line] <http://www.cati.org.co>.

²⁵ **Ibid.** 8 p.

demanda nacional; una de las razones de este crecimiento, es la dificultad de adquirir software importado, y las tareas siguientes a la adquisición como soporte y mantenimiento. Pero a pesar del aumento en estos porcentajes de producción nacional, aún persiste en muchos sectores el uso de software importado en mayor medida, por ejemplo, el sector financiero reporta; 64% aplica software importado, 3% nacional y 14% es desarrollado en casa (in house).

En cuanto al uso de líneas de outsourcing²⁶, estas también han ayudado a proveer el software de las organizaciones, pero aún existen varias que se resisten a hacer uso de ella.

En cuanto a la mano de obra involucrada, se ha identificado que existe una muy calificada, esto sin contar con la gran oferta de ingenieros en el mercado²⁷.

La demanda de software se encuentra dividida en cuatro grupos: grandes empresas, PyMes, hogares y sector público. Tanto las empresas grandes como los hogares son suplidos en su mayoría, por el mercado externo, y el sector público presenta una demanda muy baja, debido a la falta de preparación para adoptar tecnología de Información y comunicación (TIC). El mayor número de PyMes considera al Software Nacional como el más competitivo²⁸.

4.3. Estructura Administrativa

Varios autores han definido una estructura administrativa para las organizaciones. La mayoría de estas definiciones coinciden en establecer un número de niveles dividido por el tipo de

²⁶ Ibid. 10 p.

²⁷ Ibid. 12 p.

²⁸ Op. Cit. 17 p.

responsabilidades que asume cada área de la organización. Henry Mintzberg establece la siguiente definición:

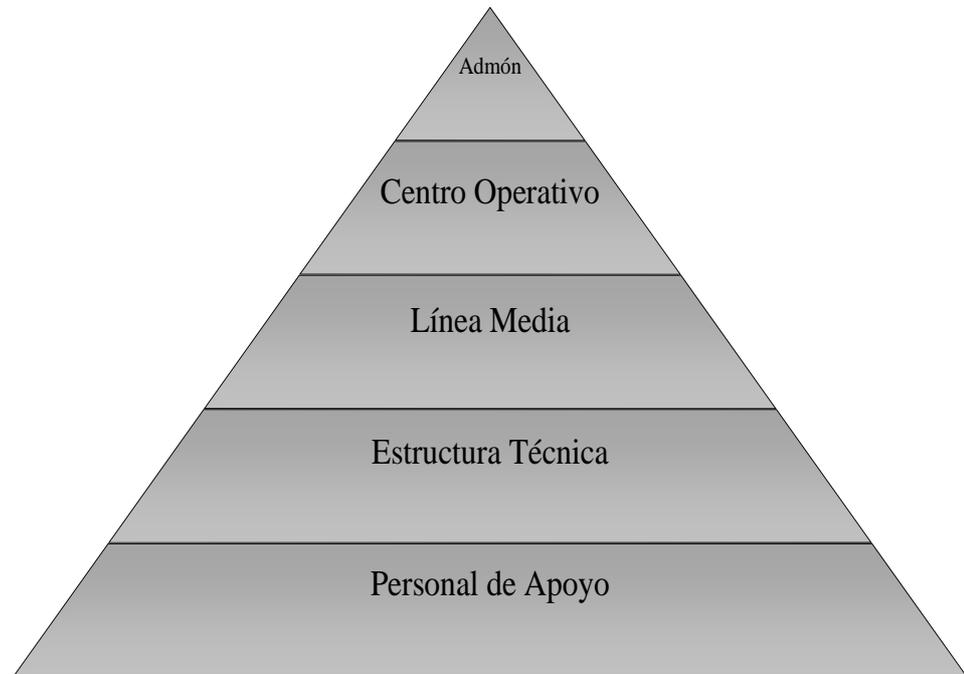


Figura 1: Componentes Básicos Organizacionales 7

Mintzberg, Henry. La Estructuración de las Organizaciones. Ed. Ariel. 1988

Mintzberg asegura que las organizaciones funcionan bajo un orden natural, y bajo este funcionamiento logran obtener una armonía funcional. En la parte inferior de la Figura 1 se encuentra el **personal de apoyo**, es decir, las personas que colaboran con varios procesos de la organización pero no toman parte de las decisiones estratégicas. En el nivel inmediatamente superior se encuentra la **estructura técnica**, donde se encuentran los analistas de sistemas que controlan el planteamiento y control de trabajo. La **Línea media** son los administradores intermedios entre el ejecutivo superior y operarios. El **centro operativo** se compone por las personas que realizan trabajos básicos de la organización. En la cúspide se encuentra el **nivel de administración superior**, donde se encuentran los gerentes generales

de las organizaciones, quienes toman las decisiones fundamentales del funcionamiento corporativo²⁹.

La siguiente estructura administrativa se encuentra definida en la Figura 2, en esta estructura se encuentra en primer lugar (de abajo hacia arriba) el área de **administración de operaciones**, donde se toman las decisiones a corto plazo basadas en reglas de negocio conocidas que suelen resultar el eventos esperados.

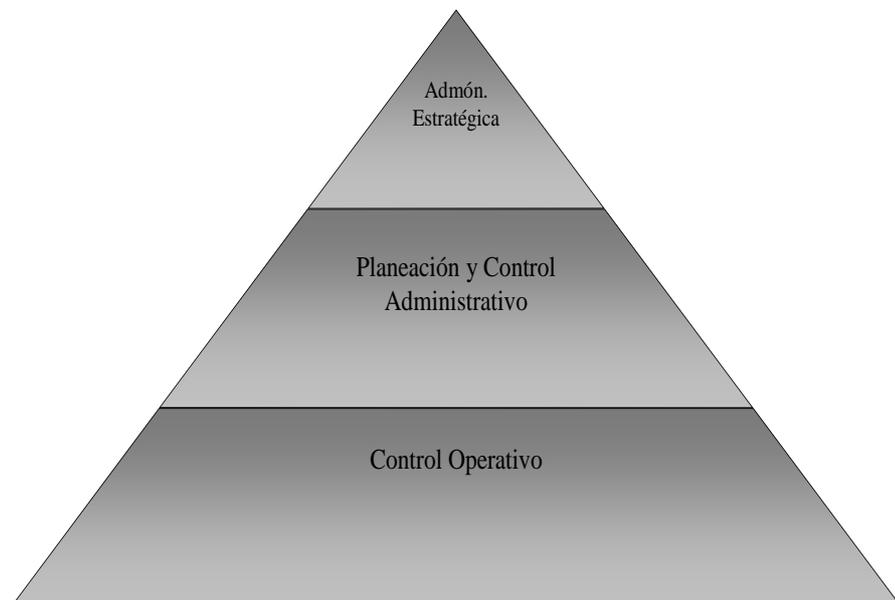


Figura 2: Niveles de administración 8

KENDALL, Kenneth. **KENDALL**, Julie. Análisis y Diseño de Sistemas de Información. Prentice Hall Hispanoamérica S.A. 1996

Se toman decisiones que intervienen en las áreas de control de inventario, envío, recepción, calendarización de trabajo y control de procesos de producción. El nivel de **Planeación y Control administrativo**, es decir la administración media, sus decisiones son tomadas a mediano plazo, ya que no cuentan con un alto nivel de certeza en la toma de decisiones, estas suelen ser menos estructuradas

²⁹ **MINZTBERG** Henry. "La estructuración de las organizaciones" 2ª Edición. Barcelona. Ariel. 1999.

que las de su nivel inmediatamente inferior. Finalmente encontramos el nivel de **Administración estratégica**, donde se toman las decisiones de mayor alcance y de largo plazo, el ambiente de toma de decisiones es bastante incierto, desde este nivel la organización es controlada como un TODO³⁰.

La siguiente figura es el resultado de la unión de las áreas comunes de ambas estructuras y será esta adaptación la que será utilizada de ahora en adelante:

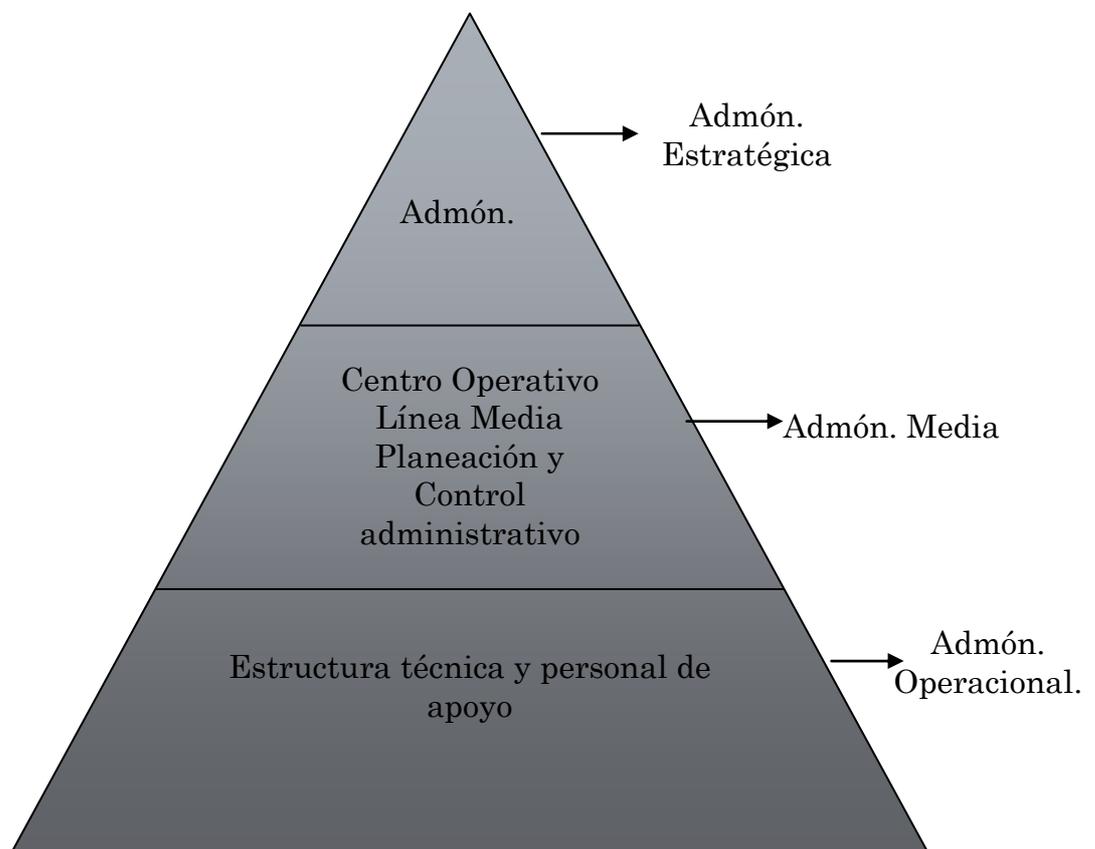


Figura 3: Niveles administrativos 9

Adaptada de: **Mintzberg**, Henry. La Estructuración de las Organizaciones. Ed. Ariel. 1988 & **KENDALL**, Kenneth. **KENDALL**, Julie. Análisis y Diseño de Sistemas de Información. Prentice Hall Hispanoamérica S.A. 1996

³⁰ **KENDALL**, Kenneth. **KENDALL**, Julie. Análisis y Diseño de Sistemas de Información. 3ª Edición. México. Prentice Hall Hispanoamérica S.A. 1996. 37 p.

Ahora bien, ¿Por qué es importante el concepto de la estructura organizacional en el contexto de Jerarquía de Componentes para PyMes? Es importante porque cada organización, por ende cada PyMe, se comporta como un sistema completo, cada uno compuesto a partir de ciertas áreas y niveles estratégicos para su correcto funcionamiento y la jerarquía de componentes, al estar orientada a la PyMe, debe atender las necesidades de cada área. Si se tiene en cuenta el concepto de estructuración administrativa en la creación de esta Jerarquía se asegura que dentro de esta última estarán consignados los elementos clave que la Pyme utilizará a lo largo y a ancho de toda la organización

4.4. Sistemas de Información Organizacionales

Un sistema de información es un conjunto formal de procesos que operan sobre una colección de datos estructurados de acuerdo con las necesidades de la empresa recopila, elabora y distribuyen la información necesaria para el funcionamiento de dicha empresa. En otras palabras, un Sistema de Información se define como:

“Una disposición de personas, actividades, datos, redes y tecnología Integrados entre sí con el propósito de apoyar y mejorar las operaciones cotidianas de una empresa, así como satisfacer las necesidades de información para la resolución de problemas y la toma de decisiones por parte de directivos de la empresa”³¹.

Los sistemas de información y en general cualquier aplicación desarrollada dentro de la organización, se analiza e implementa para cumplir determinados objetivos (requerimientos funcionales), la

³¹ **WHITTEN**, Jeffrey y otros. Análisis y Diseño de Sistemas de Información. 3ª Edición. Madrid: Mc Graw Hill, 1999. 39 p.

naturaleza de esos objetivos permiten clasificar cada sistema de información en un nivel administrativo como se muestra en la figura 4.

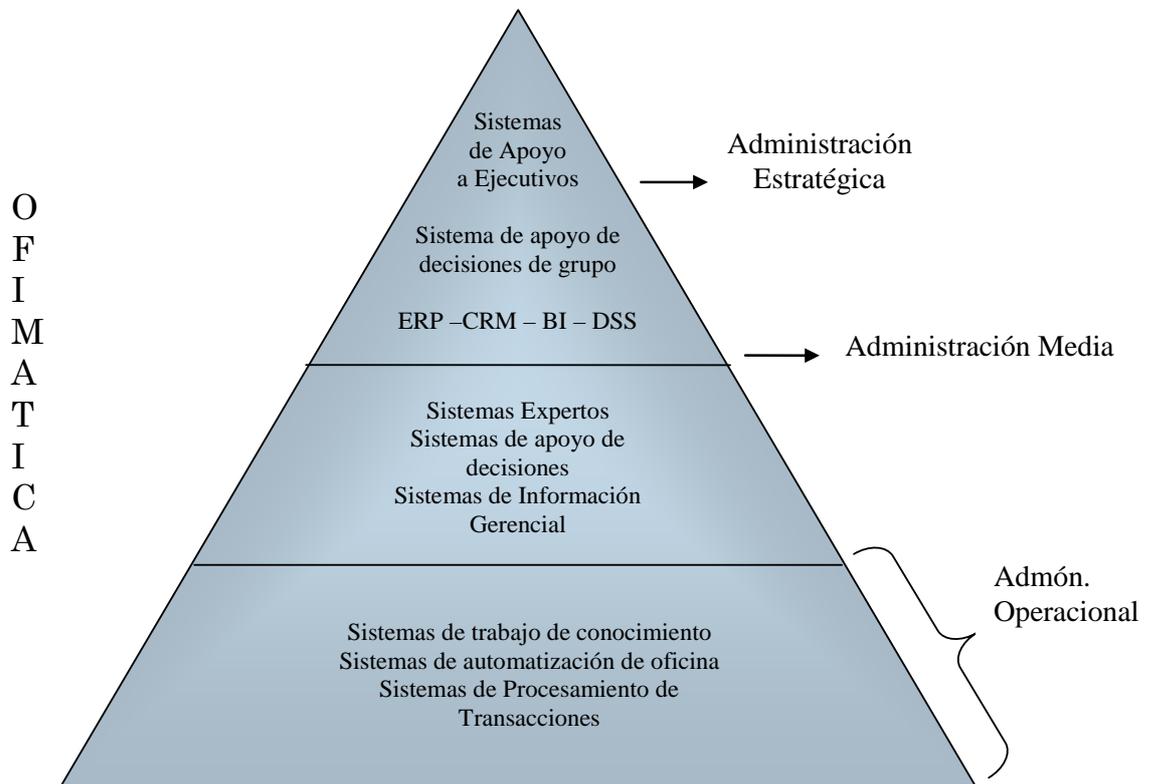


Figura 4: Sistemas de Información dentro de las organizaciones 10

Uniendo los niveles administrativos organizacionales con los sistemas de información presentes en el común de las organizaciones obtenemos:

El nivel de **Administración de Operaciones**, en esta área, se toman decisiones usando reglas preestablecidas, las cuales, ejecutadas correctamente, tienen un resultado previsible. El riesgo asociado a estas decisiones es mínimo, por el alto nivel de certeza dentro del ambiente de la toma de decisiones. En esta área suelen ser tomadas las decisiones que afectan la implementación de la calendarización del trabajo, control de inventario, control de procesos de producción,

algunos de los tipos de aplicaciones en esta área son: *sistemas de procesamiento de transacciones (TPS)*, *sistemas de automatización de oficina (OAS)*, *sistemas de manejo de conocimiento (KWS)*

La **Administración media**, realiza “decisiones de planeación y control a corto plazo sobre la manera en que son mejor asignados los recursos para satisfacer los objetivos organizacionales”, incluye los siguientes sistemas, entre otros: *sistemas expertos*, *sistemas de apoyo de decisiones (DSS)*, *sistemas de información gerenciales (MIS)*.

Finalmente, la **Administración estratégica**, en este nivel son tomadas las decisiones a largo plazo, decisiones que afectarán los niveles medio y operativo por los meses o años venideros; el ambiente de toma de decisiones es bastante incierto, esto hace necesario contar con una visión amplia del estado de la empresa, incluye entonces, los siguientes sistemas: *sistema de apoyo de decisiones de grupo (GDSS)*, *sistemas de apoyo a ejecutivos (ESS)*.

En la actualidad, algunos sistemas de administración estratégica han empezado a cobrar cada vez más importancia, debido al impacto que tienen sobre el nivel de competitividad empresarial, y son incluso aplicaciones que representaron prioridad en términos de inversión para las PyMes colombianas durante el año 2004³², ejemplo de estos sistemas son: *ERP (Planeación de Recursos Empresariales)* (39%), *CRM (Administración de Relaciones con los Clientes)* (31%), *BI y DSS: (Inteligencia para los Negocios y Sistemas de Apoyo a Decisiones)* (3%).

³² VILLATE, Ricardo. “El imperativo de la infraestructura de TI en América Latina”. Estudio Realizado por IDC: Latin America System and Network Software go to market. [on-line] <http://www.idc.com>

Este gráfico, representa entonces, los sistemas de información relacionados con cada una de las áreas de administración de las organizaciones:

Encontramos a lo largo de toda la organización aplicaciones de **Ofimática**, estas buscan perfeccionar la comunicación entre todos los trabajadores de la información³³.

Por otra parte tenemos el portafolio potencial de TIC's de una PyMe³⁴ que se compone de cuatro (4) diferentes tipos de TIC's, cada uno representa un conjunto de elementos de hardware, software y telecomunicaciones, que cumplen las siguientes tareas:

1. **TIC Transaccional:** Soportan las operaciones empresariales del día a día. Busca principalmente reducir el costo de las operaciones e incrementar la productividad.
2. **TIC de Información:** Proveen información a las áreas e individuos que la requieren internamente en la empresa. Debe soportar la toma de decisiones a nivel operativo y gerencial.
3. **TIC Estratégica:** Soportan una posición competitiva dentro del sector industrial, o posibilitar mayor innovación en productos y servicios.
4. **Infraestructura de TIC:** Es el conjunto de tecnologías que soportan todas las anteriores tecnologías. Comprende toda la infraestructura de redes internas, comunicaciones públicas, servidores, computadores de escritorio, software de operación, bases de datos.

³³ WHITTEN, Jeffrey y otros. Op. Cit. 60 p.

³⁴ BROADBENT, Mariane. WEILL, Peter. "Leveraging the new Infraestructura" .Harvard Business Scholl Press. United Stated of America. 1998.

La siguiente figura hace explícita la relación entre los diferentes niveles administrativos de las PyMes y las Tecnologías de Información y Comunicación inherentes a cada uno de estos niveles:

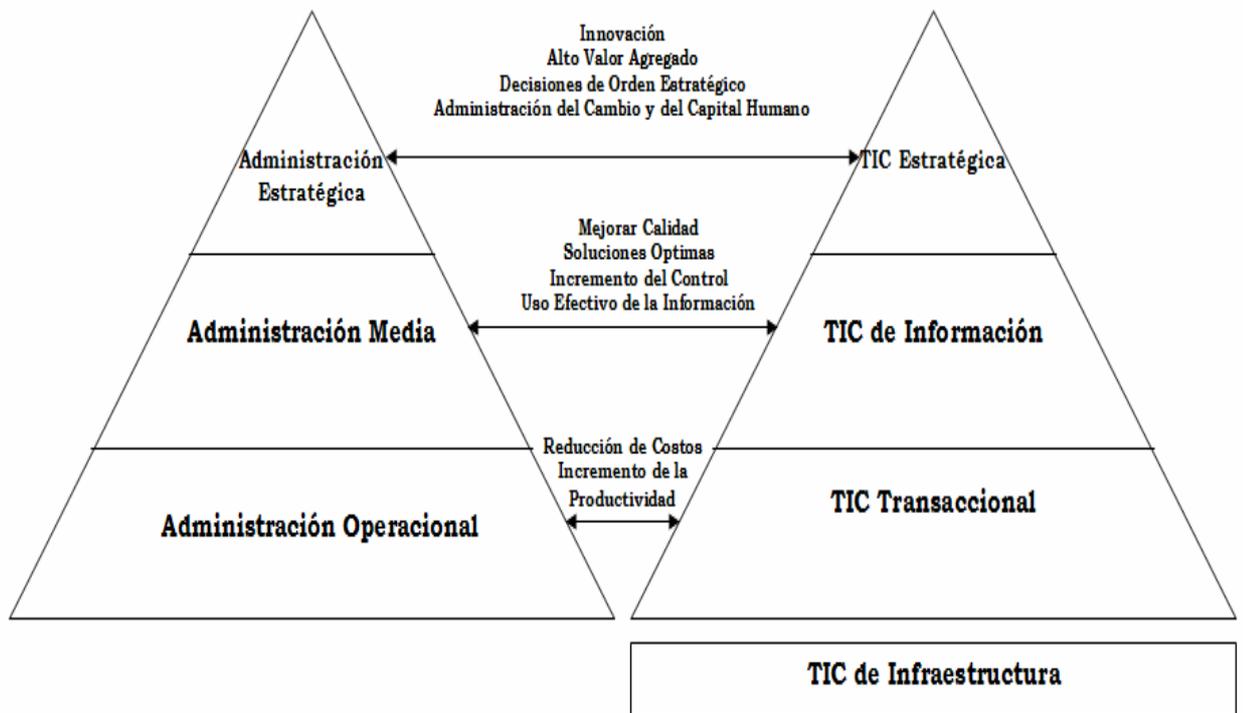


Figura 5: Portafolio Potencial de TIC's y Niveles de Admón 11

Anteriormente se expuso la importancia de la relación entre la estructura administrativa y la jerarquía de componentes, ahora es necesario explicar la relación entre los diferentes sistemas incluidos en esa estructura y esta jerarquía. Pues bien, cada nivel se encarga de atender las necesidades de las áreas que lo componen, pero ¿a través de cuáles elementos logra satisfacer esas necesidades?, la respuesta esta en los sistemas que se implementan. En otras palabras, la base de los componentes a seleccionar para componer la jerarquía se fundamentan en las necesidades que buscan satisfacer cada uno de estos sistemas.

El objetivo siguiente es estructurar la composición de estos sistemas a través de componentes de software.

4.5. Oportunidades y Amenazas del Mercado

“Ninguna empresa, en ningún país puede permitirse el lujo de desentenderse de la necesidad de competir”³⁵. Las PyMes Colombianas no son la excepción a la regla, así que deben plantearse estrategias que incrementen su nivel competitivo, para lograrlo, deben reconocer cuáles son las oportunidades y las amenazas del mercado:

Según las cifras del estudio de CATI, el índice de creatividad obtenido por el país, deja al descubierto el bajo potencial competitivo de las PyMes en cuanto a innovación tecnológica, y el índice de desarrollo tecnológico demuestra también un rendimiento deficiente.

Uno de los principales problemas identificados por el Banco Interamericano de Desarrollo (BID) es el rezago tecnológico de las PyMes, esencialmente en los siguientes aspectos:

- ❖ Bajo nivel de información y con frecuencia deficiente o negativa actitud empresarial hacia las nuevas tecnologías y sus procesos de adaptación.
- ❖ Dificultad individual para realizar investigación, desarrollo o incorporar la tecnología o los servicios tecnológicos.

³⁵ PORTER, M. “On Competition”. Boston, Harvard Business School Press, 1998.155-350.

- ❖ La inversión mínima para poner en marcha los Sistemas de Información necesario para apoyar el negocio, sigue siendo alta.

La actitud negativa empresarial, puede ser comprensible, ya que en muchos casos los procesos de implementación de nuevas tecnologías no es un proceso fácil para las organizaciones. Sin embargo, la encuesta realizada durante el período comprendido entre Agosto y Septiembre de 2004³⁶ muestra un alto interés por parte de las PyMes por adquirir nuevas herramientas tecnológicas a corto plazo.

La oportunidad está en crear nuevas tecnologías, las cuales estén al alcance de cualquier pequeña y mediana empresa, económicas y fáciles de implementar en cualquier proceso de negocio empresarial.

Ahora bien, analizando el funcionamiento interno de las PyMes, pueden identificarse algunas de las actividades que generan ventaja competitiva al ser apoyadas con Tecnologías de Información³⁷:

- ✓ Producción: Planificar y monitorear los procesos de producción mejoran el uso de insumos y el tratamiento del producto terminado por parte de la comercialización y ventas.
- ✓ Comercialización y Ventas: Controlar los productos que están y no están disponibles para venta, la relación cliente - venta y la posibilidad de contar con información histórica de la producción.
- ✓ Servicio al Cliente: Información consolidada del estado de cada venta, beneficia tanto al cliente, como a la organización.

³⁶ Estudio de campo realizado por el grupo de investigación PICS. (Ver Anexos)

³⁷ Journal of Information System and Technology Management Vol. 1, No. 1, 2004, 03-26 p.

- ✓ Mercadeo: Conocer la competencia, definir indicadores de posicionamiento de mercado, son factores que ayudan a incentivar el espíritu competitivo.
- ✓ Investigación y desarrollo: Establecer pautas de consumo y tendencias, son elementos claves para la toma de decisiones y para el control de cambios.

El interés por apoyar las PyMes es una de las principales labores que el gobierno nacional adelanta, a continuación, algunas de las entidades que secundan esta intención con algunas de las propuestas:

- ❖ Colciencias: “Financiación de Proyectos de Innovación y Desarrollo Empresarial mediante la modalidad de Cofinanciación”. Apoya proyectos de investigación, desarrollo tecnológico e innovación que se realicen entre una o varias empresas nacionales del sector productivo radicadas en Colombia, y una o varias entidades ejecutoras universidades para consolidar estrategias.
- ❖ Bancoldex: “Programa EXPOPYME” financia proyectos bajo el mecanismo de redescuento u operaciones de leasing. Por otra parte existe una línea de crédito compartida con Colciencias que financia proyectos de innovación y desarrollo tecnológico para fortalecer la competitividad de productos y servicios en el mercado mundial. Esta línea financia hasta el 80% del proyecto sin superar los 10.000 salarios mínimos mensuales vigentes (BANCOLDEX 2003).

- ❖ Ministerio de Comercio, Industria y Turismo: “PNPC: Programa Nacional de Productividad y Competitividad”. Financia programas de productividad e innovación al interior de las empresas.
- ❖ Banco Interamericano de Desarrollo: “Tecnología de Información y Comunicación para la PyMe”, buscan facilitar la adopción de TI en PyMes para fortalecer el desarrollo regional equitativo y sostenible.

Es evidente que la necesidad de incrementar la competitividad en las PyMes representa un factor clave para el mejoramiento de la economía del país, las propuestas mencionadas son sólo algunas de las cuáles el gobierno ha puesto en marcha, un gran número de esas propuestas están directamente relacionadas con el desarrollo de tecnologías de información y comunicación al interior de las organizaciones.

4.6. Resultados Obtenidos del Estudio de Campo

Durante el estudio de campo se obtuvieron otros resultados no formales. Se observó que el secreto industrial es mal entendido por las PyMes, la manera en como la información es revelada y la poca aceptación de algunas empresas a que las universidades investiguen dentro de estas es muy notable. Las PyMes no son especialistas, aunque se buscó un sector de negocio en particular, se encontró que las PyMes no buscan especializarse en una labor en particular, sino que dependiendo de la rentabilidad de sus proyectos enfocan sus esfuerzos. Las PyMes no mantienen una comunicación de mutuo beneficio con las instituciones académicas, y se vieron sorprendidas, cuando se involucraron en el desarrollo del presente trabajo ocasionando desinterés y desconfianza a

la hora de entregar datos necesarios. Esto permite concluir que la investigación en las organizaciones de negocios es casi nula.

En la sección anterior fueron expuestos algunos de los elementos que sobresalen en el mercado nacional, a continuación se muestran los principales resultados obtenidos por el Grupo de Investigación PICS realizado dentro de quince (15) PyMes bogotanas:

1. El nivel de conformidad de las PyMes hacia el software que utilizan en la actualidad confirma que aún existen elementos dentro de esta área que pueden ser sujetos a mejoras, 13.33% de la población afirma estar completamente conforme, mientras que el 66.67%, el 13.33% y 6.67% presentan un nivel conforme, neutral e inconforme respectivamente.
2. El interés demostrado por parte de las PyMes frente a la posibilidad de adquirir nuevas herramientas tecnológicas confirma que es un buen momento para ofrecer herramientas tecnológicas atractivas para el sector.
3. Las herramientas de tecnologías de información y comunicación que las PyMes buscan³⁸ son aplicaciones extensibles y modificables, las cuáles permitan verificar que las necesidades por las cuáles este fue creado se satisfacen correctamente (rastreadibilidad de requerimientos) y además sean económicas en cuanto al desarrollo y administración.
4. Las herramientas tecnológicas que buscan las PyMes son³⁹ herramientas extensibles y modificables, esto significa que el DSBC

³⁸ Criterios de Mantenimiento y Criterios de Costo (ver Anexos)

³⁹ Criterios de Desempeño (Ver Anexos)

es una muy buena alternativa, ya que la naturaleza de los desarrollos basados en componentes son precisamente la ampliación de los sistemas a nuevas funcionalidades y la renovación de las herramientas cuando esto se hace necesario.

5. Los principales obstáculos percibidos y/o encontrados en el área del desarrollo de software son principalmente el tiempo (51.67%), este resulta ser limitado, el costo en el que incurre la organización por cada desarrollo interno realizado (46.67%).

5. CLASIFICACIONES DE COMPONENTES DE SOFTWARE

Para poder construir una jerarquía de componentes es importante estudiar algunas de las clasificaciones de componentes existentes, y de acuerdo a esto, construir una jerarquía que contenga las fortalezas de cada una de ellas. Se resaltan, aquellas fortalezas que puedan cumplir con las necesidades presentadas por las PyMes en el estudio de campo; además, estas clasificaciones contienen las tecnologías TICS, de acuerdo a los niveles de sistemas de información presentados en el capítulo anterior.

5.1. Infraestructura común de componentes en capas

El *diseño de capas de componentes* que se muestra en la figura 6, incluye los siguientes componentes⁴⁰:

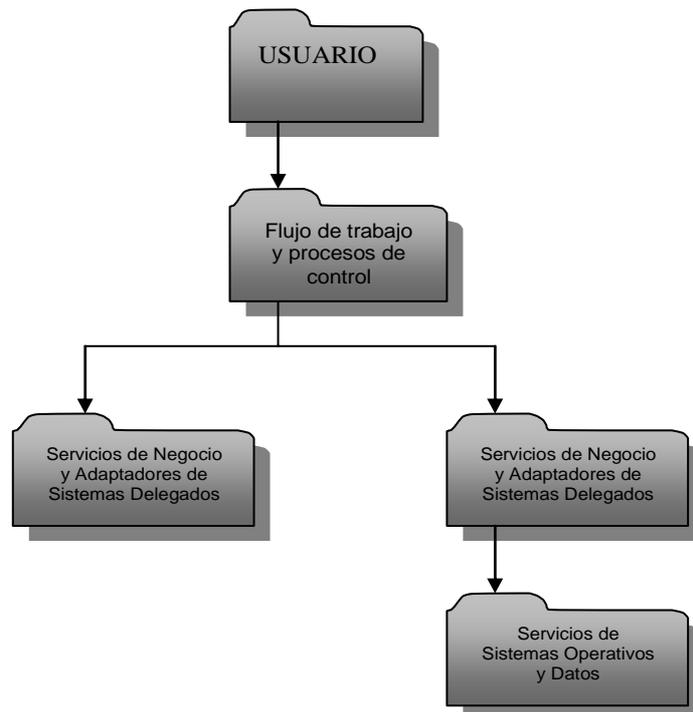


Figura 6: Infraestructura común de componentes en capas 12

⁴⁰ HEINEMAN, George. COUNCILL, William. "Component Based Software Engineering". Addison- Wesley. 2001. pág 818.

Los elementos de la capa de usuario proveen interfaces externas y el conocimiento de la interacción con el usuario (GUI – Interfaces Gráficas de usuario); esta interacción no es diferente a la representada en los casos de uso. Estos componentes requieren de servicios que se activan a través de la solicitud por parte del usuario.

La capa de Flujo de trabajo y procesos de control manejan procesos complejos, automatizados del negocio, que interactúan con servicios del negocio.

Servicios de Negocio y Adaptadores de Sistemas Legacy, donde se provee la implementación de reglas de negocio y la actividad operacional, a través de implementaciones de objetos de negocio o sistemas legacy ubicados por debajo de nuevas interfaces de componentes. Estereotipos comunes:

Finalmente encontramos la capa de Servicios de Sistemas Operativos y Datos, donde están presentes los componentes que proveen la funcionalidad que interactúa con el ambiente de almacenamiento persistente, incluyendo sistemas de administración de bases de datos y sistemas de archivos. Estos componentes son frecuentemente dependientes de las APIs (interfaces de programas de aplicación) que implementa el sistema operativo sobre el cual se ejecuta el sistema, para proveer un mayor grado de independencia se diseñan interfaces genéricas y se une la implementación de esa interface al sistema operativo en particular.

Dentro de esta definición de capas diseño de componentes por capas también se encuentra la capa de Middleware, donde residen los elementos que facilitan la integración de componentes a través de las múltiples plataformas y desarrollos.

Estas capas que permiten clasificar componentes, y basados en estas construir aplicaciones que incluyan alguno de los elementos que pertenecen a cada nivel.

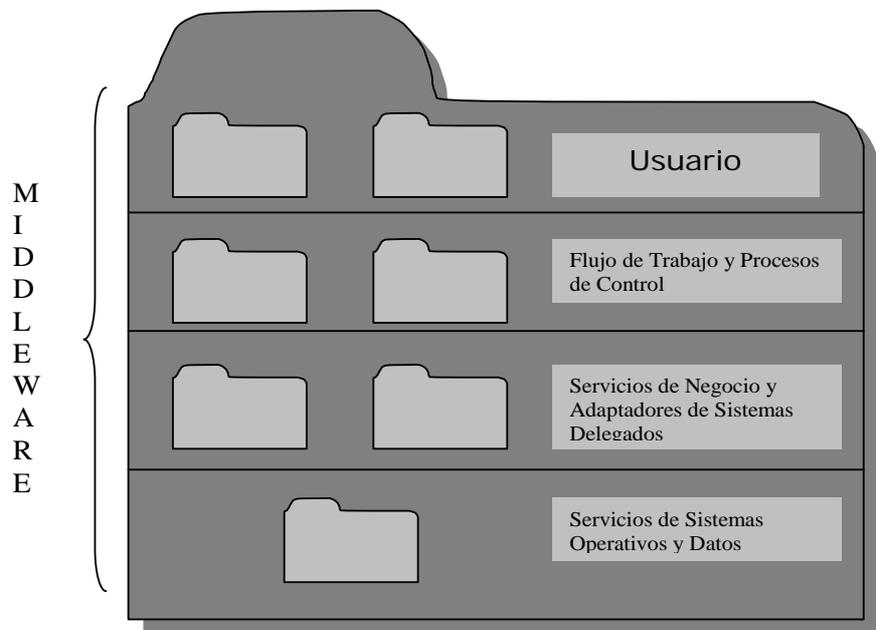


Figura 7: Clasificación de Componentes definida por medio de capas 13

5.2. Arquitectura de Sistemas de SW Basados en Componentes

La Arquitectura de los Sistemas Basados en Componentes⁴¹, incluye tres capas como se muestra en la figura 8. Cada tipo de aplicación es representada como un sistema independiente, construido por medio de componentes desde nivel técnico hasta componentes encargados de contener procesos de la lógica del negocio. Los componentes que hacen parte de cada capa pertenecen a un nivel específico, y apuntan a un negocio en particular o áreas de dominio de aplicación. Esta

⁴¹ **CAI**, Xia. **LYU**, Michael. "Component Based Software Engineering: Technologies Development Frameworks, and Quality Assurance Schemes" 4 p.

infraestructura de componentes apoya las interfaces de plataformas independientes, permitiendo flexibilidad y sistemas abiertos.

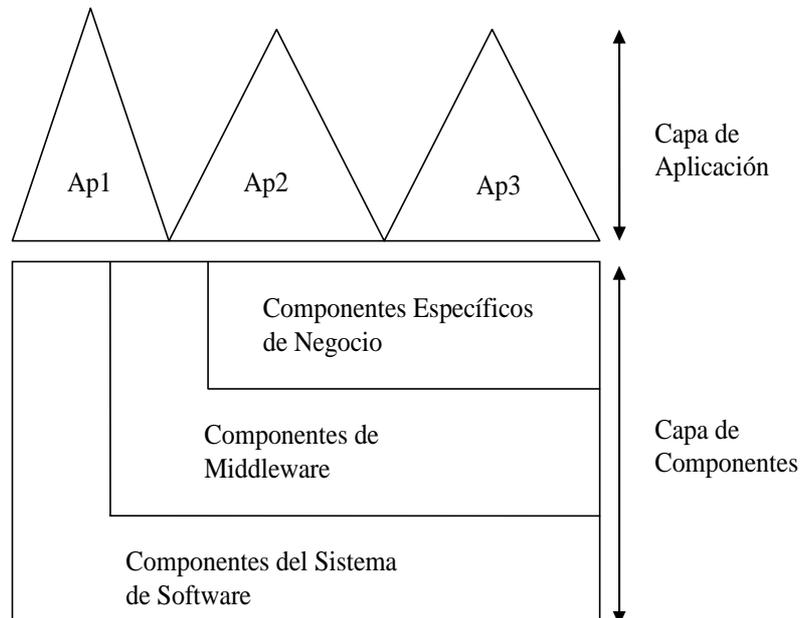


Figura 8: Capas de las aplicaciones basadas en la reutilización dirigida a la Ingeniería de Software de Negocio 14

Los *componentes específicos de negocio* contienen exclusivamente funciones del dominio especializado, son componentes de alto nivel, los cuáles, por medio de su integración al sistema satisfacen una meta de negocio especial.

Los *componentes de middleware*, son elementos que interrelacionan los diferentes dominios a los que pertenecen los componentes que hacen parte del sistema. Si el componente se encuentra más cercano a los de la capa inferior, se distinguen por ser los que administran la comunicación técnica entre los componentes de negocio, por ejemplo, aplicaciones que se encargan de llamadas a funciones remotas, ahora, si el componente trabaja en mayor escala con los componentes específicos de negocio (capa superior) son componentes administradores de bases de datos, o software que puede permitir la integración de componentes de negocio para construir aplicaciones

complejas, como un sistema de administración de flujo de trabajo, en otras palabras, los componentes de middleware apoyan específicamente aspectos técnicos de colaboración, pero no son aspectos concretos del dominio del negocio.

Los componentes del sistema de software, son componentes exclusivamente técnicos, que permiten la operabilidad del sistema.

5.3. Infraestructura de componentes orientada a la reutilización

La Reutilización e Infraestructura de Componentes de la Ingeniería de Software de Negocio⁴², es una clasificación que busca definir las diferencias claves existentes entre los componentes, su complejidad, alcance, nivel de funcionalidad, etc. Esta clasificación se enuncia a continuación:

1. Componentes GUI: Estos son los tipos de componentes que se encuentran con mayor frecuencia dentro del mercado. Los Componentes GUI, son aquellos que permiten construir interfaces para el usuario. La construcción de un componente GUI robusto exige por parte del desarrollador un alto nivel de conocimiento y compromiso con la ingeniería de software basado en componentes.
2. Componentes de Servicio: Este tipo de componentes proveen servicios comunes requeridos por diferentes aplicaciones, como acceso a bases de datos, servicios de mensajería y transacciones e integración con servicios del sistema. Una de las principales características de estos componentes, es que todos necesitan usar sistemas o infraestructura adicional para poder ejecutar su funcionalidad.

⁴² HEINEMAN, George. COUNCILL, William. Op. Cit. 82 p.

Otra función común de estos componentes es facilitar la Integración de las Aplicaciones de Negocio (EAI), por esta razón es común encontrar combinados estos componentes con herramientas EAI como Servidores de Transacciones, motores de transacción de datos y sistemas de flujo de trabajo.

Estos componentes, por si mismos no proveen funcionalidad al sistema, son menos complejos que los componentes de dominio, y más que los componentes GUI.

3. Componentes de Dominio: Estos componentes, cuando son reusables, pertenecen sólo a un dominio de aplicación específico, y no pueden ser re-utilizados fuera de él. Estos componentes son difíciles de diseñar y construir. Por hacer parte de un dominio de aplicación en particular, deben contar con una infraestructura de aplicación propia. El encargado de diseño e implementación del componente requiere un alto grado de conocimiento y experiencia dentro del dominio. Los casos de negocio que requieran dentro de su implementación componentes de dominio, requerirán tanto componentes GUI como de Servicio.

5.4. Clasificación por Facetas

Para la descripción de un componente, se tiene el modelo 3C enunciado por Pressman⁴³, que establece que la interfaz de un componente debe contener tres aspectos: concepto, contenido y contexto; sin embargo, y para la clasificación del componente hace falta

⁴³ **Pressman S** Roger. "Ingeniería de Software un enfoque practico". , Mc Graw Hill , 2002. 480-487 p.

un modelo basado en los métodos de la Biblioteconomía, que incluye tres métodos de clasificación:

1. Clasificación enumerada.
2. Clasificación por facetas.
3. Clasificación por atributos y valores.

Cada una de estas ofrece una ventaja significativa para identificar un componente, en el caso de la clasificación enumerada, ofrece un modelo jerárquico de clases, pero es necesario aplicar procesos de ingeniería de dominio para identificar las entradas a esta; adicionalmente, también puede permitir la repetición de elementos en ciertas ocasiones. La clasificación por facetas, analiza cierta área del dominio y se identifican un conjunto de características básicas (facetas) haciendo uso de un esquema de descripción [función, tipo de objeto, tipo de sistema]. La clasificación por atributos y valores se restringe a una zona del dominio y se le dan valores a estos atributos; su principal diferencia con la anterior clasificación, es que no hay límite de atributos y estos no tienen prioridades.

De las anteriores clasificaciones, puede afirmarse que la más versátil, flexible y adaptable, es la clasificación por facetas, ya que ayuda en el proceso de construcción de la Jerarquía de Componentes. Todos los modelos descriptivos utilizan el concepto de Clasificación – Dominio, que permite concluir que hay una relación de Jerarquía – Dominio, ya que la Jerarquía nace de del concepto de clasificación.

La “*clasificación por facetas*”, utiliza un esquema de descripción integrado por:

1. Función

2. Tipo de Objeto
3. Tipo de Sistema

La *función* se toma como el carácter operacional y funcional que cumple cada componente dentro de un sistema; se tienen en cuenta las generalidades descriptivas que tendrán los componentes. Esta funcionalidad es general en relación a otros sistemas, se debe comprender que la jerarquía es como un sistema compuesto por niveles, donde el nivel inferior se ve como la base sobre la cual actúan los componentes del siguiente nivel hasta llegar al último que tendrá el mayor grado de composición, lo que construye de manera genérica la jerarquía.

El segundo aspecto es el *tipo de objeto*, el cual se especifica al clasificar un componente; por ejemplo, si se tiene un componente J2EE se especificará que tipo de componente es: EJBContainer, WebContainer u otro, de igual manera, se dirá si se trata de un componente más complejo como un subprograma u otro tipo de componente como un módulo, utilizados en el lenguaje ADA.

El tercer aspecto es el *tipo de sistema*, también será definido al momento de clasificar un componente, aquí se indicará a que tipos de sistemas puede ser aplicado este componente, en que sistemas es usualmente utilizado, si obedece a algún tipo de patrón o una indicación que le permita su correcto desempeño dentro de una arquitectura basada en componentes.

Uno de los beneficios de la clasificación por facetas, es que se puede acceder a un ítem clasificado sin saber el nombre de este, ya que se puede alcanzar por medio de un entendimiento compartido de lo que este es, en términos de varias categorías de información mutuamente

excluyentes. Para buscar por medio de un software, la búsqueda de un conjunto de facetas no se hace por medio de un método exhaustivo. Por otra parte, una nueva faceta puede ser agregada en cualquier momento, sin causar confusión en la clasificación de los ítems.

- ✓ Una Clasificación por facetas, se centra en las características importantes, esenciales o persistentes de objetos, esto es de utilidad para los repositorios que cambian rápidamente.
- ✓ No es necesario saber el nombre de una categoría en especial, para la búsqueda de un elemento clasificado. Se pueden simplemente asumir características o nombres (Asunciones)⁴⁴.

5.5. Repositorios de Componentes y sus Clasificaciones.

Las clasificaciones existentes de componentes en Internet, brindan una serie de clasificaciones de componentes ordenados alfabéticamente, en categorías muy generales, enfocados a una arquitectura o tecnología específica, entre otras.

Las clasificaciones alfabéticas se complementan con las de grandes categorías, las cuales son patrocinadas y avaladas por la industria del software; por ejemplo: www.componentsource.com. Si bien, esa clasificación resulta práctica para los desarrolladores, no ayuda a situar y contextualizar los componentes dentro de un dominio y una función específica que cumple un componente. Esto ofrece como resultado búsquedas en los catálogos no depuradas, que ofrecen componentes no

⁴⁴ **TOOL** David , DataChannel, presentation at Documentation Santa Clara (conference), March 10, 1998[on-line] <http://www.kmconnection.com>

contextualizados bajo un dominio y una función operacional del componente.

Las clasificaciones por arquitecturas resultan ser de mucha utilidad para desarrolladores expertos, que de antemano saben resolver gran parte del problema basados en una arquitectura de software, estas clasificaciones por lo general se especializan en un tipo de tecnologías ya existentes. Sin embargo, para aquellos que no pueden costear un desarrollo completo y acoplado a una arquitectura específica que a su vez dependen de ciertas tecnologías, les resulta difícil el remplazar los componentes o utilizar otros que no se encuentren basados en estos tipos de diseño o tecnología.

Los sistemas basados en componentes necesitan software adicional, que no pertenece a la base del sistema en si, pero que provee servicios que son necesitados desde diferentes sistemas de aplicación como GUI, Middleware; además, de software que soporte el ciclo de vida durante el desarrollo de un sistema de aplicación. Para un sistema construido con componentes de negocio, es necesario tener un dominio cruzado de componentes, que permita una coordinación técnica dentro del sistema⁴⁵.

⁴⁵ **TUROWSKI** Klaus. **FELLNER** Klement J. "Classification Framework for Business Components" Institute for a Technical and Business Information Systems Magdeburg Germany 5 p.

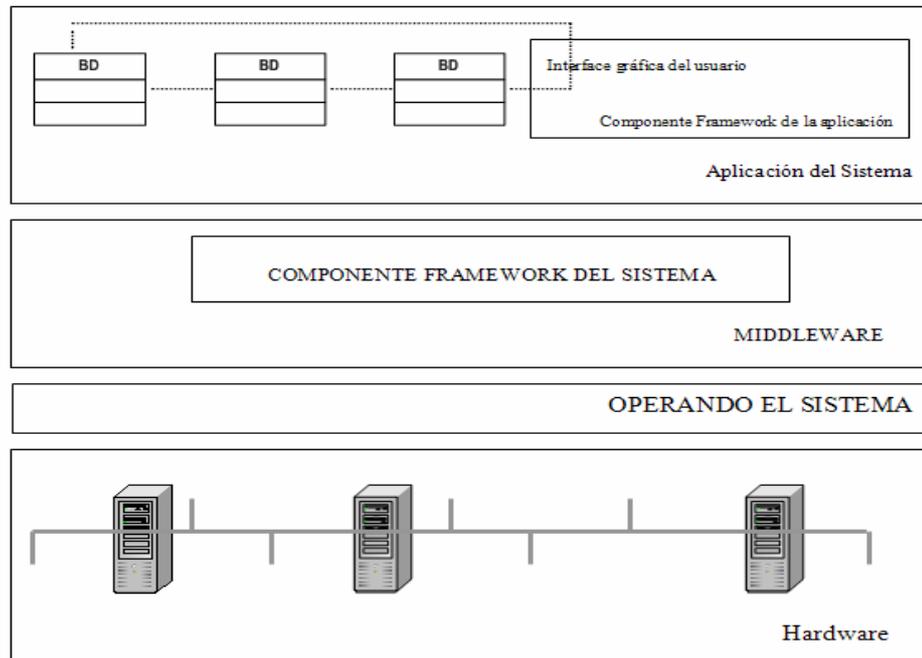


Figura 9: Arquitectura Genérica de un Sistema basado en Componentes⁴⁶

Esta clasificación de componentes, sitúa las distintas capas en las que pueden ser ubicados componentes independientemente de su origen, tecnología o arquitectura.

⁴⁶ **AKSIT**, Mehmet. **MEZINI**, Mira. **UNLAND**, Rainer. "Objects, Components, Architectures, Services and Applications for a Networked World". Ed Springer. 2002. 63 p.

6. JERARQUÍA DE DOMINIO DE COMPONENTES DE SOFTWARE PARA PYMES

La Jerarquía de componentes definida en este proyecto de investigación busca en primer lugar estandarizar el desarrollo de Software Basado en Componentes dentro de las PyMes, dando a los desarrolladores de software un lenguaje común cuando se habla de SBC. Esta jerarquía contiene una clasificación dentro de la cual, la gran mayoría de componentes existentes en el mercado pueden ser catalogados, esto es lo que permite identificar cuál o cuáles son los componentes que necesita determinada organización para implementar un sistema específico.

Tomando en cuenta los diferentes tipos de clasificaciones de componentes que existen, estas aportan diferentes conceptos que ayudan en la creación de una jerarquía de componentes. Un aspecto a tomar en cuenta en la construcción de esta jerarquía es que, aunque el DSBC no se apoya sobre una arquitectura en particular, todos los sistemas construidos a partir de este tiene niveles generales y parecidos, incluso los modelos que no se apoyan en el DSBC.

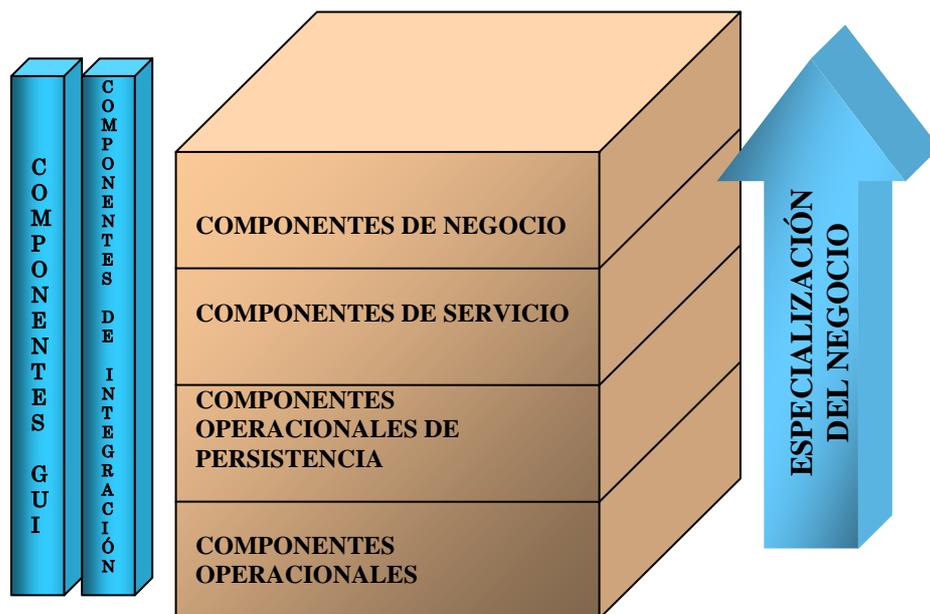


Figura 10: Jerarquía de Componentes de SW y Arquitectura de SW sugerida

6.1. Componentes de Negocio

Estos componentes son los encargados de contener la lógica de negocio especializada y contienen los módulos más representativos del dominio de la solución; esta es la razón por la cual pertenecen solo a cierto tipo de aplicaciones. El mayor valor agregado de un sistema se logra por medio de la integración de estos componentes.

6.1.1. Características:

1. Realizan tareas específicas que involucran requerimientos de dominios de aplicación especializados.
2. El éxito de la funcionalidad del componente depende en gran medida de la calidad⁴⁷ del sistema que lo soporta. Esto quiere decir que necesitan una infraestructura previa para cumplir con su propósito.
3. Pueden ser compuestos por otros componentes, módulos o subsistemas.
4. Su implementación suele ser compleja por la lógica de negocio que envuelven, ya que proveen un conjunto de servicios bien definidos en un dominio específico.
5. La complejidad y los objetivos finales de la aplicación permiten decidir si son o no este tipo de componentes necesarios para la implementación o adaptación del sistema.

⁴⁷ Calidad de un Sistema de Software:

6. El proceso de integración de este tipo de componentes requiere un proceso de Ingeniería de Dominio, el cual sirve para desarrollar y mantener modelos de dominio y arquitecturas del dominio⁴⁸.

6.1.2. Ejemplos:

Sistemas de Negocios y Adaptadores de Sistemas Legacy, Componentes de Dominio, Sistemas expertos, ARPs, ERPs, los componentes específicos de negocio.

6.2. Componentes de Servicio

Estos componentes se encargan de ejecutar procesos complejos y automatizados. Los tipos de servicios que ofrecen son requeridos por diferentes tipos de aplicaciones.

6.2.1. Características:

1. El éxito de la funcionalidad del componente depende en gran medida de la calidad del sistema que lo soporta.
2. Pueden ser compuestos por otros componentes, módulos o subsistemas.
3. Pueden administrar la interacción con un número de servicios o con uno o más componentes de negocio para completar procesos de negocio robustos.

⁴⁸ Dominio de una Aplicación: Representa todos los aspectos del problema de Usuario. Esto incluye el ambiente físico donde se ejecutará el sistema, los usuarios, y sus procesos de trabajo.

4. Pueden ser adaptados e interactuar, en varios dominios.

6.2.2. Ejemplos:

Su funcionalidad puede abarcar campos como: la administración, análisis y mantenimiento de redes, almacenamiento y análisis de datos, resolución o manejo de algoritmos, APIs especializados.

6.3. Componentes Operacionales de Persistencia

Estos componentes se encargan de interactuar con la información persistente del sistema. No envuelven lógica del negocio, al igual que los de servicio, y pueden ser requeridos en un mayor intervalo de dominios de aplicación.

6.3.1. Características:

1. La generalidad de las funciones que desarrollan hacen de estos componentes adaptables.
2. Aunque son componentes importantes para un sistema, no generan valor agregado per se.
3. Por medio de la especialización de un componente de este tipo puede obtenerse un componente de servicio.
4. Se encargan del proceso de administración de transacción entre el lugar de almacenamiento (base de datos generalmente) y el negocio.

5. Se encargan del control sobre la seguridad de la información que el sistema utiliza.

6.3.2. Ejemplos:

Paquetes de ofimática, motores de BD, Servidores de aplicación.

6.4. Componentes Operacionales

Estos componentes ofrecen la base operativa para el funcionamiento de un sistema sobre el Hardware; sobre estos se crean los modelos de arquitectura de software. Son comunes a la gran mayoría de las aplicaciones.

6.4.1. Características:

1. Representan la infraestructura primaria para un modelo más complejo de software.
2. No involucran lógica del negocio, por la generalidad de funciones que desarrollan.
3. Se encargan de la administración de recursos del hardware.

6.4.2. Ejemplos:

Sistemas Operativos (S.O), Mainframes

6.5. Componentes de Integración

Estos componentes facilitan la integración de los componentes dentro de los sistemas a través de múltiples plataformas y desarrollos. La

naturaleza de estos componentes obliga a hacer uso de ellos dentro de la mayoría de los sistemas

6.5.1. Características:

1. No envuelven lógica del negocio.
2. Apoya los procesos de flujo de información más allá de los límites de cada componente, permitiendo la interacción entre los elementos que integran el sistema.
3. La implementación de estos componentes no interfiere con el diseño o los servicios de los componentes que hacen parte del sistema, sólo compila las librerías de tipos de datos y la sintaxis del lenguaje de definición de sus entradas y salidas⁴⁹.
4. Dentro de los diferentes servicios que ofrece, no se encuentra incluido el de funcionalidad directa con usuario.

6.5.2. Ejemplos:

Actualizaciones del Sistema Operativo, Framework, Drivers de dispositivos, adaptadores, emuladores, .dll.

6.6. Componentes GUI

Estos componentes se encuentran en mayor cantidad dentro del mercado de software, se ha demostrado que no es rentable para las empresas desarrollar sus propios componentes GUI, en el año de 1999

⁴⁹ HEINEMAN, George. COUNCILL, William. Op. Cit. 267 p.

el mercado de componentes GUI ascendía a \$300 millones de dólares⁵⁰.

6.6.1. Características:

1. Proporcionan las interfaces externas de las aplicaciones y conocen el funcionamiento de la interacción con el usuario.
2. Muchos entornos de desarrollo incluyen componentes GUI, esto facilita el proceso de construcción.
3. Por si mismos, no representan un sistema funcional, ya que necesitan componentes más complejos para ejecutar las peticiones del usuario.

6.6.2. Ejemplos:

Controles visuales, reguladores de interfaces, componentes de sesión, componentes de suscripción y publicación.

La Jerarquía de Componentes de Software para PyMes se compone entonces de seis (6) niveles, como se muestra en la figura 11, aquí se sugiere un modelo de arquitectura para las aplicaciones basadas en componentes que utilicen esta clasificación. Los componentes GUI y de Integración pueden ubicarse a lo largo del DSBC. La especialización del negocio se da por la especialización de los componentes que integren una aplicación. Esto quiere decir que si una aplicación se construye utilizando componentes de negocio, está operará procesos de negocio más complejos en comparación con una aplicación que utilice componentes de servicio en el nivel superior.

⁵⁰ Ibid. 83 p.

La figura 11 integra los seis niveles de Jerarquía de componentes con los conjuntos de dominios de aplicación, entre más especializado es el componente, más reducido es su conjunto dentro los dominios de aplicación, además de acuerdo con las técnicas de análisis de dominio⁵¹, la selección de este conjunto se realiza por medio de la comprensión del dominio del sistema en particular y su relación con todos los dominios; en primer lugar, se tienen todos los dominios de aplicación, luego una primera especialización de componentes define el conjunto de la mayoría de dominios, y así sucesivamente hasta generar el conjunto de dominios de “*mi sistema*” donde se encuentran los elementos expertos de una aplicación en particular:

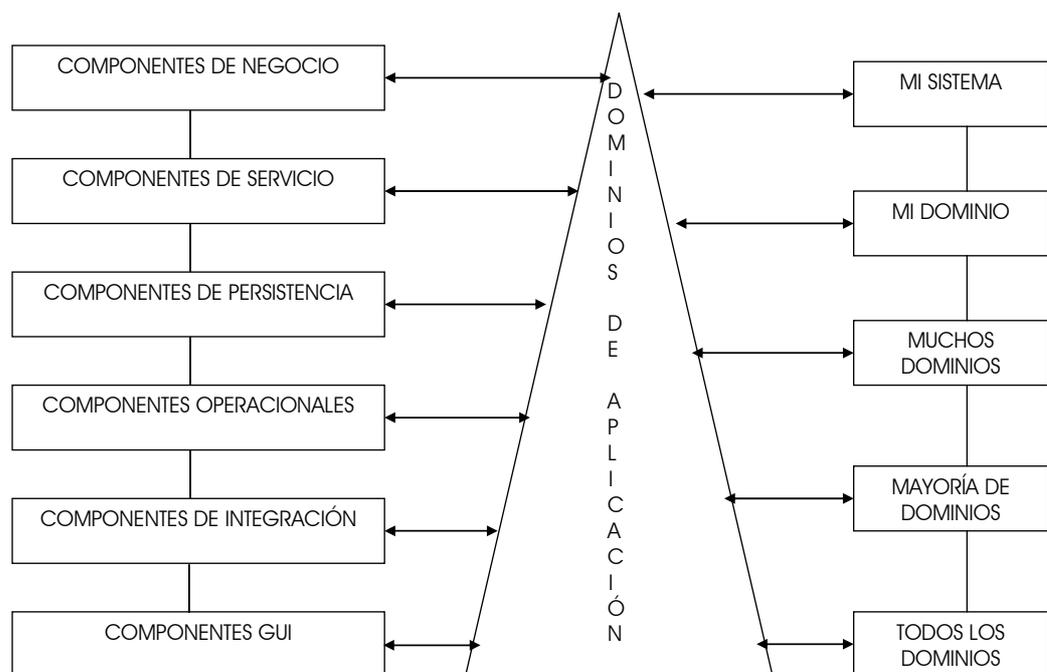


Figura 11: Jerarquía de Componentes de SW y Dominios de Aplicación

El proceso de selección del componente del sistema que se esta construyendo no es sencillo, un desarrollador no debe esperar que hayan sido construidos muchos componentes exactos para el sistema en elaboración, es más sencillo encontrar y escoger componentes que son,

⁵¹ HEINEMAN, George. COUNCILL, William. Op. Cit. 196 p.

al menos, de un dominio específico⁵². La jerarquía de componentes definida brinda un conjunto de dominios básicos donde es posible, de acuerdo a las características de cada componente y su tipo, hacer el proceso de elección más sencillo.

⁵² HEINEMAN, George. COUNCILL, William. Op. Cit. 196 p.

7. SELECCIÓN DEL NIVEL PRINCIPAL

Uno de los objetivos planteados en un principio dentro de este proyecto de investigación es seleccionar el nivel dentro de la jerarquía de componentes, el cuál represente una ventaja competitiva para las PyMes. Para la selección del nivel se tuvieron en cuenta, no sólo las necesidades tecnológicas existentes dentro de este sector sino también la capacidad de comprensión del mismo hacia la tecnología. “Inventario Tecnológico”, “Evaluación de la relación actual entre las TIC’s y PyMes” y “El compromiso del Gobierno Nacional con el desarrollo tecnológico de la PyMe”, este último tema ha sido mencionado a lo largo del documento, pero en esta sección en particular se mencionan elementos de la agenda gubernamental que afectan directamente el proceso de selección de este nivel.

7.1. Criterios de Selección

7.1.1. El compromiso del Gobierno Nacional con el desarrollo tecnológico de la PyMe

El sector de las PyMEs ha estado en la política gubernamental desde hace ya mucho tiempo, pero el proceso para que este sector sea competitivo y significativo dentro la balanza comercial colombiana ha sido lento, por este motivo, es uno de los temas prioritarios en cuanto al manejo de la economía del país.

Los planes gubernamentales de mayor alcance en el área de las TIC’s se iniciaron con el proyecto de Agenda de Conectividad, con la cual se hicieron mediciones sobre las tecnologías utilizadas por todos los sectores en Colombia. Se determinó que a Diciembre de 2003, el sector de micro establecimientos tenía los porcentajes de participación más bajos en cuanto al uso de software especializado,

sin embargo se tenía acceso a toda clase de software. Si bien los resultados de estas mediciones no fueron los esperados, fueron la base de planes que actualmente están en funcionamiento para acercar a todos los sectores a la tecnología.

El Ministerio de Industria, Comercio y Turismo, ha formulado planes y estrategias en beneficio de las PyMes desde el punto de vista tecnológico. Según el informe al congreso de 2003-2004⁵³ de este ministerio, se han realizado acciones para el programa de implantación del Fondo Colombiano de Modernización y Desarrollo Tecnológico de las Micro, Pequeñas y Medianas Empresas – FOMIPYME-, se ha realizado la cofinanciación de programas y proyectos de mejoramiento de la productividad, innovación, desarrollo tecnológico, fomento y promoción de las Mipymes exportadoras directas e indirectas o potencialmente exportadoras. Así mismo, el SENA (Servicio Nacional De Aprendizaje) también cofinanciará proyectos de transferencia tecnológica, innovación y desarrollo tecnológico. Una de las metas del gobierno es crear cadenas de productividad, entendiéndose estas como actividades que se complementan entre sí, en ese orden se han creado programas como “Movimiento Colombiano de la Productividad”, el cuál a través del ministerio en cuestión, han puesto al alcance de la PyMe componentes tecnológicos como PROMES⁵⁴, todo esto enmarcado dentro de una estrategia para promover la conectividad, con la cual se agiliza y asegura la calidad de los trámites de comercio exterior. En este programa ya se tienen resultados a nivel nacional, se han capacitado 1.300 empresarios y 351 asesores multiplicadores formados, por otro lado se ha distribuido de forma

⁵³ Ministerio de Comercio, Industria y Turismo. INFORME AL CONGRESO DE LA REPUBLICA 2004. “SECTOR COMERCIO, INDUSTRIA Y TURISMO” [On Line] <http://www.mincomercio.gov.co/>

⁵⁴ Un software similar al SCM(Supply Chain Management) para su distribución entre PyMes

gratuita alrededor de 4.000 cd's del software de Medición de Productividad-Promes

7.1.2. Inventario Tecnológico

La Federación Colombiana de Software (Fedesoft) ha sido la encargada de realizar estudios que permitan cuantificar los recursos (humanos y tecnológicos) en el campo de las TIC's. Ha estimado que existen 4.000 empresas en el área del desarrollo de software, servicios de consultoría tecnológica y canales de distribución, 75.000 empleos calificados en tecnologías de la información, 850 empresas desarrolladoras de software y 5.000 desarrolladores potenciales para trabajar en la industria. A nivel nacional, estas cifras representan una participación del 1% por parte del sector Tecnológico sobre el total de las PyMes en Colombia⁵⁵.

7.1.3. Evaluación de la relación actual entre las TIC's y PyMes

Un estudio realizado por la Universidad de los Andes⁵⁶, entregado al Ministerio de Comercio, Industria y Turismo, obtuvo algunos resultados relevantes para este proyecto, aquí se mencionan los más importantes:

- ❖ El portafolio de TIC's utilizados por las PyMes están conformados por aplicaciones de sistemas de información para las áreas administrativas, que inciden en la eficiencia, la reducción de

⁵⁵ FEDESOFTE. II Cumbre sectorial de Entidades Relaciones con Tecnologías de la información [On Line] www.fedesoft.org

⁵⁶ Centro GUIA. "Proyecto de Estudio de Casos Colombianos y un país de similares características en dos sectores productivos que permitan identificar casos de éxito y/o fracaso en la implementación de tecnologías de información y comunicación en las PyMes". Facultad de Administración, Facultad de Ingeniería. Universidad de los Andes. 2003

costos y la agilización de procesos, pero que no se enfocan hacia las ventajas competitivas que pueden ser generadas a través de estas aplicaciones.

- ❖ Un gran porcentaje del software que utilizan las organizaciones es adquirido por medio de la figura del Outsourcing, este ha sido uno de los métodos más efectivos, debido al buen nivel de las empresas que ofrecen este servicio y la buena relación establecida entre estas y las PyMes.
- ❖ La tendencia a la globalización, obliga a la PyMe a prepararse tanto para competir con empresas más preparadas en cuanto a tecnología y manejo de información, como clientes que esperan un servicio tecnológico sofisticado.
- ❖ El enfoque recomendado de las TIC's por parte de las PyMes es el *Estratégico*, ya que este permite posicionar las ventajas competitivas de las organizaciones, y en ese orden beneficiar a sus clientes. Algunas de las aplicaciones recomendadas para alcanzar estos objetivos son los sistemas internos como ERP (Planeación de Recursos Empresariales), SCM (Supply Chain Management), CRM (Administración de la Relación Clientes). Otra de las razones por las cuáles es útil hacer uso de este tipo de aplicaciones es la orientación nacional hacia las cadenas de producción⁵⁷.

⁵⁷ Ver sección: 7.1.1 El gobierno y su relación con las tecnologías de las Pymes

7.2. Selección del Nivel Principal de la Jerarquía de Componentes

Con base en todo lo anterior, puede concluirse que el nivel de la jerarquía de componentes de software que debe ser escogido por ser generador de valor agregado es: *Componentes de Negocio*, ya que en primer lugar el gobierno pretende formar PyMes capaces de competir en un mercado globalizado, esto obliga a la PyMe a optimizar sus canales de distribución, focalizar sus esfuerzos para el beneficio del cliente, mejorar la utilización de recursos tanto humanos como tecnológicos, entre otros. La selección de este nivel implica que las PyMes deben empezar a hacer un uso más efectivo de las TIC's estratégicas, no sólo por la importancia de la PyMe en el entorno económico, sino también por la importancia de las TIC's en el desempeño empresarial y la relación directa con el potencial competitivo.

Ahora bien, la relación entre las PyMes y los proveedores de software es una buena relación, de hecho el rendimiento de estos últimos han evitado que la situación tecnológica muestre un peor panorama, pero han traído consigo una estandarización del portafolio de TIC's dentro de las organizaciones. Dar a la PyMe la posibilidad de desarrollar software a su medida, haciendo uso de componentes de negocio, es permitir su diferenciación entre las PyMes de un mismo sector productivo, y esto admite crear competencias nacionales que las fortalezcan y preparen para su internacionalización.

8. GRANULARIDAD DE COMPONENTES DE SOFTWARE

8.1. Consideraciones de la Granularidad en el DSBC

Uno de los procesos más complicados en el área del desarrollo de software basado en componentes es el de selección de componentes que integran el sistema, este es un tema que concierne tanto a desarrolladores como a usuarios finales. En la mayoría de los casos, el conjunto de componentes a elegir puede ser demasiado grande, complicando el proceso de selección y adquisición; la posibilidad de conocer en forma detallada lo que ofrece y no ofrece un componente reduciría en forma notable dicho conjunto y sería aún más sencillo identificar el componente puntual.

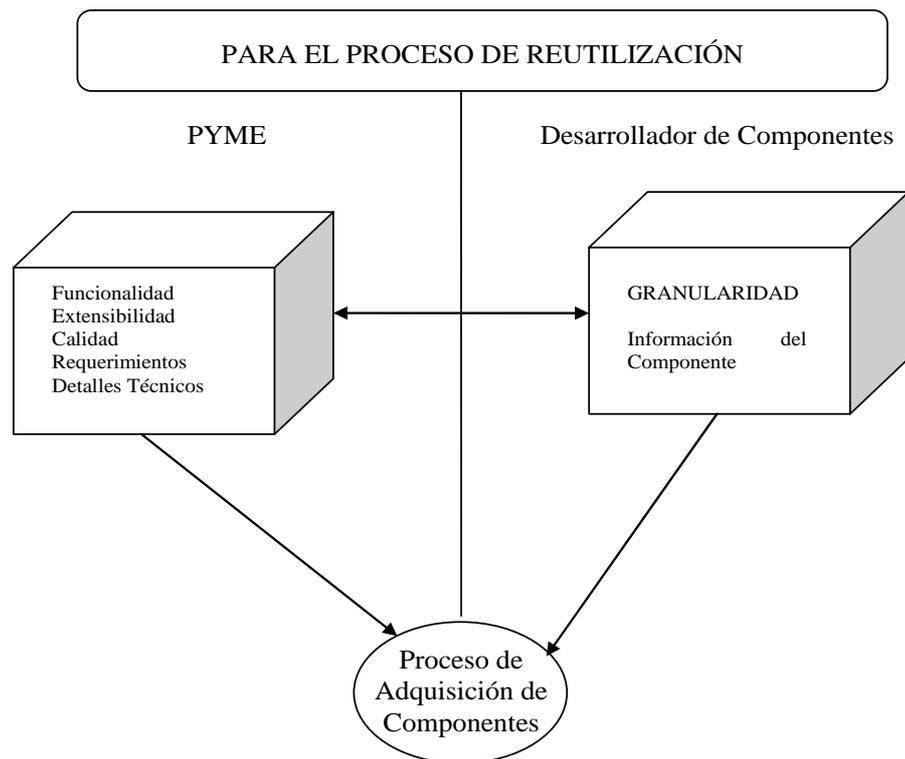


Figura 13: Elementos del proceso de reutilización⁵⁸ 15

⁵⁸ Adaptada de: KALLIO, Päivi. NIEMELÄ, Eila. "Documented Quality of COTS and OCM Components". Finland

Como se muestra en la figura 13, la información que requieren quienes se encuentran interesados en la adquisición de componentes debe ser definida, bien sea por el desarrollador o por un grupo encargado de esta tarea en específico. El conjunto de información, descrito en la figura como la funcionalidad, extensibilidad, requerimientos y detalles técnicos, que es visible para este usuario (PyMes en este caso en particular) se define como la Granularidad del Componente de Software, donde se encuentra contenida la información a exponer. La pregunta que surge es: ¿Cuál es la información que debe ponerse a disposición de las PyMes respecto a los componentes? y, ¿Cuál debe permanecer oculta? Estos dos temas se definen por medio del “grano” con el que se define la granularidad. Si un componente se encuentra descrito utilizando un *grano grueso* significa que la información que este permite conocer es a través de vistas, grupos de información de cierto nivel, el cuál no es generalmente descrito en detalle, en cambio si el grano con el que se describe es *fino* significa que la información es mucho más detallada y que ofrece información que la anterior no permite descubrir.

8.2. Definición de Granularidad

La granularidad se define como el lenguaje descriptivo que un componente expone, el cual permite introducir elementos tales como funcionalidad, restricciones, relaciones e interdependencias de los componentes dentro de los ambientes donde son requeridos.

La gran mayoría de componentes de software existentes en el mercado cuentan con grano demasiado grueso, lo cual dificulta el proceso de comprensión y retrasa el de adquisición ya que no es lo suficientemente claro si es o no el componente que se ajusta a las necesidades. Es así, como se busca definir un nivel de granularidad adecuado para las PyMes, que logrará exponer al usuario las principales características que

pueden ser de interés para este segmento en particular, y además, permitiera indagar y profundizar sobre estos elementos en la medida en la que el grano se hace más fino. Dentro de los tipos de granularidad de componentes de software investigados se destacan:

1. La granularidad definida a través de un conjunto de niveles de descripción. Las características de cada uno de los niveles son precisados de forma clara y detallada⁵⁹.
2. La granularidad definida por medio de la composición recurrente continua, donde los granos más finos de componentes se pueden combinar para obtener una descripción de grano grueso, y un componente se puede descomponer en otros de grano fino, sin especificar exactamente cuál es el tipo y el alcance de la funcionalidad de estos componentes con respecto al contexto⁶⁰.

Estas dos formas de ver la granularidad son válidas; sin embargo, la segunda necesita de un conocimiento más técnico y especializado, el cual podría llegar a afectar el entendimiento por parte de las PyMes. La primera en cambio, plantea unos niveles de detalle que pueden ser especificados y simplificados, por unidades de descripción las cuales deben ser inherentes a estos niveles.

La granularidad definida en este proyecto se precisa a partir de cuatro (4) niveles de abstracción de componentes, estos niveles son Mercadeo, Interfaz, Funcionalidad, Extensión y Técnica; el tamaño del componente no es uno de los elementos tenidos en cuenta en esta definición ya que *la “Granularidad se relaciona más con la semántica de un componente, la comprensión y la naturaleza de su interfaz, el alcance de sus responsabilidades y el nivel y la complejidad de la información que el*

⁵⁹ **STOJANOVIC**, Zoran, “A Method for Component-Based and Service-Oriented Software Systems Engineering”. 2005. 102-110 p.

⁶⁰ Ibid. 102 – 110 p.

componente maneja⁶¹”.

Estos niveles de descripción no son suficientes para delimitar los principales aspectos de un componente, es por esto que se hace uso de las *Unidades de descripción*, definidas como: “*aquello que permite identificar, describir y limitar las características contenidas dentro de cada nivel de descripción*”. Se definen a continuación los niveles de descripción antes mencionados y sus respectivas unidades⁶²:

#	NIVELES DE DESCRIPCIÓN	GRANULARIDAD	CARACTERÍSTICAS
1	DESCRIPCIÓN DE MERCADEO	UNIDADES DE CONTABILIDAD	1. Nombre
			2. Versión
			3. Tecnología
			4. Dominio de Aplicación
			5. Precio (Relación costo / beneficio)
			6. Alcance
2	DESCRIPCIÓN TÉCNICA	UNIDADES DE MANTENIMIENTO	1. Tipo de Mantenimiento (Preventivo, Correctivo, Predictivo, Adaptativo) 2. Descripción
		UNIDADES DE DESPLIEGUE	1. Características para la puesta en marcha del sistema 2. Nombre del archivo o recurso encargado del proceso de despliegue
		UNIDADES DE INSTALACIÓN	1. Descripción de recursos requeridos por el componente para instalación
3	DESCRIPCIÓN DE INTERFAZ	UNIDADES DE ABSTRACCIÓN	1. Interfaces de Componentes compatibles
			2. Servicios
			3. Componente(s) relacionado(s) y tipo de relación
4	DESCRIPCIÓN DE FUNCIONALIDAD	UNIDADES DE ADMINISTRACIÓN DEL SISTEMA	1. Nombre de la parte a administrar 2. Descripción del tipo de administración
		UNIDADES DE ANÁLISIS	1. Precondiciones - Postcondiciones (estado del sistema)
			2. Descripción de Entradas y Salida (servicios)
			3. Invariantes
			4. Tipos de Datos
			5. Atributos y Métodos (Incluye Propiedades)
6. Asociaciones y Navegabilidad entre Objetos			
5	DESCRIPCIÓN DE EXTENSIÓN	UNIDADES DE EXTENSIÓN	1. Nombre de la Unidad de análisis extensible
			2. Tipos de extensiones

Tabla 3: Elementos de representación de Unidades de Descripción 16

⁶¹ Ibid. Pág 102 - 110

⁶² Los elementos incluidos en cada unidad se describe en la tabla 3

8.2.1. Niveles y Unidades de Descripción

8.2.1.1. Descripción de Mercadeo

Representa la eficiencia de los componentes desde la perspectiva de la organización o su interés comercial. En esta descripción además de su nombre y versión, se incluyen características del dominio del componente, alcance del componente y se denota la tecnología utilizada.

Las **unidades de Contabilidad** permiten detallar este nivel, estas unidades permiten relacionar el costo / beneficio del componente en relación con la organización.

Periódicamente Windows realiza reportes por medio de terceros, apoyados sobre estudios, acerca de las cualidades que ofrecen sus productos en comparación con otros fabricantes de software (<http://www.microsoft.com/latam/hechos>); así mismo, la gran comunidad de desarrolladores de Linux presenta artículos enfocados a promocionar sus ventajas sobre otros fabricantes (www.redhat.com).

Aunque esta clase de promoción se enfoca más a técnicas de mercadotecnia entre competidores, es una aproximación a lo que debería ser una unidad de contabilidad.

8.2.1.2. Descripción Técnica:

Expone las restricciones, limitaciones del componente.

Esta descripción hace uso de las **Unidades de Despliegue**:

1. Describe los procesos que preparan las unidades para operar en un ambiente en particular.
2. Estas unidades pueden ser clases individuales con su descriptor de despliegue, generalmente se representan por medio de colecciones de clases, recursos y descriptores.
3. Se deben tener en cuenta características como: escalabilidad, seguridad, mantenibilidad, automatización del componente⁶³.

Los componentes J2EE hacen uso de archivos de despliegue, donde se describe la interacción entre componentes y contenedores, y entre los mismos componentes a través de un archivo XML llamado "*Deployment Descriptor*". Este archivo provee la información para el despliegue de componentes y la aplicación sin tener que acceder al código fuente⁶⁴.

Para otro tipo de componentes se requiere poder ofrecer al usuario un tipo de información similar a la presentada por los *Deployment Descriptor*.

Las **Unidades de Instalación**:

1. Estas unidades utilizan la configuración del despliegue y hacen posible el funcionamiento del componente en particular dentro de una configuración de hardware en específica.
2. Esta descripción esta contenida, generalmente, dentro de un árbol de archivos.

⁶³ **CHAN**, Allen, "J2EE Application Deployment Considerations", [on line] http://www.onjava.com/pub/a/onjava/2003/06/11/j2ee_deployment.html.

⁶⁴ Sun Educational Services. Developing applications for te J2EE plataform. Julio 2003. Cap 2.

La mayoría de componentes presenta algún tipo de guía, documentación, e incluso programas que ayudan a la instalación. La Herramienta Ant de Apache presenta una documentación clara y bien estructurada, por medio de la cual se puede realizar una completa y satisfactoria instalación de cada componente (<http://ant.apache.org/manual/index.html>).

Unidades de Mantenimiento:

1. Describen las actualizaciones que deben hacerse sobre el componente una vez es entregado al cliente.
2. Estas actualizaciones no deben modificar la funcionalidad requerida por el cliente en un principio.
3. Se describen las actividades que deben ejecutarse para su mantenimiento: correctivo, adaptativo, perfectivo, preventivo⁶⁵.
4. Debe especificar si necesita para su mantenimiento algún tipo de software adicional.

Por lo general los componentes de Bases de Datos, indican, que de haber alta persistencia en una o varias tablas, es recomendable realizar procesos de indexación sobre la información contenida dentro de estas. El proceso es costoso en cuanto a recursos del motor de Base de Datos y la máquina, por eso se debe realizar en algunos periodos de tiempo específicos. Algunos motores de BD incluso tienen herramientas que facilitan este proceso, en el cual se programa frecuencia de este proceso.

⁶⁵ **Somerville**, Ian, "Ingeniería del software". Addison Wesley Iberoamericana, Wilnington (EE.UU.) 1998. 683 p

8.2.1.3. Descripción de Interfaz

La interfaz se define como la abstracción del comportamiento del componente, la cual consiste en un subconjunto de interacciones relacionadas a su vez con un conjunto de restricciones las cuales pueden ocurrir bajo ciertas circunstancias.

Las unidades que permiten precisar este nivel son las **unidades de abstracción**, estas son sus principales características:

1. Abstraen la interacción entre las partes del componente.
2. Controlan la complejidad aislando las partes que forman el componente.
3. Cada unidad de abstracción es el resultado de ocultar la implementación de la funcionalidad y sólo exponer la abstracción de la interfaz.
4. Los elementos elegidos para conformar las unidades de abstracción deben ser de interés tanto para el desarrollador como para el consumidor, los elementos de poca importancia para alguna de estas partes no deben ser descritos.
5. Especifican las declaraciones de módulos, interfaces, operaciones, atributos, firmas y visibilidad.

Para la comprensión de estas unidades, se puede acudir a la definición de API: Un API (Interface de Programación de Aplicaciones) es un conjunto de especificaciones de comunicación entre componentes de software. Representa un método para conseguir abstracción en la programación, generalmente (aunque

no es necesario) entre los niveles o capas inferiores y los superiores del software. Uno de los principales propósitos de un API consiste en proporcionar un conjunto de funciones de uso general⁶⁶.

En cuanto a componentes existentes, uno de los más completos es el de JAVA, el cuál no sólo se aplica al framework, también a los componentes desarrollados sobre este framework pueden tener su propio API por medio de JavaDoc. Otros ejemplos de API son:

- Microsoft Win32 API
- SUN J2EE APIs
- API for SCSI device interfacing
- The Carbon APIs for the Macintosh OS
- Common Object Request Broker Architecture (CORBA)

8.2.1.4. Descripción de la Funcionalidad

Este nivel de descripción permite puntualizar cómo el componente realiza las funciones que expone a través de su interfaz. Esto incluye entradas y salidas, administración y análisis del componente.

Las unidades que especifican este nivel son: las **Unidades de Administración del Sistema**, cuyas características principales son:

1. Representan las partes del componente que requerirán administración explícita, la cuál va más allá del proceso de

⁶⁶ <http://es.wikipedia.org>

mantenimiento. De la constante verificación de estas partes, puede depender en alguna medida el correcto funcionamiento del sistema.

2. Las partes elegidas requerirán monitoreo posterior a la instalación.

Aunque estas unidades podrían confundirse con las unidades de mantenimiento, su diferencia radica en que estas últimas se realizan para prevenir o corregir eventos que puedan suscitar fallos en el sistema, mientras que las de administración son procesos naturales del componente, que deben ser por lo general monitoreados para saber el estado actual del sistema, sin que esto implique un eventual fallo y así manipular el componente y su estado.

Los servidores de aplicación también son componentes que poseen unidades de administración bien definidas, si se tiene en cuenta las diferencias entre JBoss, Sun One Application Server (appserver) y el Internet Information Server (IIS) de Microsoft. JBoss realiza un seguimiento a sus procesos por medio de una consola de comandos del sistema operativo en el que informa su estado actual, sin embargo la administración y el cambio de su configuración se realiza en su gran mayoría por medio de archivos XML. Por otro lado el appserver de Sun no coloca todos sus mensajes en la consola de comandos, también utiliza un archivo log el cual indica el estado del sistema con mayor detalle, además utiliza una interfaz gráfica con la cual se pueden administrar la gran mayoría de los recursos del componente. El IIS de Microsoft, tiene una consola gráfica que muestra el estado del componente y

por medio de los recursos de GUI pueden ser administrados estos recursos.

La descripción de funcionalidad incluye también las **Unidades de análisis**:

1. La definición de estas unidades deberá requerir las relaciones de acoplamiento y cohesión. Se recomienda la utilización del lenguaje UML.
2. Es recomendable la inclusión de elementos comunes al Lenguaje de Restricciones de Objetos (OCL), estos elementos son: Contratos (Invariantes, Pre y Postcondiciones), tipos de datos, atributos y métodos relacionados cada uno con sus respectivas propiedades, asociaciones y navegabilidad entre objetos.
3. Estos elementos particionan cada componente, dicha partición son la base para la selección de los lugares de extensión⁶⁷.
4. Preferiblemente se deben definir unidades pequeñas, ya que estas facilitan la comprensión del componente como la suma de sus partes.

El demo Java Pet Store de Java <http://java.sun.com/developer/releases/petstore/> es una aplicación de prueba, la cual busca demostrar las capacidades de la tecnología Java, este componente tiene una guía a través de los Java Blue Prints, los cuales, describen en detalle por medio de patrones de diseño, código y las mejores practicas de programación para Java, la

⁶⁷ Ver descripción de Extensión

manera como debe ser implementada una aplicación. La guía presentada a través de los Java Blue Prints en relación a este componente, debería ser presentada en otras tecnologías y componentes, a este tipo de descripción se refiere la unidad de análisis. Aunque para muchos fabricantes de software esto sería imposible ya que significaría la pérdida del secreto empresarial, ellos pueden adjuntar las mejores prácticas y diseños que sus productos ofrecen, para sacar el mejor provecho de estos.

8.2.1.5. Descripción de Extensión

Un componente es extendido cuando se le agrega una nueva funcionalidad, sin modificar la existente. La descripción de extensión hace uso de las unidades del mismo nombre: **Unidades de Extensión:**

1. Son los lugares donde un componente puede extender la funcionalidad existente.
2. Los lugares de extensión, están definidos dentro de las Unidades de Análisis o de Abstracción.

Una de las principales cualidades de los componentes es su extensibilidad, y esto hace necesaria su mención dentro de la descripción. Componentes como los servidores de aplicación son los que tienen los puntos más claros en cuanto a extensibilidad, cada servidor de aplicaciones especifica, como y donde deben ser extendidos los componentes, algunos servidores se especializan en ciertos servicios, otros son más fáciles de manipular por medio de una interfaz GUI, lo cierto es que cada uno de ellos presenta

una amplia documentación que permite aprovechar al máximo la extensibilidad de estos componentes.

Ejemplos:

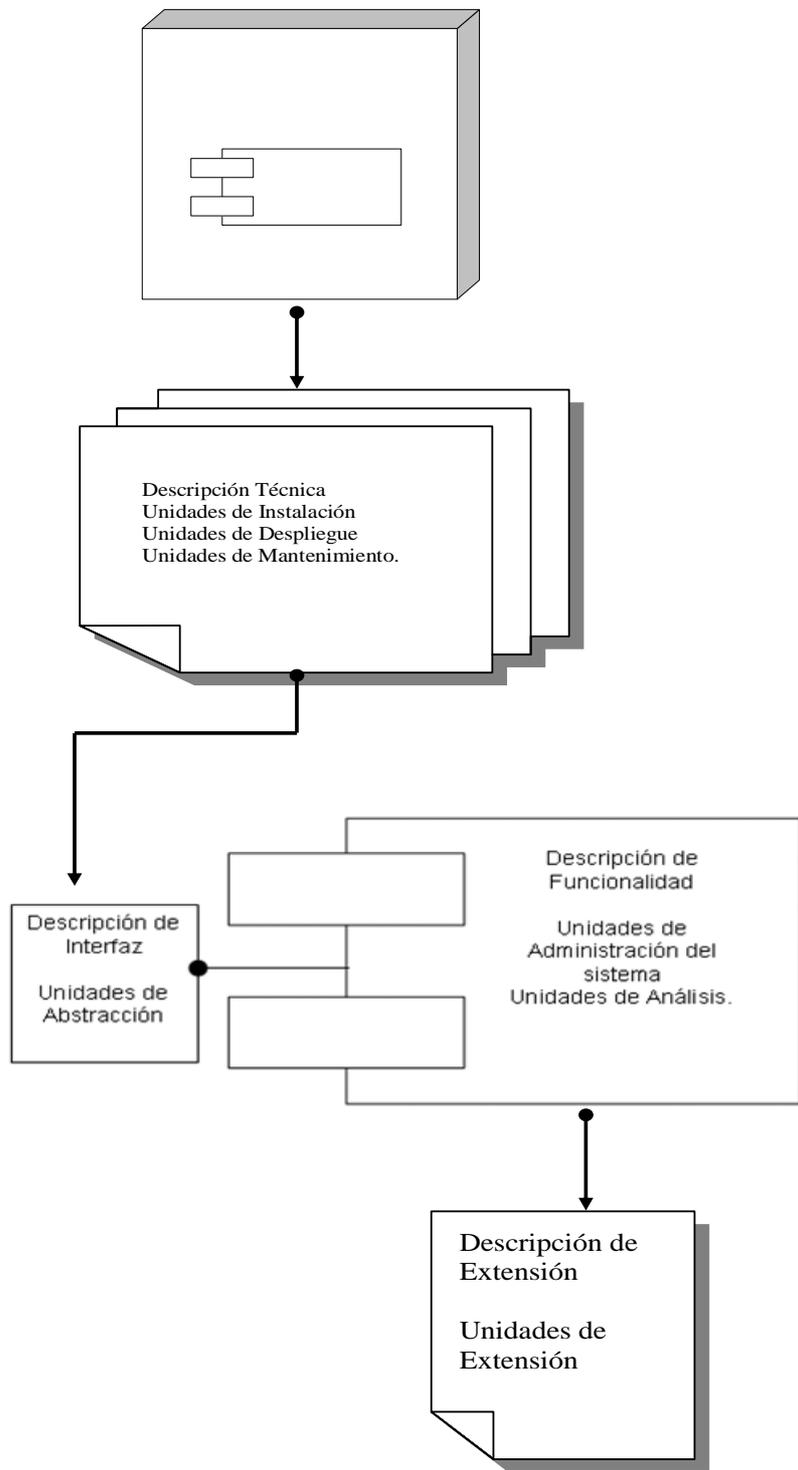
- JBoss
- Tomcat
- Sun appserver
- Internet information Server

La definición de la granularidad propuesta en este proyecto puede variar de componente a componente, bien sea por la complejidad del mismo o por otra serie de factores, es por esta razón que se define en la Tabla N°3, los niveles de descripción, unidades y elementos recomendados en la definición de cada unidad.

Se expuso entonces, cuáles son los elementos (niveles y unidades) que hacen parte de la granularidad de componentes de software. Pero, ¿Cómo un usuario puede realizar la aproximación a estas descripciones? Esta aproximación esta definida en la figura 11.

Para entender el componente, el primer acercamiento se realiza sobre la descripción de mercadeo, esta es la descripción más general de todos los niveles, no incluye un lenguaje técnico, y muestra la relación costo/beneficio, dándole a la PyMe elementos clave para la selección de uno u otro componente. Si desea conocerse más información se accede a la descripción técnica, la cuál indica al usuario en términos de unidades de instalación, despliegue y mantenimiento como debe ser instalado y mantenido el componente. Luego el usuario podrá conocer la descripción de la interface, en este nivel se pueden examinar los servicios que presta el componente a través de las unidades de abstracción, estas son manejadas de acuerdo a la complejidad y tipo de

componente, en este nivel si se hace uso del lenguaje técnico característico de los componentes, pero gracias a la abstracción, este lenguaje continúa siendo comprensible para cualquier lector. Luego de conocer qué hace el componente, el usuario podrá comprender cómo lo hace por medio del nivel de descripción de funcionalidad, donde se especifica en detalle cómo va a prestar el componente los servicios que la interfaz expone entradas, salidas, las unidades que requerirán administración especial y los elementos que permiten la construcción del componente. Finalmente, se encuentra el nivel de descripción de extensión donde se definen los lugares en los que el componente podrá extender dicha funcionalidad.



Descripc

Figura 13: Descripción de la Granularidad de un Componente de Software 17

8. RESULTADOS OBTENIDOS

1. La Jerarquía de Componentes de Software para PyMes se define a través de seis (6) niveles, cada uno de estos niveles hace parte de un dominio de aplicación en específico. Entre más especializado es el componente, más pequeño es este dominio.
2. La jerarquía de componentes está construida sobre una arquitectura genérica de sistemas basados en componentes, esta puede ser comparada y sobre lapada con la arquitectura de software de la aplicación en construcción, facilitando los procesos de selección de componentes. Todo esto, para suministrar a las PyMes herramientas que faciliten la creación de software basado en componentes.
3. El nivel de *componentes de negocio* fue seleccionado como aquel que representa una mayor ventaja competitiva para las PyMes bogotanas, esto se debe a que las características de los componentes pertenecientes a este nivel son aquellos que permiten la construcción de software especializado que puede ser utilizado como un herramienta estratégica, más que operacional.
4. La Granularidad definida consta de cinco (5) niveles de descripción, a través de los cuales puede ser accedido el componente desde sus generalidades comerciales hasta sus propiedades de extensión a través de diferentes niveles de especificación, entre más detalles desean conocerse del componente, más conocimiento técnico es requerido. De cualquier forma, buscó definirse un nivel de descripción comprensible para personas que cuenten con poco conocimiento del DSBC e incluso poco conocimiento en tecnología.

5. Todo el proceso de investigación, permitió ampliar el conocimiento en el área de DSBC, y entender la importancia que esta metodología tiene en el área del desarrollo de software.

6. En cuanto al proceso realizado para reconocer el estado actual del desarrollo de software de las PyMes en Bogotá, se descubrió que algunas empresas tenían experiencia en el DSBC y que el interés por adquirir nuevas tecnologías de información es una acción importante en la agenda empresarial. Por otra parte, es importante trabajar sobre el nivel de conformidad por parte de las PyMes hacia el software utilizado en las organizaciones, ya que el nivel de conformidad no es el ideal.

9. CONCLUSIONES

Debido al alto impacto que el concepto de “Componente de Software” ha tenido dentro del área del desarrollo de software, y áreas relacionadas, muchos conceptos y definiciones empezaron a aparecer, es por esto que se le dio importancia a la unificación de conceptos claves como: Definición, características y diseño de Componentes de Software.

Algunas de las empresas encuestadas afirmaron, de manera informal, haber trabajado con componentes de software incluso durante varios años, pero el proceso de selección de componentes no contaba con ninguna metodología, por el contrario, el proceso se ejecutaba de manera intuitiva por el grupo de desarrolladores, y la principal característica de este grupo de personas era contar con varios años de experiencia en el área de desarrollo, esto puede llevar a concluir que: un desarrollador que se inicie en el área del desarrollo basado en componentes quizás no corra con la misma suerte de poder trabajar con componentes de forma intuitiva. Por esta razón es importante crear una metodología que apoye el proceso de reconocimiento, selección e integración de componentes de software. La estructura definida en la Jerarquía, facilita la ubicación de cada componente de acuerdo a sus características y funcionalidad, estas características, unidas con los beneficios de la definición de la granularidad, apoyan claramente el proceso de selección de componentes.

Existen repositorios de Componente de Software en línea, los cuales proveen una amplia gama de componentes a cualquier tipo de organización, grupo o persona interesada; sin embargo, estos repositorios no cuentan con elementos que faciliten el proceso de comprensión de los componentes y acercamiento a estos, haciendo difícil la selección de aquellos que se ajustan a las necesidades y reduce el potencial de uso de componentes en

los desarrollos de software, la Jerarquía y Granularidad definidas en este proyecto buscan reducir estos inconvenientes.

Por medio de la estructura definida dentro de la Jerarquía, se facilita la ubicación de cada componente de acuerdo a sus características y funcionalidad. Así mismo, la manera en que fue definida la Jerarquía de componentes, como su acercamiento a través de la granularidad, facilita la construcción de un repositorio de componentes para PyMes. Por otro lado, la Jerarquía y su relación con los niveles operacionales de las PyMes, apoyan de manera más transparente el proceso de selección de componentes y creación de software.

La complejidad del lenguaje técnico que impera en las tecnologías, dificulta la comprensión por parte de los usuarios; las PyMes constituyen precisamente uno de los usuarios más importantes de esas tecnologías. Crear mecanismos que permitan simplificar este lenguaje da a las PyMes la posibilidad de concentrarse en los productos y servicios propios de la empresa, erradicando las complicaciones técnicas que puede desalentarlas en el proceso de modernización.

Aunque hace falta mejorar muchos aspectos en comparación con otros países en el área de TIC's, las PyMes Colombianas han iniciado un proceso de mejora en su infraestructura tecnológica. Seleccionar como nivel principal el de *componente de negocio* brinda a las PyMes una ventaja estratégica que puede ser de gran ayuda en una competencia globalizada, por otro lado, la creación de la Jerarquía fue diseñada evaluando en primer lugar, la estructura administrativa de la pyme, y los componentes pertenecientes a ese nivel apoyan la creación de tecnologías de información dentro del área administrativa estratégica, donde se toman las decisiones de mayor impacto a nivel organizacional.

La estructuración de la Jerarquía de componentes, así como su acercamiento a través de la granularidad, facilita la construcción de un repositorio de componentes para PyMes que apoye sus propias necesidades tecnológicas.

Las PyMes requieren estar en capacidad de satisfacer sus propias necesidades tecnológicas; la reducción de los gastos relacionados con contratos de software a través de outsourcing u otras figuras, puede representar diferentes tipos de inversión que apoyen el crecimiento económico de la PyMe.

Por otro lado, el momento que atraviesan las PyMes es un buen momento para invertir económica e intelectualmente, ya que existen varias iniciativas tanto gubernamentales como privadas, que apoyan el proceso de mejora de la infraestructura tecnológica del país y se cuenta con una mano de obra calificada y en crecimiento.

10. RECOMENDACIONES Y TRABAJO FUTURO

Para evaluar el trabajo futuro es importante analizar algunos de los elementos que representan factores decisivos al momento de determinar cuáles serán las próximas tareas a ejecutar y las recomendaciones que este trabajo puede aportar al programa de investigación de componentes de software (PICS).

Para definir la ontología de un componente de software, de acuerdo a los planteamientos del programa de investigación, es importante definir: la Jerarquía, la granularidad y la representación del componente, esto quiere decir que para especificar la ontología de componentes, en este punto del programa, hace falta definir una representación de componentes de software congruente con la Jerarquía y Granularidad precisados por este proyecto. Esta definición será el primer resultado que produzca la Jerarquía y Granularidad.

El proyecto PICS tiene como objetivo ofrecer a las PyMes herramientas tecnológicas viables por medio de las cuales puedan convertirse en empresas más modernas y competitivas, se busca alcanzar estos objetivos por medio de la definición de una metodología de desarrollo de sistemas de información basados en componentes, metodología para desarrollar componentes y su definición ontológica, pero en todo este proceso es importante identificar métodos y agentes implicados que permitan involucrar a las PyMes durante de todo el proceso. En ese orden, se hace evidente resaltar uno de los propósitos finales, tanto del programa como del proyecto de investigación, el cual es permitir que los resultados obtenidos sean realmente utilizados y apropiados por las PyMes; para que esto suceda es necesario prestar atención a los obstáculos descubiertos durante el desarrollo de este proyecto, los cuales pueden entorpecer el proceso de difusión.

En primer lugar el mecanismo de difusión seleccionado debe permitir un alto porcentaje de participación por parte de las PyMes, y de esta forma facilitar el proceso de obtención de resultados reales que beneficien su avance tecnológico, es clave que las PyMes entiendan que la modernización y la tecnología no es exclusiva para las grandes empresas, por el contrario, son herramientas que les ayudan a competir y les permite aminorar la brecha tecnológica con los países más desarrollados, para el cumplimiento de dicha tarea, sería interesante realizar un proceso de sensibilización, el cual abra espacios para la investigación y la innovación tecnológica en la PyMe. En este proceso pueden intervenir tanto universidades como entidades del gobierno.

Desde este punto de vista es interesante analizar el rol del gobierno y del sector privado en el proceso de la modernización de las PyMes. A pesar de los esfuerzos realizados por estos sectores, orientados al avance y progreso tecnológico de las PyMes Colombianas, no han logrado involucrar de manera definitiva a todas las PyMes. La desinformación, procesos demasiado burocráticos y complejos e incluso el desinterés por parte de las PyMes son algunos de los obstáculos. El gobierno debe tener continuidad en este tipo de políticas de manera que la modernización y el apoyo a las PyMes en materia tecnológica, no sea un esfuerzo de cada período de gobierno, sino que sea una verdadera política nacional.

Por otro lado, la postura que deben adoptar las Universidades debe ser proactiva, deben involucrarse con el proceso de modernización y lo que esto implica, materializar las investigaciones realizadas tanto por profesores como por estudiantes. La Jerarquía y la Granularidad obtendrán resultados tangibles y verificables; si las universidades comparten el trabajo que ya han logrado con estudiantes y profesores, a través de sus avances en el desarrollo de software y particularmente en

el desarrollo de software basado en componentes. Es claro que estos trabajos no salen de la academia; en el trabajo de campo realizado no se encontró como referencia a ninguna universidad Colombiana. ¿Por qué?, ¿que pueden perder las universidades si mantienen los derechos reservados sobre los trabajos de sus egresados de pregrado y postgrado, si estos no son consultados y puestos a prueba en entidades de negocio? (salvo algunas excepciones). Las PyMes son buenos laboratorios de prueba para estos trabajos, ya que sus necesidades en materia tecnológica son fácilmente identificables. Esta Facilidad, se debe ver reflejada al momento de identificar que componentes necesita la PyMe, utilizando la Jerarquía y que tanto son comprendidos por medio de la Granularidad.

Sin embargo, el compromiso de la Universidad con la PyMe, no debe interpretarse como una relación unidireccional, no se trata de tener un vínculo paternalista sobre las tecnologías que se le aportan a las PyMes, se debe tener claro que estas deben ser proactivas en la búsqueda de la innovación de tecnología, ellas deben estar conscientes que la superación de los obstáculos y limitaciones requieren un mayor esfuerzo, para esto es necesario que se capaciten y que empleen personal que sea capaz de integrar tecnología, en este caso componentes. Aunque es difícil la integración de este tipo de personal a las PyMes, los programas del gobierno a través del SENA pueden ayudar, capacitando a personal dentro de la empresa que asuman estos roles.

Es claro a partir de todo lo anterior, el sector Industrial y sector educativo del país necesita trabajar en proyectos de este tipo como una sola entidad, las PyMes necesitan políticas y apoyo del gobierno para poder competir internacionalmente, estas a su vez deben buscar capacitación y buscar tecnologías que les permita superar las barreras tecnológicas, las

universidades deben aportar el conocimiento logrado por sus estudiantes en sus proyectos de investigación, esto también para medir la calidad de dichos estudiantes e incluso para que estos puedan ganar prestigio en el sector industrial.

El proyecto PICS, deberá abrir canales de comunicación con las PyMes para promocionar sus resultados y ponerlos a prueba, también primordial que se realice un inventario de componentes de software que se hallan y que se estén realizando en la universidad, e incluso contactar a otras universidades para que estas puedan también aportar componentes para construir un buen repositorio, el cual no sea el logro particular de la ingeniería de sistemas de la universidad Javeriana, sino que sea el logro de la ingeniería de sistemas en Colombia. Se deben contactar las entidades gubernamentales interesadas en patrocinar este proyecto, generando la promoción, apoyo y soporte, en el sector de las PyMes y dentro del gobierno mismo.

Así será como la Jerarquía y Granularidad serán vistas como útiles, no solo por las PyMes, también por el gobierno y otras universidades; los cuales a partir del éxito que tenga la Jerarquía y Granularidad en cuanto a la comprensión que tengan las PyMes sobre los componentes, podrán evaluar su visión acerca de cómo presentar componentes, no solo a las PyMes sino también a las grandes empresas y al interior de sus organizaciones.

11. BIBLIOGRAFÍA

ABOUHAMAD, Jannette. Apuntes de investigación en Ciencias Sociales. Caracas : Universidad Central de Venezuela.

AKSIT, Mehmet. **MEZINI**, Mira. **UNLAND**, Rainer. "Objects, Components, Architectures, Services and Applications for a Networked World". Ed Springer. 2002. pág 63

BARNETT Liz, Market Overview 2003 : Component Management Tools. Mar. 2003. <http://www.forrester.com>

BERTRAN Meyer. "Significado de componentes". En **SCOUT** Amber y **CONSTANTINE** Larry ."The Unified Process Construction Phase" 102-112 p.

BROADBENT, Mariane. **WEILL**, Peter. "Leveraging the new Infraestructure" .Harvard Business Scholl Press. United Stated of America. 1998.

BROWN A. W, **WALLNAU** K. C. "The current state of CBSE", IEEE Software 15 (1998) 37-46. [on-line] <http://www.idt.mdh.se/kurser/phd/CBSE/literature/The-current-state-of-CBSE.pdf>

CAI Xia, **LYU** Michael. "Component Based Software Engineering: Technologies, Development Frameworks, and Quality Assurance Schemes" p. 373.

CALDWELL French, **HAYWARD** Simon, **MARK** Gilbert y **PHIFER** Gene. The Future of the Smart Enterprise Suite. **SHULLTE** Roy, **YEFIM** Natis. Most Composite Applications Will Need an Integration Layer, [On Line] <http://www.gartner.com>

CHAN, Allen, "J2EE Application Deployment Considerations", [On Line]
http://www.onjava.com/pub/a/onjava/2003/06/11/j2ee_deployment.html.

COLLET, Bruno. "To EAR or not to EAR: Choosing the Granularity of J2EE Deployment Units" [On Line]
<http://www.practicalsoftwarearchitect.com/articles/toearornottoear/toearornottoear.html> May 4, 2004.

DIMITRIADIS, Yannis y otros. "Middleware para CSCS: Marco de Componentes Software y Apoyo de Tecnología Grid". ETSI de Telecomunicación, Universidad de Valladolid. Noviembre. 2003. pp. 4-10.

DURHAM, James. History - Making Components: Tracing the roots of components from OOP through WS [On Line]. <http://www-106.ibm.com/developerworks/java/library/co-tmlne/>

D'SOUZA, Desmond Francis, **WILLS**, Alan Cameron. "Object components and Frameworks. The Catalysis Approach". 1998

FEDESOFTE. II Cumbre sectorial de Entidades Relaciones con Tecnologías de la información [On Line] www.fedesoft.org

FELLNER, Klement J. **TUROWSKI**, Klaus. "Classification Framework for Business Components". Faculty of Computer Science Institute for Technical and Business Information Systems Business Information Systems Magdeburg, Germany. 2000. 4-5 p.

FETTKE, Oetes. **LOOS**, Pettes. "Objects, Componentes , Achitectures, Services and Aplications for A Networked World. International Conference NetOObjectDays, NODe 2002 Erfurt, Germany, October 2002 -Revised Papers "Specification of a Business Components".

GAMMA Erich, **HELM** Richard, **JOHNSON** Ralph y **VLISSIDES** John. Design Patterns, 1995. 395 p.

GONZÁLEZ, Rafael. Programa de Investigación: Componentes de Software para PyMes. Pontificia Universidad Javeriana. Facultad de Ingeniería. 2004 p 3

GONZÁLEZ, Rafael. "Ontología de Software para PyMes". 2004. [On Line]. http://ainsuca.javeriana.edu.co/~rgonzal/pics/ontologia.html#Planteamiento_del_problema

HASSAN, Goma Georfe. "Designing Concurrent, Distributed, and Real-Time Applications with UML". Addison- Wesley. 2000. 55 p.

HEINEMAN, George. **COUNCILL**, William. "Component Based Software Engineering". Addison- Wesley. 2001. 818 p.

<http://www.componentSource.com>

<http://www.webopedia.com/>

<http://looselycoupled.com/glossary/granularity.html>

KALLIO, Päivi. **NIEMELÄ**, Eila. "Documented Quality of COTS and OCM Components". Finland

KENDALL, Kenneth. **KENDALL**, Julie. Análisis y Diseño de Sistemas de Información. 3ª Edición. México. Prentice Hall Hispanoamérica S.A.

KRUTCHEN, Philippe. **KROLL**, Per. "Rational Unified Process". 2ª Edición. Addison – Wesley Edition. 2002. 400 p.

KUNDA, Douglas, **BROOKS** Laurence. "Component-Based Software Engineering for Developing Countries: Promises and Possibilities". Department of Computer Science University of York, Heslington.

LATCHEM, Steve. "Components Infrastructures: Placing Software Components in Context" p. 265

MANTILLA, Samuel Alberto. "Capital Intelectual y Contabilidad del Conocimiento". 3ª Edición. Bogotá: ECOE Ediciones. 2004. 7 p.

"Medición de las tecnologías de la información y las comunicaciones TIC's" - Resumen ejecutivo (Diciembre 2003),[On Line] <http://www.dane.gov.co>

MINISTERIO DE COMERCIO, INDUSTRIA Y TURISMO. Informer al congreso de la Republica 2004. "Sector comercio, industria y turismo". [On Line] <http://www.mingobierno.gov.co>

Ministerio de Economía (Chile), Programa de Desarrollo e Innovación Tecnológica 2001-2005 [On Line]. [http //www.economia.cl](http://www.economia.cl)

MINZTBERG Henry. La estructuración de las organizaciones. Ariel. Barcelona. 1999.

MONTILVA, Jonás. "Desarrollo de Software Basado en Componentes". Facultad de Ingeniería. Universidad de los Andes Venezuela. [On Line] http://www.istec.org/events/ga2003/results/presentations/TDSBC_V2_Unidad_1.pdf

PARRA, José David. "Hacia una Arquitectura Empresarial basada en Servicios". [On Line] <http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art143.asp>

PRESSMAN, S Roger. "Ingeniería de Software un enfoque practico". , Mc Graw Hill , 2002. 480-487 p.

PORTER, M. "On Competition". Boston, Harvard Business School Press, 1998.155-350.

PRIETO, Manuel. MANJAVA-CAS, Ramón. LAGUNA, Miguel. "APRENDIZAJE BASADO EN LA CO-LABORACIÓN Y EL CONOCIMIENTO.

UN ENFOQUE DE INGENIERIA DE SOFTWARE". [On Line]
<http://www.formatex.org/jdc/include/getdoc.php?id=27&article=8&mode=pdf>

RODRÍGUEZ, Astrid Genoveva. La realidad de la PyME Colombiana: Desafío para el Desarrollo. Bogotá. FUNDES, 2003. 186 p.

SCOTT W, Ambler. CONSTANTINE, Larry. SMITH, Roger. "The Unified Process Elaboration Phase: Best Practices in Implementing the UP". CMP Books. Agosto 2001. 93, 102-111 pp.

SIEGEL, Jon. CORBA, Fundamentals and Programming. Ed. Wiley. 1996. 693 p.

SODHI, Jag. SODHI, Prince. Software Reuse: Domain analysis and design process. McGraw Hill. 1998

SOMERVILLE, Ian, "Ingeniería del software". Addison Wesley Iberoamericana, Wilnington (EE.UU.) 1998. 683 p

STOJANOVIĆ Zoran, A Method for Component-Based and Service-Oriented Software Systems Engineering. 2005. pp. 102-110.

SZYPERSKI, Clemens. "Component Software: Beyond Object Oriented Programming". 2ª Edición. Addison – Wesley Edition. 2002. 589 p.

Sun Educational Services. "Developing applications for the J2EE platform". Julio 2003. Cap. 1 - 2.

REYNOSO, Carlos. "Introducción a la Arquitectura de Software". Universidad de Buenos Aires. [On-Line] [Http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.asp](http://www.microsoft.com/spanish/msdn/arquitectura/roadmap_arq/intro.asp)

TAMAYO TAMAYO, Mario: El Proceso de Investigación Científica, México D.F.

TOOL, David, DataChannel, presentation at Documentation Santa Clara (conference. "Faceted Classification of Information". [On - Line] <http://www.kmconnection.com>. . Marzo 10, 1998.

TUROWSKI Klaus. **FELLNER** Klement J. "Classification Framework for Business Components" Institute for a Technical and Business Information Systems Magdeburg Germany pág 5.

VALDES, Luis Eduardo. "Situación Actual de la Informática en Colombia". CATI (Centro de Apoyo de la Tecnología Informática) <http://www.cati.org.co>.

VILLATE, Ricardo. "El imperativo de la infraestructura de TI en América Latina". Estudio Realizado por IDC: Latin America System and Network Software go to market. [on-line] [http:// www.idc.com](http://www.idc.com)

WHITTEN, Jeffrey y otros. Análisis y Diseño de Sistemas de Información. 3ª Edición. Madrid. Mc Graw Hill, 1999. 907 p.

II Cumbre Sectorial de entidades relacionadas con las Tecnologías de la Información FEDESOFIT

ANEXOS

1. Formato Encuesta

PONTIFICIA UNIVERSIDAD JAVERIANA
FACULTAD DE INGENIERIA
INGENIERIA DE SISTEMAS



CUESTIONARIO PROYECTO PICS

Fecha _____

Instrucciones: Marque con una X la respuesta escogida.

Datos de la empresa.

Nombre: _____ Dirección:

Datos encuestado.

Nombre: _____ Cargo:

Teléfono _____ E-mail:

Preguntas:

1. Exprese su conformidad o inconformidad con el software que actualmente utilizan para satisfacer las necesidades específicas de su negocio.

1. Completamente conforme.
2. Conforme.
3. Neutral.
4. Inconforme.
5. Completamente inconforme.

2. ¿Han considerado dentro de sus planes a corto plazo adquirir nuevas herramientas tecnológicas?

1. Si.
2. No.

3. ¿Si existiera una biblioteca de componentes de software que le permitiera agilizar sus desarrollos, la utilizaría?

1. Si.
2. No.

(Si la respuesta a esta pregunta es No, pasar a la pregunta 7).

4. ¿Cuáles de los siguientes aspectos tendría en cuenta para descubrir (encontrar) los componentes que usted necesita en la biblioteca?

1. Funcionalidad.
2. Tipo de Arquitectura.
3. Lenguaje de programación.
4. Autor.

5. ¿Cómo preferiría adquirir los componentes?

1. Link al proveedor principal.
2. Descargar directamente del repositorio.

6. ¿Cómo le gustaría acceder a la biblioteca de componentes?

1. Impreso.
2. Base de datos empresarial.
3. Internet.
4. Otro _____

7. ¿Qué tipo de sistema operativo utiliza?

1. Linux.
2. SCO Unix.
3. Windows.
4. Solaris.
5. OS2.
6. Otro _____

8. ¿Cuál(es) es el tipo de arquitectura de software que utilizan las aplicaciones de su empresa? ¿Número de aplicaciones que utilizan dicha arquitectura?

1. Cliente – Servidor _____
2. Multicapa _____
3. Embebidos _____
4. Centralizados _____
5. Par a Par _____
6. Distribuidos _____
7. Orientado a Servicio _____

8. Otras (Cuáles)

9. ¿Qué tipo de lenguajes utiliza dentro de la empresa?

1. C.
2. C++.
3. Java.
4. Basic.
5. Fox pro.
6. Script.
7. Otro. _____

10. Aplicaciones de software desarrolladas dentro de la organización:

Nombre de la aplicación	# Personas involucradas	Aplicaciones estándar adaptadas

11. Ordene, según la importancia, los criterios de mantenimiento que debe tener un producto de software para su empresa.

CRITERIO	DEFINICIÓN	ORDEN
Extensibilidad	Facilidad con la cuál se puede	

	agregar funcionalidad o nuevas clases al sistema	
Modificabilidad	Facilidad con la cual se puede cambiar la funcionalidad del sistema	
Adaptabilidad	Facilidad para transportar el sistema a diferentes dominios de aplicación	
Portabilidad	Facilidad para transportar el sistema a diferentes plataformas	
Legibilidad	Facilidad con la cual se puede comprender el sistema leyendo el código	
Rastreabilidad de Requerimientos	Facilidad para establecer la correspondencia entre el código y los requerimientos específicos	

Otros(Cuáles):

12. Ordene, según la importancia, los criterios de desempeño que debe tener un producto de software para su empresa.

CRITERIO	DEFINICIÓN	ORDEN
Tiempo de Respuesta	La velocidad con la cual es atendida la petición de un usuario después de haberse enviado.	
Producción	Número de tareas que puede realizar el sistema en un período de tiempo fijo	
Memoria	El tamaño del espacio que se requiere para que se ejecute el sistema	
Robustez	Capacidad de sobrevivir ante datos	

	inválidos del usuario	
Confiabilidad	Diferencia entre el comportamiento especificado y el observado	
Disponibilidad	Porcentaje del tiempo del sistema en que puede usarse para realizar las tareas normales	
Tolerancia a Fallas	Capacidad para operar bajo condiciones erróneas	
Seguridad	Capacidad para resistir ataques maliciosos	
Inocuidad	Capacidad para no poner en riesgo la vida humana, aun en presencia de errores y fallas	

Otros(Cuáles):

13. Ordene, según la importancia, los criterios de costo que debe tener un producto de software para su empresa.

CRITERIO	DEFINICIÓN	ORDEN
Costo de desarrollo	Costo de desarrollo del sistema inicial	
Costo de entrega	Costo de la instalación y del entrenamiento a los usuarios	
Costo de la actualización	Costo de trasladar los datos del sistema anterior	
Costo de mantenimiento	Costo requerido para la corrección de errores y para las mejoras al sistema	

Costo de administración	Dinero requerido para la administración del sistema	
-------------------------	---	--

Otros(Cuáles):

14. Ordene según el nivel de importancia los obstáculos percibidos y encontrados dentro del desarrollo de software interno.

1. De personal.
2. Conocimiento y capacitación.
3. Costo.
4. Tiempo.
5. Otros(Cuáles)_____

15. ¿En el último año qué actividades de capacitación se han llevado a cabo en su empresa?

Tipo de Actividad.	Intensidad Horaria.	# De Personas.

16. El software de su organización se utiliza principalmente en el área:

- a. Comercialización.
- b. Administrativa.
- c. Financiera.
- d. Producción de Bienes y Servicio.
- e. Investigación.

- f. Software Educativo.
- g. Otros.

17. Cuál es el porcentaje del presupuesto que la organización asigna al desarrollo de software

Porcentaje (%) de su inversión inicial.	SELECCIÓN
0	
1-20	
20-40	
40-60	
60-80	
>80	

18. ¿Existe alta rotación de personal dentro de la organización?

- 1. Si.
- 2. No.

19. ¿Qué métodos usan para las capacitaciones?

- 1. Relación Experto-Aprendiz
- 2. Conferencias, videos, películas, audiovisuales y similares
- 3. Instrucción directa sobre el puesto
- 4. Estudio de casos
- 5. Lectura, estudios individuales, instrucción programada

20. ¿Cuánto cuesta en promedio un proceso de capacitación?

21. ¿Conoce las ventajas del e-learning?

- 1. Si.

2. No.

22. ¿Estaría interesado en montar un programa de e-learning en su empresa?

1. Si
2. No

23. ¿Quién hace los procesos de capacitación dentro de su empresa?

1. Contrato a terceros
2. Internamente en la organización

24. ¿Se maneja algún tipo de reportes de resultados? Si ___ No ___

1. Presencia
2. Calificaciones
3. Rendimiento
4. Otra (Cuál) _____

25. ¿En qué tipo de reportes acerca de la capacitación estaría interesado después de una capacitación de personal, clientes o proveedores?

1. Reporte de calificaciones
2. Reporte de asistencia
3. Otros
(Cuáles) _____

2. Listado de Empresas Encuestadas

#	Empresa
1	Arar Representaciones

2	Comercializadora Laverde Ltda.
3	Compucentro Colombia
4	Consultoría Colombiana
5	Datasixx
6	Litigando.com
7	Información y Tecnología
8	Geosys Ltda.
9	Imprel
10	MCI Colombia
11	Sky Colombia
12	Skynet de Colombia
13	Softvalores S.A.
14	Telefónica Data
15	Universoft Ltda.

3. Tabulación de Resultados

Conformidad frente al SW	%
Completamente Inconforme	0,00%
Inconforme	6,67%
Completamente Conforme	13,33%
Neutral	13,33%
Conformidad	66,67%
<i>Total</i>	100,00%

Adquisición de Nuevas Herramientas Tecnológicas	%
NO	33,33%
SI	66,67%
<i>Total</i>	100,00%

Utilización de Biblioteca de Componentes	%
NO	6,67%
SI	93,33%
<i>Total</i>	100,00%

Aspectos para descubrir componentes en la Biblioteca	%
Lenguaje de Programación	0,00%
Autor	0,00%
Tipo de Arquitectura	6,67%
Tipo de Arquitectura y Lenguaje de Programación	6,67%
Funcionalidad, Tipo de Arquitectura y Lenguaje de Programación	6,67%
Funcionalidad y Lenguaje de Programación	13,33%
Funcionalidad	66,67%
<i>Total</i>	100,00%

Preferencia en adquisición de componentes	
Link al Proveedor Principal	33,33%
Descargar directamente del repositorio	46,67%
Pago Directo	6,67%
Indiferente	20,00%
<i>Total</i>	106,67%

Acceso a la Biblioteca de Componentes	
Impreso	0,00%
Base de Datos empresarial	20,00%
Internet	73,33%
Otros	0,00%
<i>Total</i>	93,33%

Sistema Operativo	%
SCO Unix	0,00%
Solaris	0,00%
OS2	0,00%
Linux	6,67%
Windows y Solaris	6,67%
Linux y Solaris	6,67%
Linux, SCO Unix y Windows	6,67%
Linux, Windows, Solaris y OS2	6,67%
SCO Unix y Windows	13,33%
Linux y Windows	13,33%
Windows	40,00%
<i>Total</i>	100,00%

Arquitectura de Software	%
Embebidos	0,00%
Par a Par	0,00%
Distribuidos	0,00%
Orientado a Servicio	0,00%
Centralizados	6,67%
Cliente - Servidor, Embebidos y Centralizados	6,67%
Cliente - Servidor, Multicapa, Embebidos, Centralizados, Par a Par, Distribuidos y Orientado a Servicio	6,67%
Multicapa	13,33%
Cliente - Servidor y Multicapa	13,33%
Cliente - Servidor	53,33%
<i>Total</i>	100,00%

Tipo de Lenguajes	
C	20,00%
C++	6,67%
Java	66,67%
Basic	26,67%
Fox Pro	26,67%
Script	20,00%
PHP	13,33%
C#	6,67%
Cobol	6,67%
SQL	6,67%
Delphi	6,67%
asp	6,67%
<i>Total</i>	213,33%

Criterios de Mantenimiento	%
Adaptabilidad	36,67%
Portabilidad	43,33%
Legibilidad	56,67%
Rastreabilidad	58,89%
Modificabilidad	63,33%
Extensibilidad	65,56%

Criterios de Desempeño	%
Tolerancia a Fallas	43,70%
Inocuidad	45,93%
Robustez	46,67%
Seguridad	46,67%
Memoria	50,37%
Confiabilidad	51,11%
Disponibilidad	51,85%
Producción	54,81%
Tiempo de Respuesta	67,41%

Criterios de Costo	%
Costo de Entrega	53,33%
Costo de Actualización	56,00%
Costo de Mantenimiento	57,33%
Costo de Administración	60,00%
Costo de Desarrollo	61,33%

Obstáculo	%
De Personal	30,00%
Conocimiento y Capacitación	38,33%
Costo	46,67%
Tiempo	51,67%

Areas de uso del SW	%
Administrativa	0,00%
Investigación	0,00%
Software Educativo	0,00%
Comercialización	6,67%
Financiera	6,67%
Administrativa y Financiera	6,67%
Comercialización, Administrativa, Financiera y Pr. Bienes y Servicios	6,67%
Administrativa, Financiera y Pr. Bienes y Servicios	6,67%
Financiera e Investigación	6,67%
Comercialización, Administrativa y SW Educativo	6,67%
Otros	6,67%
Redes y Telecomunicaciones	13,33%
Producción de Bienes y Servicios	33,33%
<i>Total</i>	100,00%

Porcentaje del Presupuesto al Desarrollo de SW		%
Ns/Nr		20,00%
1%- 20%		26,67%
20%-40%		26,67%
0%		6,67%
40%-60%		6,67%
60%-80%		6,67%
>80%		6,67%
<i>Total</i>		100,00%

Rotación de Personal		
SI		26,67%
NO		73,33%
<i>Total</i>		100,00%

Métodos de Capacitación		
a. Relacion Experto Aprendiz		46,67%
b.Conferencias, Videos, Peliculas, Audiovisuales, o similares		20,00%
c.Instrucción Directa Sobre el Puesto		13,33%
d.Estudio de Casos		6,67%
e.Lectura, estudios Individuales, instrucción programada		33,33%
f. Web Training		6,67%
<i>Total</i>		126,67%

Ventajas de e-learning		
SI		66,67%
NO		33,33%
<i>Total</i>		100,00%

Interés en adquirir e-learning		
SI		53,33%
NO		46,67%
<i>Total</i>		100,00%

Quién hace los procesos de capacitación		
Contrato con terceros		40,00%
Internamente en la organización		66,67%
<i>Total</i>		106,67%

Manejo de Reportes de resultados	
Presencia	13,33%
Calificación	26,67%
Rendimiento	33,33%
<i>Total</i>	73,33%

Administración de Recursos	
Reporte de calificaciones	40,00%
Reporte de asistencia	26,67%
Rendimiento	20,00%
Otros	13,33%
<i>Total</i>	100,00%