



**UNIVERSIDAD NACIONAL  
AUTÓNOMA DE MÉXICO**

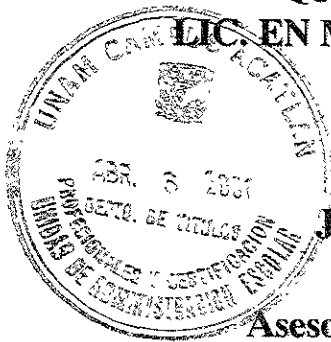
**ESCUELA NACIONAL DE ESTUDIOS  
PROFESIONALES "ACATLÁN"**

**TÉCNICA DE APOYO PARA EL DESARROLLO DE  
LA DOCUMENTACIÓN EN LA PROGRAMACIÓN  
ORIENTADA A OBJETOS**

**Seminario Taller Extracurricular  
Análisis De La Planeación**

**Trabajo De Investigación  
Que Para Obtener El Título De:**

**LIC. EN MATEMÁTICAS APLICADAS Y  
COMPUTACIÓN**



**PRESENTA:  
Juan Carlos Guijosa Solís**

**Asesor: Act. Hugo Reyes Martínez**





Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Quedar seducido para siempre por el mundo de las ideas;

eso hace la educación ...

Edith Hamilton.

---

---

# Agradecimientos

A mis padres:

Por enseñarme lo valioso de la vida.

A Marcela:

Por ser un apoyo incansable.

A Joel:

En su memoria, yo se que le hubiera gustado ver esto.

A mis profesores:

Act. Luz María Lavín

Lic. Juan Torres

Lic. Gerardo Roldán

Mtro. Ignacio Lizárraga

Por ser también mis amigos.

En especial, a mi asesor del presente trabajo de investigación el Act. Hugo Reyes, por confiar siempre en mi y guiarme con paciencia al éxito.

Gracias a todos ustedes ...

---

# Contenido

|   |    |
|---|----|
| Introducción  | IX |
| Capítulo I La Programación Orientada a Objetos                    | 1  |
| 1.1 Lenguajes de Programación                                     | 3  |
| 1.2 Acerca de la Programación Orientada a Objetos                 | 6  |
| 1.2.1 Ventajas  | 9  |
| 1.2.2 Desventajas   | 11 |
| 1.2.3 Abordando la Problemática                                   | 12 |
| Conclusiones del Capítulo I                                       | 19 |
| Fuentes de Consulta   | 21 |
| Capítulo II Técnicas de Documentación                             | 23 |
| 2.1 Acerca de las Técnicas de Documentación                       | 25 |
| 2.2 Técnicas, Modelos y Procesos de Desarrollo                    | 27 |
| 2.3 Técnica de Documentación Propuesta                            | 50 |
| 2.3.1 Organización  | 51 |
| 2.3.2 Estructuras   | 52 |
| 2.3.3 Diagramas   | 52 |
| 2.3.4 Formatos  | 53 |
| Conclusiones del Capítulo II                                      | 55 |
| Fuentes de Consulta   | 57 |
| Capítulo III Explotación de la Técnica de Documentación Propuesta | 59 |
| 3.1 Solución con Programación Orientada a Objetos                 | 61 |
| 3.1.1 Descripción de la Solución                                  | 63 |
| 3.1.2 Documentación Propuesta                                     | 66 |
| 3.1.3 Evaluación de la Solución                                   | 72 |
| 3.2 Escenarios  | 74 |
| Conclusiones del Capítulo III                                     | 81 |
| Fuentes de Consulta   | 83 |

## **Contenido, Continuación ...**

|                                     |      |
|-------------------------------------|------|
| Conclusiones Generales .....        | XIII |
| Fuentes de Consulta Generales ..... | XV   |
| Glosario de Términos .....          | XIX  |

## Introducción

En la actualidad una de las áreas más candentes en la industria y en el ámbito académico para la *informática* es la programación orientada a objetos. La orientación a objetos promete mejoras de amplio alcance en la forma de diseño, desarrollo y mantenimiento del *software* ofreciendo solución a los problemas y preocupaciones que han existido desde el comienzo en el proceso de desarrollo de *aplicaciones de sistemas*: la falta de portabilidad del código y reusabilidad, código que es difícil de modificar, ciclos de desarrollo largos y técnicas de codificación no intuitivas entre otras.

El concepto de programación orientada a objetos no es nuevo, *lenguajes* clásicos como SmallTalk\* se basan en ella. Dado que la programación de este estilo se ocupa en la idea natural de la existencia de un mundo lleno de objetos y que la resolución del problema que se pueda plantear se realiza en términos de objetos, se dice que un lenguaje de programación está basado en objetos si soporta estas entidades como una característica fundamental del mismo.

Un lenguaje orientado a objetos también cuenta con tres características principales: debe ser basado en objetos, como se menciono anteriormente; además en *clases* para representar las entidades y con la capacidad de poder adquirir la funcionalidad de *herencia* para estas clases. Muchos lenguajes de programación pueden cumplir uno o dos de estos puntos; muchos menos cumplen los tres, sin embargo la barrera más difícil de sortear es usualmente la herencia informática.

En general se dice que el elemento fundamental de la programación orientada a objetos, obviamente y como su nombre lo indica, es precisamente el objeto y se puede definir al mismo como un conjunto complejo de datos y programas que poseen estructura y forma parte de una organización.

Es importante ubicar el ámbito orientado a objetos dentro del mundo de la computación, para de este modo entender la esencia fundamental del mismo; tema que se aborda ampliamente en el primer apartado de la presente investigación.

\* La idea de que los datos y sus operaciones son una misma cosa fue de las primeras lecciones del proyecto "Simula", realizado por Nygaard y Dahl en 1981; estos autores afirmaron que en muchas aplicaciones las variables y procedimientos forman una colección y servían como "entidades de programación naturales". Posteriormente y gracias al mismo proyecto surge la opción de utilizar "clases de programación", a través de "Modula-2", para finalmente operar de lleno con el concepto de programación orientada a objetos mediante el "Smalltalk<sup>™</sup>".

SETHI, Ravi. Lenguajes de Programación. Conceptos y Constructores.  
Ed. Addison-Wesley Iberoamericana S.A., E.U.A., p. 11; 1992.

El contar con una *herramienta* de programación adecuada para las exigencias que presenta el mundo contemporáneo dentro del área de desarrollo de aplicaciones, (como la programación orientada a objetos) no implica que la tarea este terminada, es también de suma importancia documentar los procesos de las aplicaciones generadas con la misma, debido a que como cualquier lenguaje de programación y de acuerdo a sus propias características presenta cierta complejidad al buscar representar los problemas de la vida cotidiana en un ambiente orientado a objetos.

Planteamiento del problema: hoy en día la cantidad de información que se utiliza es de grandes proporciones y en muchas ocasiones se puede desbordar; situación que no le es ajena a la orientación a objetos; sin embargo, las distintas técnicas de documentación existentes auxilian a ordenar y describir claramente la información, lo que implica poder contar con datos relevantes oportunamente mediante el uso de un documento adecuado. Por ello al conjuntar una herramienta de programación eficaz para la creación de aplicaciones de sistemas, como la orientación a objetos y acompañada con su respectiva documentación creada con la técnica propuesta, se enriquece y se dota a las empresas con aplicaciones de sistemas hechas a la medida y de fácil mantenimiento, de este modo los creadores de aplicaciones y los mismos *usuarios* contribuyen con un mejor desempeño de sus funciones dentro de cualquier organización. Esta inquietud muestra precisamente el objetivo de la presente investigación, el cual se enuncia de la siguiente manera: Proponer una técnica de apoyo que auxilie en el desarrollo de la documentación que describa el funcionamiento de las aplicaciones generadas por medio de la programación orientada a objetos.

La técnica de documentación propuesta que contribuye al desarrollo de la documentación para las aplicaciones orientadas a objetos, debe retomar los aspectos de negocio que enseña el método de información Mapping®, también las cuestiones de análisis y diseño (modelado) que otorga el análisis estructurado moderno, los elementos de organización y técnicos que aborda el lenguaje unificado para la construcción de modelos y ordenando todas las reglas, usando un formato predefinido para ser congruentes; con todo esto, se obtiene como resultado un documento flexible, acorde a las particularidades del ambiente orientado a objetos, que describe claramente el funcionamiento de la aplicación y de este modo se ubica rápidamente la información clave.

Cuando un creador de aplicaciones de sistemas ó usuario experto, se apoya con documentación descriptiva que muestra la columna vertebral de su código, se logra entender claramente todos los procesos generados en su programa y con toda precisión y/o oportunidad, se encuentra de inmediato la información deseada, para de este modo responder a cualquier exigencia que se pueda presentar en un marco donde el cambio es la principal constante.



La descripción de la técnica de documentación que se propone se desarrolla a lo largo del segundo apartado para después aplicarla en el tercer y último apartado.

El resultado del uso de una técnica de documentación, para la creación de documentos que describan los aspectos relevantes de cualquier aplicación de sistemas, implica que cualquier usuario del mismo se desempeñe con “calidad”<sup>1</sup> dentro de su organización. Precisamente, la técnica de documentación responde a las necesidades del trabajo de calidad de cualquier usuario; actualmente para que un desarrollo de sistemas pueda llegar a un desenvolvimiento eficaz, se resuelve al considerar todos los momentos del sistema, de acuerdo al instante exacto y los retoma como aspectos distintos pero ligados y con una “orientación interactiva”<sup>2</sup>, que para el momento del sistema en *producción* la técnica de documentación adecuada, propicia una descripción eficaz para un ambiente de trabajo de calidad.

La aplicación de la técnica de documentación se propone efectuarla en la *implantación* de la solución de sistemas, tomando en cuenta las fases de *análisis* y *diseño* y de acuerdo a una orientación interactiva. Durante la etapa final de producción, el documento creado, le da el mantenimiento a los desarrollos y como una manera de resultado cumple con requerimientos de actualización y retroalimentación de los sistemas de aplicaciones.

<sup>1</sup> “Modelo de administración para la calidad. Principios: revoluciona los procesos administrativos. Técnicas: reducción de errores, inventarios, reducir el costo de mantenimiento, formar círculos de calidad con grupos de trabajo para obtener alternativas de solución”.

TORRES Lovera, Juan. Seminario: Análisis de la Planeación.  
UN.A.M., E.N.E.P. Acatlán, México; julio del 2000.

<sup>2</sup> “Existe una cuarta orientación: la interactiva. Esta última orientación que considera al pasado, al presente y al futuro como aspectos diferentes pero inseparables, de la problemática para la que se planea; se concentra en todas las orientaciones al mismo tiempo. Está basada en la creencia de que si no se toman en cuenta los tres aspectos temporales de una problemática, el desarrollo será obstruido”.

ACKOFF, Russell L. Planificación de la Empresa del Futuro.  
Ed. Limusa, México, p. 73; 1983.

# Capítulo I

## La programación orientada a objetos

**OBJETIVO:** En el presente capítulo se explicarán los fundamentos suficientes que conforman lo que se denomina como POO\*, no sin antes; señalar los conceptos básicos del mundo de las *computadoras* para entender y ubicar claramente el objeto de estudio. Por último se aborda la problemática de la investigación propuesta, para poder plantear la solución acorde con el método adecuado para el mismo.







Se comprende la POO como una técnica que sirve para desarrollar aplicaciones por medio de una computadora; las aplicaciones las realiza el *programador* ó el usuario experto en la técnica de programación que se refiere.

Antes de abordar ampliamente el tema de la POO, se señalan algunos conceptos básicos de la informática.

A través de un conjunto de instrucciones ó programas, todos los dispositivos que conforman una computadora funcionan adecuadamente. Una computadora se puede definir como la interacción de dos componentes: el *hardware* y el *software*, cada uno de estos tiene razón de ser gracias al otro, sin uno de ellos el otro no tendría porque existir. Los programas ó aplicaciones de sistemas, pueden estar integrados dentro de la computadora para el funcionamiento de todos los dispositivos de la misma, ó ser creados por el desarrollador de aplicaciones. Las aplicaciones de sistemas creadas por el programador, no son más que instrucciones generadas por el mismo, para la explotación y aprovechamiento de los datos que se almacenan en los equipos de cómputo. Estas aplicaciones se desarrollan con herramientas que se denominan: lenguajes de programación.

Por otro lado, dentro de las técnicas de desarrollo para el diseño de software y como parte de los lenguajes de programación, se ubica la POO. En la actualidad la POO es muy común debido a las necesidades que presenta el entorno de los sistemas, tanto en cuestiones de presentación, como de versatilidad y confianza; sin embargo, esta técnica de desarrollo presenta alguna complejidad y es necesario ordenar claramente sus procedimientos para poder llegar a una aplicación de sistemas eficaz.

\* Siglas de programación orientada a objetos

| CONTENIDO   |   |
|---|---|
|   | Capítulo I                                    |
|   | Lenguajes de programación                     |
|   | Acerca de la programación orientada a objetos |
|   | Ventajas                                      |
|   | Desventajas                                   |
|  | Abordando la problemática                     |

Una aspecto importante de la POO; es su contribución al desarrollo de *aplicaciones gráficas*. Estos tipos de aplicaciones cubren los requerimientos de las nuevas *soluciones a la medida*, para cualquier organización. Los nuevos sistemas de aplicación, entre otras cosas, tratan de *satisfacer inquietudes, inclusive de los usuarios más quisquilloso*. Esto precisamente genera oportunidades inmensas para las aplicaciones gráficas; ya que además de cumplir con necesidades primarias, también cubren requerimientos de presentación.

Algunos expertos en la programación de gráficos como Ben Ezzell, dicen: “ La programación orientada a objetos está íntimamente relacionada con los gráficos. Los objetos aportan posibilidades gráficas nuevas y poderosas que podrían ser demasiado complejas de incluir desde el punto de vista de la programación convencional y en la mayor parte de los programadores”<sup>1</sup>. Este razonamiento muestra la diferencia más notoria que existe para los usuarios de aplicaciones entre: la “programación convencional”, llamada de este modo por Ezzell y la programación orientada a objetos. (Sin embargo, existen otras diferencias importantes que son sumamente notorias para el programador ó usuario experto y las cuales se abordan ampliamente en el siguiente apartado)

Ezzell, menciona como principal ventaja para la programación orientada a objetos su característica gráfica y la distingue de otras herramientas; además de que hace hincapié en la importancia que toman aplicaciones de este tipo en la actualidad. Los razonamientos de Ezzell, tienen gran importancia debido a que precisamente, para que en la actualidad una aplicación presuma de ser eficaz, tiene que contar, entre otras cosas, con una presentación gráfica adecuada<sup>2</sup>.

A continuación y como parte del siguiente apartado, primeramente se menciona el concepto de lenguaje de programación, para después revisar las características de la programación orientada a objetos y de este modo entender claramente la funcionalidad de la herramienta.

<sup>1</sup> EZZELL, Ben. *Programación de gráficos en Turbo C++*. Ed. Addison-Wesley Iberoamericana S.A., E.U.A., p. 359; 1993.

<sup>2</sup> “Con el incremento de la popularidad de las aplicaciones gráficas muchos programas ( y programadores) han renunciado a los menús de texto y a los botones rotulados a favor de las aplicaciones basadas en los *iconos*, como las abanderadas por la computadoras Apple®, por Microsoft® Windows™ y por *Presentation Manager*. En muchos casos, como sucede con Apple® Macintosh™, donde los rótulos de texto aparecen casi como si fueran *ocurrencias tardías*, los *iconos gráficos* han sido llevados hasta sus extremos, cosa que muchos consideran más un obstáculo que una ayuda. En otros casos, tales como *Presentation Manager*, los iconos aparecen principalmente como elementos informativos más que como elementos interactivos primarios. Independientemente de los gustos o las antipatías, los iconos son elementos muy corrientes en la programación gráfica, pudiéndose utilizar en cualesquiera de las formas que dicten las preferencias y las aplicaciones. Si usted encuentra a los iconos molestos o de mal gusto, no se sienta en la obligación de utilizarlos. Si usted los encuentra molestos, entonces todos aquellos que utilicen sus aplicaciones probablemente encuentren molesta la forma en que usted los ha empleado. Tome o deje los iconos en función de sus deseos, pero recuerde que son elementos válidos en la programación gráfica y que aparecerán muy a menudo en el futuro”

*Ibid.*, p. 459.

## 1.1 Lenguajes de programación

Para realizar aplicaciones justas a las necesidades que se presentan en el mundo de la informática, se recurre a los lenguajes de programación, los cuales han ido evolucionando conjuntamente con el desarrollo del hardware y por supuesto, del hombre. Se puede pensar que los programas de aplicación están sujetos al hardware existente; sin embargo, la realidad es otra y es que gracias a estos programas de aplicación o software, que cada vez exigen mejor desempeño en los equipos de cómputo, el hardware ha tenido que evolucionar vertiginosamente.

Los lenguajes de programación se manifiestan como la liga de comunicación que existe entre las computadoras y los seres humanos mediante la programación. Los programas creados con los lenguajes de programación le indican a la computadora la tarea a realizar y como efectuarla. Los primeros lenguajes de programación, eran complicados debido a que la tecnología únicamente permitía el desarrollo de aplicaciones haciendo uso del llamado *lenguaje máquina*. El lenguaje máquina se puede denominar como una manera de comunicación con el hardware basado en el *sistema binario* ó *código de máquina*. (Sistema de compuertas, abierto y cerrado) Naturalmente es obvio deducir que las aplicaciones basadas en sistemas binarios solían ser casi siempre más complicadas para su desarrollo, que el mismo problema que se quisiera resolver en su momento; por lo tanto, la técnica de programación con lenguaje máquina era muy costosa en su fondo aunque no tanto en la forma.

Una alternativa que surgió conjuntamente con el lenguaje de máquina fue el *lenguaje ensamblador*, el cual tiene la característica de manejar *identificadores*, en lugar de códigos; también se basa en la manipulación directa de valores de *memoria* y localidades de memoria. Las tecnologías del lenguaje de máquina y el lenguaje ensamblador forman parte de lo que se denomina como *lenguajes de bajo nivel*.

Actualmente se facilita el desarrollo de aplicaciones gracias a los llamados *lenguajes de alto nivel*, los cuales utilizan *comandos* de fácil entendimiento y normalmente se escriben con frases del vocabulario inglés. (GO TO, IF, THEN. Instrucciones en *Basic*.)

En su momento, los beneficios de los lenguajes de alto nivel son bien recibidos por todos los que tenían que ver algo dentro de la computación; esto debido a que la notación de los programas es más fácil de leer; como en las expresiones aritméticas, de tal forma que el programa podía escribirse en una notación cercana a la que describía originalmente el problema, facilitando el desarrollo del mismo. De alguna manera, los lenguajes del alto nivel también contribuyeron al crecimiento de los usuarios de la computadora.

Los lenguajes de programación actuales llegan a ser más sencillos gracias a su representación estructurada y su naturaleza de compatibilidad con cualquier computadora; además, es posible optimizar sus instrucciones, lo que permite una edición de programas con costos bajos, con reglas universales en su desarrollo y se convierten en una herramienta abierta para cualquier usuario de computadora.

En general; Ravi Sethi, quien es uno de los grandes investigadores de los principios de la computación, concluye en sus estudios lo siguiente: “un lenguaje de programación es una extensión de la *máquina* en que se apoya y un programa es una extensión del lenguaje de programación”<sup>3</sup>.

Gráficamente se puede entender la afirmación de Sethi, a través de la siguiente figura. (Véase figura No.1)

En la figura No. 1 cada burbuja amplía las posibilidades de la capa siguiente:

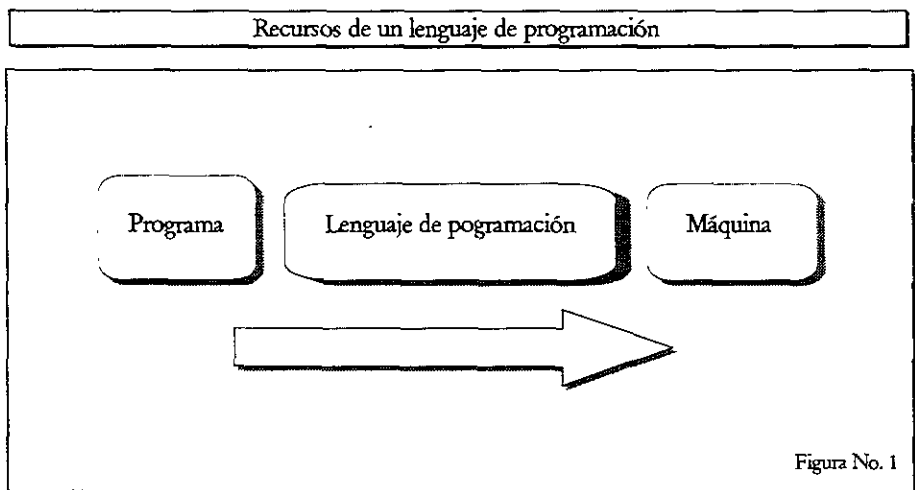
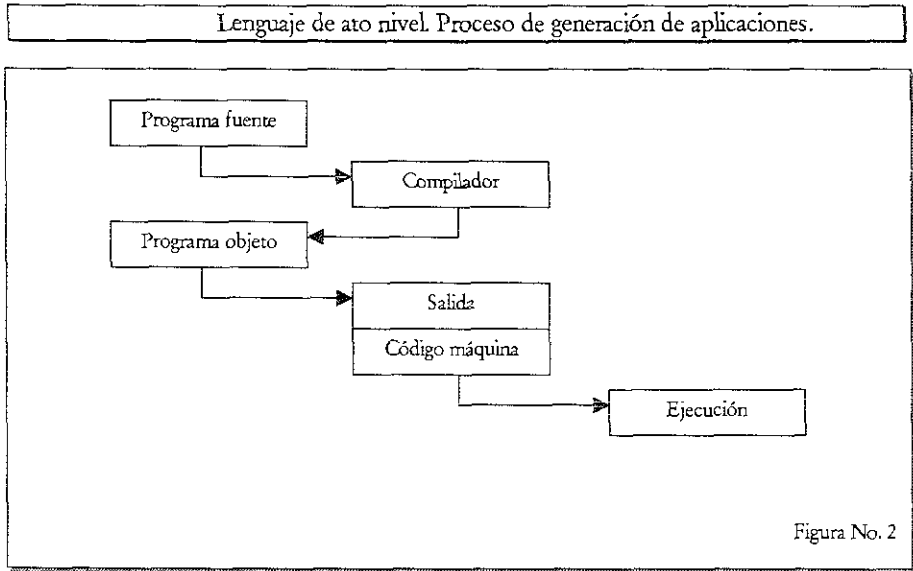


Figura No. 1

Otra característica importante de los lenguajes de alto nivel y de los llamados *lenguajes de cuarta generación*, es su proceso para el desarrollo de aplicaciones; proceso que se define de la siguiente manera:

<sup>3</sup> SETHI, Ravi, *Lenguajes de Programación, Conceptos y Constructores*. Ed. Addison-Wesley Iberoamericana S.A., E.U.A. p. 11; 1992.

Primeramente se genera el *programa fuente* con instrucciones estructuradas y entendibles por el programador, posteriormente se compila el mismo mediante el uso de un *compilador* que esta inherente al lenguaje de programación; para así obtener un *programa objeto* el cual contiene código máquina, que finalmente la computadora utiliza para la ejecución de la aplicación generada. (Véase figura No.2)



Los *lenguajes interpretes* han pasado a segundo termino debido al tiempo que utilizan estos al editar un programa. Siempre será más conveniente que se interprete y se compile el código fuente al final de la edición del programa y una sola vez; como se manejan los lenguajes de alto nivel, que compilar línea por línea, forma como operan los lenguajes interpretes. Algunos lenguajes de alto nivel de gran utilidad son: *Cobol, Pascal, Basic, Fortran* y *C*.

En la actualidad y como una nueva opción de programación; surgen las tecnologías orientadas a objetos. Estas tecnologías utilizan instrucciones similares a los lenguajes de alto nivel, pero con la particularidad de asignar código fuente a objetos muy específicos dentro del ambiente de la organización y de acuerdo al estilo del diseño orientado a objetos; con el propósito de crear aplicaciones que reflejen las actividades de la organización de una forma tangible. Los procesos y hechos de la vida diaria dentro de una organización se reflejarán, con base a la asignación de responsabilidades a los objetos creados virtualmente con POO.

La tecnología orientada a objetos se puede prestar a debates, en función de que algunas personas mencionan que solo es una herramienta creada por un enorme aparato de mercadotecnia; es decir, solo es una "moda", que se debe a intereses muy particulares; sin embargo, es justo mencionar que como toda herramienta nueva para desarrollo y en general como todo lo nuevo dentro de la informática; es una propuesta, que busca estándares de acuerdo a las exigencias del mundo contemporáneo, que procura la creación de sistemas robustos, en ambientes agradables y que representen claramente los procesos en la vida diaria en las organizaciones del ser humano.

Es necesario abordar ampliamente el concepto de la POO para la comprensión del mismo; tema que se desarrolla a continuación.

## 1.2 Acerca de la programación orientada a objetos

"El diseño orientado a objetos y la programación orientada a objetos representan un cambio de perspectiva con respecto a la programación estándar por procedimientos. En lugar de pensar en el flujo del programa desde la primera hasta la última línea de código, se debe pensar en la creación de objetos: componentes autocontenidos de una aplicación que tienen funcionalidad privada además de la funcionalidad que se puede exponer al usuario"<sup>4</sup>.

La afirmación anterior revela la importancia que toma la POO en el ámbito de la informática, al venir a cambiar los esquemas antes fijados y contribuir al desarrollo de *aplicaciones Robustas*.

La técnica de POO permite un cambio de visión para los programadores ya que pasamos de la programación estándar por procedimientos en donde el flujo de información comienza en la primer línea del programa y termina en la última; a un estilo de programación que hace uso de entidades llamadas objetos, los cuales tienen la característica del agrupamiento privado de código fuente, (encapsulamiento de código) y se ejecutan solamente en caso de ser requeridos dichos objetos; a su vez que no afectará a los objetos de su alrededor, a menos que se deseará lo contrario.

Se visualiza a un objeto dentro de la POO, como un conjunto de datos y programas que poseen estructura y forman parte de la organización del sistema. Se puede decir que cada objeto actúa como una entidad distinta y se generan dentro de un sistema, en el cual forma parte, cómo una función integra dentro del mismo, manejando un entorno modular y de autosuficiencia respecto a otros objetos; además tiene encapsulado el código fuente, lo que permite manejar atributos privados y asociados al mismo objeto.

---

<sup>4</sup> MICROSOFT® , Visual FoxPro™ , Manual del Programador. MICROSOFT® , E.U.A., p. 37; 1996.

De alguna manera los objetos lo que buscan es la representación de la realidad tangible; así al definir el objeto casa, este tiene características como: color, tamaño, habitaciones; y distintas funciones como: lugar para habitar, descansar, convivir, etcétera. Dentro de estos objetos se combinan funciones, características y estados para formar la entidad u objeto. Esta es exactamente la idea peculiar del entorno a la POO para representar las diferentes instancias de la realidad.

Como se puede observar la abstracción es la principal característica de la POO, al representar las situaciones y estancias de la vida cotidiana a objetos con particularidades bien definidas. Se trata de atraer necesidades del mundo real a un mundo de razonamiento lógico, en el entorno de la informática. De esta manera los problemas se entienden y solucionan adecuadamente, con una técnica avanzada para el desarrollo de aplicaciones robustas.

En suma, la POO tiene la particularidad de manejar los siguientes tres conceptos: (encapsulamiento, herencia y polimorfismo)

### 1) Encapsulamiento:

Se refiere a la agrupación de los datos de un programa y es el mecanismo con que se ocultan los datos, la estructura interna y los detalles del desarrollo de una aplicación orientada a objetos.

### 2) Herencia:

Se refiere a la característica de retomar atributos de un objeto superior a un objeto de menor nivel, siempre y cuando estén relacionados y formen una misma clase. En resumen, se dice que a esta particular característica de adoptar atributos y funciones se le llama: *herencia de clases*.

### 3) Polimorfismo:

Los programas orientados a objetos pueden realizar operaciones de dos o más objetos, con una misma instrucción (llamada); para ello usan operaciones polimórficas.

Estos conceptos forman la parte fundamental de la POO y acompañados con su representatividad gráfica contribuyen a la formación de aplicaciones orientadas a objetos de fácil desarrollo<sup>5</sup>.

---

<sup>5</sup> LARMAN, Craig. *UML, v Patrones, Introducción al Análisis y Diseño Orientado a Objetos*. Ed Prentice Hall, México. pp 499-501, 1999.



Ahora bien, también a los objetos se le asocian propiedades del mismo, eventos, métodos y relaciones, características que complementen la funcionalidad íntegra de las aplicaciones orientadas a objetos y quedan definidas como sigue:

### **Propiedades:**

Las propiedades de un objeto son los atributos ó características especiales que lo definen, por ejemplo: nombre, tamaño, color, altura, peso, etcétera. Es la forma del objeto.

### **Eventos:**

Los eventos son las acciones que realiza un objeto y que se encuentran inherentes al mismo, por ejemplo: iniciar el objeto, llamarlo, mostrarlo, activarlo, destruirlo, etcétera. Los eventos son las acciones iniciadas por el usuario ó el sistema y que son reconocidas por el objeto.

### **Métodos:**

Los métodos son los procedimientos ó funciones que realiza un objeto. Generalmente estos procedimientos son editados por el programador.

### **Relaciones:**

Estas permiten que el objeto se inserte en la organización ó solución y están formadas esencialmente por *primeros* hacia otros objetos.

Los eventos también se identifican como los *mensajes* ó llamadas que se realizan dentro de nuestro ámbito de estudio; esto debido a que no son otra cosa que situaciones que se realizan gracias a alguna acción y en un período definido.

Es también necesario destacar que las propiedades, los métodos y las relaciones cuentan con la ventaja de compartir atributos a sus “descendientes” en la organización, así un objeto “padre” puede asignar propiedades, métodos y relaciones a la instancia más cercana a él y de nivel inferior. Esto es posible gracias a la manera en como se ordenan las instancias u objetos dentro de un ambiente orientado a objetos; es decir, se ordenan éstos con base a la jerarquía que tienen los mismos dentro de la organización. (De modo jerárquico)<sup>6</sup>

---

<sup>6</sup> *Ibid.*, p. 348.

Finalmente se puede decir, que el estilo de POO lleva al programador a la clasificación de objetos en clases y subclases, agrupadas en un organigrama de clases. Las clases heredan características de las clases superiores e interactúan gracias a las "variables públicas"<sup>7</sup>, que permiten el desenvolvimiento de los sistemas de aplicación y conservando siempre su integridad de datos.

Las ventajas y desventajas que se pueden encontrar en un entorno de objetos informáticos será el tema a tratar a continuación.

### 1.2.1 Ventajas

Las aplicaciones generadas con POO son muy frecuentes hoy en día y contribuyen a la creación de sistemas amigables y de fácil mantenimiento para los cambios que se presentan continuamente dentro de cualquier organización. Los desarrollos de aplicaciones de sistemas deben contemplar representaciones tangibles y gráficas, de acuerdo a la realidad; para un mejor entendimiento de los mismos y con un agradable desenvolvimiento. Con aplicaciones de este tipo, los creadores de los sistemas y los mismos usuarios, contribuyen con un mejor desempeño de sus funciones dentro de cualquier organización.

Los requerimientos que se mencionan para una eficaz aplicación de sistemas, pueden ser cubiertos por cualquier desarrollo orientado a objetos elaborado adecuadamente y forman parte de la fortaleza que proporciona la técnica de programación. Detallando la misma se cuenta con lo siguiente:

#### Alcances:

La POO cumple con desarrollos de amplio alcance en la forma de diseño, desarrollo y mantenimiento de software ofreciendo soluciones a largo plazo. Por ser una herramienta que tiene acción directa con los distintos *sistemas operativos* en los que se desenvuelve, también se muestra como una solución de gran alcance en el mundo de la informática, ya que de igual manera puede integrarse a los distintos ambientes de programación y recicla su código para distintos procedimientos, lo que evita redundancia en los mismos.

---

<sup>7</sup> El tipo de variable que se utiliza en un desarrollo orientado a objetos se define dentro del mismo objeto si se prefiere de este modo ó en el entorno del programa principal; de acuerdo a las necesidades del programador y también dependiendo de la herramienta que se utilice. Se pueden definir variables públicas para que se usen en todo el entorno del programa ó definir las como privadas y únicamente se puedan desenvolver dentro de un objeto de la aplicación. Ejemplo para Visual FoxPro™ versión 5.0 de Microsoft®. La variable "a" al definirla como pública es de la siguiente manera: PUBLIC A; por otro lado para una variable "a" privada se define así: PRIVATE A.

“La herencia como recurso de los lenguajes para definir una nueva clase de objetos como una extensión de clases previamente definidas. La nueva clase hereda las variables y operaciones de la clase anterior. La POO crea un ambiente en el cual los objetos dinámicos interactúan entre sí mediante el envío de mensajes. Los mensajes corresponden a las invocaciones de procedimientos”<sup>8</sup>. La herencia le da a la técnica de programación un alcance amplio para describir procedimientos y aprovechar recursos de objetos definidos con anterioridad; para esto cada objeto funciona como una “caja negra” con la cual el resto del programa actúa, a través de una interfaz. Esta interfaz no es otra cosa que declaraciones de tipos, variables ó procedimientos que permiten ligar y dar una gran amplitud a todos los elementos de una Solución orientada a objetos.

### **Portabilidad y reutilizable:**

El encapsulamiento de código fuente permite la portabilidad de código y puede ser reutilizable en otra área de la organización e incluso en otra organización. Los distintos procedimientos encapsulados en los objetos de una aplicación orientada a objetos extienden los operadores integrados de cualquier desarrollo y de ellos mismos; es decir, los procedimientos permiten ignorar a los demás objetos y se pueden entender operaciones aisladas, además de también contar con la oportunidad de extender las posibilidades de los procedimientos como parte integra de la POO.

Los objetos operan como módulos que dividen un programa en piezas manejables, lo que permiten operar entidades dinámicas y reutilizables para el sistema de aplicación.

### **Seguridad:**

El código fuente en cápsula, permite el ocultar el código fuente, además de que se requieren autorización a cualquier petición, retomando ideas de sistemas operativos tan confiables como el mismo “Unix”<sup>9</sup>.

La creación de especificaciones por medio de abstracción de datos, es un ideal que siempre busca la POO; el concepto de ocultamiento de código fuente e independencia del mismo sugiere que los objetos están empaquetados de manera que los detalles de una solución son concretos y sólidos. La disposición de información nunca se presenta desde afuera, solo es visible para el usuario que tiene el acceso ignorando a los demás procedimientos ó datos y reprimiendo accesos innecesarios cubriendo necesidades integras de seguridad para cualquier solución orientada a objetos.

<sup>8</sup> SETHI, Ravi. *Op. Cit.* p. 212

<sup>9</sup> El sistema operativo UNIX, presenta una modalidad de multiusuario y con un énfasis de seguridad importante, en donde el administrador del sistema toma el mando de los accesos al mismo, con una herramienta confiable.

La tecnología orientada a objetos sigue proliferando hoy en día y aún hay mas opciones, gracias a la adopción de herramientas como el "Java"<sup>10</sup>. Esta tecnología abre todo un mundo de oportunidades a los arquitectos, analistas y diseñadores.

## 1.2.2 Desventajas

La técnica de POO requiere de un amplio conocimiento en construcción de modelos, además de recursos importantes de hardware, para que las aplicaciones generadas con esta técnica no se vean afectadas en su desempeño<sup>11</sup>.

Los usuarios con poca experiencia en el ámbito de orientación a objetos requieren de un gran esfuerzo para entender la nueva filosofía de utilizar objetos y principalmente para el manejo de herencia informática y evitar redundancia al desarrollar aplicaciones. Tal vez la creación de especificaciones abstractas verdaderas ó abstracción de datos es un ideal, al tratar de representar hechos de la vida real, un ideal que difícilmente se alcanza. La independencia para representar un objeto, de manera que pueda cambiar, sin afectar el resto del programa, depende íntegramente de un buen análisis orientado a objetos.

"El proverbio: El hábito no hace al monje, se aplica perfectamente a la tecnología de objetos. El hecho de conocer un lenguaje orientado a objetos (Java, por ejemplo) y además tener acceso a una rica biblioteca (como la de Java), es un primer paso necesario pero insuficiente para crear sistemas de objetos. Se requiere además analizar y diseñar un sistema desde la perspectiva de los objetos"<sup>12</sup>. (Es importante intentar llevar las entidades de la realidad a un mundo informático), Larman dice que es necesario tener un análisis adecuado para poder llegar a un diseño eficaz y de acuerdo a un enfoque de objetos informáticos; finalmente lo que menciona Larman se puede observar que aplica para todas las técnicas de programación; sin embargo dentro de la POO es de suma importancia el utilizar un enfoque abstracto y un énfasis importante hacia un análisis y diseño orientado a objetos. En realidad aquí esta la clave del presente trabajo de investigación.

<sup>10</sup> "Java™ es un lenguaje de programación cuya importancia está creciendo rápidamente, no solamente por los componentes para Internet que proporciona, sino también por su potencia en el campo de las aplicaciones orientadas a objetos".

AGULLO, Soliveres Pedro. *Java Vs. C++: Un Estudio Comparativo de Java.*

Véase: <http://www.ctv.es/USERS/pagullo/poo.htm> México; 2000.

<sup>11</sup> Para un adecuado desempeño de cualquier aplicación orientada a objetos, se requiere como mínimo equipos con procesador 486, con 16 Megabytes de memoria R.A.M. y 250 Megabytes en disco duro. Obviamente estas características de equipo pueden variar dependiendo del tamaño de la aplicación y el ambiente de trabajo en el cual se opere el sistema. (Red, multusuario, personal, etcétera...)

<sup>12</sup> LARMAN, Craig *Op. Cit.* p. 3

Es difícil encontrar desventajas en una técnica moderna, de mucho uso; pero a la vez poco comprendida por la mayoría de los usuarios. En general se observan grandes ventajas más que desventajas al aplicar una herramienta tan útil como la presente; en la cual, lo único que queda es explotarla y ser creativos haciendo uso de mucho "arte" en algunas de sus manifestaciones. Como por ejemplo dentro del manejo de clases.

Una vez que se definieron los aspectos más relevantes en lo que se refiere: primeramente a la informática, después a los lenguajes de programación y por último a la POO. Es tiempo de hablar acerca del objeto de estudio. ¿Cuál es?, ¿Para qué? y ¿Cómo?; es decir, el "Know-How" del trabajo de investigación. Mismo que se trata inmediatamente.

### 1.2.3 Abordando la problemática

Los cambios constantes de tecnología, los ambientes gráficos de los sistemas, la necesidad de crear aplicaciones robustas para los usuarios; son entre otros, algunos de los principales aspectos que debe tomar en cuenta el programador al realizar una aplicación de sistemas.

Muchas veces dentro de las organizaciones existe capacidad de desarrollo, pero no se cuenta con herramientas técnico-administrativas flexibles para darle soporte y seguimiento a los proyectos en desarrollo ó que se encuentran en producción; además, tampoco se le da la importancia que se merece a las técnicas de documentación.

De aquí la inquietud de facilitar herramientas adecuadas de software, con su respectiva documentación, para que acompañen a las aplicaciones y así poder evitar retrasos costosos en la operación de las organizaciones.

La tecnología orientada a objetos le resuelve al programador muchas de las necesidades que requieren las aplicaciones de sistemas actuales. Sin embargo; esto implica ocuparse de otras implicaciones, que no se deben pasar por alto y requieren de comprensión adecuada para poder realizar una aplicación eficaz y acorde a las exigencias del mundo contemporáneo.

Por otro lado, es importante señalar lo siguiente: "¿Por qué los administradores deben planear?. Por que ello da dirección, reduce el impacto del cambio, minimiza el desperdicio y la redundancia y fija los estándares para facilitar el control"<sup>13</sup>. A partir de estos señalamientos que indica acertadamente Robbins, se delimita la investigación propuesta, para darle dirección al trabajo y plantear al mismo tiempo la problemática que se presenta conjuntamente con la solución.

<sup>13</sup> ROBBINS, Stephen P. *Administración: Teoría y Práctica*. Ed. Prentice Hall, México, p. 114; 1994.

En busca de soluciones alternativas para la presente investigación es útil saber lo siguiente: “planear es ocuparse por encontrar el mejor método para lograr un resultado accidental”<sup>14</sup>. Es necesario definir dentro de la planeación el tipo de problema al que se enfrenta, para así poder entender todo el entorno que gira alrededor del mismo y buscar la solución.

Todo esto lleva a una solución que mejora y facilita el logro de las metas dentro de la planeación de cualquier organización y para el trabajo de investigación propuesto.

Las características del problema de estudio se desenvuelve de la siguiente manera y de acuerdo a las variables que se definen a continuación:

- Ambito técnico 50% del problema
- Ambito administrativo 20% del problema
- Ambito jurídico 10% del problema
- Ambito económico 10% del problema
- Factor tiempo 10% del problema

#### **Variables controladas:**

Técnicas de documentación, herramientas de desarrollo, mantenimiento de aplicaciones, personal asignado en el área, tiempo disponible y presupuesto otorgado.

#### **Variables no controladas:**

Presupuesto, estructura de la organización, normatividad, situaciones jurídicas y reingenierías.

Las variables se definen de este modo en una primera fase de estudio del problema. A lo largo del desarrollo de la técnica de documentación propuesta y de acuerdo a las necesidades de la misma organización, las variables pueden tomar distintos matices, además de que también se muestran en función de los diferentes escenarios que se puedan manifestar dentro de una organización y los cuales se analizarán a detalle a lo largo del apartado 3.2.

<sup>14</sup> BIERCE, Ambrose. *Cit. Por. ACKOFF, Russell L. Planeación de la Empresa del Futuro*  
Ed. Limusa, México, p. 41; 1983

Sintetizando el problema y de acuerdo a las características del mismo se define lo siguiente:

El problema de estudio se ubica dentro del área de sistemas de cualquier organización y afecta directamente a los usuarios de tecnologías orientadas a objetos. Este se refiere a la necesidad de adoptar una técnica de documentación versátil que permita describir los procesos de aplicaciones creadas por medio de la POO, para así generar un documento eficiente que enriquezca las mismas aplicaciones.

Es indispensable para poder proponer una técnica útil, el conocer como se construyen los Modelos orientados a objetos y precisar las técnicas de documentación existentes, tanto las diseñadas especialmente para la POO, como las que son dirigidas a la administración de las organizaciones.

Finalmente es de cuidado para el estudio, saber como se encuentran integradas las negociaciones de los sistemas que se desarrollan y también de los que se encuentran en producción, para poder aplicar la técnica de documentación propuesta y sin descuidar, por supuesto, las políticas de la organización.

Todo lo que se señala del problema de estudio y su ámbito de estudio trata de establecer un "enfoque de sistemas"<sup>15</sup>, de acuerdo a lo que se dijo en el seminario análisis de la planeación: "El objetivo del enfoque de sistemas es estudiar los recursos que una sociedad tiene para pensar mejor sobre sus sistemas"<sup>16</sup>. Para ello es importante considerar el objeto de estudio como un sistema que esta dentro de otros más grandes que forma parte de un todo y que se encuentra expuesto a un medio ambiente.

De este modo ubicamos el ámbito de estudio, el cual se considera principalmente técnico-administrativo. Los alcances de la investigación y la solución propuesta de acuerdo a la problemática ya antes mencionada; se especifica funcional para un periodo de cinco años.

El momento que vive la POO define una etapa de crecimiento y consolidación, además con el ritmo de adaptación tecnológica en México, se pretende una revisión de la técnica sugerida después del periodo señalado. Al contar con una técnica que permita describir los procedimientos realizados con POO, se pretende también proponer una herramienta que se adecue a las nuevas aplicaciones que se desarrollen, independientemente del tipo de tecnología que se utilice.

<sup>15</sup> CHURCHMAN, C. West. El Enfoque de Sistemas para la Toma de Decisiones. Diana. México, 1995.

<sup>16</sup> REYES, M. Hugo. Seminario Análisis de la Planeación. UNAM, E.N.E.P. Acatlán, México; marzo del 2000.

Indudablemente que los costos de operación se reducen en un 20% del presupuesto otorgado al área respectiva aproximadamente y todo ello gracias a la independencia que obtiene cualquier área de sistemas, de cualquier apoyo externo ó interno a la misma y se logra obtener un control sobre los procesos que operan y un conocimiento de los aspectos más importantes de la funcionalidad de sus aplicaciones.

Esto se justifica con el uso único del personal que labora dentro del área y de acuerdo a que solamente se utilizará al personal asignado para la operación de las aplicaciones y se evitan gastos en apoyo técnico-administrativo, por parte de personal distinto al área competente. Los nuevos desarrollos y cambios de fondo a las aplicaciones necesitan del personal especializado; sin embargo, la herramienta propuesta funciona adecuadamente con el proceso de mantenimiento de las aplicaciones y sin descuidar la fase que le antecede a la de productivo.

Después del periodo determinado se puede revisar la propuesta y en caso de requerir una actualización, es posible, gracias al énfasis de adaptabilidad que se propone para la técnica de documentación propuesta. Una vez revisando el entorno de la problemática, se puede citar la solución de acuerdo a una acción *prospectiva*; también retomando algunos aspectos relevantes de una acción *retrospectiva* y fijando algunas innovaciones acordes a la solución, como se plantea a continuación:

### Soluciones:

- Promover políticas de capacitación para todos los usuarios que actúan con aplicaciones orientadas a objetos.
- Aplicando y proponiendo elementos viables de técnicas de documentación, para describir modelos orientados a objetos.
- Describir todos los procesos de las aplicaciones con POO, utilizando diagramas y formatos predefinidos y dentro de los límites que permite el ambiente orientado a objetos.
- Localizar los elementos importantes de las aplicaciones, para de este modo ubicar inmediatamente las variables más relevantes en el momento oportuno.

La solución a la problemática tiene una *orientación interactiva*. Se considera esta orientación como la más idónea para buscar la solución óptima a la problemática que se presenta y acorde con lo que dice Ackoff: "Existe una cuarta orientación: la interactiva. Esta última orientación que considera al pasado, al presente y al futuro como aspectos diferentes, pero inseparables, de la problemática para la que se planea; se concentra en todas las orientaciones al mismo tiempo. Está basada en la creencia de que si no se toman en cuenta los tres aspectos temporales de una problemática, el desarrollo será obstruido"<sup>17</sup>.

---

<sup>17</sup> ACKOFF, Russell L. Planificación de la Empresa del Futuro  
Ed Limusa, México, p. 73; 1983



Ackoff menciona la mejor opción, de acuerdo al momento en que se encuentra el objeto de estudio, para de este modo planear adecuadamente la solución. (Véase solución propuesta dentro del Capítulo No. II) En definitiva el retomar todos los aspectos relevantes de cualquier desarrollo de sistemas es fundamental para un buen desempeño del mismo y por consiguiente es indispensable considerar toda la historia del desarrollo y de esta forma se entiende claramente la necesidad de la autosuficiencia de operación de sistemas dentro de cualquier organización.

Tomado el *modelo administrativo* que se adecua al ámbito de estudio y de acuerdo a la solución presentada Es necesario destacar el modelo de *administración para la calidad*.

El modelo administrativo; llamado de administración para la calidad, será el que se retome de acuerdo al tipo de problemática que se plantea y para la solución de la misma. La administración para la calidad favorece desde sus principios, técnicas, personal que ejecuta y hasta sus agregados para establecer completamente la metodología adecuada para el trabajo de investigación. (Véase Cuadro No.1)

De acuerdo al modelo de administración para la calidad, es necesario ocuparse de las instancias siguientes:

***Acción retrospectiva:***

- Uso de modelos estructurados.

***Acción prospectiva:***

- Control total de los desarrollos creados.
- Creación de documentos sencillos y digeribles para entender el funcionamiento de las aplicaciones generadas con POO.
- Solución inmediata a problemas en momentos críticos.

***Innovaciones:***

- Representación flexible de modelos orientados a objetos.
- Documentación íntegra e independiente.
- Herramienta adecuada para el mantenimiento de aplicaciones.
- Bitácora de avance de desarrollo.

**Indicadores:**

- Productividad: Tiempo de producción.  
Mano de obra = Producto / Horas-Hombre.
- Gestión y financieros: Rentabilidad.  
Rendimiento sobre inversión = Utilidad neta / Capital contable
- Evaluación de desempeño.  
Evaluación con base al registro de acontecimiento y momentos críticos.

Al retomar los distintos lapsos del modelo presentado se refleja claramente las necesidades que se manifiestan y acorde con lo que menciona Ackoff. “El concepto de planeación podrá comprenderse con los conceptos derivados de sus orientaciones temporales. La orientación hacia el pasado es llamada reactiva ó retrospectiva, hacia el presente inactiva, para el futuro preactiva ó prospectiva y la que considera todas las orientaciones interactiva”<sup>18</sup>.

Sin lugar a duda y gracias a los elementos que se mencionan en el apartado de “Abordando la problemática”; es tiempo de describir la técnica de documentación propuesta, misma que se desarrollo en el capítulo dos; que a continuación se presenta.

Antes de describir la técnica de documentación propuesta, para las aplicaciones creadas mediante POO; se menciona la metodología que se consideran necesarias y suficientes para enriquecer la técnica propuesta.

---

<sup>18</sup> *Ibid.*, p. 72-73.

|  |
|--|
| <b>Modelo de administración para la calidad*</b> |
|--|

**Principios:**

- La empresa esta compuesta por conocimientos, habilidades y actitudes para producir.
- Reducción de errores. (Cumple con necesidades del cliente y propone innovaciones)
- El modelo se basa en los procesos productivos, administrativos y humanos; da singular importancia a planear, ejecutar, observar y medir.
- El aumento de salario se basa en evaluación de méritos individuales y de grupo.
- Se busca conservar al personal, en especial cuando se tienen problemas económicos.
- Revoluciona los procesos administrativos.

**Técnicas:**

- Reducción de errores.
- Inventarios, reducir el costo de mantenimiento, formar círculos de calidad con grupos de trabajo para obtener alternativas de solución.

**Ejecución:****i) Dirección:**

- Escucha y recibe alternativas por parte del equipo de trabajo.
- Es responsable de implementar la administración para la calidad.
- Tiene el papel de comunicación entre la alta dirección y los trabajadores.

**ii) Supervisión:**

- *Participa en la solución de errores y presenta alternativas de solución practicas e inmediatas.*

**iii) Trabajadores:**

- Se presenta para la satisfacción de todas las necesidades y evita errores.

**Agregados:**

- Analiza los problemas hasta encontrar la causa para tomar medidas coherentemente.
- Los aumentos de salarios son generales.
- Problemas humanos en el trabajo: se manifiesta con la comprensión de grupo para conjuntar diversos criterios.

Cuadro No. 1

\* TORRES, L. Juan. Seminario: Análisis de la Planeación. UNA.M., E.N.E.P.Acaztlán, México; julio del 2000.

## Conclusiones del Capítulo I

La abstracción es la clave de un programa de sistemas y esto se manifiesta en especial en las aplicaciones generadas con programación orientada a objetos; “La programación orientada a objetos se potencializa al transformar a la programación como una extensión natural del pensamiento. La idea es ir evolucionando con el tiempo al igual que el aprendizaje humano pero no por parte de la tecnología de objetos sino por el usuario del paradigma”<sup>19</sup>. Es conveniente ir cada vez más allá al desarrollar cualquier aplicación de Sistemas, las Soluciones orientadas a objetos son precursoras de un cambio de programación y pasan de un mundo informático estructurado con programación estructurada que parte del diseño top-down\*, a otro con entidades particulares y generales en donde se comparten propiedades, funcionalidad, relaciones llamadas herencia entre objetos informáticos y la finalidad es llevar las necesidades de la vida real a un mundo informática para su representación y solución.

El ambiente orientado a objetos permite a los desarrolladores de aplicaciones de Sistemas escribir Soluciones de forma que estén organizadas en la misma manera que el problema y que se busca representar con un modelo informático. La filosofía de esta técnica de programación es principalmente la asignación de responsabilidades de funcionalidad.

La flexibilidad de la orientación a objetos es la principal ventaja de la herramienta, al tener relación entre procedimientos que manipulan entidades y los cambios de cualquiera de estas entidades se verán siempre reflejados en cualquier lugar donde los datos aparezcan, proporcionando estabilidad de la representación de la solución.

Actualmente las herramientas de diseño de aplicaciones convencionales que parten de la idea de empezar el desarrollo del código en la primer línea y de terminar en la última, no favorecen el mantenimiento, ampliación y rehusos pues dispersan los datos por todo el sistema y esto se puede reflejar críticamente en cualquier desarrollo. Como parte de una necesidad de “Revolución industrial del software”, La programación orientada a objetos representa el objetivo de la programación al reunir soluciones a partir de una colección de subcomponentes prefabricados, más que cualquier otra tecnología de programación.

La orientación a objetos es más que un simple conjunto de lenguajes de programación nuevos, sino es una nueva forma de pensar acerca de lo que significa la información dentro de una computadora.

<sup>19</sup> Véase: <http://sol.info.unlp.edu.ar/~luciod/objcour>. Argentina; 2000.

<sup>20</sup> La programación y diseño estructurado top-down, parte de la idea de escribir su código a partir de la primer línea y terminarla en la última; como al leer un libro, sin extensiones de otra naturaleza del programa.

El uso de una “representación artística” de la vida real es la constante de la orientación a objetos como principal innovación y atractivo de la técnica.

Existe hoy la necesidad inagotable de trabajar con “calidad”,<sup>20</sup> (Véase Cuadro No.1, Página No.18) en todos los ámbitos de cualquier organización y al estudiar soluciones técnico-administrativas no se puede olvidar de ello. La orientación a objetos como parte fundamental de cualquier usuario dentro de las áreas de sistemas contribuye a desarrollar y operar aplicaciones de sistemas de calidad.

La programación orientada a objetos, también surge en un ambiente donde las organizaciones cambian continuamente y es necesario contar con aplicaciones robustas que se adecuen a las exigencias del mundo contemporáneo; todo esto fortalece el desarrollo de aplicaciones basada en entidades informáticas.

El principio del buen camino se inicia con el uso de una herramienta adecuada de programación; (como lo es la orientación a objetos) sin embargo, el trayecto es largo y para poder llegar a la meta es necesario buscar el apoyo con un documento apropiado para la misma, situación que se analiza a lo largo del siguiente capítulo con la técnica de documentación que se propone, para la creación de un documento eficaz que describa la esencia de las aplicaciones. (Véase Capítulo No.II)

---

<sup>20</sup> TORRES, L. Juan. Op. Cit.

## Fuentes de Consulta

- ACKOFF, Russell L. Planificación de la Empresa del Futuro. Ed. Limusa, México; 1983.
- ACKOFF, Russell L. Un Concepto de Planeación de Empresas. Ed. Limusa, México; 1972.
- BOOCH, Grady. Diseño Orientado a Objetos. Ed. Prentice Hall, México; 1998.
- CEBALLOS Sierra, Francisco J. Programación Orientada a Objetos con C++. Ed. Alfaomega, México; 1998.
- COX, Brad J. Programación Orientada a Objetos: Un Enfoque Evolutivo. Ed. Addison-Wesley, Wilmington Delaware, E.U.A.; 1993.
- CHURCHMAN, C. West. El Enfoque de Sistemas para la Toma de Decisiones. Ed. Diana, México; 1995.
- DEVIS Botella, Ricardo. Programación Orientada a Objetos en C++. Ed. Paraninfo, Madrid España; 1993.
- EZZELLI, Ben. Programación de Gráficos en Turbo C++. Ed. Addison-Wesley Iberoamericana S.A., E.U.A.; 1993.
- GONZALÉZ Pérez, Manuel. Programación Orientada a Objetos con Borland Pascal™. Ed. Anaya multimedia, Madrid España; 1994.
- JOYANES Aguilar, Luis. Programación Orientada a Objetos. Ed. McGraw Hill Interamericana, S.A., España; 1998.
- LARMAN, Craig. UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos. Ed. Prentice Hall, México; 1999.
- MORALES Lozano, Angel. Programación Orientada a Objetos: Aplicaciones con Smalltalk™. Ed. Paraninfo, Madrid España; 1993.
- ROBBINS, Stephen P. Administración: Teoría y Práctica. Ed. Prentice Hall, México; 1994.
- SETHI, Ravi. Lenguaje de Programación, Conceptos y Constructores. Ed. Addison-Wesley Iberoamericana S.A., E.U.A.; 1992.

## Fuentes de Consulta, Continuación

- VOSS, Greg. Programación Orientada a Objetos: Una Introducción. Ed. McGraw Hill, México; 1994.
- MICROSOFT®. Visual FoxPro™, Manual del Programador. Microsoft®, E.U.A.; 1996.
- REYES Martínez, Hugo. Seminario: Análisis de la Planeación. U.N.A.M., E.N.E.P. Acatlán, México; marzo del 2000.
- TORRES Lovera, Juan. Seminario: Análisis de la Planeación. U.N.A.M., E.N.E.P. Acatlán, México; julio del 2000.
- AGULLO, Soliveres Pedro. Java Vs. C++: Un Estudio Comparativo de Java. <http://www.civ.es/USERS/pagullo/poo.htm>, México; 2000.
- Véase: <http://solinfo.unlp.edu.ar/~luciod/objcour>. Argentina; 2000.

# Capítulo II

## Técnicas de documentación









**OBJETIVO:** A lo largo del capítulo se describen algunas técnicas de documentación existentes, tanto para el ámbito de la programación orientada a objetos, como para el ámbito administrativo. Posteriormente se propondrá una técnica para construir documentos flexibles que ayuden a describir las aplicaciones realizadas con programación orientada a objetos, para que de este modo, se pueda contar con soluciones eficaces y acordes con las exigencias del mundo contemporáneo.

Utilizando la técnica de documentación propuesta se tendrá la capacidad de presentar documentos que se lean y entiendan fácilmente; también, que permitan ubicar la información clave rápidamente; es decir, en el momento oportuno y que auxilien en la toma de decisiones adecuadas, para elevar la productividad dentro de cualquier organización.

Un aspecto de suma importancia que se debe de tomar en cuenta; es el manejo de información. Frecuentemente se presenta que la información de hoy es obsoleta mañana. Por otro lado, la cantidad de datos que generalmente se utiliza, es de grandes proporciones y en muchas ocasiones se sale de control. Las técnicas de documentación auxilian a ordenar y describir claramente la información; lo que implica poder contar con datos relevantes y desechar la información inoperante.

Existen diversas técnicas de documentación, tanto las que son dirigidas a los entornos administrativos ó de negocios, como las que están dirigidas a entornos más especializados y/o técnicos; como lo son las tecnologías orientadas a objetos. De acuerdo a las exigencias del mundo actual, en cuanto a la *calidad* de las aplicaciones que se desarrollan, tanto para soluciones pequeñas u hogareñas, como para las grandes soluciones; se requiere de un soporte amplio, dirigido a todos los operadores de las mismas y que se acople a ciertas situaciones particulares.

En la actualidad la documentación que acompañe a las aplicaciones debe estar dirigida tanto al usuario especializado como al *operador*. Esta razón es la que puede marcar la diferencia entre una buena aplicación de una aplicación de calidad; ya que, de nada sirve el tener un sistema suficiente para las necesidades de la organización, si en momentos agraviantes no es posible resolver problemas; debido entre otras cosas, a la falta de un documento que soporte los procesos de la operación de los sistemas, para que se resuelvan las contingencias en el momento oportuno.

| CONTENIDO   |  |
|---|--|
|    | Capítulo II                                |
|    | Acerta de las técnicas de documentación    |
|    | Técnicas, modelos y procesos de desarrollo |
|    | Técnica de documentación propuesta         |
|   | Organización                               |
|  | Estructuras                                |
|  | Diagramas                                  |
|  | Formatos                                   |



Todos los métodos de documentación se aplican de acuerdo a necesidades existentes en las organizaciones. Las técnicas de documentación que se analizan, en el apartado 2.2, (Véase Página No.27) son las que se consideran primordiales para el objeto de estudio.

Es necesario al hablar de calidad y en particular dentro de un ámbito técnico-operativo de cualquier organización, de las “normas ISO 9000”<sup>1</sup>, las cuales tiene como principal objetivo crear una cultura de calidad. La técnica de documentación propuesta es una fiel muestra del trabajo de calidad e independientemente de que un sistema este certificado con alguna *norma ISO 9000*, se complementa con lo ya existente y puede inclusive ser suficiente para un sistema de aplicaciones, acorde con características de estandarización y eficacia dentro de la organización en que se desenvuelve.

<sup>1</sup> ¿Qué son las normas ISO? : “ISO” es una palabra que deriva del Griego “isos”, que significa “igual” o “estándar”. Curiosamente la palabra se convirtió en el nombre de la International Standard Organization (ISO), es decir, la Organización Internacional de Estandarización. Esta organización es una federación de alcance mundial integrada por cuerpos de estandarización nacionales de 130 países, uno por cada país. Tiene su sede en Ginebra, Suiza, y entre sus miembros se encuentran el Reino Unido y los países de casi toda Europa, Japón, los Estados Unidos, Canadá y varios países en América Latina. Su objetivo es promover mundialmente el desarrollo de normas con el fin de mejorar la eficiencia en la manufactura, el comercio, la comunicación, la operación, la productividad, la calidad y la reducción de costos en las empresas de todo el mundo, para facilitar el intercambio internacional de bienes y servicios. Asimismo, busca la cooperación de sus participantes en la esfera de lo intelectual, científico, tecnológico y económico.

Para lograr lo anterior, la organización ha establecido las normas genéricas ISO 9000, una familia de normativas internacionales orientadas a la gestión y administración de la calidad en las empresas, dichas normas se agrupan en 20 rubros distintos. A pesar de que la organización fue establecida oficialmente el 23 de febrero de 1947, la primera edición de la serie de normas ISO 9000 se publicó en 1987, mientras que la de ISO 14000 fue publicada en 1996.

¿Cuáles son las normas ISO? : Existen numerosas guías, cuatro normativas ISO 9000 y una ISO 14000 de especificaciones. Cada área o empresa selecciona aquella normativa que cubra el alcance de sus productos, servicios y operaciones:

- ISO 9001 Modelo para el aseguramiento de la calidad en el diseño y desarrollo.
- ISO 9002 Modelo para el aseguramiento en calidad en producción e instalación, estableciendo la prevención, detección y corrección de problemas durante la producción e instalación.
- ISO 9003 Modelo para el aseguramiento de la calidad en la inspección final y prueba.
- ISO 9004 Es una guía para sistemas de aseguramiento de calidad.
- ISO 14000 Serie de normas para la gestión ambiental, mismas que proveen el marco de trabajo a los negocios para demostrar su compromiso con la responsabilidad ambiental.

La Norma ISO 9000-1987 define como “sistema de calidad” a: “la organización, estructura, responsabilidad, procedimientos, procesos y recursos para implementar la gestión de calidad”. Es así como el mercado, en busca de la calidad y superar barreras comerciales, adopta para sus productos estas normativas, habiéndose transformado en un símbolo de calidad que muchos fabricantes muestran en sus productos.

## 2.1 Acerca de las técnicas de documentación

Como se menciona en el apartado anterior dentro de las organizaciones es de suma importancia el contar con documentos flexibles que describan los procesos ó aplicaciones, para poder resolver problemas en momentos críticos. Es necesario revisar y tomar en cuenta los aspectos más importantes de las técnicas de documentación que servirán de base, para proponer una técnica viable para las exigencias del mundo contemporáneo.

De acuerdo a las características de la POO y con base a las necesidades actuales que se presentan en las organizaciones, se propone estudiar las siguientes técnicas:

- Método de información Mapping.
- Análisis estructurado moderno.
- Lenguaje unificado de construcción de modelos.

Las técnicas enunciadas contribuyen al desarrollo de una técnica viable para la tecnología orientada a objetos. Al retomar los aspectos de negocio que enseña el método de información Mapping \*, las cuestiones de análisis y diseño (modelado) que otorga el análisis estructurado moderno \*\*, los elementos de organización y técnicos que aborda el lenguaje unificado para la construcción de modelos \*\*\* y ordenando todas las reglas para ser congruentes. De todo esto se obtiene como resultado; una técnica eficaz, flexibles y acorde a las necesidades del ambiente orientado a objetos.

“El enfoque de information Mapping enseña un método para pensar y escribir que tiene una amplia aplicación en los documentos de negocios. Algunas de sus aplicaciones son: procedimiento y métodos, entrenamientos y políticas”<sup>2</sup>.

Con la anterior afirmación se puede entender cual es la finalidad y hasta donde puede llevar un método adecuado de documentación. Las técnicas de documentación sugieren una manera de analizar y organizar las cosas; tal vez, un estilo de pensar. Sin embargo se trata también de crear una herramienta adecuada y con cierta flexibilidad para que se pueda ajustar a las necesidades del usuario y teniendo una mente objetiva para poder adaptar y describir las aplicaciones acertadamente

\* (MIM), Siglas de Método de Información Mapping.

\*\* (AEM), Siglas de Análisis Estructurado Moderno

\*\*\* (UML), Siglas en inglés de Unified Modeling Language.

<sup>2</sup> HORN, Rober E. *Desarrollo de Procedimientos Políticas y Documentación*. InfoMap®, Lexanton MA., p. 1-2, 1991.

Horn además señala: “ El método de information Mapping permite al escritor integrar el uso que el lector le dará al material con el propósito del escritor al escribirlo. Se concentra en dar a la información la secuencia adecuada al uso del lector y en incluir los detalles que el lector necesita”<sup>3</sup>. Al realizar cualquier documento que describa los procesos de la información, la claridad y la flexibilidad del mismo, es siempre lo fundamental para que cualquier usuario, de cualquier aplicación de sistemas se beneficie y fundamente todas sus actividades con un desempeño eficaz.

Por otra parte el análisis estructurado moderno de Yourdon contribuye al desarrollo de la documentación para aplicaciones orientadas a objetos al representar con modelos los procesos de la organización. “Gran parte de la labor que desempeñara como analista involucra el modelado del sistema que desea el usuario”<sup>4</sup>. “¿Por qué se construyen modelos? ¿Por qué no se construye simplemente el sistema mismo? La respuesta es que podemos construir modelos de manera tal que enfatizamos ciertas propiedades críticas del sistema, mientras que simultáneamente desacentuamos otros de sus aspectos. Esto nos permite comunicarnos con el usuario de una manera enfocada, sin distraernos con asuntos y características ajenas al sistema. Y si nos damos cuenta de que nuestra comprensión de los requerimientos del usuario no fue la correcta (o de que el usuario cambió de parecer acerca de sus requerimientos), podemos hacer cambios en el modelo o desecharlo y hacer uno nuevo, de ser necesario”<sup>5</sup>.

La representación, es fundamental al documentar cualquier desarrollo de sistemas y Yourdon lo señala acertadamente.

“El lenguaje unificado de construcción de modelos, nació como una notación estándar de la construcción de modelos; por ello le será de gran utilidad familiarizarse con él”<sup>6</sup>. Larman hace hincapié en la estandarización de la notación dentro de la representación y esto es fundamental dentro del ambiente orientado a objetos.

Al retomar las aportaciones muy particulares de las técnicas del método de información Mapping, el análisis estructurado moderno y el lenguaje unificado de construcción de modelos, todas ellas estándares dentro de su género y agrupando todos estos conceptos con un sentido lógico y con formatos adecuados, se propone una técnica también estándar para lograr representar cualquier programa orientado a objetos, flexible y de acuerdo a requerimientos de calidad para cualquier organización.

<sup>3</sup> HORN, Rober E. *Op. Cit.* p. 2-3.

<sup>4</sup> YOURDON, Edward. *Análisis Estructurado Moderno*. Ed. Prentice-Hall Hispanoamericana, S.A., México, p. 73; 1993.

<sup>5</sup> *Ibid.* p. 73.

<sup>6</sup> LARMAN, Craig. *UML v Patrones, Introducción al Análisis y Diseño Orientado a Objetos*. Ed. Prentice Hall, México, p. xix; 1999.

## 2.2 Técnicas, modelos y procesos de desarrollo

### Método de información Mapping \*

El enfoque de información Mapping, es un estilo para procesar la información y tiene aplicaciones dentro de los documentos de negocios; como por ejemplo en:

- Procedimientos y métodos.
- Entrenamientos y
- Políticas.

El método proporciona habilidades para analizar, organizar información y los objetivos del método son:

- Analizar, organizar y presentar claramente la información y
- Elaborar documentos que tengan las siguientes características:
  1. Se lean y entiendan fácilmente.
  2. Faciliten la búsqueda de la información clave.
  3. Ayuden a que las organizaciones eleven su productividad y
  4. Ayuden a que las organizaciones mejoren la capacidad para la toma de decisiones.

Principalmente el método está dirigido a las personas que diseñan y escriben métodos y procedimientos, materiales técnicos de referencia y documentos de proyectos. Todo esto, por supuesto que contribuye enormemente para describir cualquier programa orientado a objetos.

En general el MIM, presenta la información en unidades modulares basadas en su propósito ó función. Estas unidades tienen la particularidad de poder aislarse y cambiar ó actualizarse fácilmente y en el momento indicado. En especial dentro del ambiente orientado a objetos las unidades modulares de información de acuerdo al método de información Mapping son congruentes con los objetos que se crean en un ambiente de programas orientados a objetos; aquí se manifiesta la importancia y la relación fundamental entre el método de información Mapping y la programación orientada a objetos.

\* HORN, Rober E. Op. Cit

La esencia del método de información Mapping, esta dentro de siete principios que a continuación se enumeran y los cuales son llamados de comunicación.

- Fragmentación
- Relevancia
- Etiquetado
- Consistencia
- Gráficas integradas
- Detalle accesible
- Jerarquía

Todos los elementos mencionados funcionan íntegramente para crear documentos flexibles y ágiles que acompañan la administración de negocios y también validos para tomarlos en cuenta al documentar cualquier desarrollo orientado a objetos. Para poder entender las características de la técnica, es indispensable analizar cada principio a detalle.

#### 1) Fragmentación

Los documentos deben de manejar la información en ideas pequeñas y manejables. De acuerdo a las investigaciones se sugieren hasta " $7 \pm 2$ "<sup>7</sup> unidades de información ó ideas. Ejemplo:

---

Los módulos del seminario taller extracurricular análisis de la planeación que se imparte en la UNAM\*, campus E.N.E.P.\*\* Acatlán son:

- La administración en México y el enfoque de sistemas.
- Concepto de planeación y la toma de decisiones.
- Planeación e instrumentos de la administración.
- Planeación de las organizaciones.
- Modelos de evaluación, predicción y prospectiva.

---

<sup>7</sup> HORN, Rober E. *Op. Cit.* p. 3-A-2

\* Sigla de Universidad Nacional Autónoma de México.

\*\* Sigla de Escuela Nacional de Estudios Profesionales.

La información se encuentra fragmentada de acuerdo a los temas mencionados para el seminario lo que implica el manejo sencillo y organizado de los datos y de manera sencilla.

## 2) Relevancia

Los documentos deben contener una sola idea dentro de un mismo fragmento; es decir, colocar la información de acuerdo a la clase a la que pertenece. Ejemplo:

---

El primer módulo del seminario taller extracurricular: análisis de la planeación, que se imparte en la E.N.E.P. Acatlán, se refiere al estudio del enfoque sistemático.

---

Organizando la información de acuerdo a su relevancia implica poder encontrar la información rápidamente.

## 3) Etiquetado

Después de organizar las ideas en unidades manejables, el documento debe contener etiquetas para cada unidad de información, como lo muestra el ejemplo:

### Análisis de la planeación

---

Módulo 1 La administración en México y el enfoque de sistemas.

---

Módulo 2 Concepto de planeación y la toma de decisiones.

---

Módulo 3 Planeación e instrumentos de la administración.

---

Módulo 4 Planeación de las organizaciones.

---

Módulo 5 Modelos de evaluación, predicción y prospectiva.

---

Gracias a la técnica de etiquetado, también se logra ubicar la información clave acertadamente.

#### 4) Consistencia

Los documentos deben tener palabras, etiquetas, formatos, organización y secuencias similares para temas semejantes. Este principio es importante para organizar la información adecuadamente de acuerdo a un solo criterio; fundamental al representar un Ambito orientado a objetos.

#### 5) Gráficas integradas

Los documentos deben usar diagramas, tablas, ilustraciones, etc. como una parte integral del mismo y no como un complemento que se agrega después de terminar de escribir; es decir, las gráficas son parte misma de la información y no únicamente un elemento de adorno en el documento.

#### 6) Detalle accesible

Los documentos se escriben al nivel del detalle que haga que la información que el lector necesita sea accesible y que el documento sea útil para cualquier persona. Solo escribir lo suficiente y no redundar en la inútil.

#### 7) Jerarquía de fragmentos y etiquetas

Los documentos tienen que agrupar los fragmentos ó elementos relevantes de información en forma jerárquica y poner una etiqueta a cada uno de los grupos más grandes.

Las investigaciones que realiza el MIM indican que las personas normalmente recuerdan mejor cantidades grandes de información si están organizadas en grupos relacionados que contengan de  $7 \pm 2$  elementos; además también leen y encuentran rápidamente la información, si cada elemento es etiquetado. Lo valioso de cualquier documento siempre será la flexibilidad del mismo y de acuerdo a este principio es como el método de información Mapping justifica su efectividad, para utilizarlo en cualquier aplicación orientada a objetos y es el primer elemento que se integra la técnica de documentación propuesta.

## Análisis estructurado moderno \*

El análisis estructurado moderno es una herramienta interesante para los creadores de sistemas de información ó los mismos analistas; es importante repasar los conceptos relevantes de la herramienta para aprovechar algunos elementos y aplicarlos a la técnica de documentación que se propone.

La técnica del análisis estructurado moderno es tan útil para el ámbito orientado a objetos; así como lo es el método de información Mapping, el cual se reviso en el apartado anterior. Las dos técnicas se complementan para presentar un documento integro.

El análisis estructurado moderno, no es más que una versión actualizada y mejorada de lo que es el "modelo clásico" <sup>8</sup> del análisis estructurado. De acuerdo como lo menciona Yourdon: "Obviamente, serán otros los que terigan que juzgar los resultados. Pero sí pienso que es necesario un libro que actualice algo del material clásico del análisis estructurado que se publicó por primera vez a finales de la década de los 70's. Cuando Tom DeMarco escribió *Structured Analysis and Systems Specification*, y Chris Gane y Trish Sarson escribieron *Structured Systems Analysis: Tools and Techniques*, no existían los lenguajes de programación de cuarta generación y no había herramientas de creación de prototipos disponibles para los creadores de sistemas. Las computadoras personales no existían en aquellos días, a excepción de algunas de las máquinas primitivas de Apple® y Radio Shack®. No había productos de software para estaciones de trabajo que pudieran auxiliar al analista de sistemas en la creación de diagramas de flujo de datos" <sup>9</sup>. Tal vez, es necesario señalar, que aún, el análisis estructurado moderno propuesto por Yourdon, tiene algunos años de haber sido creado, que son significativos para el ambiente de la informática; sin embargo, la técnica es de mucha ayuda en este momento y es muy utilizada, gracias al énfasis de adaptabilidad; que es algo de lo que se busca al integrar parte del método de información Mapping y la representatividad del análisis estructurado moderno en la técnica de documentación que se describe a partir del apartado 2.3. (Véase Página No.48)

De acuerdo con lo que menciona Yourdon, en cuanto a la actualización del modelo clásico (el análisis estructurado) y su complemento, que viene siendo el análisis estructurado moderno; se sabe que al hacer uso del análisis estructurado moderno, también se toman aspectos relevantes del modelo clásico.

\* YOURDON, Edward. *Op. Cit.*

<sup>8</sup> DEMARCO, Tom. *Structured Analysis and System Specification*.  
Ed. Prentice Hall, Englewood Cliffs, N.J., 1979

GANE, Christopher P. *Análisis Estructurado de Sistemas*.  
Ed. El Ateneo, México, 1987

<sup>9</sup> YOURDON, Edward. *Op. Cit.* p. vii.



A lo largo de todo el estudio se mencionan los aspectos más importantes de ambos modelos y la relación con la programación orientada a objetos. Gracias al uso de módulos dentro del análisis estructurado moderno se logran analogías con la orientación a objetos; es decir, la relación que existe entre un módulo y un objeto respectivamente es parte fundamental para el desarrollo de un documento orientado a objetos.

La finalidad primordial del análisis estructurado moderno es difundir la importancia del análisis de sistemas y también a su vez satisfacer las necesidades dentro de cualquier organización de los sistemas creados por el hombre; aquí la importancia de retomar los aspectos relevantes del análisis estructurado moderno y también es necesario que se contemplen las siguientes cuatro consideraciones que menciona Bertalanffy:

- 1) "Entre más especializados sea el sistema, menos capaz es de adaptarse a circunstancias diferentes.
- 2) A medida de que el sistema es más grande, también son más los recursos que se deben asignar al mismo para el mantenimiento diario.
- 3) Los sistemas siempre forman parte de sistemas mayores y se pueden dividir en sistemas menores.
- 4) Los sistemas siempre crecen" <sup>10</sup>.

Los cuatro aspectos de los sistemas forman parte de lo que se busca controlar dentro de cualquier organización y es indispensable reconocerlos al proponer una herramienta de documentación.

Para lograr contar con documentación que describa los procedimientos y estructuras principales de cualquier sistema de aplicación, es necesario retomar aspectos del análisis estructurado moderno y complementarlo con la herramienta de documentación propuesta y durante la fase de la implementación del sistema de aplicación.

Las personas que interactúan con los sistemas de aplicación son distintas y actúan en distintos tiempos. Es necesario considerar a todas ellas para proponer un documento flexible y que pueda ser utilizado en el momento de la implantación de la aplicación.

<sup>10</sup> BERTALANFFY, Ludwig Von. Tendencias en la Teoría General de Sistemas. Ed. Alianza, Madrid España; 1978.

El personal a considerar es el siguiente, todos ellos con su responsabilidad muy particular para la aplicación de sistemas:

- Auditores.
- Administradores.
- Analistas de sistemas.
- Diseñadores de sistemas.
- Programadores.
- Personal de operaciones y
- Usuarios.

Los principales elementos para complementar la técnica de documentación propuesta y que forman parte del análisis estructurado moderno son:

- Diagramas de flujo de datos. (DFD)
- Diccionarios de datos. (DD)
- Diagramas de entidad-relación. (DER)

Cada una de ellas buscan una representación mediante modelos de los procesos desarrollados y forman parte importante para lograr describir los Procesos orientados a objetos.

### Diagramas de flujo de datos

Este elemento es uno de los tres aspectos gráficos del análisis estructurado moderno a considerar y es de lo más relevante dentro del modelado de la técnica de documentación. Los diagramas de flujo de datos revelan las funciones del sistema mediante la relación de *conductos* y *tanques de almacenamiento de datos* ó *burujas*. Algo importante de destacar es que estos diagramas existieron desde “los últimos 70 años para construir modelos de flujo de trabajo de organizaciones. Es importante tener esto en mente: los diagramas de flujo de datos no sólo se pueden utilizar para modelar sistemas de procesos de información, sino también como manera de modelar organizaciones enteras, es decir, como una herramienta para la planeación estratégica y de negocios”<sup>11</sup>

<sup>11</sup> YOURDON, Edward. Op. Cit., p. 158  
GANE, Christopher P. Op. Cit.

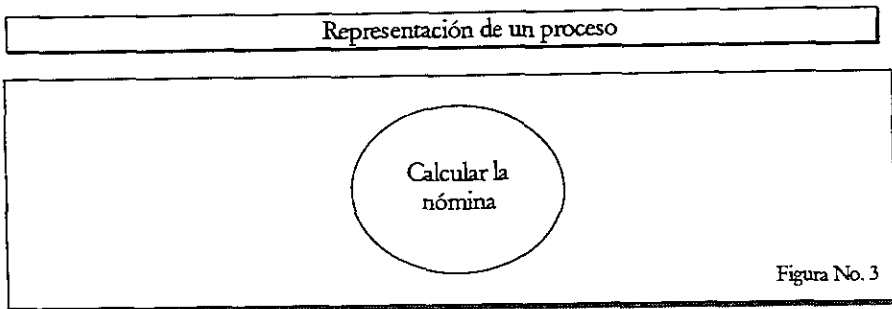
Los componentes principales para representar un sistema mediante un modelo de diagramas de flujo de datos son los siguientes:

### Proceso

Es el primer componente de un diagrama de flujo de datos,<sup>12</sup> se representa mediante una burbuja y la función del proceso es mostrar como las entradas de información se traducen en salidas; es decir se representan las funciones del sistema. (Véase Figura No.3)

Al representar cualquier burbuja es similar a la representación de cualquier objeto dentro de la programación orientada a objetos de aquí la importancia de las burbujas de información. Los objetos para cualquier desarrollo orientado a objetos pueden ser congruentes con los procesos en los diagramas de flujos de datos. Es así como se liga la idea de representar los objetos con los procesos definidos en el análisis estructurado moderno.

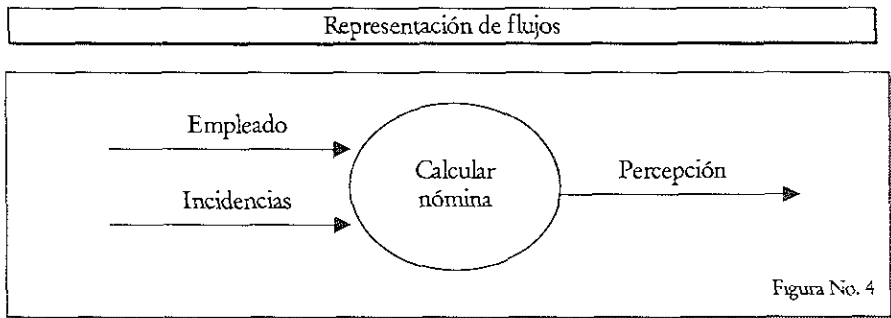
La parte medular de la técnica de documentación propuesta se presenta precisamente en la representación de los procesos y se refleja cuando se relacionan las responsabilidades obtenidas en cualquier aplicación orientada a objetos con los procesos; esto quiere decir que las responsabilidades otorgadas para cualquier objeto se pueden modelar con una burbuja ó proceso.



<sup>12</sup> YOURDON, Edward. Op. Cit. p. 160.

## Flujo

Se refiere al flujo de la información y se representa con una flecha que entra y sale del proceso. <sup>13</sup> (Véase Figura No.4)



La delimitación de responsabilidades para cualquier objeto dentro de un ambiente de objetos se logra modelar con flechas de dirección. (Flujos de información) Acertadamente Yourdon logra representar funciones y esto se adecua perfectamente a un modelado de objetos, necesario al documentar cualquier aplicación y fundamental para la orientación a objetos.

## Almacén

El almacén son los archivos de datos que se encuentran dentro del sistema y se representan mediante dos líneas paralelas, acompañadas de la etiqueta ó nombre del almacén. <sup>14</sup> (Véase Figura No.5)

<sup>13</sup> YOURDON, Edward. Op. Cit., p. 162.

GANE, Christopher P. Op. Cit.

DEMARCO, Tom. Op. Cit.

<sup>14</sup> YOURDON, Edward Op. Cit., p. 168.

---

 Representación de un almacén de datos
 

---



Empleados

Figura No. 5

El almacenamiento de datos de cualquier aplicación se modela para un entendimiento de la ubicación exacta de los datos almacenados.

## Terminador

El último componente que integra un diagrama de flujo de datos es el terminador,<sup>15</sup> el cual se representa mediante un rectángulo y proporciona información de las entidades externas que se comunican con el sistema que se describe. (Véase Figura No.6)

---

 Representación para el terminador
 

---



Departamento de  
contabilidad

Figura No. 6

El elemento llamado terminador puede delimitar todo el entorno orientado a objetos, situación fundamental para poder establecer las fronteras de cualquier sistema y también delimitar enteramente responsabilidades del mismo.

---

<sup>15</sup> YOURDON, Edward *Op. Cit.* p. 175.

El uso de los elementos mostrados del análisis estructurado moderno es parte de la técnica de documentación propuesta, al lograr representar los contenedores de información de cualquier aplicación de sistemas en forma sencilla y en especial para las desarrolladas con programación orientada a objetos.

### Diccionario de datos

“ El diccionario de datos de frase casi se define. El diccionario de datos es un listado organizado de todos los datos pertinentes al sistema, con definiciones precisas y rigurosas para que tanto el usuario como el analista tengan un entendimiento común de todas las entradas, salidas, componentes de almacenes y cálculos intermedios”.<sup>16</sup>

Los diccionarios de datos; en pocas palabras, describen todos los datos detallados de la aplicación de sistemas y en un lenguaje común para todos los participantes del sistema y sus principales características son:

- Describe los flujos y almacenes que se muestran en los DFD.
- Proporciona la información al detalle de los paquetes de información.
- Especifica las unidades y los valores relevantes de la información.
- Señala los detalles de la información, mediante un lenguaje conocido por todos los participantes del mismo.

La idea de describir el significado de algunos elementos de las aplicaciones orientadas a objetos se fundamenta con base a la complejidad de las mismas. Precisamente los diccionarios de datos buscan describir lo suficientemente razonable para todos los participantes de las aplicaciones de sistemas y de este modo llegar a un entendimiento común de todos los términos usados dentro de cualquier desarrollo.

El nivel de complejidad que se desea manejar en cualquier descripción de las aplicaciones orientadas a objetos se puede modular gracias al uso del diccionario de datos. Los sistemas deben ser descritos de un modo adecuado para el entendimiento de cualquier usuario del mismo y así se satisfacen necesidades intrínsecas de los mismos.

---

<sup>16</sup> YOURDON, Edward Op. Cit., p 212.

GANE, Christopher P. Op. Cit.

DEMARCO, Tom. Op. Cit.

La notación que se utiliza para la elaboración de los diccionarios de datos es la siguiente: (Véase Cuadro No.2)

Notación para el diccionario de datos

| <i>Símbolo</i> | <i>Descripción</i>                             |
|----------------|--|
| =              | Está compuesto de ...                          |
| +              | Y...   |
| ( )            | Opcativo (puede estar presente o ausente).     |
| { }            | Iteración.                                     |
| [ ]            | Selecciona una de varias alternativas.         |
| **             | Comentario.                                    |
| @              | Identificador. (Campo llave), para un almacén. |
|                | Separa opciones alternativas...                |

Cuadro No. 2

Al elaborar un diccionario de datos se describe como lo muestra el siguiente ejemplo. Para una entidad llamada "Empleado":

**Empleado = Nombre + Domicilio + Expediente +Puesto**

**Nombre = Paterno + Materno + Nombre + (Segundo Nombre)**

**Paterno = {Carácter}**

**Materno = {Carácter}**

**Nombre = {Carácter}**

**Segundo Nombre = {Carácter}**

**Carácter = {A-Z|a-z| |}**

Con esto se observa y se entiende que existe dentro de alguna aplicación de sistemas un *campo* de nombre "empleado", con tres registros llamados: "nombre", "domicilio", "expediente" y "puesto"; cada registro con sus propias características indicadas dentro de la descripción del diccionario de datos y que en conjunto forman parte de una entidad.

## Diagrama de entidad-relación

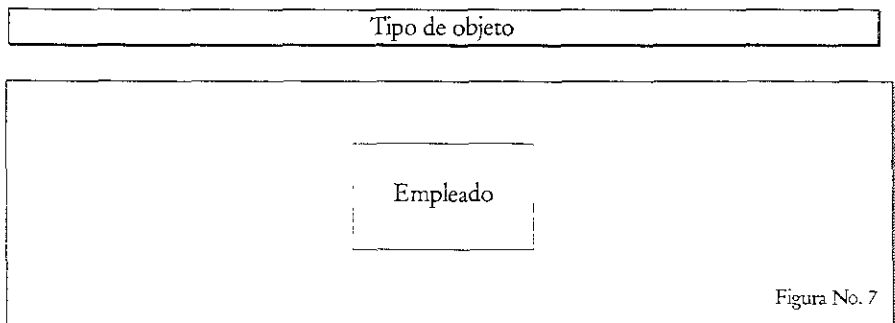
Es tiempo de abordar el tema de los diagramas de entidad-relación, última técnica que proporciona el análisis estructurado moderno para ser contemplada por la técnica de documentación propuesta. “El diagrama de entidad-relación es una herramienta efectiva de modelado para comunicarse con el grupo de administración de base de datos. Basándose en la información presentada por el diagrama de entidad-relación, el grupo de administración de base de datos puede ver el tipo de *datos*, *índices* o *apuntadores* que se necesitarán para llegar de manera eficiente a los registros de *bases de datos*.”<sup>17</sup> De acuerdo con lo que menciona Yourdon, los diagramas de entidad-relación son muy útiles para representar los sistemas con modelos de base de datos, algo de lo cual resulta ser de mucha trascendencia para la mayoría de las aplicaciones existentes y por supuesto acorde con necesidades actuales de desarrollos orientados a objetos.

En el diagrama de entidad-relación se consideran cuatro componentes principalmente:

- Tipos de objetos.
- Relaciones.
- Indicadores asociativos de tipo.
- Indicadores de supertipo.

### 1) Tipo de objeto

Se representa mediante un rectángulo y contiene un conjunto de objeto(s) del mundo real. (Véase Figura No.7)



<sup>17</sup> YOURDON, Edward *Op. Cit.*, p. 261



El modelado de un tipo de objeto es muy cercano al mundo real y es idéntico a cualquier objeto característico en una aplicación orientada a objetos, situación que también la considera el análisis estructurado moderno mediante este tipo de diagramas de entidad-relación.

## 2) Relaciones

La forma como se conectan los objetos entre sí es mediante las relaciones y se representa mediante un rombo. (Véase Figura No.8)

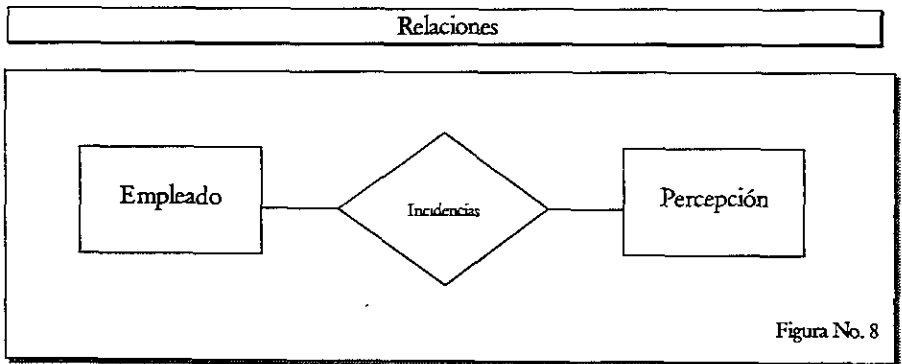


Figura No. 8

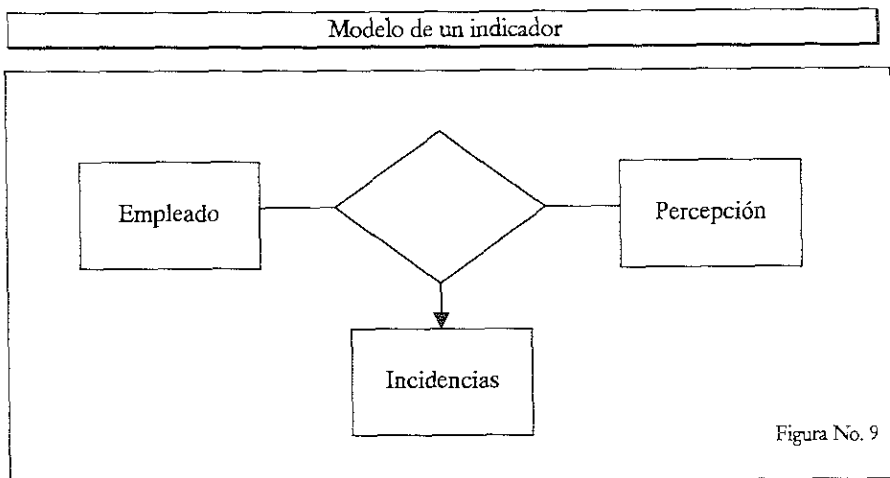
Al hablar de relaciones también se habla de ocurrencias que existen entre diversos objetos; siempre existirá alguna(s) conexión(es) entre objetos, sí y solo sí, hay por lo menos una ocurrencia que los afecte directamente. Yourdon señala: " Es importante reconocer que la relación representa un conjunto de conexiones. Cada instancia de la relación representa una asociación entre cero o más ocurrencias de un objeto y cero o más ocurrencias del otro. " <sup>18</sup>

El término de objeto que utiliza Yourdon al hablar de las relaciones o actividades que existen entre objetos modelados, demuestra la similitud que existe entre la descripción modular del análisis estructurado moderno y el ambiente orientado a objetos. Es posible relacionar identidades de un programa orientado a objetos con objetos de un análisis estructurado moderno, situación que favorece al modelado de una Solución orientada a objetos.

<sup>18</sup> YOURDON, Edward *Op. Cit.* p. 264.

## 4) Indicadores asociativos de tipo

Los indicadores de tipo sirven para representar lo que opera como objeto y relación a la vez dentro de cualquier aplicación de sistemas. (Véase Figura No.9)

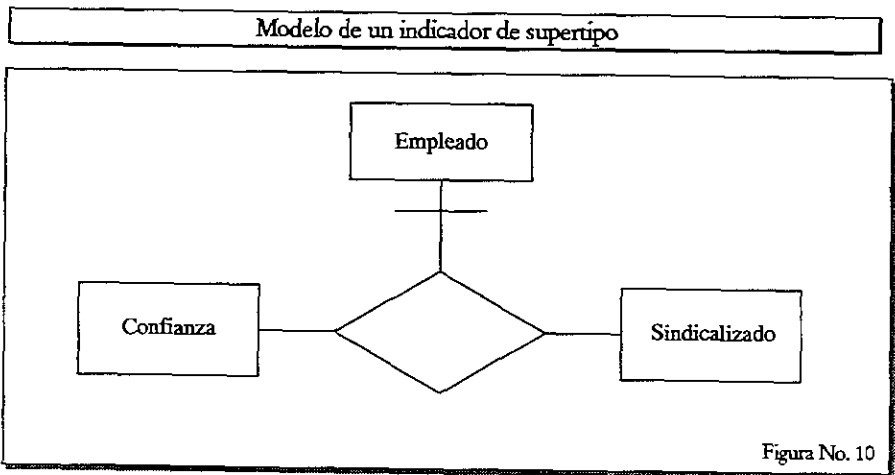


La representación de la doble función se refleja cuando al rombo se le agrega una línea dirigida hacia un nuevo rectángulo con la etiqueta respectiva. El resultado de la relación entre dos objetos se representa como un indicador. En el ejemplo mostrado, (Figura No. 9) las ocurrencias entre el objeto “empleado” y el objeto “percepción” son las “incidencias”, también se denominan indicadores asociativos de tipo.

Los indicadores representan dentro de un programa orientado a objetos la respuesta a las actividades de una identidad, el modelado de los mismos reflejan gráficamente las salidas de cualquier proceso.

## 5) Indicadores de supertipo

Los indicadores de supertipo son los tipos de objetos de una ó más categorías, conectadas por una relación, que tiene una barra horizontal como distintivo y sin líneas dirigidas. (Véase Figura No.10)



Los indicadores de supertipo también tienen una gran contribución para modelar cualquier Solución orientada a objetos, al describir adecuadamente las diversas categorías ó clases existentes, para las diversas entidades. En la Figura No.10 el objeto "Empleado" tiene dos clases para las entidades "confianza" y "sindicalizado".

Las clases en un programa orientado a objetos son parte esencial y algo que los distingue de otras tecnologías; situación muy favorable para el modelado de indicadores de supertipo.

Todos los elementos del análisis estructurado moderno que se refieren, forman parte de la técnica de documentación propuesta y colaboran en el modelado de cualquier programa orientado a objetos. Para el personal involucrado con soluciones orientadas a objetos es importante visualizar el desarrollo sobre el cual se trabaja y el modelado que proporciona el análisis estructurado moderno es flexible y suficiente para describir cualquier aplicación de sistemas.

Antes de que se hable de la técnica de documentación propuesta es indispensable mencionar la técnica del lenguaje unificado de construcción de modelos.<sup>19</sup>

<sup>19</sup> "El UML (lenguaje unificado de construcción de modelos) se define como un lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software. Es un sistema notacional (que, entre otras cosas, incluye el significado de sus notaciones) destinado a los sistemas de modelados que utilizan conceptos orientados a objetos".

LARMAN, Craig, *Op. Cit.* p. 15.

## Lenguaje unificado de construcción de modelos \*

El lenguaje unificado de construcción de modelos, nació como una notación estándar de la construcción de modelos; por ello es de gran utilidad en la actualidad. Mediante la técnica de notación UML, se entiende como diseñar y analizar tecnologías orientadas a objetos; la técnica también tiene como objetivo primordial crear mejores diseños orientados a objetos creados con programación orientada a objetos y para ello recurre a principios explicables y a la *heurística*.

“La habilidad más importante en el análisis y el diseño orientado a objetos es asignar eficientemente las responsabilidades a los componentes del software”<sup>20</sup>; es decir, la asignación de responsabilidades implica el ejercer funciones sobre alguna entidad u objeto.

En general, se dice que la esencia del análisis y el diseño orientado a objetos, consiste en situar el dominio de un problema y su solución lógica dentro de la perspectiva de los objetos. (Cosas, conceptos ó entidades) Durante este análisis orientado a objetos se trata de identificar y describir los objetos dentro del dominio del problema.

Al recordar que los objetos en la programación orientada a objetos, tiene *atributos y métodos*. Así durante el diseño orientado a objetos, se procura definir los objetos lógicos del problema que finalmente se implementan en un lenguaje orientado a entidades.

Para empezar a describir la técnica del lenguaje unificado de construcción de modelos primeramente se definen los conceptos importantes que la conforman y que ayudan a construir una técnica de documentación integral y flexible.

Elementos a destacar del lenguaje unificado de construcción de modelos:

- Casos de uso
- Modelo conceptual
- Diagramas de colaboración
- Diagramas de diseño de clases

\* FOWLER, Martin. *UML Gota a Gota: Actualizado para Cubrir la Versión 1.2 OMG UML Standard*. Ed. Addison Wesley Longman, México; 1999

ERIKSSON, Hans-Erik. *UML Toolkit*. Ed. Wiley, New York E U.A.; 1998.

LARMAN, Craig. *Op. Cit.*

<sup>20</sup> LARMAN, Craig. *Op. Cit.* p. 5.

Al ubicar el UML, dentro de las distintas fases del desarrollo de sistemas y de acuerdo a las herramientas que utiliza, se puede definir de la siguiente manera: (Véase Cuadro No.3)

| Lenguaje unificado de construcción de modelos |  |  |
|---|--|--|
| <i>Fase</i>                                   | <i>Modelo</i>  | <i>Herramientas</i>  |
| Planeación y elaboración                      | <ul style="list-style-type: none"> <li>• Casos de uso</li> </ul>   | <ul style="list-style-type: none"> <li>• Entorno</li> <li>• Requerimientos</li> <li>• Descripción de procesos</li> <li>• Clasificación y programación</li> </ul> |
| Desarrollo:<br>1.- Análisis                   | <ul style="list-style-type: none"> <li>• Modelo conceptual</li> </ul>  | <ul style="list-style-type: none"> <li>• Construcción del modelo</li> <li>• Asociaciones</li> <li>• Atributos</li> <li>• Glosario de términos</li> </ul>         |
| 2.- Comportamiento del sistemas               | <ul style="list-style-type: none"> <li>• Diagrama de secuencia</li> <li>• Contratos</li> </ul>                         | <ul style="list-style-type: none"> <li>• Eventos y operaciones</li> <li>• Contratos para operaciones</li> </ul>  |
| 3.- Diseño                                    | <ul style="list-style-type: none"> <li>• Diagramas de colaboración</li> <li>• Diagramas de diseño de clases</li> </ul> | <ul style="list-style-type: none"> <li>• Diagrama de colaboración</li> <li>• Responsabilidades</li> <li>• Visibilidad</li> <li>• Diseño de clases</li> </ul>     |

Cuadro No. 3

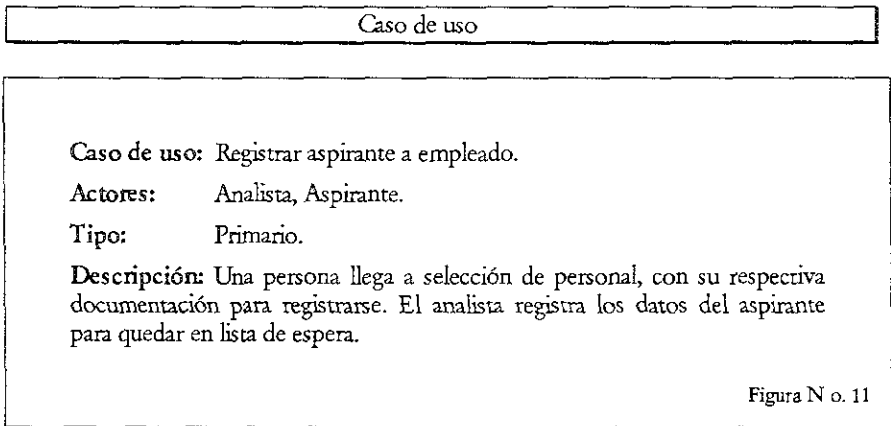
### 1) Casos de uso

Al hablar del caso de uso se habla del entorno y los factores externos que participan en los procesos. Dichos procesos pueden expresarse en descripciones narrativas y con un formato de estructura de prosa. Es necesario describir los requerimientos del caso de uso ya que representan las descripciones de las necesidades de un sistema para identificar y documentar lo que se requiere.

Al desarrollar los requerimientos se recomienda lo siguiente:

- Explicar el panorama general.
- Describir a los *dientes*.
- Describir las metas.
- Proponer las funciones del sistema y
- Proponer los atributos del sistema.

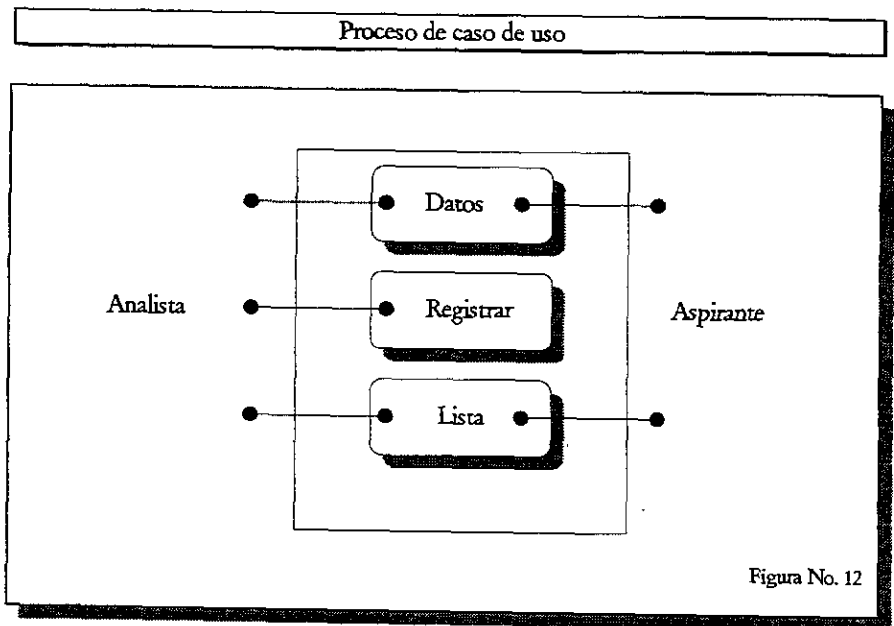
Ejemplo de un proceso, para un caso de uso:



El proceso presentado en la Figura No.11, también puede representarse gráficamente. En la representación gráfica se observa claramente a los actores de los sistemas y la relación que hay entre éstos <sup>21</sup>. ( Véase Figura No.12)

<sup>21</sup> LARMAN, Craig *Op. Cit.*

FLOWLER, Martin. *UML Got a Goot: Actualizado para Cubrir la Versión 1.2 OMG UML Standard*. Ed. Addison Wesley Longman, México, 1999.



En general los diagramas de los casos de uso, tienen por objeto ofrecer una descripción contextual que permite conocer rápidamente los actores externos de un sistema y las formas básicas en que lo utilizan.

Otro aspecto para la fase de planeación y elaboración. Es la programación y clasificación de los casos de uso; para que de esta manera mas adelante se pueda instrumentar la fase del análisis y diseño.

Para el UML , la programación se realiza en ciclos y de acuerdo a los requerimientos de caso de uso. Lo que se busca con esto, es no tener casos de uso muy complejos, que dificulten la descripción de los mismos.

Para la técnica de documentación propuesta, la identificación de los casos de uso es indispensable desde el punto de vista de la representación jerárquica, aspecto se suma importancia para la programación orientada a objetos, ya que siempre el identificar jerarquías lleva a las soluciones de herencia informática, importante dentro del Modelado de aplicaciones orientadas a objetos.<sup>22</sup>

<sup>22</sup>LARMAN, Gaig. *Op. Cit.* p. 47-82.

Es necesario clasificar los casos de uso de acuerdo a su jerarquía; es decir, la clasificación es acorde con la influencia que tiene los mismos sobre el sistema. Para ello se recomienda considerar lo siguiente:

- Los que tienen una fuerte repercusión en el diseño del sistema.
- Destacar los que contienen importante información.
- Detectar las funciones de riesgo, urgentes ó complejas.
- Verificar los que consideran tecnología de punta.
- Detectar los procesos primarios de la línea de negocio y
- Los que afectan directamente a los ingresos ó costos de la organización.

La calificación que corresponde a la clasificación de los casos de uso que se exponen es con base a la escala: alto, mediano y bajo.

### Modelo conceptual

El modelo conceptual representa los conceptos del problema en el mundo real. Consiste en explicar a los desarrolladores de aplicaciones los conceptos significativos en un dominio del problema; es decir, es la herramienta más importante a crear durante el análisis orientado a objetos.

Una característica primordial en el análisis ó investigación orientada a objetos es descomponer el problema en conceptos u objetos individuales. “En términos informales el concepto es una idea, cosa u objeto. Es un lenguaje más formal, podemos considerarlo a partir de su símbolo, intensión y extensión”.<sup>23</sup> En general el concepto se describe de la siguiente forma, según Larman:

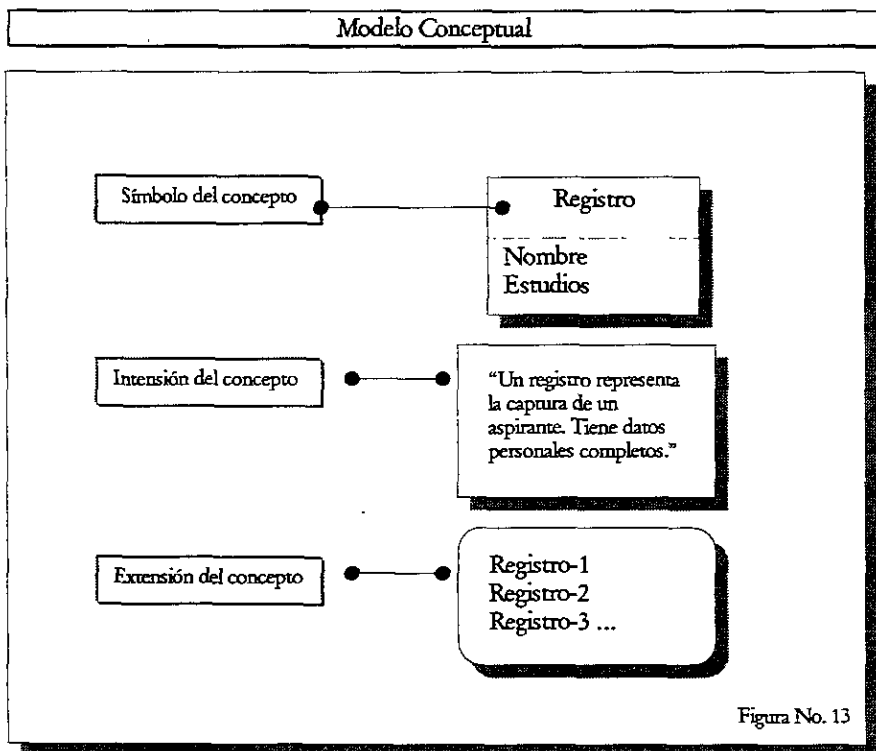
- Símbolo: palabras ó imágenes que representan un concepto.
- Intensión: la identificación del concepto y
- Extensión: el conjunto de ejemplos a que se aplica el concepto.

Describir un modelo de concepto, dentro del análisis orientado a objetos, se distingue la división de responsabilidades por objetos ó conceptos; en cambio para el análisis estructurado moderno la división se realiza con base a funciones ó responsabilidades.

<sup>23</sup> LARMAN, Craig. *Op. Cit.* p 89.



Cuando se crea un modelo conceptual, por lo regular la vista del símbolo y de la intensión de un concepto es el aspecto de mayor interés práctico. (Véase Figura-No.13)



#### Diagramas de colaboración \*

Los diagramas de colaboración muestran gráficamente el flujo de mensajes entre las entidades y la invocación de métodos.

\* "Debido a la especialización de los diagramas de colaboración, se sugiere no entrar a detalle y recomendarlos solo para las personas que quieran conocer mas a fondo del lenguaje unificado de construcción de modelos. (UML)"

LARMAN, Craig. *Op. Cit.* p. 15.

FOWLER, Martin. *Op. Cit.*

El diagrama de colaboración, conjuntamente con los diagramas de secuencia; dos tipos de diagramas que utiliza el lenguaje unificado de construcción de modelos, forman parte de lo que se denominan iteraciones de un sistema.

Este tipo de diagramas, auxilian a expresar como los objetos interactúan, a través de mensajes para realizar tareas. En pocas palabras la herramienta explica gráficamente las interacciones existentes entre las instancias y las clases del modelo de éstas.<sup>24</sup>

#### Diagramas de diseño de clases \*

Se refiere a los métodos que utilizan las entidades y que se deben definir, así como las conexiones entre las mismas. Estos tipos de diagramas describen las especificaciones de las clases de software y de las interfaces.<sup>25</sup>

Finalmente y a continuación, es momento de describir la técnica propuesta que auxilie en la creación de un documento flexible, que describa las aplicaciones de sistemas desarrolladas con programación orientada a objetos.

A partir de las herramientas analizadas a lo largo de lo que va del Capítulo II y organizando todas ellas, con base a un formato adecuado se logra el objetivo buscado. (Véase Página No. 50)

\* “Debido a la especialización de los Diagramas de diseño de Clases, se sugiere no entrar a detalle y recomendarlos solo para las personas que quieran conocer mas a fondo del lenguaje unificado de construcción de modelos. (UML)”

LARMAN, Craig *Op. Cit.* p. 15.

FOWLER, Martin. *Op. Cit.*

<sup>24</sup> LARMAN, Craig *Op. Cit.* p 167-18

<sup>25</sup> *Ibid.* p 255-270

## 2.3 Técnica de documentación propuesta

Durante el desarrollo del Capítulo II, se han descrito las técnicas de documentación existentes y que servirán de base para el desarrollo de la nueva técnica que se propone. Es tiempo de empezar a describir la nueva técnica de documentación que acompañara a las aplicaciones generadas con programación orientada a objetos, mediante un documento flexible y apropiado para las mismas.

El método de información Mapping, contribuye con su esencia de administración de negocios, para ello se toma en cuenta lo siguiente:

- La fragmentación de los documentos.
- La relevancia de los documentos.
- El etiquetado de los documentos y
- La jerarquía de fragmentos y etiquetas.

Ahora bien, el análisis estructurado moderno, que auxilia en el análisis, diseño e implantación para modelar aplicaciones de sistemas, aporta las siguientes herramientas:

- Diagramas de flujo de datos.
- Diccionario de datos y
- Diagramas de entidad-relación.

El lenguaje unificado de construcción de modelos, también contribuye con algunas herramientas para el desarrollo de la técnica que se propone.

Gracias al modelado de aplicaciones con programación orientada a objetos y de acuerdo a las necesidades que presentan las mismas aplicaciones de sistemas, es conveniente tomar en cuenta los siguientes elementos:

- Descripción de los casos de uso y
- El Modelo conceptual.

Las herramientas descritas son indispensables para desarrollar el documento propuesto; sin embargo, es necesario conjuntarlas adecuadamente y para ello se toman en cuenta cuatro aspectos relevantes:

- Organización
- Estructuras
- Diagramas y
- Formatos

Para entender la forma de operar de estos cuatro ingredientes agregados a la técnica de documentación propuesta se explica lo siguiente:

### 2.3.1 Organización

Al organizar eficazmente el documento propuesto, primeramente es indispensable describir los casos de uso, usando etiquetas y de acuerdo a la relevancia de la información; por supuesto, sin descuidar la fragmentación de la misma.

Se hace uso primeramente de un elemento del lenguaje unificado de construcción de modelos (casos de uso) y se complementa con elementos del método de información Mapping, (fragmentación, relevancia, etiquetado y jerarquía de la información) de tal forma que las aplicaciones se especifican textualmente y ordenadamente.

No olvidar que la información se escribirá en orden de importancia, de acuerdo a la jerarquía que muestra la misma dentro de la aplicación de sistemas que se desea describir.

Los elementos empleados hasta este momento son los siguientes y en ese orden:

1. Casos de Uso
2. La fragmentación de los documentos.
3. La relevancia de los documentos.
4. El etiquetado de los documentos y
5. La jerarquía de fragmentos y etiquetas

Precisamente, este orden y los elementos enunciados, proporcionan el principio de un documento flexible y que describe, (textualmente) las aplicaciones de sistemas orientados a objetos en forma precisa.

### 2.3.2 Estructuras

Las estructuras que son suficientes para la construcción de un documento de sistemas ágil, las proporciona el lenguaje unificado de construcción de modelos y el análisis estructurado moderno. (Modelo conceptual y diccionario de datos, respectivamente)

Para tener estructuras simples y acordes con aplicaciones de sistemas eficientes, se emplean las siguientes herramientas:

1. El Modelo conceptual
2. Diccionario de datos

El modelo conceptual describe todos los objetos que utiliza el sistema y de acuerdo a la relevancia para el mismo. Ahora bien, el diccionario de datos solo describirá a un nivel superior, cada uno de los objetos.

Es necesario recomendar el uso del diccionario de datos, pero sin profundizar demasiado en las descripciones de los sistemas y solo tanto como sea necesario. Para la creación de un documento suficiente y acorde con aplicaciones de sistemas contemporáneas; es suficiente con modelar la información a un primer nivel únicamente y no se caerá en redundancias inoperantes, evitando manejar información sin relevancia.

### 2.3.3 Diagramas

Para el desarrollo de la técnica, es necesario utilizar los diagramas de flujo de datos y los diagramas de entidad-relación. (Herramientas del análisis estructurado moderno)

Los diagramas de flujo de datos describen gráficamente los procesos de las aplicaciones; sin embargo, es importante darle un giro a la definición de los mismos; ya que, como se sabe, para el análisis estructurado moderno estos diagramas describen las funciones dentro de sus procesos; ahora también tendrán la utilidad para asignar *responsabilidades* ó conceptos que se detallan en el sistema de aplicación.<sup>26</sup>

La definición de descripción de conceptos, también se puede aplicar a los diagramas de flujo de datos, depende mucho de la relevancia del mismo para el sistema. Por su parte, los diagramas de entidad-relación, tienen la representación más abstracta de las aplicaciones de sistemas y al modelar objetos con la misma filosofía que lo opera el ámbito orientado a objetos, repercute enormemente en el modelado ideal para una aplicación orientada a objetos.<sup>27</sup>

El uso del modelado mediante diagramas de flujo de datos y de entidad-relación, ofrece un documento flexible para describir programas orientados a objetos. Es indistinto el orden de aplicación de las herramientas; sin embargo, es necesario utilizar al menos una de ellas.

### 2.3.4 Formatos

Los formatos que se proponen permiten describir la situación actual del sistema ó la situación estable del mismo y también fungirán como bitácora de movimientos, en dado caso de que se encuentre en desarrollo la aplicación. La característica principal del formato propuesto, es la flexibilidad del mismo y debe contener los siguientes elementos:

- Fecha
- Momento del sistema. (Status)
- Texto
- Estructuras
- Modelos

<sup>26</sup> YOURDON, Edward. *op. cit.* p. 211-226

<sup>27</sup> *Ibid.*, p 260-28

El formato debe ser uniforme y ser homologado dentro de la organización.  
(Véase Figura No.14)

| Formato para la técnica de documentación |  |
|--|--|
| <b>Fecha</b>                             | i) Registro del período o fecha de la aplicación                           |
| <b>Status</b>                            | i) Momento en el que se encuentra la aplicación                            |
| <b>Textos</b>                            | i) Casos de Uso<br>ii) Fragmentación, relevancia, etiquetado, jerarquizado |
| <b>Estructuras</b>                       | i) Modelo Conceptual<br>ii) Diccionario de Datos                           |
| <b>Modelos</b>                           | i) Diagramas de Flujo de Datos<br>ii) Diagramas de entidad-relación        |

Figura No. 14

Para lograr la aplicación de la técnica de documentación propuesta, es importante no olvidar cada uno de los elementos mencionados a lo largo del Capítulo II. No sobra señalar, que ante todo, la apertura que tiene la técnica de documentación propuesta, es el elemento principal para lograr una herramienta útil para la construcción de un documento flexible que acompañe las aplicaciones orientadas a objetos, en un mundo actual, cambiante y global.

## Conclusiones del Capítulo II

Existen diversas técnicas de documentación que auxilian a la administración de negocios y para la descripción y modelado de soluciones de sistemas; sin embargo, para contar con una técnica de documentación completa, es necesario darle apertura a nuevas ideas e integrar las mismas. Situación idónea para la creación de una herramienta que apoye a la descripción de aplicaciones creadas con programación orientada a objetos.<sup>28</sup>

La técnica de documentación propuesta en esta investigación, cumple con las exigencias del mundo contemporáneo y se presenta como una opción flexible y ágil, para apoyar a la solución de problemáticas particulares de cualquier organización y sobre todo, soluciones oportunas.

El resultado de la técnica de documentación propuesta, contribuye a la “administración de calidad”;<sup>29</sup> mediante la capacitación de cualquier usuario en las aplicaciones de sistemas y especialmente en la estandarización de los procedimientos de las empresas, razón de ser de la presente investigación. En general, cualquier usuario de las aplicaciones de sistemas, al apoyarse con documentos como el propuesto contribuirán, dentro de su organización, con un mejor desempeño de sus funciones y sobre todo, realizándolas con calidad.

Por último y como parte fundamental de la técnica de documentación propuesta para la creación de documentación flexible y que sea acorde a las necesidades del mundo actual, es necesario ubicarla y señalar el momento de acción de la misma. En los desarrollos orientados a objetos la esencia del *análisis y diseño* consiste en situar el dominio de un problema y su solución lógica dentro de la perspectiva de entidades informáticas y se procura ante todo de identificar y describir éstas entidades, dentro del dominio del problema identificado. El resultado de esto, es la descripción del desarrollo orientado a objetos, mismo que se sitúa en la parte de la *construcción o implementación* del sistema; sin olvidar todas las anteriores etapas y de acuerdo a una “orientación interactiva”,<sup>30</sup> fundamental para una solución de sistemas eficaz y de calidad suficiente para las exigencias de las organizaciones actuales. (Véase Figura No.15)

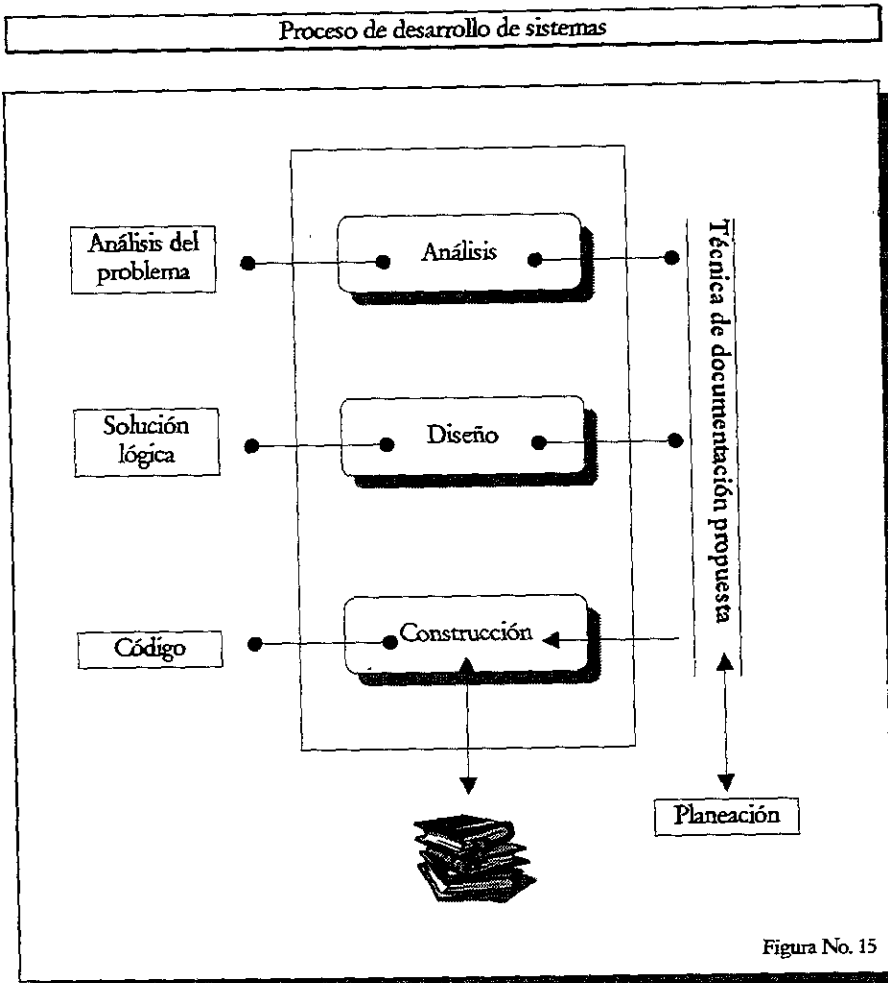
<sup>28</sup> HORN, Rober E *Desarrollo de Procedimientos Políticas y Documentación*. InfoMap®, Lexington MA; 1991.

YOURDON, Edward. *Análisis Estructurado Moderno*. Prentice-Hall Hispanoamericana, S.A., México; 1993.

LARMAN, Craig. *UML y Patrones. Introducción al Análisis y Diseño Orientado a Objetos*. Prentice Hall, México; 1999  
(Véase Página No 25-26)

<sup>29</sup> TORRES, L. Juan. *Seminario Análisis de la Planeación*. U.N.A.M., E.N.E.P.Acatlán, México; julio del 2000.  
(Véase Cuadro No 1 Página No.18)





<sup>30</sup> La solución a la problemática tiene una orientación interactiva. Se considera esta orientación como la más idónea para buscar la solución óptima a la problemática que se presenta y acorde con lo que dice Ackoff: "Existe una cuarta orientación: la interactiva. Esta última orientación que considera al pasado, al presente y al futuro como aspectos diferentes, pero inseparables, de la problemática para la que se planea; se concentra en todas las orientaciones al mismo tiempo. Está basada en la creencia de que si no se toman en cuenta los tres aspectos temporales de una problemática, el desarrollo será obstruido".

ACKOFF, Russell L. Planificación de la Empresa del Futuro. Ed. Limusa, México, p. 73; 1983.

## Fuentes de Consulta

- ACKOFF, Russell L. Planificación de la Empresa del Futuro. Ed. Limusa, México; 1983.
- ACKOFF, Russell L. Un Concepto de Planeación de Empresas. Ed. Limusa, México; 1972.
- BERTALANFFY, Ludwig Von. Tendencias en la Teoría General de Sistemas. Ed. Alianza, Madrid España; 1978.
- DEMARCO, Tom. Structured Analysis and System Specification. Ed. Prentice Hall, Englewood Cliffs N.J.; 1979.
- FOWLER, Martin. UML Gota a Gota: Actualizado para Cubrir la Versión 1.2 OMG UML Standard. Ed. Addison Wesley Longman, México; 1999.
- GANE, Christopher P. Análisis Estructurado de Sistemas. Ed. El Ateneo, México; 1987.
- LARMAN, Craig. UML y Patrones. Introducción al Análisis y Diseño Orientado a Objetos. Ed. Prentice Hall, México; 1999.
- YOURDON, Edward. Análisis Estructurado Moderno. Ed. Prentice Hall, México D.F.; 1993.
- ERIKSSON, Hans-Erik. UML Toolkit. Ed. Wiley, New York E.U.A.; 1998.
- HORN, Rober E. Desarrollo de Procedimientos Políticos y Documentación. Information Mapping®, Lexington MA; junio 30, 1991.
- MORALES, Ana Cecilia. Red TELMEX, Metropolitano. ©Red TELMEX, México; 2000.
- TORRES, L. Juan. Seminario: Análisis de la Planeación. U.N.A.M., E.N.E.P. Acatlán, México; julio del 2000.
- AGULLO, Soliveres Pedro. Java Vs. C++: Un Estudio Comparativo de Java. <http://www.ctv.es/USERS/pagullo/poo.htm>, México; 2000.
- AYALA Mojica, Pedro. Programación Orientada a Objetos (Conceptos Básicos). [http://www.lasalle.edu.co/~payalam/p\\_objeto.html](http://www.lasalle.edu.co/~payalam/p_objeto.html), México; 2000.

## Fuentes de Consulta, Continuación ...

- GORDILLO, Silvia. Desarrollo de Lenguajes de Consultas para Sistemas Hypermediales. <http://www.lfia.info.unlp.edu.ar/spanish>, Argentina; 2000
- Véase: <http://sol.info.unlp.edu.ar/~luciod/objcour>, Argentina; 2000.
- Véase <http://www.ctv.es/USERS/pagullo/poo.htm>. México; 2000.

## Capítulo III

### Explotación de la técnica de documentación propuesta

**OBJETIVO:** El presente capítulo busca la implementación de la técnica de documentación propuesta, para una aplicación orientada a objetos y de este modo adquirir un mejor entendimiento de la misma y así contribuir con una eficaz descripción de las soluciones orientadas a objetos dentro de cualquier organización.







Las aplicaciones orientadas a objetos buscan ser robustas de acuerdo a las exigencias del mundo contemporáneo, motivo por el cual es necesario desarrollarlas de acuerdo a necesidades muy particulares de cualquier organización y enriquecerlas con documentos apropiados, que sean digeribles para cualquier usuario del sistema y con base a las políticas establecidas dentro de las mismas organizaciones.

Para desarrollar un documento flexible que se adecue al ámbito de la POO debe contemplar los siguientes aspectos:

- La organización
- Las estructuras apropiados
- Los diagramas necesarios y
- Los formatos homologados.

Estos aspectos que se señalan son de suma importancia para lograr el objetivo buscado; que es el obtener un documento capaz de soportar los embates del medio ambiente que puede sufrir cualquier aplicación y responder a las exigencias de cualquier organización y en condiciones extremas. Sin olvidar la capacidad de autosuficiencia que puede obtener el usuario con las especificaciones técnicas del documento que se propone.

Es importante no olvidar el grado de flexibilidad que debe tener el documento propuesto para eso se señalarán muchas formas de presentar el mismo y de acuerdo a las necesidades de los usuarios. También es necesario mantener el interés del documento y actualizarlo adecuadamente, motivo por el cual la creatividad puede ser de mucha ayuda para el desarrollo y explotación de la técnica de documentación sugerida.

| CONTENIDO   |   |
|---|---|
|   | Capítulo III                                  |
|   | Solución con programación orientada a objetos |
|   | Descripción de la solución                    |
|   | Documentación propuesta                       |
|   | Evaluación de la solución                     |
|  | Escenarios                                    |

En el instante de adoptar la técnica de documentación propuesta y por consecuencia, obtener un documento que acompañe soluciones orientadas a objetos; es necesario buscar la homologación del mismo dentro de la organización que afecta.

El estandarizar los documentos y homologarlos, es parte esencial del trabajo eficaz y con calidad. "La estandarización de la calidad"<sup>1</sup> se puede lograr con la inclusión de la técnica de documentación propuesta, ya que se obtiene de la misma un documento estándar, flexible y que propicia el trabajo de calidad.

---

<sup>1</sup> "¿Por qué es importante estandarizar la calidad? La calidad de un producto depende de muchas variables, tales como el calibre de los componentes o materiales usados; el tipo de equipamiento usado en el diseño, producción, prueba y embarque; el equipo de calibración y los procedimientos de mantenimiento empleados, el entrenamiento y experiencia del personal de producción y supervisión; así como las condiciones ambientales de producción. Por esta razón, en un mundo global, donde los productos y servicios de una empresa pueden estar en cualquier país, es fundamental mantener un sistema de calidad internacional."

MORALES, Ana Cecilia. Red TELMEX, Metropolitano. © Red TELMEX, México, p. 8; 2000.

Precisamente al trabajar con una técnica de documentación como la que se propone en el presente trabajo de investigación, se logra un entorno de calidad y contribuye a mejorar el desempeño de cualquier usuario dentro de las organizaciones.

### 3.1 Solución con programación orientada a objetos

La solución que se presenta a continuación, esta desarrollada con POO y muestra las ventajas de un ambiente orientada a objetos; por supuesto, acompañada con la técnica de documentación propuesta, para de este modo entender el funcionamiento de la misma solución.

El documento que se propone debe ser flexible; es decir, no muy complejo para que pueda ser leído por la mayoría de los usuarios de la solución. Debe también ser agradable a la vista de cualquier persona, ágil en la presentación del mismo y de fácil acceso para cualquier usuario de la aplicación.

De acuerdo a una solución de necesidades muy particulares partimos del siguiente ejemplo:

**Sistema:** Consulta de los datos de la nómina y estadísticas de las filiales que conforman un grupo comercial.

**Problema:** Se requiere presentar toda la información de los ingresos que perciben mensualmente los empleados que conforman un grupo comercial; además de mostrar la posición que tienen en el mercado, con base a encuestas adquiridas para poder realizar estudios de posición del empleado y del puesto.

Para entender el problema y la solución planteada es necesario contar con toda la información posible para poder describir claramente la solución orientada a objetos tratando de no perder el mínimo de detalle y así lograr una descripción adecuada.

Indudablemente que el personal involucrado dentro del proceso de la solución tendrá siempre una idea más clara de la situación establecida, para el entorno de la problemática que se plantea, en comparación con cualquier individuo ajeno a la misma. Precisamente por este señalamiento, es conveniente conocer el ámbito externo de la aplicación para diseñar un documento completo y que sea suficiente para cualquier necesidad que se pueda presentar dentro de la organización.

Otro aspecto importante a reconocer por parte de las personas que se ubican directamente dentro de cualquier solución propuesta; es que ellas entenderán rápidamente cada uno de los argumentos que se puedan establecer y al utilizar un documento eficaz que describa los procesos fundamentales, tendrán todavía más capacidad de asimilar cualquier situación que se les pueda presentar en sus labores diarias.

Precisamente, entre los diversos aspectos que se deben remarcar con la técnica de documentación sugerida, es la adaptabilidad que obtiene el lector del documento desarrollado con la técnica de documentación propuesta; que precisamente contribuye a un fácil entendimiento de cualquier proceso dentro de la solución y es uno de los aspectos relevantes a remarcar y uno de los puntos que fortalecen la técnica de estudio sugerida. Esto quiere decir, que cualquier persona que establece contacto con el documento propuesto, logrará entender los aspectos más relevantes de la solución orientada a objetos.

El documento a desarrollar, con la técnica de documentación propuesta, se puede empezar a escribir a partir de una aplicación que esta en producción o plantearlo conjuntamente con el análisis y el diseño orientado a objetos; sin embargo, aunque no se establece un momento para empezar a describirlo, precisamente por la flexibilidad que se supone es característica de la técnica de documentación; sí es recomendable planear con tiempo el desarrollo del mismo y no cuando sea demasiado tarde y se encuentre la solución desbordada por los continuos cambios que requiere en momentos críticos.

Lo ideal para la elaboración del documento, es planear el desarrollo y terminación del documento con tiempo; para así, no caer en el error de pensar en el después de que sea necesario. Hay que adelantarse a los hechos y programarlo en el momento oportuno, para que este exista antes de que se le necesite preferentemente.

La técnica de documentación también es realmente simple, lo que ocasionará que la elaboración del documento sea de manera inmediata; si se compara con el desarrollo completo de toda la solución. Esto significa, que la propuesta es sencilla para la elaboración del documento sugerido y fácil para la aplicación y entendimiento del mismo.

En medio de tanto tecnicismo y complejidad de algunas soluciones orientadas a objetos; lo común, sencillo y práctico, además de universal siempre será de lo más valiosos para cualquier desarrollo. En efecto la técnica de documentación propuesta esta llena de estas virtudes y se consolida oportunamente, dentro del cualquier medio en el que se aplique. No hay que menospreciar lo sencillo e universal, ya que precisamente estas características sugieren aplicaciones orientadas a objetos eficaces y desempeño adecuado de profesionistas.

### 3.1.1 Descripción de la solución

El *proyecto* de la aplicación se realiza con Visual FoxPro™ versión 5.0 de Microsoft®, esta herramienta presenta un entorno orientado a objetos muy versátil y con la confianza de utilizar *tablas* compatibles con muchas otras herramientas de aplicación.

La elección del lenguaje de programación orientado a objetos, se realiza con base a lo siguiente:

#### Familiaridad

Con el uso de un lenguaje. Esto no implica que aprender el nuevo lenguaje de objetos sea difícil. A veces puede cegar al programador el hecho de que los lenguajes den giros fundamentalmente de diseño para adaptarse a la programación orientada a objetos; sin embargo, al tener claros conceptos de diseño de aplicaciones se puede programar en el lenguaje que se prefiera.

#### Plataformas

Según el hardware en que se ejecute la aplicación, puede que varíe la disponibilidad de lenguajes.

#### Otros

Entre otros aspectos; es importante al seleccionar una herramienta de programación: el tiempo de programación en relación con el de ejecución, el acceso a código existente para su reutilización, los entornos de programación existentes, las características de entrega, si es lenguaje interprete se necesita solo el *run-time*, mientras que los compiladores generan un archivo ejecutable.

El Visual FoxPro™ versión 5.0, favorece totalmente al desarrollo de la solución de sistemas propuesta; aunque se deja a elección del lector, que seleccione la herramienta de programación que se adecue a sus necesidades muy particulares.<sup>2</sup>

---

<sup>2</sup> “Hay lenguajes orientados a objetos que implementan los conceptos de la programación orientada a objetos de diversas formas. Existen lenguajes de programación orientados a objetos denominados: Puros: Todo en su ambiente de objetos; Híbridos: Son en general los lenguajes estructurados con un acoplamiento de programación orientada a objetos. Entre otros tenemos a Smalltalk®, Eiffel®, C++™, VisualBasic™, VisualFoxPro™ Java®, etcétera. Cada uno de ellos con sus características particulares.”

Véase: <http://sol.info.unip.edu.ar/~luciod/objicour>. Argentina; 2000.



De acuerdo al desarrollo de la aplicación de sistemas; la solución queda establecida en el siguiente proyecto:

### **Sistema**

Consulta de los datos de la nómina y estadísticas de las filiales que conforman un grupo comercial.

### **Nombre**

Sistema de consultas y estadísticas.

### **Problema**

Se requiere presentar toda la información de los ingresos que perciben mensualmente los empleados que conforman un grupo comercial; además de mostrar la posición que tienen en el mercado, con base a encuestas adquiridas para poder realizar estudios de posición del empleado y del puesto.

### **Generalidades**

La aplicación que se describe a continuación esta realizada a la medida de un grupo de empresas comerciales, de acuerdo a sus necesidades particulares y se retoma únicamente para ejemplificar la técnica de documentación propuesta y distinguir los beneficios que proporciona el contar con una herramienta de tal naturaleza.

La organización del "proyecto"<sup>3</sup> muestra que se tienen cuatro tablas, con siete *formas*, un *reporte*, una clase, dos programas, un *menú* y cinco *gráficos*. (Véase Cuadro No.4)

El proyecto pretende mostrar al usuario de la solución orientada a objetos, la situación actual de los empleados dentro del grupo determinado y también estadísticas de mercado; para de este modo, poder ubicar adecuadamente los empleados dentro del grupo.

---

<sup>3</sup> "Un proyecto dentro de la programación orientada a objetos consta de componentes independientes que se almacenan como archivos individuales.."

MICROSOFT®. *Visual FoxPro™. Manual del Programador.*  
Microsoft®, E.U.A., p. 407-413; 1996.

## Descripción del proyecto I \*

| Proyecto I |  |
|------------|--|
| Fecha      | 15 de enero del 2001   |
| Status     | En producción  |
| Datos      | CLAVE<br>ENC_2000<br>PROYEC00<br>PROYEC99                              |
| Documentos | CLAVE<br>BUS_INI<br>FORMA_1<br>FORMA_11<br>FORMA_12<br>RESUL<br>REPTOT |
| Clase      | MISCLAS  |
| Código     | PRINCIPAL<br>CORRIGE   |
| Menú       | MENUPRIN   |
| Otros      | CLOSE.BMP<br>DELETE.BMP<br>FIND.BMP<br>NEW.BMP<br>FOX.BMP              |

Cuadro N o. 4

\* MICROSOFT®. Op. Cit.MICROSOFT®. Microsoft Visual Tools Suite. Microsoft®, E.U.A.; 1993.Véase <http://www.geocities.com/SiliconValley/Lakes/4725>. México;2000.

### 3.1.2 Documentación propuesta

De acuerdo a la técnica de documentación que se propone para poder entender cualquier aplicación orientada a objetos, primeramente es necesario identificar todas las entidades fundamentales *necesarias y ordenarlas de acuerdo a su importancia ó su jerarquía dentro de la misma aplicación*<sup>4</sup>.

Ejemplo: La aplicación orientada a objetos del Proyecto\_I, contiene lo que se menciona a continuación:

|           |   |
|-----------|---|
| Proyecto: | Proyecto_I  |
| Tablas    | Clave<br>Enc_2000<br>Proyec00<br>Proyec99           |
| Formas    | Bus_Ini<br>Forma_1<br>Forma_11<br>Forma_12<br>Resul |
| Reporte   | RepTot  |
| Programas | Principal<br>Corrige                                |
| Menú      | MenuPrin  |
| Otros     | Close.bmp<br>Delete.bmp<br>Find.bmp<br>New.bmp      |

De este modo, se tiene perfectamente identificados los principales argumentos de la aplicación y de acuerdo a lo que se recomienda.

<sup>4</sup> HORN, Rober E. Op. Cit.

La organización de la solución se realizó primeramente tomando como base lo siguiente:

- Fragmentación.
- Relevancia.
- Etiquetado y
- Jerarquía de fragmentos y etiquetas.

Ahora bien de acuerdo al siguiente código del programa principal:

```
*****
** Programa: Principal.prg **
*****
Set Talk Off
Set Century On
Set Date To British
Set Confirm On
Set Escape Off
Set Echo Off
Set Status Off
Set System Menu Off
On Error Do Corrige With tipoerror
Close All
Clear All
Public oApp, Ejecutable, cUser, Exped, fil, fil1, aa, mm, dd, tipo, c, tipoerror
Public dato, baja, mesOk, mtemp, opc
Store space(1) To fil, fil1, dato, baja, mtemp
Store 'C' To tipo
Store 0 To opc, tipoerror, exped
Store .F. To mesOk, ejecutable
Store space(1) To cUser
aa=Substr(Dtoc(Date()),7,4)
mm=Substr(Dtoc(Date()),4,2)
dd=Substr(Dtoc(Date()),1,2)
f_act=dd+mm+aa
Set Classlib To misclas
oApp=CreateObject("ENTORNO")
oApp.Setup
```

```

Do Memuprin.mpr
oApp.claveacceso
If !Empty(cUser)
    Read Events

Endif
If !Ejecutable
    OAPP.Cleanup
Endif
Close All
Release All Extended
Clear All
Set Talk On
Set Century Off
*****
***          Fin          ***
*****

```

El programa principal muestra entre otras cosas que existe una variable que contiene la fecha actual; llamada "f\_act", que esta compuesta de "dd+mm+aa" y de acuerdo a lo siguiente:

```

F_act = ** Fecha actual establecido por el sistema operativo **
aa = ** Año actual establecido por el sistema operativo **
mm = ** Mes actual establecido por el sistema operativo **
dd = ** Día actual establecido por el sistema operativo **
F_act = aa+mm+dd
aa = {Carácter}
mm = {Carácter}
dd = {Carácter}
Carácter = {0-9}

```

El diccionario de datos muestra claramente las características de las variables que ocupamos en el programa y de este modo sabemos como identificarlas y cual es la finalidad de las mismas dentro de la solución <sup>5</sup>.

<sup>5</sup> YOURDON, Edward *Op. Cit.* p. 212.

GANE, Christopher P. *Op. Cit.*

DEMARCO, Tom. *Op. Cit.*

Otro momento importante para documentar la aplicación lo sugiere el diagrama de flujo de datos.

De acuerdo al siguiente código del menú, que es otro de los componentes de la solución, se establece el código que a continuación se muestra:

```

*****
** Menú: MenuPrin.mpr **
*****
SET SYSMENU TO
SET SYSMENU AUTOMATIC
DEFINE PAD _04o0mgpsl OF _MSYSMENU PROMPT " Consultas" COLOR SCHEME
3 ;
    KEY ALT +C, " " ;
    MESSAGE " Consulta de las nóminas de las filiales"
DEFINE PAD _04o0mgpt4 OF _MSYSMENU PROMPT " Consolidados" COLOR
SCHEME 3 ;    KEY ALT +C, " " ;
    MESSAGE " Consolidados de las filiales"
DEFINE PAD _04o0mgpta OF _MSYSMENU PROMPT " Reportes" COLOR SCHEME
3 ;
    KEY ALT +R, " " ;
    MESSAGE " Informes"
DEFINE PAD _04o0mgpte OF _MSYSMENU PROMPT " Utilidades" COLOR
SCHEME 3 ;
    KEY CTRL+U, " CTRL+U" ;
    MESSAGE " Utilidades de la aplicación"
DEFINE PAD _04o0mgptj OF _MSYSMENU PROMPT " Salir" COLOR SCHEME 3 ;
    KEY CTRL+S, " CTRL+S" ;
    SKIP FOR IEMPTY(WONTOP());
    MESSAGE " Abandonar la aplicación"
ON PAD _04o0mgpsl OF _MSYSMENU ACTIVATE POPUP consultas
ON PAD _04o0mgpt4 OF _MSYSMENU ACTIVATE POPUP consolidad
ON PAD _04o0mgpta OF _MSYSMENU ACTIVATE POPUP reportes
ON PAD _04o0mgpte OF _MSYSMENU ACTIVATE POPUP utilidades
ON PAD _04o0mgptj OF _MSYSMENU ACTIVATE POPUP salir
DEFINE POPUP consultas MARGIN RELATIVE SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF consultas PROMPT " Consultas de las filiales" ;
    KEY CTRL+C, " CTRL+C" ;
    MESSAGE " Consulta de las nóminas de las filiales"
DEFINE BAR 2 OF consultas PROMPT " Consulta de encuesta 1999" ;
    KEY CTRL+E, " CTRL+E" ;
    MESSAGE " Consulta de la encuesta de compensaciones 1999"
ON SELECTION BAR 1 OF consultas ;
    DO _04o0mgput ;

```

```

    IN LOCFILE("\FILIALES\MENUPRIN" ,"MPX;MPR|FXP;PRG" ,"DÓNDE
está MENUPRIN?")
ON SELECTION BAR 2 OF consultas ;
    DO _04o0mgpv6 ;
    IN LOCFILE("\FILIALES\MENUPRIN" ,"MPX;MPR|FXP;PRG" ,"DÓNDE
está MENUPRIN?")
DEFINE POPUP consolidad MARGIN RELATIVE SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF consolidad PROMPT "Consolidados de las filiales" ;
    KEY CTRL+F, "CTRL+F" ;
    MESSAGE "Consolidados de las filiales"
ON SELECTION BAR 1 OF consolidad ;
    DO _04o0mgpvu ;
    IN LOCFILE("\FILIALES\MENUPRIN" ,"MPX;MPR|FXP;PRG" ,"DÓNDE
está MENUPRIN?")
DEFINE POPUP reportes MARGIN RELATIVE SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF reportes PROMPT "Impresión de reportes" ;
    KEY CTRL+R, "CTRL+R" ;
    MESSAGE "Informes"
ON SELECTION BAR 1 OF reportes ;
    DO _04o0mgpwf ;
    IN LOCFILE("\FILIALES\MENUPRIN" ,"MPX;MPR|FXP;PRG" ,"DÓNDE
está MENUPRIN?")
DEFINE POPUP utilidades MARGIN RELATIVE SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF utilidades PROMPT "Configurar impresora" ;
    MESSAGE "Configuración de la impresora entorno Windows"
DEFINE BAR 2 OF utilidades PROMPT "\-"
DEFINE BAR MST_CALCU OF utilidades PROMPT "Calculadora" ;
    KEY CTRL+A, "CTRL+A" ;
    MESSAGE "Calculadora"
DEFINE BAR 4 OF utilidades PROMPT "\-"
DEFINE BAR 5 OF utilidades PROMPT "Reindexar archivos" ;
    MESSAGE "Reindexación de archivos de la aplicación"
DEFINE BAR 6 OF utilidades PROMPT "\-"
DEFINE BAR 7 OF utilidades PROMPT "Clave de acceso" ;
    MESSAGE "Cambiar la clave de acceso"
ON SELECTION BAR 1 OF utilidades =SYS(1037)
DEFINE POPUP salir MARGIN RELATIVE SHADOW COLOR SCHEME 4
DEFINE BAR 1 OF salir PROMPT "Salir de la aplicación" ;
    KEY CTRL+S, "CTRL+S" ;
    SKIP FOR !EMPTY(WONTOP());
    MESSAGE "Abandonar la aplicación"
ON SELECTION BAR 1 OF salir CLEAR EVENTS

PROCEDURE _04o0mgput
Do Form Forma_1

PROCEDURE _04o0mgpv6
Do Form Bus_inj

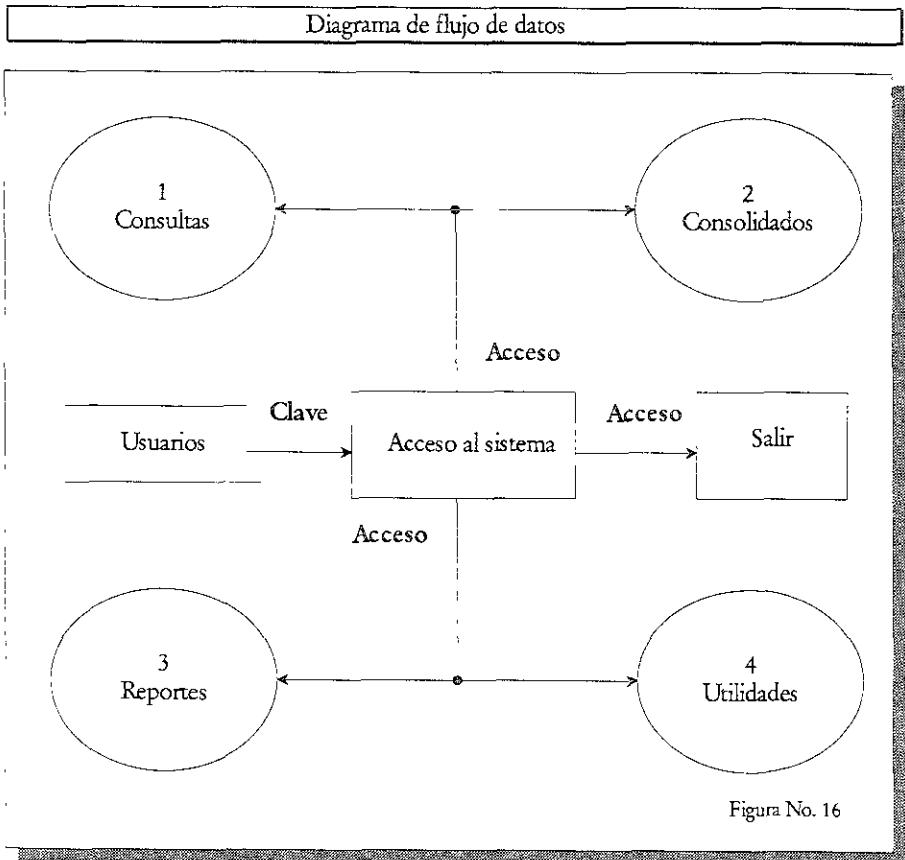
```

```
PROCEDURE_04o0mgpvu
*Do Form Buscar With "Sueldos"
```

```
PROCEDURE_04o0mgpwf
*Do Form Buscar With "datos"
```

```
*****
***      Fin      ***
*****
```

El código se describe en el siguiente diagrama de flujo de datos. (Véase Figura No. 16)





Es obvio darse cuenta de que siempre es más difícil entender el procedimiento mediante el código y para ello es necesario tener al menos conocimientos básicos de la herramienta en la cual se ha realizado la solución; que analizar el funcionamiento de la misma a través del diagrama de flujo de datos, el cual proporciona una idea general del funcionamiento de la solución <sup>6</sup>.

De aquí la importancia de proporcionar el diagrama de flujo de datos o el diagrama de entidad relación para poder percibir en forma general y a un nivel superior el funcionamiento de la solución y tener una idea general de la aplicación.

### 3.1.3 Evaluación de la solución

Como se ha venido mencionando a lo largo del presente trabajo de investigación y como parte medular de una solución orientada a objetos, lo que se busca siempre es obtener una solución integral, acorde con las necesidades del mundo contemporáneo; por supuesto, la técnica de documentación propuesta que genera un documento para la aplicación orientada a objetos y que se ubica dentro de la programación orientada a objetos, cumple este requerimiento de adaptabilidad a las exigencias del mundo actual.

La programación orientada a objetos siempre tiene la necesidad de expresar soluciones completas para las necesidades de cualquier organización, esto de entrada cubre la necesidad de una aplicación justa a la medida dentro de cualquier organización.

Al complementar las aplicaciones orientadas a objetos, con un documento flexible que describe sus procedimientos e independientemente de la documentación propia que contenga la solución y que forme parte del análisis y diseño orientado a objetos; se tendrá una propuesta completa para cualquier problema de funcionamiento de la aplicación que pueda surgir, en un mundo también inestable.

De acuerdo a la técnica de documentación que se propone para poder entender cualquier aplicación orientada a objetos, primeramente es necesario identificar todas las entidades fundamentales necesarias y ordenarlas de acuerdo a su importancia ó su jerarquía dentro de la misma aplicación.

---

<sup>6</sup> YOURDON, Edward. *Op. Cit.*, p. 158.  
GANE, Christopher P. *Op. Cit.*

De acuerdo a la técnica de documentación que se propone para poder entender cualquier aplicación orientada a objetos, primeramente es necesario identificar todas las entidades fundamentales necesarias y ordenarlas de acuerdo a su importancia ó su jerarquía dentro de la misma aplicación.

El desempeño adecuado de cualquier solución y en especial de las orientadas a objetos depende indiscutiblemente de que se haga acompañar de un documento que contenga los siguientes cuatro aspectos:

- Organización
- Estructuras
- Diagramas
- Formatos

Aspectos que se cubren con las distintas herramientas que proporcionan el método de information Mapping, el análisis estructurado moderno y el lenguaje unificado para construcción de modelos; así como los formatos y el énfasis de adaptabilidad que se propone. Organizando adecuadamente estas herramientas propician un desempeño de la solución inmejorable y adecuada a cualquier ritmo de trabajo al que sea expuesta.

La técnica de documentación propuesta se resume como un estándar de calidad, hecho a la medida de la organización y contribuye con un 20% de ahorro aproximadamente, en gastos de operación dentro del área de sistemas de cualquier organización.

El usuario especializado y el usuario terminal de las aplicaciones orientadas a objetos señalan que su trabajo es eficaz y el desempeño es adecuado cuando consultan un documento flexible para ubicar la información clave en el momento deseado, lo cual les resuelve entre el 80% y 100% de los problemas cotidianos dentro de sus áreas respectivas y en la fase de producción de los sistemas de aplicación orientados a objetos. ( Véase Figura No.15, Página No. 56)

### 3.2 Escenarios

Kees Van Der Heijden afirma: "El propósito definitivo del planificador de escenarios es crear una organización más adaptable que reconozca el cambio y la incertidumbre, y que los utilice para su propia ventaja. De manera tradicional, éste ha sido el tema de una disciplina conocida como dirección estratégica".<sup>7</sup> De acuerdo a lo que afirma Kees es importante ubicar los distintos escenarios en los que se desenvuelve el objeto de estudio y la interacción que tiene la técnica de documentación propuesta dentro de los periodos de turbulencia. De esta manera también se contribuye a la dirección estratégica dentro de cualquier organización.

La justificación, que precisamente fortalece el uso del documento creado con la técnica de documentación propuesta, se refleja inmediatamente en los periodos de turbulencias de la organización y su desempeño ofrece ventajas para los integrantes de la solución, de acuerdo a su posición dentro del mismo proceso; por ello es importante ver el comportamiento de la documentación creada dentro de los distintos escenarios.

Para empezar a planear con escenarios primeramente se establecen las variables que intervienen dentro de los distintos escenarios y se nombran las dos variables principales:

#### Variables:

- Cambios (Variable principal)
- Tiempo (Variable principal)
- Presupuesto (Variable dependiente)
- Solución (Variable dependiente)
- Tecnología (Variable dependiente)
- Documentación (Variable dependiente)

Las variables principales como los cambios que se requieran realizar con base a cierto tiempo; son variables no controladas que se tratan de controlar de acuerdo al documento que se propone, sin olvidar que existe la parte siempre inevitable de la incertidumbre y que al buscar controlarla esta disminuye. No se sabe nunca el verdadero futuro, solo se controla lo humanamente posible y se prevé.

<sup>7</sup> VAN DER HEIJDEN, Kees. *Escenarios, El Arte de Prevenir el Futuro*. Ed. Panorama, México, p. 27; 1998.

Los cambios que se presentan para cualquier solución orientada a objetos y el tiempo, son las variables principales e independientes y de ellas dependen las demás variables.

El presupuesto que se otorga a un área o departamento de sistemas, la solución de sistemas, la tecnología que se requiere para actuar en los distintos períodos y la documentación propuesta; se desempeñan al compás de lo que marquen las variables principales (cambios y tiempo) y generan una calificación dependiendo del instante exacto en que se ubiquen.

Las calificaciones se asignan acorde con distintos adjetivos y quedan definidas en el siguiente tabulador: (Véase Figura No. 17)

| Tabulador de adjetivos |              |
|------------------------|--------------|
| Descripción            | Calificación |
| Ideal                  | 100 %        |
| Adecuado               | 80 %         |
| Suficiente             | 60 %         |
| Justo                  | 40 %         |
| Insuficiente           | 20 %         |

Figura No. 17

Una vez quedando establecidas las distintas variables que interactúan en los diferentes escenarios, las variables primordiales de los mismos y los valores que pueden obtenerse; es tiempo de describir las condiciones que propician las situaciones posibles y también asignar probabilidades de ocurrencias; por supuesto, con base al desempeño de las variables principales.

Para lograr abarcar los distintos comportamientos de las variables que se definieron anteriormente es necesario utilizar **cuatro escenarios**. Los distintos escenarios se presentan dependiendo de las siguientes condiciones:

### Condiciones :

- Primer escenario llamado “2001 odisea del espacio”, en esta situación las condiciones son de inestabilidad extrema tanto dentro del entorno de la organización como en todo el entorno social pero sujeto a tiempos razonables para actuar; lo que contribuye a compensar fuerzas y actuar únicamente en el momento oportuno.
- Segundo escenario, también llamado “Tornado”. Situación con condiciones de inestabilidad extrema tanto dentro del entorno de la organización como en todo el entorno social y sujeto a tiempos limitados para actuar; lo que implica buscar alternativas ágiles y flexibles para lograr obtener resultados satisfactorios y en el momento oportuno. Son tiempos difíciles en los que el trabajo de calidad contribuye enormemente a salir de la problemática que se pueda presentar.
- “Rocky”, tercer escenario. Hay calma relativa dentro del entorno de la organización y tiempos limitados para actuar; sin embargo, la calma aparente permite lograr los objetivos a tiempo con base a herramientas adecuadas de trabajo.
- Cuarto y último escenario: Situación de calma aparente dentro del entorno de la organización y tiempo razonable para actuar lo que genera un total control sobre cualquier problemática que se presente. Tiempo de control y estabilidad; escenario que recibe el nombre de “Superman”.

Las probabilidades de ocurrencia quedan establecidas de acuerdo al comportamiento de las dos principales variables definidas anteriormente (cambios y tiempo) y generan las calificaciones muy particulares para las demás variables dependientes y que corresponden a los distintos escenarios descritos. De esta manera la probabilidad de ocurrencia queda definida como sigue:

### Ocurrencias :

- Primer escenario: De acuerdo a las calificaciones obtenidas, la probabilidad de establecerse esta situación es del 24 %.
- Segundo escenario: De acuerdo a las calificaciones obtenidas, la probabilidad de establecerse esta situación es del 18 %.
- Tercer escenario: De acuerdo a las calificaciones obtenidas, la probabilidad de establecerse esta situación es del 30 %.
- Cuarto escenario: De acuerdo a las calificaciones obtenidas, la probabilidad de establecerse esta situación es del 28 %.

Véase la Figura No. 18: Diagrama de escenario en diferentes períodos y con frecuencia de cambios también distinta:

Diagrama de escenarios\*

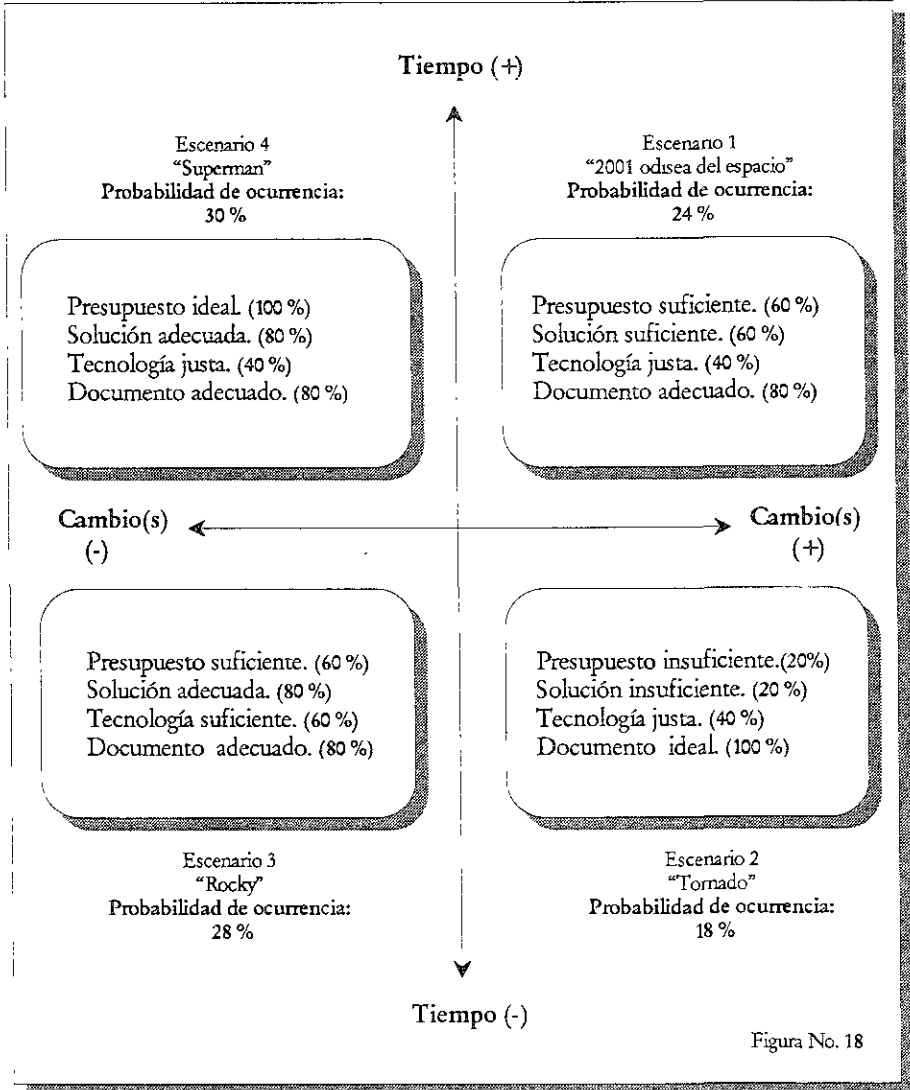


Figura No. 18

\* VANDER HEIJDEN, Kees. *Escenarios, El Arte de Prevenir el Futuro*. Ed. Panorama, México; 1998.

Para identificar los distintos escenarios se gira sobre el plano, como lo hacen las manecillas del reloj y se ubican los cuadrantes; es decir, el escenario de situación de más cambios en el menor tiempo, es el segundo. Por lo tanto, el cuarto escenario es el último cuadrante y así respectivamente.

La figura No. 18, plantea las situaciones en las cuales a partir de una solución orientada a objetos, se observan distintos escenarios (cuatro para ser exactos); escenarios en los que la documentación ideal se presenta en el peor momento que se plantea; (segundo escenario) lo que le da fortaleza a la aplicación de la técnica de documentación propuesta.

Kees también concluye que: “Por lo tanto, la planeación de escenarios es una herramienta natural de pensamiento para su uso en una conversión estratégica. Así mejora la aptitud a la organización en dos niveles:

- En el largo plazo, mediante el desarrollo de un sistema de organización más robusto, mejor capacitado para soportar las sacudidas inesperadas que enfrentará.
- En un término más corto, una creciente adaptabilidad por medio de una observación más hábil del entorno empresarial.”<sup>8</sup>

La necesidad de apertura en todas las instancias para la concepción de la planeación estratégica mediante la participación de todos los integrantes de la organización con base a sus necesidades, expectativas y experiencias, son factores importantes dentro de la planeación de escenarios; es decir, es necesario la cooperación del equipo para concebir una planeación de escenarios integral en un mundo de cambios continuos.

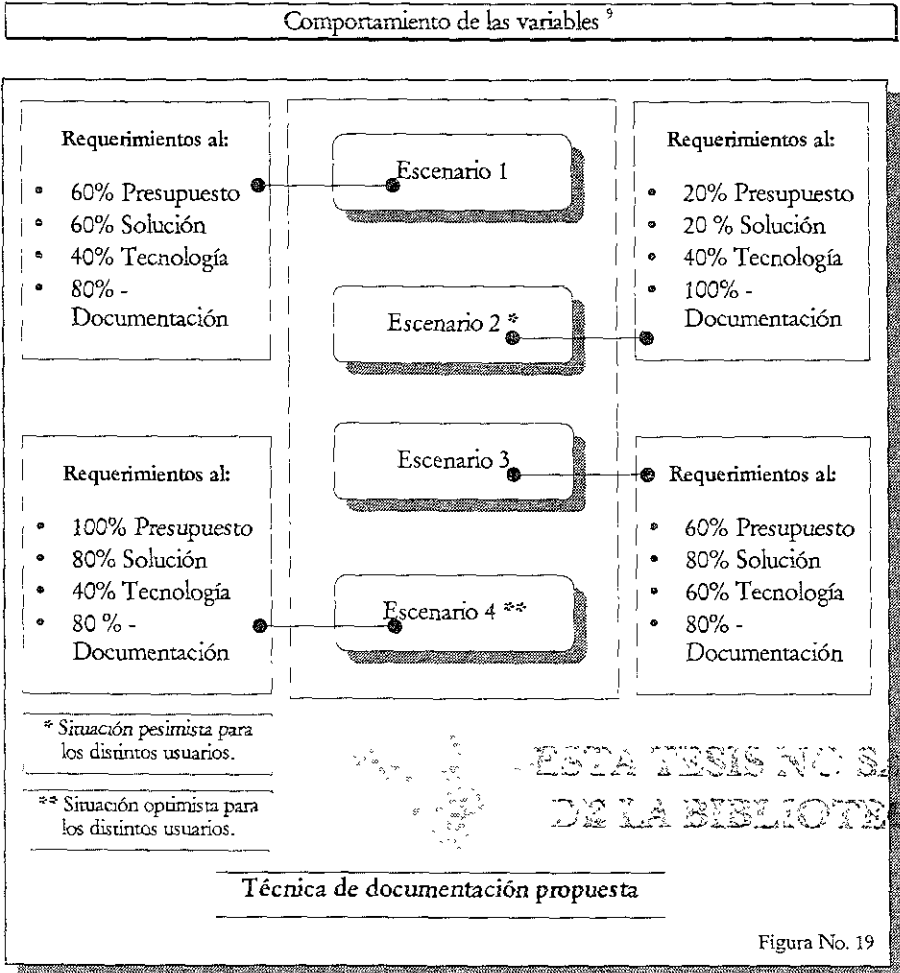
La técnica de documentación propuesta crea un documento que se integra completamente a la apertura que requiere la planeación estratégica y para ser exactos en su complemento llamado planeación de escenarios; con base a su énfasis de adaptabilidad para cualquier situación que se presenta y su apertura a todos los integrantes del proceso para lograr una herramienta de aplicación integral.

Las variables que afectan el ámbito de estudio se muestran con un presupuesto de **insuficiente** a **ideal** en épocas inestables, pero en tiempos estables se reflejan como **suficiente** y sin desperdiciar recursos; lo que implica el tener la oportunidad de manejar un presupuesto sobrado del 20%, con respecto al presupuesto original. (Véase Página No. 15) Se mantiene un índice de maniobra flexible para los tiempos de austeridad y se demuestra que en tiempos difíciles, el factor humano ante todo, responderá con calidad y oportunidad, gracias a la documentación eficaz que fue creada con una herramienta oportuna y que propicia el cumplimiento de las necesidades particulares de cualquier organización.

---

<sup>8</sup> *Ibid.*, p. 55,56.

Es evidente que los usuarios de las aplicaciones generadas con programación orientada a objetos y a todos los niveles se ven afectados directamente con el comportamiento de las variables mencionadas; (Véase Pagina No. 74) sin embargo y dependiendo del desempeño de las mismas variables en los distintos escenarios, existen situaciones favorables para los usuarios y otras no son tanto. Por ello es importante observar el reflejo de las variables de acuerdo a los requerimientos del personal y con base a la técnica de documentación propuesta. (Véase Figura No. 19)



9 La situación optimista queda conformada por el cuarto escenario, mientras que la situación pesimista es el segundo escenario y finalmente la situación realista se basa en los escenarios uno y tres. Todo esto acorde con lo reflejado en la Figura No. 18 y como complemento de la misma.



“Por sí solo, el proceso formal de planeación no puede producir cambios. Se requiere más. Sin importar lo sofisticado que sea el proceso de planeación formal, la mayor parte de la toma de decisiones ocurre por medio de contactos informales en los cuales ocurre la mayoría de la conversación estratégica. Los planificadores de escenarios hábiles tomarán en cuenta esto y pretenderán influir en la acción por medio de estos canales”.<sup>10</sup>

Finalmente y de acuerdo a lo que Kees Van Der Heijden afirma es importante no encartonar los procesos en los cuales se mueven las decisiones, promover siempre el toque fino de la improvisación y de la flexibilidad, para de este modo obtener mejores resultados dentro de las organizaciones. Por supuesto que este señalamiento también da fortaleza a la técnica de documentación propuesta, mediante el documento creado, para resolver incidencias en los desarrollos orientados a objetos y ¿por qué no?, en cualquier tipo de desarrollo de sistemas que se requiera controlar.

---

<sup>10</sup> *Ibid.*, p. 262.

## Conclusiones del Capítulo III

La industria esta absorta con la revolución informática. Para cualquier organización inmersa en un mundo global, responder en el momento oportuno y con calidad, es necesario para ser competitivo en cualquier ámbito. Las soluciones de “plataformas”<sup>11</sup> o “estandarización de soluciones,”<sup>12</sup> son parte indispensable de cualquier mundo empresarial.

La técnica de documentación propuesta contribuye al desarrollo de cualquier solución orientada a objetos al proporcionar documentación necesaria para un desempeño eficaz. La documentación obtenida propicia ayuda tanto para el momento del desarrollo de la misma solución, como en la fase de producción. Todos los participantes se ven favorecidos con la misma ya que establece un trabajo estandarizado y pretender ser una herramienta de plataforma.

La capacitación de todos los usuarios de los sistemas, es parte fundamental para poder desempeñarse adecuadamente cualquier función. La descripción y modelado de todas sus tareas ejercidas, son elementos indispensables para un desempeño de cualquier trabajo con “calidad,”<sup>13</sup> variable indispensable en el mundo actual. Aquí radica la importancia de contar con documentación que describa los procesos de cualquier organización de una manera clara, precisa y con calidad; es decir, eficaz.

La técnica de documentación propuesta contribuye al desarrollo de cualquier organización y plantea escenarios muy prometedores en tiempos de calma y también situaciones de suficiencia y poco ajetreo en tiempos de turbulencias. (Véase Figura No. 18) Precisamente este atributo de la herramienta propuesta, es la justificación principal para ser usada dentro de cualquier *planeación estratégica* de la organización y forma parte indispensable en la fase de producción de un sistema de aplicación y en particular de los generados con programación orientado a objetos.

11 “La creación de un estándar para la fabricación de automóviles, (plataformas iguales en todas las compañías) es la solución a las demandas actuales del mercado de autos. Soluciones integrales se requieren en un mundo tan competitivo. Al establecimientos de estándares para la construcción de autos se le denomina: *Tecnología de Plataforma.*”

Canal 22, México. La industria del Automóvil. T.V. Londinense. Londres, Inglaterra; 2000.

12 MORALES, Ana Cecilia. Red TELMEX, Metropolitano. © Red TELMEX, México, p. 6-8; 2000.

13 TORRES, L. Juan. Seminario. Análisis de la Planeación. UNA.M, E.NE.P.Acatlán, México; julio del 2000 (Véase Cuadro No. 1 en la Página No. 18)

## Fuentes de Consulta

- ACKOFF, Russell L. Recreación de las Corporaciones, Un Diseño Organizacional para el Siglo XXI. Ed. Oxford University Press, México; 2000.
- DEMARCO, Tom. Structured Analysis and System Specification. Ed. Prentice Hall, Englewood Cliffs N.J.; 1979.
- GANE, Christopher P. Análisis Estructurado de Sistemas. Ed. El Ateneo, México; 1987.
- VAN DER HEIJDEN, Kees. Escenarios, El Arte de Prevenir el Futuro. Ed. Panorama, México; 1998.
- YOURDON, Edward. Análisis Estructurado moderno. Ed. Prentice Hall, México D.F.; 1993.
- HORN, Rober E. Desarrollo de Procedimientos Políticas y Documentación. Information Mapping®, Lexinton MA; junio 30, 1991.
- MICROSOFT®. Microsoft Visual Tools Suite. Microsoft®, E.U.A.; 1993.
- MICROSOFT®. Visual FoxPro™, Manual del Programador. Microsoft®, E.U.A.; 1996.
- MORALES, Ana Cecilia. Red TELMEX, Metropolitano. ©Red TELMEX, México; 2000.
- TORRES, L. Juan. Seminario: Análisis de la Planeación. U.N.A.M., E.N.E.P. Acatlán, México; julio del 2000.
- CANAL 22, México. La Industria del Automóvil. T.V. Londinense. Londres, Inglaterra; 2000.
- Véase <http://sol.info.unlp.edu.ar/~luciod/objcour>. Argentina; 2000.
- Véase <http://www.geocities.com/SiliconValley/Lakes/4725>. México; 2000.

## Conclusiones Generales

Una de las herramientas para desarrollar y diseñar software es la programación orientada a objetos, tal vez no sea la idónea para todas las organizaciones; sin embargo, sí es de las más utilizadas en la actualidad en un gran número de organizaciones, y esto es debido principalmente a sus interfaces gráficas como característica relevante de cualquier desarrollo orientado a objetos.

Un experto y precursor de la filosofía orientada a objetos como Larman dice: “La tecnología de objetos es un área muy prometedora, pero su potencial no se aprovechará plenamente si no se poseen las habilidades apropiadas”<sup>1</sup>. Larman habla de una necesidad de describir claramente las funciones que se generen con programación orientada a objetos y afirma también que esta técnica de programación promete mucho para el mundo de la computación. Mediante el presente trabajo de investigación se reconocen las afirmaciones antes mencionadas de Larman y además se presenta solución a la problemática planteada estableciendo lo siguiente:

- i) Primeramente es conveniente entender claramente como funciona el entorno orientado a objetos.
- ii) También es fundamental documentar la aplicación orientado a objetos con una técnica que contemple lo que a continuación se menciona:
  - Es necesario utilizar elementos de descripción de una técnica de documentación de negocios. (Como el método de información Mapping®)
  - El análisis estructurado moderno de Yourdon proporciona diagramas para representar la funcionalidad de las entidades adecuadamente.
  - Al utilizar el lenguaje unificado para la construcción de modelos, (UML, siglas en inglés) se fincan bases importantes para describir adecuadamente la información clave.
  - Con el uso de un formato predefinido y organizando todas las herramientas utilizada se logra encontrar la información rápidamente.
  - Una presentación agradable de la documentación de la aplicación de sistemas para todos los usuarios, es fundamental para un buen desempeño del documento.
  - El énfasis de adaptabilidad, estandarización y flexibilidad del documento que describe el funcionamiento de la aplicación; es ante todo, el ingrediente fundamental para un desempeño adecuado de cualquier documento de sistemas.

---

<sup>1</sup> LARMAN, Craig. UML v Patrones. Introducción al Análisis y Diseño Orientado a Objetos. Ed. Prentice Hall, México, p. xxii; 1999.

Todos estos elementos forman parte esencial de la presente investigación, para que una técnica de documentación presuma de ser eficaz y de esta manera poder describir la información clave adecuadamente, de cualquier aplicación de sistemas y en especial de las generadas con programación orientada a objetos. Estos argumentos dan respuesta a la hipótesis planteada en la introducción de la presente investigación. (Véase Página No. III)

La eliminación de la rigidez en cualquier documento para describir aplicaciones, la representación gráfica y la apertura para cualquier idea, es la parte fundamental de la presente investigación y lo que da origen a una "Técnica de Plataforma"<sup>2</sup>, que viene muy de acuerdo al momento que se vive; es decir, se vive en un mundo en donde si algo es seguro, es de que nada es estático, todo es cambiante, se exige "calidad"<sup>3</sup> para cualquier trabajo y se requiere actuar en el momento oportuno; ya que como se dice... "El tiempo es oro".

Los costos de operación se reducen en un 20%, en el peor de los casos para el área de sistemas de cualquier organización y esto se logra al utilizar la técnica de documentación propuesta, todo esto gracias entre otras cosas a la autosuficiencia que se obtiene con un documento flexible que describa la columna vertebral de los procesos generados en programación orientada a objetos.

En principio se plantea la posibilidad de describir acertadamente cualquier programa orientado a objetos, con una técnica de documentación oportuna; sin embargo y gracias a que se retoman aspectos importantes de herramientas para describir la administración de los negocios, además elementos de análisis y diseño que promueve el análisis estructurado moderno y en su forma anterior, elementos también del análisis estructurado clásico y finalmente elementos del lenguaje unificado de construcción de modelos, entre otras cosas de organización y adecuación. Gracias a todo esto, se puede también describir soluciones desarrolladas con cualquier herramienta de programación y no solo las generadas con ambiente orientado a objetos; todo ello, debido entre otras cosas al énfasis de apertura que propone la técnica de documentación expuesta a lo largo del presente trabajo de investigación.

---

<sup>2</sup> "La creación de un estándar para la fabricación de automóviles, (plataformas iguales en todas las compañías) es la solución a las demandas actuales del mercado de autos. soluciones integrales se requieren en un mundo tan competitivo. Al establecimientos de estándares para la construcción de autos se le denomina: *Tecnología de Plataforma.*"  
Canal 22, México. *La industria del Automóvil*. T.V. Londinense. Londres, Inglaterra; 2000.

<sup>3</sup> TORRES, L. Juan. *Seminario Análisis de la Planeación*. U.N.A.M., E.N.E.P. Acatlán, México; 22 de julio del 2000.  
(Véase Cuadro No. 1 en la Página No. 18)

## Fuentes de Consulta Generales

- ACKOFF, Russell L. Planificación de la Empresa del Futuro. Ed. Limusa, México; 1983.
- ACKOFF, Russell L. Recreación de las Corporaciones. Un Diseño Organizacional para el Siglo XXI. Ed. Oxford University Press, México; 2000.
- ACKOFF, Russell L. Un Concepto de Planeación de Empresas. Ed. Limusa, México; 1972.
- BERTALANFFY, Ludwig Von. Tendencias en la Teoría General de Sistemas. Ed. Alianza, Madrid España; 1978.
- BOOCH, Grady. Diseño Orientado a Objetos. Ed. Oxford University Press, México; 1998.
- CEBALLOS Sierra, Francisco J. Programación Orientada a Objetos con C++. Ed. Alfaomega, México; 1998.
- COX, Brad J. Programación Orientada a Objetos: Un Enfoque Evolutivo. Ed. Addison-Wesley, Wilmington Delaware, E.U.A.; 1993.
- CHURCHMAN, C. West. El Enfoque de Sistemas para la Toma de Decisiones. Ed. Diana, México; 1995.
- DEMARCO, Tom. Structured Analysis and System Specification. Ed. Prentice Hall, Englewood Cliffs NJ.; 1979.
- DEVIS Botella, Ricardo. Programación Orientada a Objetos en C++. Ed. Paraninfo, Madrid España; 1993.
- EZZELL, Ben. Programación de Gráficos en Turbo C++. Ed. Addison-Wesley Iberoamericana S.A., E.U.A.; 1993.
- FOWLER, Martin. UML Gota a Gota: Actualizado para Cubrir la Versión 1.2 OMG UML Standard. Ed. Addison Wesley Longman, México; 1999.
- GANE, Christopher P. Análisis Estructurado de Sistemas. Ed. El Ateneo, México; 1987.
- GONZÁLES Pérez, Manuel. Programación Orientada a Objetos con Borland Pascal<sup>TM</sup>. Ed. Anaya Multimedia, Madrid España; 1994.

## Fuentes de Consulta Generales, Continuación ...

- JOYANES Aguilar, Luis. Programación Orientada a Objetos. Ed. McGraw Hill Interamericana, S.A., España; 1998.
- LARMAN, Craig. UML y Patrones, Introducción al Análisis y Diseño Orientado a Objetos. Ed. Prentice Hall, México; 1999.
- MORALES Lozano, Angel. Programación Orientada a Objetos: Aplicaciones con Smalltalk™. Ed. Paraninfo, Madrid España; 1993.
- ROBBINS, Stephen P. Administración: Teoría y Práctica. Ed. Prentice Hall, México; 1994.
- SETHI, Ravi. Lenguaje de Programación, Conceptos y Constructores. Ed. Addison-Wesley Iberoamericana S.A., E.U.A.; 1992.
- VAN DER HEIJDEN, Kees. Escenarios, El Arte de Prevenir el Futuro. Ed. Panorama, México; 1998.
- VOSS, Greg. Programación Orientada a Objetos: Una Introducción. Ed. McGraw Hill, México; 1994.
- YOURDON, Edward. Análisis Estructurado Moderno. Ed. Prentice Hall, México D.F.; 1993.
- ERIKSSON, Hans-Erik. UML Toolkit. Ed. Wiley, New York E.U.A.; 1998.
- HORN, Rober E. Desarrollo de Procedimientos Políticas y Documentación. Information Mapping®, Lexinton MA; junio 30, 1991.
- MICROSOFT®. Microsoft Visual Tools Suite. Microsoft®, E.U.A.; 1993.
- MICROSOFT®. Visual FoxPro™, Manual del Programador. Microsoft®, E.U.A.; 1996.
- MORALES, Ana Cecilia. Red TELMEX, Metropolitano. ©Red TELMEX, México; 2000.

## Fuentes de Consulta Generales, Continuación ...

- REYES Martínez, Hugo. Seminario: Análisis de la Planeación. UN.A.M., E.N.E.P. Acatlán, México; marzo del 2000.
- TORRES Lovera, Juan. Seminario: Análisis de la Planeación. UN.A.M., E.N.E.P. Acatlán, México; julio del 2000.
- CANAL 22, México. La Industria del Automóvil. T.V. Londinense. Londres, Inglaterra; 2000.
- AGULLO Soliveres, Pedro. Java Vs. C++: Un Estudio Comparativo de Java. <http://www.ctv.es/USERS/pagullo/poo.htm>, México; 2000.
- AYALA Mojica, Pedro. Programación Orientada a Objetos (Conceptos Básicos). [http://www.lasalle.edu.co/~payalam/p\\_objeto.html](http://www.lasalle.edu.co/~payalam/p_objeto.html), México; 2000.
- GORDILLO, Silvia. Desarrollo de Lenguajes de Consultas para Sistemas Hypermediales. <http://www-lifia.info.unlp.edu.ar/spanish>, Argentina; 2000.
- Véase <http://sol.info.unlp.edu.ar/~luciod/objcour>. Argentina; 2000.
- Véase <http://www.ctv.es/USERS/pagullo/poo.htm>. México; 2000.
- Véase <http://www.geocities.com/SiliconValley/Lakes/4725>. México; 2000.



# Glosario de Términos

| <i>Término</i>                   | <i>Descripción</i>  |
|----------------------------------|---|
| Análisis                         | Distinción de las partes de un sistema para llegar a conocer sus principios y elementos.  |
| Aplicaciones de sistemas         | Se hablan de programas elaborados en una computadora con el fin de resolver problemas de procesamiento de información digital.    |
| Aplicaciones gráficas            | Tipo de visualización que permite al usuario elegir de entre diferentes opciones mediante representaciones visuales.              |
| Apuntador                        | Dispositivo capaz de manipular instrucciones u objetos visuales por medio de un programa.   |
| Base de datos                    | Conjunto de datos organizados para su almacenamiento dentro de la memoria de una computadora.                                     |
| Basic, C, Cobol, Fortran, Pascal | Lenguajes de programación de alto nivel.  |
| Campo                            | Información almacenada en la computadora, registro informático.   |
| Clases                           | Define las estructuras de datos y rutinas de un objeto informático.   |
| Cliente                          | Usuario de un sistema de aplicación para el cual se realiza un desarrollo.  |
| Código de máquina                | Instrucciones que se utilizan para que la computadora realice operaciones basadas en direcciones electrónicas de memoria.         |
| Comandos                         | Instrucciones que procesa la computadora mediante lenguajes de programación.  |
| Compilador                       | Un compilador crea una lista de instrucciones que son procesadas por la computadora, ( código objeto) basándose en código fuente. |
| Computadora                      | Máquina digital y automática auxiliar para el tratamiento de información.   |
| Diseño                           | Es la descripción de una aplicación de sistemas mediante la construcción de algoritmos.   |
| Forma                            | Contenedor de objetos informáticos.   |
| Gráfico                          | Imagen de dos y tres dimensiones creados por computadora.   |
| Hardware                         | Componentes físicos de una computadora.   |
| Herencia                         | Capacidad de adquirir atributos de una entidad informática superior o de mayor grado.   |

## Glosario de Términos, Continuación ...

| <i>Término</i>                | <i>Descripción</i>  |
|-------------------------------|---|
| Herramienta                   | Técnica o instrumento para usarse en el desarrollo de aplicaciones de sistemas.   |
| Heurística                    | Arte inventivo del programador para crear aplicaciones de sistemas; creatividad.  |
| Icono                         | Imagen gráfica mostrada en la pantalla de una computadora que representa un objeto informático para el usuario.                             |
| Identificadores               | Instrucciones predeterminadas dentro de un lenguaje de programación y que sirven para codificar una aplicación.                             |
| Implantación                  | Fase final del desarrollo de una aplicación de sistemas.  |
| Índice                        | Orden de una base de datos creado para el control de la misma.  |
| Informática                   | Conjunto de técnicas que hacen posible el procesamiento de datos mediante una computadora.  |
| Lenguajes                     | Instrucciones entendibles por el programador para comunicarse con la computadora y realizar operaciones.                                    |
| Lenguaje de alto nivel        | Instrucciones entendibles por el programados para comunicarse con la computadora y generalmente se estructuran con el idioma de su creador. |
| Lenguaje de bajo nivel        | Abstracciones para comunicarse con la computadora mediante instrucciones de un lenguaje mnemónico.  |
| Lenguaje de cuarta generación | Lenguaje estructurado e interpretado de fácil ampliación y ofrece una alta funcionalidad en espacios muy reducidos.                         |
| Lenguaje ensamblador          | Instrucciones para comunicarse con la computadora mediante localidades electrónicas de memoria.   |
| Lenguaje interprete           | La traducción de un código fuente a un código máquina la realiza un interprete, compilación de instrucciones.                               |
| Lenguaje máquina              | Instrucciones para comunicarse con la computadora basado en el sistema de numeración binario.   |
| Máquina                       | Máquina digital y automática auxiliar para el tratamiento de información.   |
| Memoria                       | Lugar donde se almacena y distribuye la información digitalizada.   |

# Glosario de Términos,

## Continuación

| <i>Término</i>       | <i>Descripción</i>   |
|----------------------|--|
| Mensaje              | Se refiere al envío de instrucciones en una computadora, intercambio de instrucciones mediante el uso de objetos informáticos. . |
| Menú                 | Selección de instrucciones dentro de una computadora.  |
| Objeto               | Es una instancia dentro de un programa que se puede usar como una variable; entidad informática.                                 |
| Operador             | Ejecutor de la aplicación de sistemas o del equipo de cómputo.   |
| Programador          | Es la persona que escribe ó modifica instrucciones que ejecuta una computadora.  |
| Programa fuente      | Instrucciones generadas por el programador de aplicaciones de sistemas.  |
| Programa objeto      | Instrucciones generadas por un compilador dentro de una computadora.   |
| Presentation manager | Interfaz gráfica de usuario que procede del entorno de Windows y basado en MsDOS.  |
| Producción           | Aplicación de sistemas que utilizan los usuarios, en productivo.   |
| Sistema binario      | Sistema de numeración basado en dos estados (Abierto y Cerrado), que es entendible por la computadora.                           |
| Software             | Programas ó instrucciones que contiene y ejecuta una computadora.  |
| Unix                 | Sistema operativo multiusuario basado en el protocolo de comunicación TCP/IP.  |
| Usuario              | Es la persona que hace uso de los sistemas de aplicaciones o de las herramientas de informática con ciertas limitaciones..       |