



15 872748  
**UNIVERSIDAD DON VASCO, A. C.**

**INCORPORACION No. 8727-48 A LA  
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

## **ESCUELA DE INFORMATICA**

29/07/79  
" **DESARROLLO DE UN PROGRAMA PARA LA ELABORACIÓN  
DE ANÁLISIS FINANCIEROS PARA EL ÁREA DE  
CONTABILIDAD DE LA EMPRESA  
PROMOTORA Y CONSTRUCTORA ZIRAHUEN, S.A. DE C.V.,  
DE LA CIUDAD DE URUAPAN, MICHOACÁN "**

## **SEMINARIO DE INVESTIGACIÓN**

**QUE PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN INFORMÁTICA**

**P R E S E N T A :  
OMAR ALBERTO NARANJO GARCÍA**



**UNIVERSIDAD  
DON VASCO, A.C.**

**URUAPAN, MICHOACÁN 1979.**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**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.

# INDICE GENERAL

	Pág.
<b>CAPITULO 1 EL ANÁLISIS FINANCIERO</b>	
1.1 Las finanzas .....	8
1.1.1 La información financiera .....	10
1.2 Los estados financieros .....	12
1.3 Métodos de análisis .....	15
1.3.1 Método de razones financieras .....	16
1.3.2 Método de porcentos integrales .....	22
1.3.3 Método gráfico .....	24
1.3.4 Punto de equilibrio .....	26
1.3.5 Método de aumentos y disminuciones .....	28
1.3.6 Método de tendencias .....	31
1.3.7 Método de origen y aplicación de recursos .....	33
<b>CAPITULO 2 LOS LENGUAJES DE PROGRAMACIÓN</b>	
2.1 Los lenguajes de programación .....	39
2.2 Historia de los lenguajes de programación .....	40
2.3 Historia de los lenguajes orientados a objetos .....	46
2.4 Ventajas de los lenguajes orientados a objetos .....	51
2.5 Conceptos básicos de la programación orientada a objetos .....	52
<b>CAPITULO 3 ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS</b>	
3.1 Antecedentes de la ingeniería de software .....	57

3.2 Principios de la ingeniería de software .....	58
3.3 Elementos de la ingeniería de software .....	59
3.4 Cualidades del software .....	59
3.4.1 Cualidades externas .....	60
3.4.2 Cualidades internas .....	60
3.5 Mecanismos básicos de la orientación a objetos .....	62
3.5.1 Objeto .....	63
3.5.1.1 Relación entre objetos .....	64
3.5.2 Clase .....	66
3.5.2.1 Relación entre clases .....	66
3.5.3 Mensajes y métodos .....	71
3.6 La notación .....	72
3.6.1 Diagrama de clases .....	72
3.6.2 Diagrama de objetos .....	75
3.6.3 Diagrama de módulos .....	76
3.6.4 Diagrama de estados de transición .....	78
3.7 Microproceso para el desarrollo .....	80
3.7.1 Identificar clases y objetos .....	81
3.7.2 Identificar la semántica de clases y objetos .....	83
3.7.3 Identificar las relaciones entre clases y objetos .....	83
3.7.4 Implementación de clases y objetos .....	84

## **CAPITULO 4 CASO PRACTICO**

4.1 Metodología .....	86
4.2 Datos generales de la empresa .....	87
4.3 Planteamiento del problema .....	87
4.3.1 Características del programa .....	89
4.4 Justificación .....	89
4.4.1 Lenguaje de programación .....	89
4.4.2 Autor (metodología) .....	90
4.5 Análisis y diseño orientado a objetos .....	91
4.5.1 Definición del problema y descripción de la solución .	91
4.5.2 Identificación de objetos .....	92
4.5.3 Identificación de métodos .....	93
4.5.4 Definición de interfaces entre objetos .....	94
4.5.5 Diagrama de objetos .....	95
4.5.6 Diagrama de clases .....	96
4.5.7 Diagrama de estados de transición .....	97
4.5.8 Diagrama de módulos .....	98
4.5.9 Diccionario de datos .....	99
<b>CONCLUSIONES</b> .....	<b>101</b>
<b>BIBLIOGRAFÍA</b> .....	<b>103</b>

## INTRODUCCIÓN

El análisis de estados financieros es una tarea que actualmente es realizada por la mayoría de las empresas para conocer su situación económica, a fin de tomar las medidas adecuadas que garanticen a la empresa una situación económica estable. La empresa "Promotora y Constructora Zirahuen S.A. de C.V.", ubicada en la ciudad de Uruapan, Michoacán, en su departamento de Contabilidad actualmente elabora éste análisis de estados financieros manualmente, lo que en ocasiones trae como consecuencia pérdidas económicas para la empresa, ya que dicha información no es generada de forma eficiente, veraz y oportuna para la empresa y por lo tanto para la adecuada toma de decisiones, lo que significa pérdidas para la organización. Este problema afecta a los usuarios encargados de elaborar los informes y al gerente general para la acertada toma de decisiones, los usuarios están conscientes del problema que esto representa, ya que en ocasiones no se genera la documentación en el momento adecuado y no se tiene la información al día, existiendo pérdida de tiempo en elaborar tales reportes; para esto actualmente no se ha realizado algún proyecto para mejorar y agilizar las actividades en esta área de la institución, por lo que a mi parecer considero que una de las soluciones para resolver este problema, es el de realizar un estudio que conste de un análisis y diseño para desarrollar un programa hecho a la medida que cubra con las necesidades de esta área y realice tales análisis

financieros de forma automatizada. Con la solución al problema planteado anteriormente la institución será capaz de tener un mejor control de la información y de elaborar sus reportes en el momento en que requiera, brindándole al empresario reportes completos, veraces y oportunos para la buena toma de decisiones; también es de gran importancia para mí, ya que puedo llevar a la práctica los conocimientos que he obtenido a través de mi formación como estudiante y como persona.

El objetivo principal que se persiguió en esta investigación es el de desarrollar un programa hecho a la medida para el departamento de contabilidad, de la empresa antes mencionada, para lo cual se requirió identificar las necesidades de la institución, para posteriormente elegir un método y paradigma para la realización del análisis y diseño. El desarrollo adecuado de un programa hecho a la medida dependerá básicamente del estudio y análisis de necesidades.

El método utilizado en este proyecto es el explicativo, debido a que no se llegó hasta la implantación, sólo se describe el proceso para llegar al diseño más apropiado del programa para la institución en estudio. La técnica que utilizada fue la entrevista, ya que el número de personas que laboran en esta área de la institución es reducido, dando así la facilidad a los entrevistados de expresarse más ampliamente y obtener algunos detalles importantes que fueron útiles en ésta investigación.

La presente investigación consta de tres capítulos teóricos y un caso práctico; en el capítulo uno se analizará cual es la importancia de las finanzas, sus objetivos, los métodos que se puede aplicar para llevar a cabo el análisis financiero y sobre que documentos se puede aplicar, este capítulo es la base principal para la realización de este proyecto; el segundo capítulo presenta una breve historia de los lenguajes de programación, desde los años 50's hasta los 90's, tanto de lenguajes estructurados, como lenguajes orientados a objetos, así como algunas ventajas de estos últimos en comparación con los primeros, de igual forma se menciona la finalidad para la que fueron creados, dando con ésto una visión más clara para determinar el lenguaje que más apropiado y aplicarlo en el desarrollo del software de esta investigación; el tercer capítulo se trata lo referente al análisis y diseño orientado a objetos, por lo que se analizarán los conceptos básicos que maneja este paradigma, así como algunas herramientas, como lo es la ingeniería de software, que nos ayuda a crear programas más eficientes y de mejor calidad, por último se estudiará un microproceso para llevar a cabo este análisis y diseño orientado a objetos; culminando así con lo planteado desde un principio, el desarrollar un programa a la medida para la elaboración de análisis financieros, en el cuarto capítulo de esta investigación, que es el caso práctico.



## **CAPITULO 1**

### ***EL ANÁLISIS FINANCIERO***

---

Todas las empresas a medida que ha transcurrido el tiempo se han dado cuenta de que cada vez es más importante mejorar sus políticas administrativas para poder ser más productivas y obtener mayores utilidades, pero para esto se requiere de información financiera oportuna, veraz y confiable, que auxilie a los empresarios a fortalecer la toma de decisiones.

El análisis y estudio de estados financieros, permite recabar esa información financiera mediante un examen minucioso de cada una de las partidas que integran los rubros de los estados financieros, a fin de conocer como ha venido desarrollándose dicha empresa y tomar las acciones más adecuadas para el curso de la misma.

En este capítulo se estudiará primeramente el concepto de las finanzas, así como su importancia dentro de una empresa, pasando posteriormente al apartado de los estados financieros, en donde se analizará tanto su concepto, como algunos de los estados financieros más comunes, y por último se describen algunos de los métodos utilizados para llevar acabo el análisis de estados financieros.

## 1.1 LAS FINANZAS.

*Concepto:*

“Las finanzas son un cuerpo de hechos, principios y teorías relacionadas con la búsqueda y aplicación del dinero, ya sea por individuos o por empresas públicas o privadas”.

(CALVO, 1993:102)

Por lo tanto, como se menciona en el concepto anterior, la aplicación de las finanzas es general.

Las finanzas tienen como objetivo principal lo siguiente:

- *Allegarse de recursos.*- consiste en el estudio de opciones y la elección más conveniente de acuerdo con los requerimientos de la empresa.
- *Aplicación de dichos recursos.*- implica la utilización de los recursos de acuerdo a la correcta inversión que otorgue el mayor rendimiento a la empresa; ya que la finalidad de ésta consiste en la maximización de utilidades.

Hoy en día las finanzas representan una de las áreas más importantes en la empresa, debido a que a través de las actividades realizadas por ésta área, se puede contar con información veraz y confiable que sirva de sustento, tanto para la correcta toma de decisiones, como para definir las acciones a realizar por parte de la empresa.

Las finanzas dentro de la empresa juega uno de los papeles más importantes que es realizado generalmente por el gerente o el administrador, que son las personas encargadas de la toma de decisiones; sin embargo, para esta toma de decisiones se debe tomar como base la información que presenten todos los departamentos de la empresa, la cual la mayoría de las veces resulta insuficiente e incompleta, situación que no se presenta si se cuenta con un departamento de finanzas.

Por lo tanto, el responsable de la administración financiera realiza prácticamente dos funciones enfocadas a obtener el máximo rendimiento de la inversión de los propietarios o inversionistas.

- Por una parte tiene la responsabilidad de allegarse de los fondos necesarios para hacer frente a los compromisos contraídos por parte de la empresa.

- Por otra parte, debe de buscar la obtención de utilidades invirtiendo el efectivo en actividades que garanticen rendimientos atractivos.

Una vez analizada la importancia de las finanzas y el papel que juegan dentro de la empresa, se estudiará en que consiste la información financiera.

### **1.1.1 LA INFORMACIÓN FINANCIERA.**

La información financiera o análisis financiero se centra principalmente en el estudio de estados financieros, con la finalidad de conocer el desarrollo de las operaciones de una empresa.

*Concepto:*

“El análisis de los estados financieros es el método por el cual se llega al conocimiento de cada uno de los elementos que forman dichos estados, a fin de poder estudiar su comportamiento dentro de la estructura financiera de una empresa”.

(GALVEZ, 1992:89)

El análisis financiero consiste en la separación y estudio de cada uno de los apartados que conforman un estado financiero, a fin de conocer sus

orígenes, los cambios sufridos y causas, con el objeto de tener un panorama general, claro y preciso sobre la situación financiera de la empresa.

El análisis de los estados financieros es un trabajo arduo que consiste en realizar una serie de operaciones matemáticas encaminadas a calcular los saldos de una empresa; con base en lo anterior podemos concluir que el análisis e interpretación de estados financieros, consiste en conocer la base de las operaciones de una empresa mediante el estudio de los estados financieros, a fin de obtener conclusiones que nos permitan dar una opinión sobre la buena o mala política administrativa de la empresa en estudio.

Las tres áreas generales de estudio en el análisis financiero de una empresa son:

1. *Liquidez*. Consiste en contar con los recursos necesarios para cubrir las deudas de una empresa. Para determinar la solvencia de una empresa se realiza un estudio de su situación a corto plazo.
2. *Rentabilidad*. Es la capacidad que tiene una empresa para generar utilidades. Generalmente son los accionistas las personas más

interesadas en la rentabilidad de una empresa, esta rentabilidad debe de estar acorde al riesgo que se asuma.

3. *Estabilidad.* Consiste en que una empresa garantice su desarrollo normal y continuo en sus actividades; con el estudio de la misma se determina si la empresa está en condiciones de hacer frente a sus obligaciones futuras.

## **1.2 LOS ESTADOS FINANCIEROS.**

Los estados financieros son documentos generados generalmente en forma periódica, elaborados basándose en datos obtenidos de los registros contables de las actividades realizadas por una empresa, durante un periodo o fecha determinada y que refleja la situación financiera de la misma.

Los estados financieros tienen como objetivo principal presentar un informe sobre la situación financiera de una empresa en una fecha o periodo determinado, y que sirvan de sustento para la toma de decisiones, satisfaciendo así las necesidades de información de las empresas, tanto públicas como privadas.

A continuación se presentan algunos de los estados financieros más utilizados, así como una breve descripción de cada uno de ellos, en los cuales se pueden aplicar los diversos métodos que existen para realizar el análisis financiero, con el objeto de conocer la situación financiera de una empresa.

### *Balance General*

“Estado que muestra la situación financiera de una empresa a una fecha determinada. Este estado también es conocido como Estado de Situación Financiera debido a que incluye todos los derechos y obligaciones que tiene una empresa, siendo éste estado el más completo que se formula”.

(CALVO, 1993:7)

### *Estado de Resultados*

“Estado que muestra las operaciones realizadas por una empresa durante un periodo determinado, éste estado es importante por que nos muestra paso a paso la forma en que se llegó al resultado final (utilidad o pérdida)”.

(CALVO, 1993:7)

### *Estado de Costo de Producción y Costo de Ventas*

“Estado financiero que muestra detalladamente todos los rubros que conforman el costo de producción y/o costo de lo vendido en un periodo determinado, el resultado de este estado nos ayuda a determinar el precio de venta con un margen de utilidad bruta adecuado y llegar así al resultado final deseado”.

(CALVO, 1993:7)

### *Estados Comparativos*

“Estados utilizados para determinar las variaciones sufridas como consecuencia de la comparación de estados financieros homogéneos, pero que se refieren a fechas diferentes”.

(CALVO, 1993:7)

### *Estado de Capital de trabajo*

“Estado que muestra la diferencia entre el activo circulante y el pasivo circulante a una fecha determinada. Este estado nos muestra realmente con lo que la empresa cuenta como propio”.

(CALVO, 1993:8)



### *Estado de Movimiento de las Cuentas de Capital*

“Estado que muestra la diferencia entre el Capital Social y el Capital Contable, este estado también es conocido como Estado del Superávit”.

(CALVO, 1993:8)

### *Estado de Cambios en la Situación Financiera en Base al Flujo de Efectivo*

“Es un estado que nos informa sobre los recursos con los que contó la empresa y cuál fue la aplicación de dichos recursos durante un periodo determinado”.

(CALVO, 1993:9)

## **1.3 MÉTODOS DE ANALISIS.**

Los métodos de análisis se pueden clasificar en dos grupos, métodos horizontales y métodos verticales. Los primeros consisten en realizar un estudio de la información financiera de varios años de la empresa; y los métodos verticales se refieren a que las cifras obtenidas del estudio corresponden únicamente a un sólo ejercicio. Algunos de los métodos más utilizados en el análisis de estados financieros son:

### **1.3.1 MÉTODO DE RAZONES FINANCIERAS.**

Las razones financieras son el medio más utilizado para analizar la información financiera de las empresas, consiste en relacionar una partida con otra, como por ejemplo relacionar los activos circulantes con los pasivos circulantes, ambos constituyen un conjunto de partidas. También se podrán relacionar partidas entre sí de un estado financiero, o bien, de estados financieros distintos.

Las razones financieras se pueden utilizar para analizar estados financieros históricos, situaciones actuales o como herramienta útil para la planeación financiera, que sirve para evaluar el desempeño y las tendencias de una empresa.

Las razones financieras no son limitadas, se pueden utilizar en forma ilimitada y variable de acuerdo a las necesidades de las empresas, y tienen como finalidad ofrecer datos concretos y veraces que nos faciliten el mejor entendimiento de las cantidades reflejadas en los estados financieros.

Las formulas que a continuación se presentan para realizar el análisis de estados financieros por el método de las razones financieras, se

tomaron de apuntes obtenidos en la materia de Finanzas I, impartida por la L.C. Elvia Lorena Torres Alejandre.

### *Razones de liquidez*

Las razones de liquidez se refieren al monto; a como se encuentra compuesto el pasivo y la relación que guarda con el activo circulante, siendo éste último, el origen de recursos con que cuenta la empresa para hacer frente a sus obligaciones contraídas.

- *Relación del circulante*

$$\frac{\text{Activo circulante}}{\text{Pasivo circulante}}$$

Esta razón muestra el efectivo que tiene la empresa en activo circulante para cubrir cada peso de pasivo circulante.

- *Prueba del ácido*

$$\frac{\text{Activo circulante- Inventarios}}{\text{Pasivo circulante}}$$

Esto significa, el efectivo que se tiene de activo disponible para cubrir cada peso de las obligaciones contraídas a corto plazo.

- *Rotación de cuentas por pagar*

$$\frac{\text{Costo de ventas}}{\text{Cuentas por pagar}}$$

La rotación de cuentas por pagar nos indica el número de veces que se aplico un crédito con los proveedores.

- *Rotación del activo total*

$$\frac{\text{Ventas netas}}{\text{Activo total}}$$

Esa razón nos indica el margen de utilidad que se tiene por cada peso invertido en la empresa.

- *Rotación del activo fijo*

$$\frac{\text{Ventas netas}}{\text{Activo fijo}}$$

La rotación del activo fijo señala el porcentaje que se genera de utilidad en base a lo que se tiene invertido en el activo fijo (estructura de la empresa).

- *Rotación de cuentas por cobrar*

$$\frac{\text{Ventas netas}}{\text{Cuentas por cobrar}}$$

Esta razón indica el número de veces que los clientes utilizaron los créditos, entre menor sea el número de veces es mejor para la empresa.

- *Rotación de inventarios*

$$\frac{\text{Costo de ventas}}{\text{Inventarios}}$$

La rotación de inventarios es la velocidad de movimiento de los inventarios en la empresa, a mayor rotación de inventarios mayores ventas.

#### *Razones de palanca financiera*

- *Deuda total a activo total*

$$\frac{\text{Deuda total}}{\text{Activo total}}$$

Esta razón nos muestra que de todo lo que tiene la empresa el porcentaje que no es de ella.

- *Multiplicador del capital*

$$\frac{\text{Activo total}}{\text{Capital contable}}$$

Esta razón expresa lo que ha generado la empresa adicional, después de lo que representa el capital contable, por cualquier otra fuente.

- *Razón de cobertura*

$$\frac{\text{Utilidad de operación}}{\text{Interés + Gtos. financieros + [ ( Pagos de amort. ) / ( 1- Tarifa impto ) ]}}$$

La razón de cobertura nos indica como repercuten los intereses y los impuestos en las utilidades de la empresa e indican cuantas veces se llegan a cubrir éstos.

#### *Razones de rentabilidad*

- *Margen de utilidad en las operaciones*

$$\frac{\text{Utilidad antes de impuestos e intereses}}{\text{Ventas netas}}$$

Esta razón nos indica la utilidad de operación que han generado las ventas netas sin considerar intereses.

- *Margen de utilidad neta*

$$\frac{\text{Utilidad neta}}{\text{Ventas netas}}$$

El resultado de esta razón nos indica que porcentaje de las ventas representa una utilidad neta sin considerar los gastos de venta y de administración.

- *Rendimiento sobre utilidad de operación*

$$\frac{\text{Utilidad antes de impuestos e intereses}}{\text{Total activos}}$$

Esto nos indica que porcentaje de utilidad se generó con las operaciones de la empresa.

- *Rendimiento neto sobre activos*

$$\frac{\text{Utilidad neta}}{\text{Total activos}}$$

Representa la utilidad neta que generó la inversión total que se tiene en la empresa.

- *Rendimiento sobre el capital*

$$\frac{\text{Utilidad neta}}{\text{Capital contable}}$$

Esta razón indica el porcentaje que se generó de utilidad por el dinero invertido por los socios o accionistas.

### **1.3.2 MÉTODO DE PORCIENTOS INTEGRALES.**

El método de porcientos integrales consiste en determinar la proporción que guarda cada uno de los elementos que integran los estado financieros con respecto al total.

Este método es aplicable en aquellos casos en los cuales se desea conocer la importancia que tiene la parte de un todo en relación con dicho todo, por lo que los resultados quedan expresados en porcientos, se puede utilizar en cualquier estado financiero, sin embargo, por lo general se maneja principalmente en el Estado de Situación Financiera y Estado de Resultados.

La ventaja en la utilización de este método es de que se comprende más fácilmente cual es la importancia de cada uno de los apartados que



integran un estado financiero, sin embargo, es aconsejable manejar este método para obtener conclusiones aisladas, ya que el hablar en términos de porcentajes es fácil caer en conclusiones erróneas.

Aplicando este método al estado de resultados, se considera a las ventas netas como el 100% y se relacionarán las partidas restantes con dichas ventas netas, como lo muestra el ejemplo siguiente:

Ventas	120,000	100%
Costo de ventas	<u>70,000</u>	<u>58%</u>
Utilidad Bruta	50,000	42%
Gastos de operación		
Gastos de venta	10,000	8%
Gastos de Admón.	<u>10,000</u>	<u>8%</u>
Total gastos	<u>20,000</u>	<u>16%</u>
Utilidad de operación	30,000	26%
Gastos financieros	<u>25,000</u>	<u>21%</u>
Utilidad antes de ISR	5,000	5%

### **1.3.3 MÉTODO GRAFICO.**

La aplicación de este método consiste en representar en forma gráfica el análisis realizado a un estado financiero, ofreciendo una mayor facilidad de percepción e interpretación de la situación financiera de la empresa, dado que las variaciones se perciben con mayor claridad.

La representación gráfica para este método puede ser tan variada como el analista lo desee, de acuerdo a los fines que pretenda y a su predilección por las gráficas, de entre las formas más utilizadas para la representación de este método están:

- Las gráficas de pastel.
- Las gráficas de barras.
- Las gráficas de tipo cartesiano.
- Las gráficas circulares.

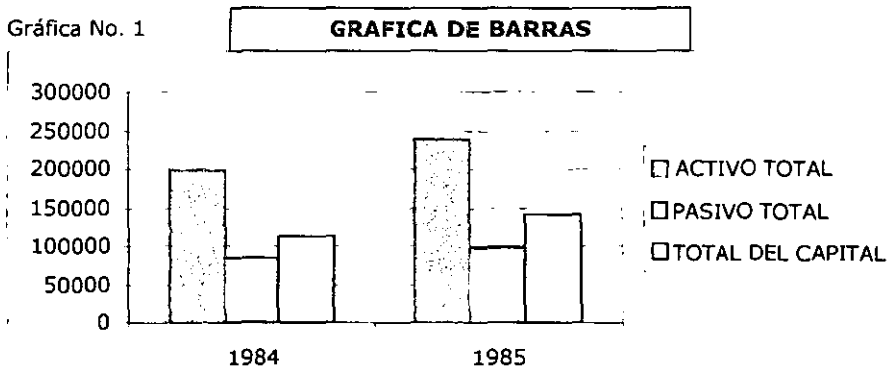
Por lo regular la utilización de las gráficas se realiza con la finalidad de destacar información importante, proporcionándonos con ello una idea más clara de los rubros de los estados financieros en estudio.

La Gráfica No. 1 y la Gráfica No. 2 son formas de representar los rubros que integran los estados financieros utilizando el método gráfico, tomando como base los datos de la siguiente tabla.

	AÑO		AUMENTO O DISMINUCION DURANTE 1985	
	1984	1985	IMPORTE	% DE CAMBIO
	ACTIVO CIRCULANTE	143300	183500	40200
ACTIVO TOTAL	197100	238800	41700	21.16
PASIVO CIRCULANTE	72700	73000	300	0.41
PASIVO TOTAL	84500	97100	12600	14.91
TOTAL DEL CAPITAL	112600	141700	29100	25.84

(MORENO,1995:326)

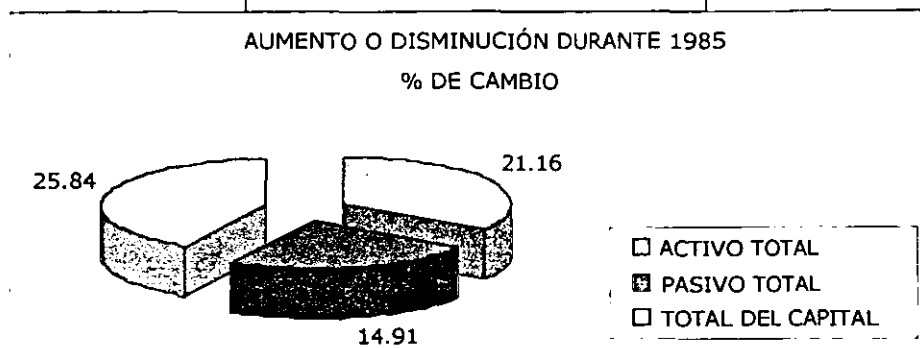
Gráfica No. 1



Elaboración propia.

Gráfica No. 2

## GRAFICA DE PASTEL

**1.3.4 PUNTO DE EQUILIBRIO.**

En términos financieros podemos decir que consiste en predeterminar un importe en el cual una empresa no sufra pérdidas ni obtenga ganancias, es decir, es el punto en que las ventas de la empresa son iguales a los costos y los gastos.

Para aplicar este método se debe conocer el comportamiento de los costos, ingresos y gastos, separando lo que son gastos variables y gastos fijos.

Los gastos fijos se generan a través del tiempo independientemente del volumen de producción, como puede ser la depreciación en línea recta,

los salarios, las rentas, etc., por el contrario los gastos variables se generan basándose en el volumen de producción, como la mano de obra utilizada en la elaboración de los productos y la materia prima utilizada en los productos.

El punto de equilibrio puede determinarse de dos formas:

1. *Por fórmula:*

$$PE = CF / 1 - CV$$

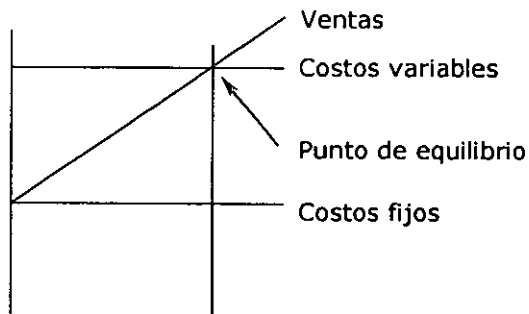
En donde:

PE = punto de equilibrio

CF = costos fijos

CV = costos variables

2. *Por gráfica:*



### **1.3.5 MÉTODO DE AUMENTOS Y DISMINUCIONES.**

Este método consiste en la comparación de cada uno de los rubros financieros homogéneos, pero referentes a distintas fechas, obteniéndose como resultado las diferencias por aumento o disminución, los cuales se deben analizar para determinar sus causas que las motivaron.

Los requisitos para llevar a cabo dicha comparación son:

- Deben ser iguales los periodos de los estados financieros que se van a comparar.
- Que haya sido consistente la aplicación de los principios de contabilidad en ambos periodos de los estados financieros a comparar.
- Se debe contar además con información complementaria o en su caso otros estados financieros.

El resultado de conocer las diferencias de aumentos o disminuciones por la comparación de estados financieros, aunque nos proporcionen una idea sobre el desarrollo de las actividades de la empresa, se requiere de

investigar más a fondo las causas que originaron dichas variaciones y contestar algunas preguntas como:

- La disminución en caja y bancos indica que existe una política de cobranza flexible, o que se han utilizado más eficientemente los fondos en el pago de los pasivos.
- El aumento en clientes indica que han incrementado las ventas, o que no existe una política de cobranza eficiente.
- El aumento en inventarios indica que existe una mala administración en las compras, o que existe un aumento en las ventas.
- La disminución de documentos por pagar indica que se han liquidado oportunamente, o que se han acordado su pago a largo plazo y se pasaron al pasivo fijo.

A continuación se muestra un balance comparativo de los ejercicios al 31 de Diciembre tanto del año 1985, como del año 1986, en donde se puede apreciar más claramente la aplicación de este método. Cabe resaltar que la columna de diferencia es la que se debe de analizar detenidamente, a fin de conocer su origen.

<b>EMPRESA "X", S.A.</b>			
<b>BALANCE COMPARATIVO DE LOS EJERCICIOS AL 31 DE DICIEMBRE 1985 Y 1986.</b>			
	<b>MILES DE PESOS</b>		<b>DIFERENCIA</b>
	<b>1985</b>	<b>1986</b>	
<b>ACTIVO</b>			
<b>CIRCULANTE</b>			
Caja y bancos	2181	104	2077 -
Inversiones en valores	109	203	94 +
Documentos por cobrar	343	57	286 -
Clientes	12407	14123	1716 +
Inventarios	9225	13780	4555 +
	<b>24265</b>	<b>28267</b>	<b>4002 +</b>
<b>FIJO</b>			
Maquinaria y equipo	442	584	142 +
Equipo de transporte	200	279	79 +
Muebles y enseres	50	37	13 -
	<b>692</b>	<b>900</b>	<b>208 +</b>
<b>OTRO ACTIVO</b>			
	1456	925	531 -
<b>SUMA ACTIVO</b>			
	<b>26413</b>	<b>30092</b>	<b>3679 +</b>
<b>PASIVO</b>			
<b>CIRCULANTE</b>			
Proveedores	1204	1806	602 +
Acreedores diversos	1292	1682	386 +
Documentos por pagar	5872	2518	3354 -
Gastos y otros por pagar	1601	2223	622 +
	<b>9973</b>	<b>8229</b>	<b>1744 -</b>
<b>FIJO</b>			
Documentos por pagar	5678	8318	2640 +
Acreedores diversos	4210	6137	1927 +
	<b>9888</b>	<b>14455</b>	<b>4567 +</b>
<b>CAPITAL</b>			
Capital social	6050	6050	
Reserva legal	211	271	60 +
Reserva de reinversión		51	51 +
Resultados	291	1036	745 +
	<b>6552</b>	<b>7408</b>	<b>856 +</b>
<b>PASIVO Y CAPITAL</b>			
	<b>26413</b>	<b>30092</b>	<b>3679 +</b>

(GALVEZ, 1992:100)



### 1.3.6 MÉTODO DE TENDENCIAS.

La aplicación de este método trata de contrarrestar las deficiencias que presenta el método de aumentos y disminuciones en el que se comparan solamente dos ejercicios contables, este método permite la comparación de dos o más ejercicios, mostrando de esta forma, las tendencias de una empresa hacia situaciones determinadas, en cuanto a los rubros que se analicen.

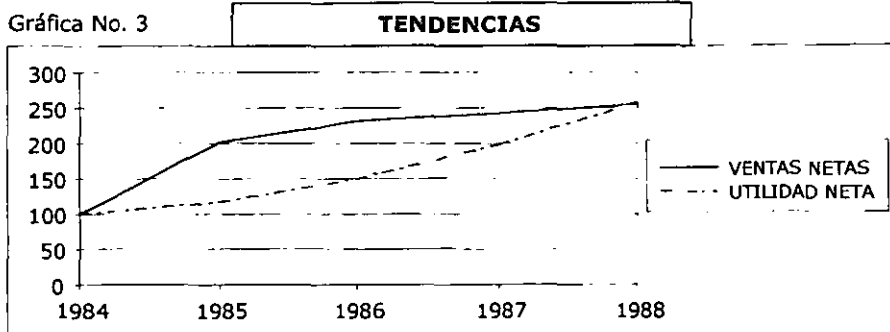
Para la aplicación de este método se requiere de aplicar la estadística y la representación gráfica; ésta último nos proporciona una apreciación más clara y objetiva de los resultados arrojados por este método.

La siguiente tabla ejemplifica los resultados de este método:

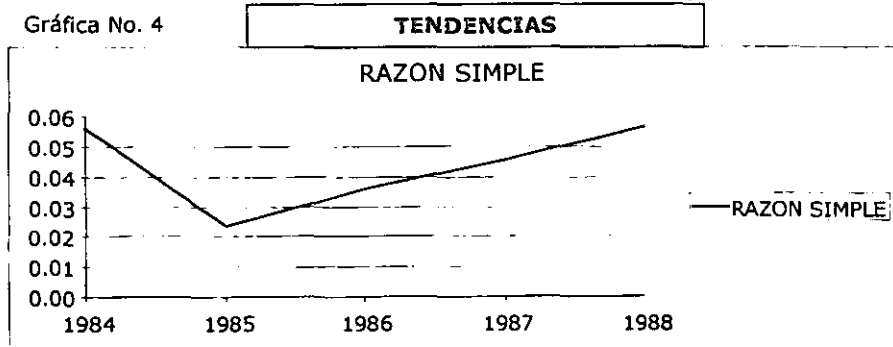
AÑO	VENTAS NETAS		UTILIDAD NETA		RAZON SIMPLE
	VALOR ABSOLUTO	VALOR RELATIVO	VALOR ABSOLUTO	VALOR REALTIVO	
1984	21740	100	1214	100	0.06
1985	43994	202	1034	117	0.02
1986	50322	231	1818	150	0.04
1987	52810	243	2400	198	0.05
1988	55600	256	3150	259	0.06

(GALVEZ, 1992:113)

En la Gráfica No. 3 se muestra la tendencia de las ventas y la utilidad tomando como base los valores relativos de la tabla anterior; y la Gráfica No. 4 muestra la tendencia de la razón simple.



(GALVEZ, 1992:114)



(GALVEZ, 1992:115)

### 1.3.7 MÉTODO DE ORIGEN Y APLICACIÓN DE RECURSOS.

Este estado también es llamado Estado de Cambios en la Situación Financiera, y su objetivo es mostrar los recursos que ha obtenido la empresa y la aplicación de los mismos en un periodo de tiempo determinado, permitiéndonos conocer los cambios que ha sufrido en su situación financiera con motivo de las actividades realizadas.

#### *Origen de recursos*

Se consideran como fuente de recursos los siguientes:

- *Utilidades obtenidas.*- es la utilidad resultante de un ejercicio contable y que se ve reflejada como un aumento del capital.
- *Aumentos en el capital social.*- este aumento se da por nuevas aportaciones al capital por parte de los socios.
- *Aumento del pasivo.*- este aumento se presenta cuando la empresa utiliza recursos ajenos a ella como préstamos, compras a crédito, etc.
- *Disminución del activo.*- este caso se da por la venta de mercancías, por lo que se verá reflejado en la disminución de los inventarios pero puede repercutir en el aumento de otros activos como pueden ser caja o bancos.

## *Aplicación de recursos*

La aplicación de recursos se puede presentar por las siguientes situaciones:

- *Pérdida en los resultados del ejercicio contable.*- pérdidas en los resultados de un ejercicio contable y que se ve reflejado en la disminución del capital.
- *Disminución en el capital social.*- esta disminución puede presentarse por un retiro de capital por parte de los socios o inversionistas de la empresa.
- *Disminución del pasivo.*- este caso se da por el pago a los proveedores de la empresa.
- *Aumento del activo.*- esta disminución puede ser por la salida de recursos para la compra de mercancías, viéndose esto reflejado en un aumento en los inventarios de la empresa.

Para formular el estado de origen y aplicación de recursos se requiere de elaborar un estado comparativo con las características mencionadas en el método de aumentos y disminuciones, como se muestra a continuación.

<b>EMPRESA "X", S.A.</b>			
BALANCE COMPARATIVO DE LOS EJERCICIOS AL 31 DE DICIEMBRE 1985 Y 1986.			
	MILES DE PESOS		DIFERENCIA
	1985	1986	
<b>ACTIVO</b>			
<b>CIRCULANTE</b>			
Caja y bancos	2181	104	2077 -
Inversiones en valores	109	203	94 +
Documentos por cobrar	343	57	286 -
Clientes	12407	14123	1716 +
Inventarios	9225	13780	4555 +
	<b>24265</b>	<b>28267</b>	<b>4002 +</b>
<b>FIJO</b>			
Maquinaria y equipo	442	584	142 +
Equipo de transporte	200	279	79 +
Muebles y enseres	50	37	13 -
	<b>692</b>	<b>900</b>	<b>208 +</b>
<b>OTRO ACTIVO</b>			
	1456	922	534 -
<b>SUMA ACTIVO</b>			
	<b><u>26413</u></b>	<b><u>30089</u></b>	<b><u>3676 +</u></b>
<b>PASIVO</b>			
<b>CIRCULANTE</b>			
Proveedores	1204	1806	602 +
Acreedores diversos	1292	1682	386 +
Documentos por pagar	5872	2518	3354 -
Gastos y otros por pagar	1601	2223	622 +
	<b>9973</b>	<b>8229</b>	<b>1744 -</b>
<b>FIJO</b>			
Documentos por pagar	5678	8318	2640 +
Acreedores diversos	4210	6137	1927 +
	<b>9888</b>	<b>14455</b>	<b>4567 +</b>
<b>CAPITAL</b>			
Capital social	6050	6050	
Reserva legal	211	271	60 +
Reserva de reinversión		51	51 +
Resultados	291	1033	742 +
	<b>6552</b>	<b>7405</b>	<b>853 +</b>
<b>PASIVO Y CAPITAL</b>			
	<b><u>26413</u></b>	<b><u>30089</u></b>	<b><u>3676 +</u></b>

(GALVEZ, 1992:108)

<b>EMPRESA "X", S.A.</b>		
<b>ESTADO DE CAMBIOS EN LA SITUACION FINANCIERA DEL EJERCICIO</b>		
<b>COMPRENDIDO DEL 1 DE ENERO AL 31 DE DICIEMBRE DE 1986.</b>		
<b>ORIGEN DE RECURSOS</b>		
<i>RECURSOS PROPIOS</i>	<b>IMPORTE</b>	<b>PORCENTAJE</b>
Utilidad del ejercicio	742	7
Reserva de inversión	51	1
Reserva legal	60	1
Caja y bancos	2077	21
Documentos por cobrar	286	3
Muebles y enseres	13	
Otros activos	534	5
	<b>3763</b>	<b>38</b>
<i>RECURSOS AJENOS</i>		
Proveedores	602	6
Acreedores diversos	386	4
Gastos y otros por pagar	622	6
Documentos por pagar L.P.	2640	27
Acreedores L.P.	1927	19
	<b>6177</b>	<b>62</b>
<b>TOTAL DE RECURSOS OBTENIDOS</b>	<b>9940</b>	<b>100</b>

(GALVEZ, 1992:109)

<b>APLICACION DE RECURSOS</b>		
<i>AUMENTO DE ACTIVOS</i>	<b>IMPORTE</b>	<b>PORCENTAJE</b>
Inversión en valores	94	1
Clientes	1716	17
Inventarios	4555	46
Maquinaria y equipo	142	1
Equipo de transporte	79	1
	<b>6586</b>	<b>66</b>
<i>DISMINUCION DE PASIVO</i>		
Documentos por pagar	<b>3354</b>	<b>34</b>
<b>TOTAL DE RECURSOS APLICADOS</b>	<b>9940</b>	<b>100</b>

(GALVEZ, 1992:110)

Como pudimos observar en el presente capítulo, el contar con departamento de finanzas es realmente importante para todas las empresas, ya que éste es el encargado de llevar a cabo las mejores estrategias a fin de allegarse de recursos y por otra parte de buscar las mejores opciones que permitan a la empresa realizar una correcta inversión que maximice sus utilidades.

El análisis financiero es una actividad importante, realizada por el departamento de finanzas de una empresa para conocer su situación financiera y de esta forma contar con información veraz, confiable y oportuna, para estar en condiciones de tomar decisiones acertadas sobre el curso de la misma.

Cabe mencionar que los métodos de análisis, así como los estados financieros vistos en este capítulo son sólo algunos de los más utilizados por las empresas, sin embargo su aplicación y utilización será conforme a las necesidades y requerimientos de cada empresa en particular.

## CAPITULO 2

### ***LOS LENGUAJES DE PROGRAMACIÓN***

---

La historia de los lenguajes de programación está ligada íntimamente con el invento de la computadora, por lo que no puede existir uno sin el otro; desde los primeros lenguajes inventados en los años 50's hasta la fecha, la mayoría de los lenguajes se han desarrollado para cubrir ciertas necesidades de sus desarrolladores o inventores, por lo que existen lenguajes orientados a aplicaciones científicas, para aplicación en los negocios, aplicados a cuestiones matemáticas, lenguajes de inteligencia artificial y lenguajes de propósito general.

En este capítulo veremos cómo han ido evolucionando los lenguajes de programación, desde los lenguajes tradicionales hasta los lenguajes orientados a objetos, mencionando algunos aspectos importantes de cada uno de ellos, así como algunas ventajas que presentan dichos lenguajes orientados a objetos en comparación con los estructurados y el propósito para el que fueron desarrollados.

Esto servirá para posteriormente elegir el lenguaje más apropiado, de acuerdo con sus características y las necesidades del proyecto requerido, para su aplicación en el caso práctico de esta investigación.



## **2.1 LOS LENGUAJES DE PROGRAMACIÓN.**

Un lenguaje es un medio a través del cual nos podemos comunicar con otras personas con el objeto de transmitir un mensaje; dicho medio puede ser escrito (utilizando letras o caracteres), oral, a través de señas, gestos, etc.

Un lenguaje de programación consiste en un conjunto de símbolos, caracteres y reglas de uso, que permiten al usuario comunicarse con las computadoras.

Actualmente existe una gran variedad de lenguajes que van desde uso o aplicación especial hasta lenguajes de propósito general. Además, los lenguajes de programación cuentan con un conjunto de instrucciones que permiten realizar las operaciones más comunes de entrada, salida y almacenamiento, mediante programas.

Un programa es una secuencia de pasos que le indican al hardware (computadora), que operaciones debe ejecutar para realizar una tarea específica.

Sin embargo, para que la computadora pueda ejecutar dicha secuencia de instrucciones, primero deberán ser éstas alimentadas al sistema en un lenguaje que pueda ser entendido por dicho sistema.

## **2.2 HISTORIA DE LOS LENGUAJES DE PROGRAMACIÓN.**

Los lenguajes de programación se dividen básicamente en tres grupos, que son:

- Lenguaje máquina.
- Lenguaje ensamblador.
- Lenguajes de alto nivel.

### *Lenguaje Máquina*

Este lenguaje es sin duda el más elemental y el único que las computadoras entienden directamente, y mediante el cual realizan todas sus operaciones, consta de 1 y 0, (también conocido como aritmética binaria); en este caso el programador debe colocar cada instrucción como patrones de unos y ceros. Esto es claramente tedioso y muy susceptible a errores humanos por lo que actualmente casi nadie programa a este nivel.

## *Lenguaje Ensamblador*

Estos lenguajes surgen en la década de los cincuentas con la finalidad de facilitar la tarea de los programadores, desarrollándose códigos nemotécnicos de tres letras, en donde se traducían en códigos binarios los comandos o instrucciones introducidas al sistema por los programadores.

Estos lenguajes tienen grandes ventajas con respecto a los lenguajes máquina, ahorran tiempo, requieren de menos atención a detalles, se tienen menos errores y son más fáciles de modificar.

Aunque el ensamblador es uno de los lenguajes más poderosos en cuanto a velocidad, carece de generalidad, ya que cada microprocesador tiene un conjunto propio de instrucciones que no es compatible con otros procesadores, esto llevo a los programadores a desarrollar nuevos lenguajes de programación que sí permitieran su transportación de una máquina a otra la misma secuencia de instrucciones, los lenguajes de alto nivel.

## *Lenguaje de Alto Nivel*

Estos lenguajes están orientados al usuario por lo que son relativamente fáciles de aprender y de esta forma programar con ellos, además los lenguajes de alto nivel tienen normalmente la característica de transportabilidad, de tal forma que un programa puede ser transportado fácilmente de una máquina a otra.

Generalmente para programar en estos lenguajes se utilizan palabras en inglés como OPEN, PRINT, CLOSE, que equivalen a muchas instrucciones de lenguaje máquina, los comandos se introducen a través del teclado a un programa particular, el cual los analiza y los convierte a instrucciones que puedan ser entendidas por la computadora.

La traducción de instrucciones en este tipo de lenguajes se puede realizar mediante compiladores o intérpretes. Con un compilador, el programa completo es traducido a instrucciones máquina antes de ser ejecutado, de tal forma que es mucho más veloz y eficiente que los intérpretes.

En el caso de los intérpretes cada vez que un programa repite una instrucción, el traductor se ve obligado a re-traducir dicha instrucción a

secuencias de códigos que pueda entender la computadora. Siendo esto un factor que los hace ineficientes en términos de velocidad de ejecución.

En la siguiente tabla se muestra en breve resumen de cómo han ido evolucionando los lenguajes de programación de alto nivel, desde el primer lenguaje comercial de computadoras creado por IBM orientado a aplicaciones científicas llamado Fortran, hasta Pascal diseñado originalmente como herramienta de enseñanza. Por otra parte gracias al interés personal de Ritchie y Thompson (laboratorios Bell), crearon UNIX (sistema operativo), y el lenguaje C, el cual se ha convertido en el estándar de los lenguajes de programación.

También existen lenguajes específicos para determinadas áreas de la ciencia como LISP y PROLOG que son los lenguajes más conocidos para el desarrollo de la Inteligencia Artificial.

Principales lenguajes de alto nivel:

<b>LENGUAJE</b>	<b>ORIGEN DEL NOMBRE</b>	<b>AÑO</b>	<b>USOS/COMENTARIOS</b>
FORTTRAN	Formula Translator	1954	Diseñado para aplicaciones científicas
LISP	List Processing	1956	Lenguaje para Inteligencia Artificial
COBOL	Common Business Oriented Language	1959	Usado en aplicaciones de negocios
ALGOL	Algorithmic Language	1960	Primer lenguaje de programación estructurado

APL	A Programming Language	1961	Intérprete, aplicado a cuestiones matemáticas
PL/1	Programming Language One	1964	Combina características de Cobol, Fortran y Algol
BASIC	Beginners All-Purpose Symbolic Instruction Code	1965	Popular en la década de los 70. Usado por principiantes
LOGO	Derivado de la palabra griega logos	1968	Usado en la enseñanza de programación a niños
FORTH	Fourth Generation Language	1970	Intérprete, útil cuando los recursos de la máquina son limitados
PASCAL	Blaise Pascal	1971	Compilado, estructurado, de propósito general
C	Predecesor del lenguaje B, creado en los laboratorios Bell	1972	Compilado, de propósito múltiple, estructurado, portable
PROLOG	Programming in Logic	1972	Desarrollado para investigación en Inteligencia Artificial
ADA	Augusta ADA Byron	1979	Derivado de Pascal, usado por la milicia
MODULA-2	Modular Language (segunda fase de Pascal)	1980	Enfatiza la programación modular

(LOPEZ, 1995:35)

Debido a los continuos avances en la miniaturización de los circuitos integrados y al avance general día con día de la ciencia, es claro que los investigadores han aumentado no solo la eficiencia y velocidad de los lenguajes de programación, sino que también se han preocupado por hacerlos más amigables para los usuarios, por lo que a partir de la década de los 90s, se ha ido desplazando la programación estructurada para dar paso a un nuevo concepto la Programación Orientada a Objetos (POO), siendo esta actualmente la tendencia de los lenguajes de programación.

El modelo de la programación orientada a objetos difiere del modelo de la programación estructurada, en cuanto a que la programación estructurada se centra básicamente en los datos y procedimientos sin limitaciones, estos datos están estructurados de tal forma que pueden ser manejados por un conjunto de procedimientos independiente y modificable.

A diferencia de la programación tradicional en la programación orientada a objetos, a los programas se les considera como conjuntos de una única entidad básica (objeto), los cuales combinan los datos con los procedimientos que actúan sobre ellos (métodos), además dichos objetos pueden recibir peticiones e interactuar enviando mensajes con los demás.

Otro punto es la organización jerárquica de objetos mediante clases, lo cual permite que los datos y métodos de una clase anterior sean heredados por sus descendientes.

En base a los puntos mencionados anteriormente realmente existe una diferencia entre ambos, ya que los lenguajes orientados a objetos soportan los mecanismos de objetos, clases, métodos, mensajes y herencia, siendo este último uno de los mecanismos más potentes que no puede ser soportado por los lenguajes tradicionales.

## 2.3 HISTORIA DE LOS LENGUAJES ORIENTADOS A OBJETOS.

Al igual que los lenguajes tradicionales, los lenguajes orientados a objetos empiezan a desarrollarse en los años 50's, por lo que algunos de los lenguajes de programación que han contribuido a su evolución son los siguientes:

*LISP (List Processing)*, en los años 50. Lenguaje de inteligencia artificial que maneja el concepto de ligadura dinámica, además de contar con un entorno de desarrollo interactivo.

*SIMULA*, desarrollado en los años 60. Lenguaje para programar simulaciones, contribuyó con el concepto de clase y el de herencia.

*CLU*, en los años 70; desarrollado en el Instituto Tecnológico de Massachusetts (MIT), contribuyó con el manejo de abstracción de datos en forma de tipos abstractos de datos.

*SMALLTALK*, fue el principal proyecto que impulsó el desarrollo comercial de los lenguajes de programación, incluía el concepto de clases y una interfaz gráfica que soportaba la manipulación directa del



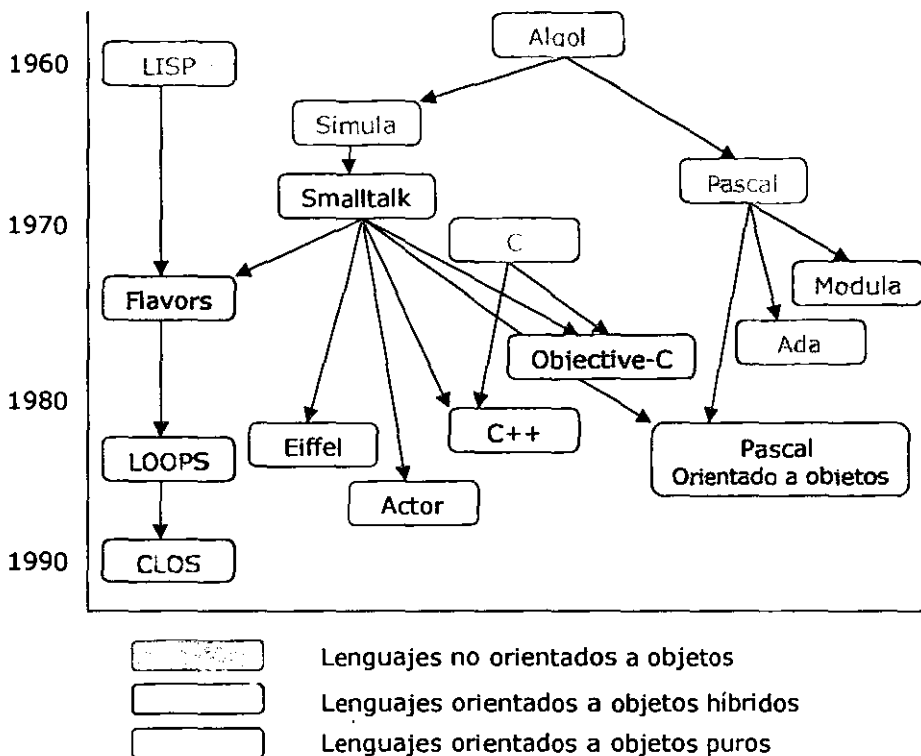
programador, sin embargo no fue sino hasta 1981 en que culminó su lanzamiento como SMALLTALK-80.

C, en los años 80, considerado como un lenguaje extremadamente popular debido a su portabilidad en todas las plataformas, razón por la cual existió una gran atención hacia la programación orientada a objetos y la justificación del mayor empleo de C++.

*PASCAL orientado a objetos*, en 1985 se le incorporaron a Pascal las primeras extensiones orientadas a objetos, y fue desarrollado con velocidad y eficacia como requisitos subyacentes.

En la Figura 2.1 se muestra un breve resumen de la evolución que han tenido los lenguajes orientados a objetos, desde la década de los cincuentas hasta los años noventas, mencionando también los lenguajes que no son orientados a objetos, que lenguajes orientados a objetos son híbridos (el concepto de híbrido constituye un lenguaje que surgió de la combinación de lenguajes que no son orientados a objetos, pero que sin embargo presentan las características de la orientación a objetos) y los lenguajes orientados a objetos puros.

Figura 2.1 Evolución de los lenguajes orientados a objetos.



(Winblad,Edward,King, 1993:62)

Actualmente existen lenguajes de programación orientados a objetos más sofisticados y que ofrecen varias herramientas que permiten al programador generar cada vez programas más complejos, rápidos, seguros y flexibles, ofreciendo entornos más amigables y sencillos para el usuario, como son:

*DELPHI/400* es la única herramienta de desarrollo Windows visual y de alto rendimiento que combina la facilidad y productividad del diseño visual basado en componentes, la potencia del compilador de código original optimizador de 32 bits más rápido del mundo y la flexibilidad de una arquitectura de base de datos abierta y dimensionable en un entorno robusto orientado a los objetos.

Muchas empresas fabricantes de herramientas de desarrollo prometían como solución una herramienta de desarrollo rápido de aplicaciones robusta, rápida, segura y flexible, pero sólo *DELPHI/400* de Borland cumple esas promesas.

*JAVA* es un lenguaje de programación orientado a objetos (OOP) similar al C++, el cual tiene la capacidad de crear aplicaciones que pueden ser colocadas dentro de una página web. Los programas Java consisten exclusivamente de clases y sus métodos. Los requerimientos de Java para la declaración de clases, métodos escritos y el tipo de seguridad hacen la programación más compleja del JavaScript

*BORLAND C++ 5.0* ofrece a los desarrolladores, el C++ más productivo para Windows 95 y NT y proporcionando la mejor solución a los actuales cambios en el campo del desarrollo de aplicaciones; migración a

sistemas operativos de 32 bit, cambio en los estándares industriales y la necesidad de herramientas de desarrollo más rápidas y fáciles de usar.

*MICROSOFT VISUAL BASIC*, a mediados de 1989 apareció un lenguaje que promete una facilidad de uso superior a la de cualquier otro, aunque no tiene el poder de lenguajes como PASCAL y C, permite al usuario desarrollar aplicaciones rápida y eficientemente, se trata de Visual Basic un lenguaje orientado a objetos.

Visual Basic define cada uno de los elementos con los que se construye una interfaz de tal forma que si se desea añadir al programa un cuadrículado, no hay más que colocarlo sobre una forma y especificar lo que se denominan propiedades, estos son algunos descriptores del propio control, tal como su posición en la forma, su nombre, su color, su tamaño, etc., cada uno de los controles, así como las formas reaccionan ante acciones del usuario, es por ello que el lenguaje se considera orientado a objetos.

Visual Basic permite el manejo de bases de datos compatibles con Access, dBase, Foxpro, entre otras, esto se logra mediante un control denominado DATA CONTROL, con el cual se puede acceder a los datos con poca programación.

Microsoft Visual Basic 5.0 es una nueva herramienta de desarrollo de aplicaciones bajo ambiente Windows que brinda una total flexibilidad, requiere de poco conocimiento de programación y se puede adquirir a un costo muy bajo.

En este contexto podemos decir que Visual Basic es una aplicación interactiva para realizar desarrollos muy poderosos permitiendo crear archivos ejecutables bajo Windows y está basado en el diseño y propiedades de diversos objetos dentro de las ventanas, es decir, programación orientada a objetos.

## **2.4 VENTAJAS DE LOS LENGUAJES ORIENTADOS A OBJETOS.**

Algunas de las ventajas más importantes que presentan los lenguajes orientados a objetos en relación con los lenguajes tradicionales son las siguientes:

- Con la programación orientada a objetos los programas tienen menos líneas de código.
- Programar ya no significa solo escribir líneas de código sino desarrollar modelos utilizando clases.

- Ofrece un modelo más natural del mundo real.
- Las bibliotecas de clases predefinidas, son un elemento de los lenguajes orientados a objetos.
- El método general de reducir el código mediante la programación de las diferencias con la herencia.
- Las subclases pueden heredar las estructuras de datos y los métodos de las clases existentes.
- Los lenguajes orientados a objetos benefician al programador soportando el desarrollo del software modular.

## **2.5 CONCEPTOS BÁSICOS DE LA PROGRAMACIÓN ORIENTADA A OBJETOS.**

En la programación orientada a objetos se manejan conceptos que difieren totalmente de los lenguajes tradicionales, los cuales son:

"Clase.- define un conjunto de objetos de tiempo de ejecución, o instancias (modelos), cada uno de los cuales se caracteriza por las

operaciones, llamadas métodos, y el valor de sus datos internos, llamados variables de instancia”.

(Winblad,Edward,King, 1993:69)

*“Objeto.-* contienen un conjunto de procedimientos compartidos llamados métodos. Todos los objetos de la misma clase tienen el mismo conjunto de métodos disponibles para ellos”.

(Winblad,Edward,King, 1993:70)

*“Métodos.-* los métodos de un objeto se invocan solamente con un mensaje adecuado”.

(Winblad,Edward,King, 1993:70)

*“Mensaje.-* el mensaje puede invocar un método para realizar cálculos, acceder y cambiar variables instancia del objeto, o enviar mensajes a otros objetos como parte de su respuesta”.

(Winblad,Edward,King, 1993:70)

*“Herencia.-* permite la capacidad de ampliar y refinar los tipos de datos y la funcionalidad sin duplicar o incluso tener que acceder al código fuente original”.

(Winblad,Edward,King, 1993:71)

"*Polimorfismo*.- los lenguajes orientados a objetos permiten la creación de una jerarquía de objetos con nombre de métodos comunes para operaciones similares conceptualmente, pero que están realizadas de manera diferente para cada una de las clases de la jerarquía. Como consecuencia, cuando diferentes objetos reciben el mismo mensaje, éste puede provocar acciones totalmente distintas".

(Winblad,Edward,King, 1993:73)

"*Ligadura dinámica*.- es el proceso por el cual se proporciona al usuario de una rutina la dirección de la rutina".

(Winblad,Edward,King, 1993:74)

"*Herencia múltiple*.- con la herencia múltiple una subclase puede heredar variables instancia y métodos de más de una clase simple; proporciona, además, un medio de combinar o de unir clases diferentes".

(Winblad,Edward,King, 1993:75)

"*Biblioteca de clases*.- es un conjunto de clases utilizadas para una tarea determinada de programación. Cuanto más coincida una biblioteca de clases con una aplicación, menos modificación se necesita".

(Winblad,Edward,King, 1993:78)



Los avances desde los lenguajes de alto nivel, a finales de los años 50, hasta los modernos lenguajes orientados a objetos, aun están rodeados de un paradigma que asusta al usuario promedio, sin embargo su uso es de aplicación general y actualmente es la tendencia de los lenguajes de programación, por lo que cada día tienen más auge y constantemente los desarrolladores se esfuerzan cada vez más por poner al alcance de los usuarios lenguajes cada vez más sencillos de manejar, pero más poderosos, de tal manera que las aplicaciones cada vez se pueden ser más poderosas y se pueden desarrollar en un menor tiempo de horas de trabajo, pero sin perder la calidad.

Una vez analizados algunos de los lenguajes de programación tanto estructurados como orientados a objetos, podemos observar que éstos últimos permiten manejar algunos conceptos como la herencia, el polimorfismo, mensajes, objetos, métodos, etc., los cuales permiten al programador crear o desarrollar código en menos líneas, más sencillos, permite la reutilización de código, etc., conceptos que no se pueden manejar con los lenguajes estructurados.

Por lo anterior, en el siguiente capítulo se tratará de documentar el análisis y diseño orientado a objetos, tratando de abarcar más a detalle los conceptos de la orientación a objetos.

## **CAPITULO 3**

### ***ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS***

---

Todo desarrollo de sistemas puede estructurarse en forma general en dos componentes principales, el análisis de sistemas y el diseño de sistemas.

El análisis y diseño de sistemas se refiere al proceso de estudiar una situación con la finalidad de mejorarla mediante nuevos métodos y procedimientos, ya sea reemplazando o complementado el ya existente.

En este capítulo estudiaremos algunas de las cualidades que debe reunir el análisis y diseño orientado a objetos, tanto internas como externas, así como algunas de las herramientas, como lo es la ingeniería de software, que ayudan a desarrollar programas más eficientes y eficaces y de mayor calidad, así como lo referente al análisis y diseño orientado a objetos y sus mecanismos básicos.

El análisis orientado a objetos se puede definir como un método que examina los requisitos desde la perspectiva de las clases y objetos que se encuentran en el vocabulario del dominio del problema.

El diseño orientado a objetos es un método que abarca el proceso de descomposición orientado a objetos y una notación para describir los modelos lógico y físico, así como los modelos estáticos y dinámico del sistema que se diseña.

A continuación analizaremos primeramente en que consiste la ingeniería de software y de que manera ésta nos puede auxiliar en el desarrollo de programas.

### **3.1 ANTECEDENTES DE LA INGENIERIA DE SOFTWARE.**

La ingeniería de software surge como una necesidad de escribir mejores programas, además de proveer técnicas y herramientas para el desarrollo de dichos programas.

La ingeniería de software tiene como objetivos los siguientes:

- Tener la certeza de que el código fuente final esté libre de errores.
- Evaluar la semántica (reglas de interpretación).
- Elaboración de programas de mayor calidad y que estén bien estructurados.

- Mayor grado de eficiencia y eficacia en el software, que proporcione resultados veraces.

### **3.2 PRINCIPIOS DE INGENIERIA DE SOFTWARE.**

Los principios de la ingeniería de software consisten en una serie de conceptos o estándares aplicables al desarrollo de programas (análisis y diseño) que debe reunir el software una vez terminado, los cuales son:

*Modularidad.*- consiste en separar un problema por partes, agrupándolas por características similares y darles solución a cada una de ellas.

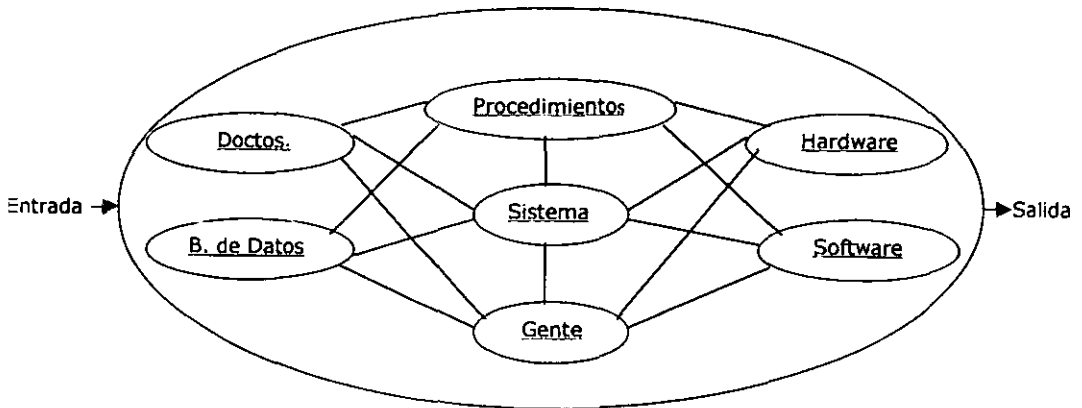
*Abstracción.*- es la capacidad de representar un esquema sin perder la idea de lo que se quiere representar con los requerimientos mínimos o indispensables.

*Anticipación al cambio.*- consiste en prever cambios que se le puedan hacer al sistema y tener así un software de mayor calidad.

*Rigor y formalismo.*- el rigor es la fuerza especial para que se cumplan las normas, (tener un mismo estilo de programación); el formalismo consiste en dar la importancia que requiere el software.

### 3.3 ELEMENTOS DE LA INGENIERÍA DE SOFTWARE.

Los elementos que considera la ingeniería de software para el desarrollo de programas se muestran en la siguiente figura, la cual fue tomada de los apuntes de la materia de Informática IV, impartida por el I.S.C. Manuel López Gallegos, en donde se representa que para llevar a cabo el desarrollo de cualquier sistema deben de relacionarse los procedimientos, los documentos, el hardware, el software, las bases de datos, los sistemas y la gente para alcanzar el software deseado.



### 3.4 CUALIDADES DEL SOFTWARE.

Las cualidades que debe reunir el software en el análisis y diseño orientado a objetos son internas y externas.

### **3.4.1 CUALIDADES EXTERNAS.**

Las cualidades externas son aquellas que le dan la apariencia al programa para su uso final por parte del usuario, y las cuales se consideran más importantes, estas son:

*Desempeño.*- es la actitud de eficiencia que valora el usuario y que se conoce como "performan".

*Interfaz amigable.*- es un punto de comunicación que siempre esta disponible, en la cual no existe una saturación de datos y permite un ambiente agradable para el usuario.

### **3.4.2 CUALIDADES INTERNAS.**

Las cualidades internas son aquellas que tienen que ver con la técnica y estilo de programación, estándares, normas, etc.

*Correctud.*- consiste en si un valor es o no aceptado, la correctud de un programa se mide por la aplicación correcta de las ya establecidas normas de sintaxis y semántica.

*Confiabilidad.*- predecir cuales son las acciones que va a realizar el programa ante determinadas circunstancias. Este punto esta basado en la correctud y exactitud.

*Robustez.*- capacidad de mantenerse confiable y correcto independientemente de los cambios del medio ambiente.

*Verificabilidad.*- corresponde a la capacidad para cuestionar y auditar las operaciones con que trabaja, en comparación con los resultados que obtiene para cerciorarse que son correctos.

*Mantenibilidad.*- que el software se pueda ajustar o modificar de acuerdo a las necesidades, engloba también el concepto de edición: añadir, suprimir, modificar o actualizar, etc.

*Reparabilidad.*- es la capacidad que se tiene para corregir errores en un sistema. La reparabilidad no incluye al mantenimiento.

*Evolutibilidad.*- es la capacidad de adaptación del software con el paso del tiempo.

*Reusabilidad.*- es la capacidad de escribir código que sirva para varias actividades.

*Portabilidad.*- el código fuente debe de estar pensado en forma genérica de tal forma que no tenga cambios cuando se quiera correr en otras plataformas y hardware.

*Entendibilidad.*- tratar de escribir el código de manera sencilla y entendible.

*Productividad.*- lograr producir más con los mismos recursos, lograr que el código sea eficiente con los requerimientos que se necesitan realmente.

*Visibilidad.*- el código fuente debe de tener un orden agradable a la vista, (la indentación es uno de sus elementos).

### **3.5 MECANISMOS BÁSICOS DE LA ORIENTACIÓN A OBJETOS.**

Los conceptos básicos de la orientación a objetos son los objetos, mensajes y los métodos, clases y variables de instancia y herencia. Todos los sistemas que se describen mediante la orientación a objetos contienen estos mecanismos esenciales.



### **3.5.1 OBJETO.**

Es una unidad automática que contiene datos (atributos) y las instrucciones (funciones), que operan sobre dichos datos. Así dentro de los objetos residen los datos de los lenguajes como pueden ser: números, matrices, cadenas de caracteres, registros, así como cualquier función, instrucción o subrutina que opere sobre ellos.

#### *Estado de un objeto.*

El estado de un objeto abarca todas las propiedades (normalmente estáticas) del mismo, más los valores actuales (normalmente dinámicos) de cada una de esas propiedades.

#### *Comportamiento de un objeto*

El comportamiento consiste en como actúa y reacciona un objeto en términos de sus cambios de estados y en paso de mensajes. El comportamiento de un objeto depende básicamente de los métodos del mismo objeto.

#### *Operaciones de un objeto*

Algunos de los servicios u operaciones que puede realizar un objeto son los siguientes:

1. *Modificador*.- es una operación que altera el estado de un objeto, haciendo que cambie de un estado a otro.
2. *Selector*.- es una operación que permite acceder al estado de un objeto, pero sin alterar o modificar su estado.
3. *Iterador*.- es una operación que permite acceder a todas las partes de un objeto llevando un orden preferentemente establecido.
4. *Constructor*.- es la operación que crea un objeto y/o lo puede inicializar si este ya existe.
5. *Destructor*.- es una operación que permite liberar el estado de un objeto y/o lo destruye.

Identidad: la identidad es aquella propiedad de un objeto que lo distingue de todos los demás objetos.

### **3.5.1.1 RELACIÓN ENTRE OBJETOS.**

Los objetos contribuyen al comportamiento de un sistema al colaborar con otros objetos. La relación entre dos objetos abarca las suposiciones que cada uno realiza acerca del otro, incluyendo qué operaciones se pueden realizar y cuál es el comportamiento que se obtiene. Los tipos de relaciones entre objetos de interés en el análisis y diseño orientado a objetos son:

## *Enlaces*

Un objeto colabora con otros objetos a través de sus enlaces con éstos, es decir, un enlace denota la asociación por la cual un objeto utiliza los servicios de otro objeto, o a través de la cual un objeto puede comunicarse con otro.

El paso de mensajes entre dos objetos es generalmente unidireccional, aunque en ocasiones puede ser bidireccional. Un objeto puede desempeñar uno de los tres siguientes papeles como participante de un enlace:

- Actor      Un objeto que puede operar sobre otros objetos, pero no se opera sobre él por parte de otros objetos.
- Servidor    Un objeto que no opera sobre otros objetos, sólo otros objetos operan sobre él.
- Agente     Un objeto que opera sobre otros objetos y además otros objetos pueden operar sobre él.

## *Agregación*

Mientras que los enlaces denotan relaciones igual-a-igual o cliente/servidor, la agregación denota una jerarquía todo/parte, con la capacidad de ir desde el todo hasta sus partes.

### **3.5.2 CLASE.**

Es una descripción genérica de un conjunto de objetos que se pueden representar. Cada clase cuenta con datos y métodos que resumen las características comunes de un conjunto de objetos. La definición de clases significa contar con un depósito común de código reutilizable, con la finalidad de no volver a expresarlo cada vez que se utilice.

Una clase debe poseer una interfaz que puede ser:

1. *Pública*.- la puede acceder cualquiera de los objetos tengan o no relación.
2. *Protegida*.- declaración accesible únicamente dentro de la misma clase.
3. *Privada*.- declaración accesible sólo desde la propia clase y por clases amigas.

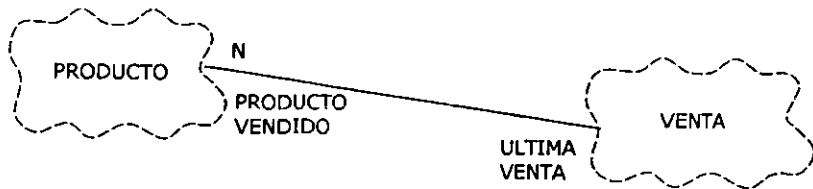
#### **3.5.2.1 RELACIÓN ENTRE CLASES.**

Al igual que los objetos, las clases no pueden existir aisladamente. En los lenguajes de programación existen varios enfoques para establecer relaciones de generalización/especialización, todo/parte y asociación. En

particular la mayoría de los lenguajes orientados a objetos dan soporte para alguna combinación de las siguientes relaciones:

### *Asociación*

Dependencia semántica, interpretación que se da a las ligas de los objetos. En la siguiente figura se muestra una asociación simple entre la clase producto y la clase venta, en donde la clase producto refleja los productos que se venden como parte de una venta y la clase venta muestra la transacción por la cual varios productos acaban de venderse, de acuerdo a lo anterior, vemos que también existe una comunicación bidireccional entre ambas clases.



(BOOCH,1996:125)

### *Cardinalidad*

Es la relación de comportamiento bidireccional desde un objeto a otro. La cardinalidad entre clases puede ser:

- Uno a uno
- Uno a muchos
- Muchos a muchos

### *Herencia*

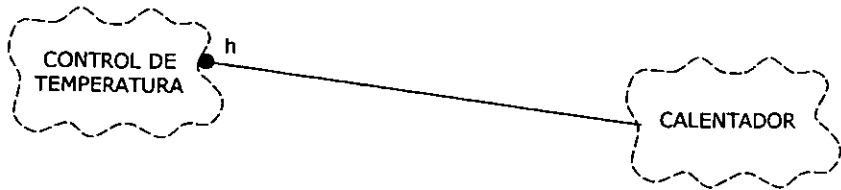
Es un mecanismo que se transmite bajo ciertas características a otro elemento. La herencia permite compartir automáticamente métodos y datos entre metaclasses, clases y objetos, siendo esta herencia en un solo sentido. La herencia puede ser :

- Simple: cuando un objeto puede heredar datos y métodos de una clase.
- Múltiple: se refiere a la posibilidad de que un objeto pueda adquirir los datos y métodos de más de una clase, como:

metaclassa —→ clase —→ objeto.

### *Agregación:*

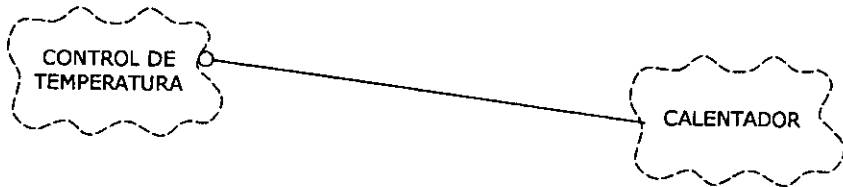
Es cuando se realiza una unión entre un par de clases, permitiendo esa unión sumar las estructuras de las clases. En la siguiente figura la clase ControlTemperatura muestra el todo y una de sus partes es una instancia de la clase Calentador.



(BOOCH,1996:147)

### *Uso*

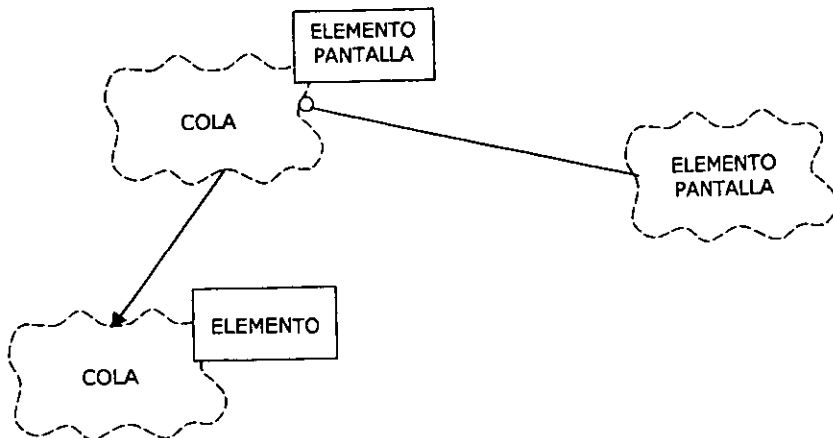
Es cuando se obtienen funciones específicas de otra clase. En la siguiente figura la clase ControlTemperatura usa los servicios de la clase Calentador.



(BOOCH,1996:149)

### *Instanciación*

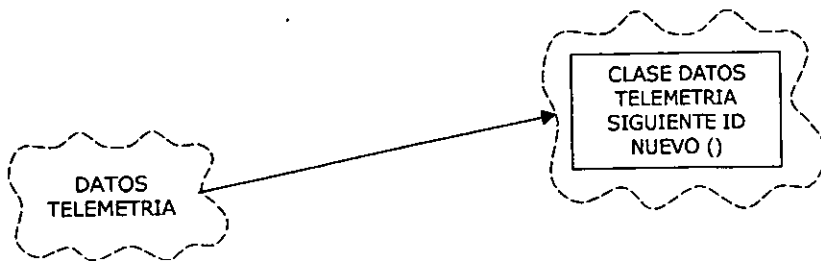
Creación de instancias, tomar las características de una clase. La siguiente figura muestra las relaciones entre la clase parametrizada Cola, su instanciación para ElementoPantalla y su instancia correspondiente ColaElemento.



(BOOCH,1996:151)

*Metaclase*

Es una clase de clases, es decir, es una clase cuyas instancias son, ellas mismas, clases.



(BOOCH,1996:153)



En la figura anterior observamos que se define una variable de clase `SiguienteId` para la metacase `DatosTelemetría`.

### **3.5.3 MENSAJES Y MÉTODOS.**

En la orientación a objetos, los objetos tienen la posibilidad de actuar, esto es cuando reciben un mensaje, que es, una solicitud que pide al objeto que se comporte de alguna forma; cuando estos programas son ejecutados los objetos reciben, interpretan y responden a mensajes procedentes de otros objetos. El conjunto de mensajes al que un objeto puede responder se llama protocolo del objeto.

Los métodos residen en los objetos y determinan como debe actuar el objeto cuando recibe un mensaje y las variables de instancia almacenan información o datos locales en el objeto, de tal forma que los métodos se ejecutan en respuesta a los mensajes y manipulan los valores de las variables de instancia.

Los mensajes que recibe el objeto son el unico medio que conectan al objeto con el mundo exterior y solamente éste puede disponer de los datos del interior del mismo para su manipulación. De acuerdo a lo anterior la orientación a objetos fomenta la modularidad haciendo muy

claras las fronteras entre objetos, aclara la comunicación entre ellos mismos y oculta los detalles de la realización.

Finalmente cuando se ejecuta un programa orientado a objetos, primero se crean los objetos cuando se necesitan; segundo, los mensajes se mueven de un objeto a otro a medida que se procesa información y finalmente, los objetos se borran cuando ya no son necesarios y se recupera memoria.

### **3.6 LA NOTACIÓN.**

La notación consiste en un medio para capturar razonamientos acerca del comportamiento y la arquitectura de un sistema. A continuación veremos la notación propuesta por Grady Booch, para representar los escenarios.

#### **3.6.1 DIAGRAMA DE CLASES.**

El diagrama de clases es utilizado para mostrar la existencia de clases y sus relaciones en la visión lógica de un sistema. Los elementos esenciales para este diagrama son las clases y sus relaciones básicas.

*Clases.* La siguiente figura muestra el icono utilizado para representar una clase en un diagrama de clases.







(BOOCH,1996:205)

En ciertos diagramas de clases es necesario exponer algunos de los atributos y operaciones asociados con una clase, ya que representan una visión abreviada de la especificación completa de la clase, que sirve como punto de declaración para todos sus miembros. Según la siguiente sintaxis un atributo puede tener un nombre, una clase o ambos, y opcional un valor por defecto.

- A nombre de atributo solamente
- : C clase de atributo solamente
- A : C nombre y clase del atributo
- A : C = E nombre, clase y valor por omisión del atributo

*Relaciones entre clases.* Las conexiones esenciales entre clases incluyen las relaciones de asociación, herencia, posesión y uso cuyos iconos se muestran a continuación:

	Asociación
	Herencia
	Posesión
	Uso

Cada uno de ellos puede contener una etiqueta textual que documente el nombre de la relación o sugiere su propósito, los nombres deben de ser únicos dentro de su contexto. La asociación se conecta mediante expresiones sustantivas (empleo) denotando la naturaleza de la relación.

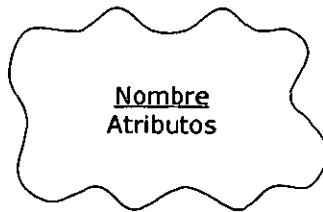
Adicionalmente se puede expresar la cardinalidad de las asociación utilizando la siguiente sintaxis:

- 1 Indica que sólo está asociada a un elemento, hereda, tiene o usa un elemento
- N Representa un número ilimitado, incluyendo el 0
- 0..N Usa desde 0 hasta ilimitado
- 1..N Usa desde 1 hasta ilimitado
- 0..1 Indica 0 o 1
- 3..7 Indica un rango específico
- 1..5,8 Indica un rango específico o un número exacto

### 3.6.2 DIAGRAMA DE OBJETOS.

Los diagramas de objetos sirven para representar la existencia de objetos y sus relaciones en el diseño lógico de un sistema. Los dos elementos esenciales en este tipo de diagramas son los objetos y sus relaciones.

*Objetos.* La siguiente figura muestra el icono para representar un objeto.



(BOOCH,1996:239)

El nombre de un objeto sigue la sintaxis de los atributos y puede escribirse en cualquiera de las 3 formas siguientes:

- A Solamente nombre del objeto
- : C Solamente nombre de la clase
- C : A Nombre y clase del objeto

*Relaciones entre objetos.* Los objetos interactúan a través de sus enlaces con otros objetos y se representa mediante la siguiente figura:

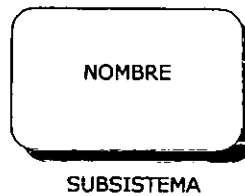
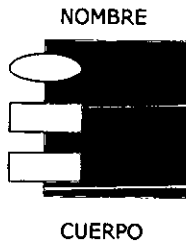
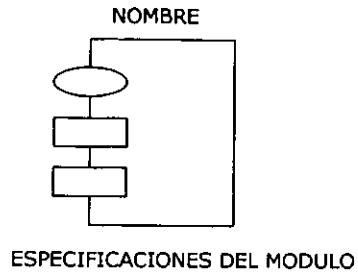
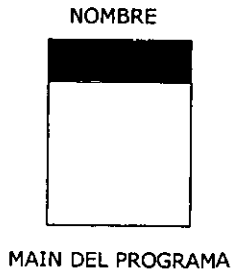
Un enlace es una instancia de una asociación. Puede existir un enlace entre objetos sólo si existe una asociación entre sus clases correspondientes. Como se muestra a continuación un enlace se puede adornar con una serie de mensajes, cada mensaje consta de los siguientes tres elementos:

- D Símbolo de sincronización que denota la dirección de la invocación
- M Invocación de operación o despacho de evento
- S Opcionalmente, un número de secuencia

### **3.6.3 DIAGRAMA DE MÓDULOS.**

Este tipo de diagrama sirve para mostrar la asignación de clases y objetos a módulos en el diseño físico de un sistema. Un solo diagrama representa una vista de la estructura de módulos de un sistema. Los dos elementos esenciales en este tipo de diagrama son los módulos y sus dependencias.

*Módulo.* Son iconos para representar varios tipos de módulos. Los tres primeros denotan archivos, distinguidos por su función, el cuarto denota un archivo que contiene la raíz del programa.



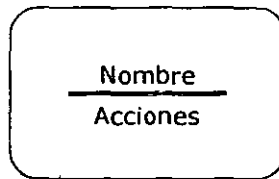
(BOOCH,1996:245)

*Dependencia.* La relación existente entre dos módulos es una dependencia de compilación, representado por una línea que apunta al módulo respecto al cual existe la dependencia.

### 3.6.4 DIAGRAMA DE ESTADOS DE TRANSICIÓN.

Es utilizado para mostrar la situación de actividad de una clase dada, teniendo un evento que causa la transición de un estado a otro, teniendo como resultado un cambio de estado. Los elementos esenciales en este tipo de diagrama son los estados y las transiciones entre estados.

*Estados.* El estado de un objeto representa los resultados acumulados de su comportamiento, la siguiente figura muestra el icono que se utiliza para representar un estado específico:

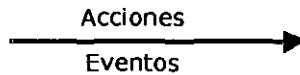


(BOOCH,1996:231)

Se requiere un nombre para cada estado, todo nombre de estado debe ser único en el ámbito que lo encierra. Para ciertos estados es útil exponer las acciones asociadas a ellos.



*Transición de estados.* Un evento es algún suceso que puede causar un cambio de estado en un sistema, a esto se le llama transición de estados y se representa con el siguiente icono:



Los mecanismos mencionados anteriormente conforman la base de lo que es el paradigma orientado a objetos, sin embargo existen además otros conceptos que resumen las ventajas de este método de orientación a objetos, que son:

*Encapsulación:* se le denomina de esta forma al conjunto de métodos y datos dentro de un objeto, de tal forma que el acceso a esos datos solamente se puede llevar a cabo a través de los métodos del objeto.

*Abstracción:* consiste en una buena identificación y definición de funciones para crear programas con funciones estandarizadas que ejecuten tareas específicas.

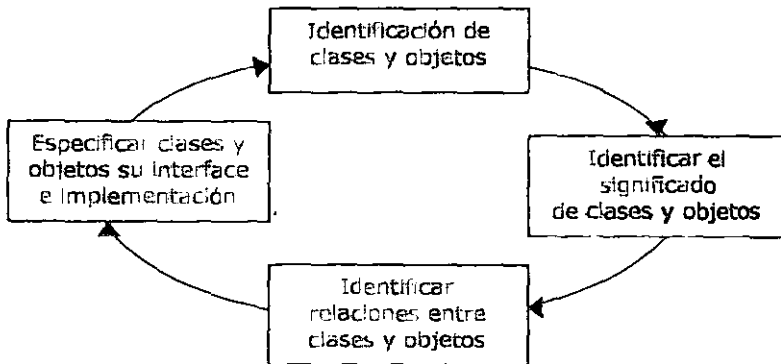
*Polimorfismo:* es la capacidad de relacionar objetos de clase diferentes mediante la herencia, capaces de responder de forma distinta a una misma llamada de atención.

ESTA TERCERA PARTE  
DE LA BIBLIOTECA.

*Persistencia:* consiste en que todo objeto debe de tener la capacidad para extraer de sí mismo su contenido, a partir de un flujo de datos de entrada generalizados y también de insertar su propio contenido en el flujo de datos de salida generalizado.

### 3.7 MICROPROCESO PARA EL DESARROLLO (ANÁLISIS Y DISEÑO).

La siguiente figura muestra un modelo con una serie de pasos a seguir para realizar el análisis y diseño orientado a objetos y tiende a definir las siguientes actividades:



(BOOCH,1996:269)

- Identificar las clases u objetos a un nivel dado de abstracción.
- Identificar la semántica de las clases y los objetos, significado de cada uno de ellos (servicios que va a brindar esa clase, metodos, etc.)

- Identificar las relaciones entre clases y objetos (especificar herencia, jerarquía, etc.)
- Definición abstracta del código (servicios) que van a representar las características.

### **3.7.1 IDENTIFICAR CLASES Y OBJETOS.**

EL proposito de identificar clases y objetos es establecer las bondades del problema que se maneja. Es el primer paso en la visualización de la descomposición de un sistema en desarrollo bajo la metodología O.O.

Descubrir las abstracciones que son propias del dominio del problema, cuales van a formar parte de la representación y cuales no.

En esta etapa se genera un diccionario que contiene nombre de los objetos, sus métodos, nombre de la clase, atributos y métodos de cada una de las clases, este diccionario también sirve como referencia para las demas clases.

Características para identificar el dominio de un problema.

- Construcción de un modelo genérico del dominio mediante la consulta a expertos del dominio del problema.
- Examinar sistemas existentes dentro del dominio del sistema y representar su entendimiento en un formato comun.
- Identificar similitudes y diferencias entre los sistemas mediante la consulta del dominio a expertos del sistema.
- Refinar un modelo genérico para acoplar los modelos existentes.

#### Abstracción de clases y objetos:

- Los objetos deben ser nombrados con frases objetivas.
- Las clases deben ser nombradas con adjetivos comunes.
- Las operaciones de modificación deben ser nombradas con verbos activos.
- Las operaciones de selección deben implicar una consulta y deben utilizar verbos que involucren la forma ser o estar.
- Utilizar una notación estandar para los nombres y que sean autoconsistentes.
- Identificación del mecanismo entrada, proceso, salida.

### **3.7.2 IDENTIFICAR LA SEMÁNTICA DE CLASES Y OBJETOS.**

Identificar la semántica de las clases y objetos para establecer el comportamiento y atributos de cada abstracción identificados en el paso anterior y se definen las responsabilidades de cada uno de los elementos.

Las responsabilidades de cada elemento se añaden al diccionario de datos.

### **3.7.3 IDENTIFICAR LAS RELACIONES ENTRE CLASES Y OBJETOS.**

Solidificar las bondades y el reconocimiento de los colaboradores con cada una de las abstracciones definidas, el producto de este punto nos dará el Diagrama de clases, Diagrama de objetos y Diagrama de módulos.

La descripción de los escenario consiste en identificar dentro del diseño:

- Las clases existentes y sus relaciones entre ellas.
- Los mecanismos que hacen que las clases y objetos colaboren.
- Donde deben ser declaradas las clases y los objetos.

Los escenarios se construyen a partir de cuatro tipos de diagramas:

- Diagramas de clases.
- Diagramas de objetos.
- Diagramas de módulos.
- Diagramas de de estados de transición.

### **3.7.4 IMPLEMENTACIÓN DE CLASES Y OBJETOS.**

Proveer de un refinamiento de abstracciones existentes para desarrollar la abstracción de la implementación de las clases, objetos, semánticas y relaciones. Producto de esta fase: código del programa.

Como se pudo observar en este capítulo el análisis y diseño orientado a objetos ofrece bastantes ventajas que los sistemas estructurados, suele producir menos líneas de código, más reutilizable y tiende a producir sistemas más flexibles a los cambios, fijando su atención en los elementos más estables del sistema que son los objetos; aunque actualmente no existe un método de diseño orientado a objetos que sea ampliamente aceptado, si existe una clara aceptación en la reutilización, modularidad y encapsulación, así como una fuerte combinación entre

diseño y código, los cuales forman la parte integral de los sistemas orientados a objetos.

Por lo tanto, el propósito del análisis es proveer una descripción general del problema, siendo este completo, consistente y leible, además de que debe ser revisable (sujeto a cambios).

En lo que respecta al propósito del diseño, consiste en crear una arquitectura que pueda ser implementada estableciendo tácticas y políticas que deben ser usadas por los diversos elementos del sistema.

Por último, cabe mencionar que el método orientado a objetos es más intuitivo, no solo para el analista sino también para el usuario final, ya que su enfoque de organización y comprensión esta basado en la forma de pensar de los seres humanos.

## **CAPITULO 4**

### **CASO PRÁCTICO**

---

En este capítulo abarcaremos todos los temas vistos con anterioridad con el propósito de elaborar un análisis y diseño adecuado, para crear el prototipo de un programa que elabore los análisis financieros de manera automatizada, rápida y eficientemente.

#### **4.1 METODOLOGÍA.**

La metodología empleada en este trabajo de investigación es el explicativo, el cual está enfocado a responder a las causas de los eventos físicos o sociales y se centra en explicar el porqué ocurre un fenómeno y en qué condiciones se presenta, o porqué dos o más variables están relacionadas; la elección de este método es debido a que no se llegará hasta la implantación del sistema, sino que se describirá todo el proceso para llegar al diseño mas apropiado de un software acorde a las necesidades de la institución en estudio.

Para el levantamiento de información se realizaron varias entrevistas con el personal involucrado en el desarrollo del sistema, debido a que el número de personas no era muy extenso; en dichas entrevistas se



estableció el identificar claramente las necesidades del departamento de contabilidad de la empresa en estudio, para la elaboración de sus análisis financieros, que documentos de entrada se requerían para alimentar al sistema, los métodos de análisis financieros utilizados, los cuales se aplicarían a los estados financieros y por último la forma de los reportes, en cuanto a información, diseño, distribución, etc.

#### **4.2 DATOS GENERALES DE LA EMPRESA.**

A continuación hablaremos sobre la empresa, su giro, el departamento en estudio y el planteamiento del problema.

Datos generales de la empresa:

Nombre: Promotora y Constructora Zirahuén S.A. de C.V.

Domicilio: Salazar No. 11-C, Col. La Magdalena

Ciudad: Uruapan, Michoacán

Giro: Construcción de obra civil, eléctrica y electromecánica

#### **4.3 PLANTEAMIENTO DEL PROBLEMA.**

Área: Departamento de Contabilidad

La empresa en estudio requiere de la automatización de algunos de los procedimientos o actividades que realiza con más frecuencia en su departamento de Contabilidad; el procedimiento sujeto a estudio en éste trabajo de investigación consiste en agilizar la elaboración de los análisis financieros, el cual consiste en la separación y estudio de los rubros que conforman un estado financiero, a fin de conocer la situación financiera de una empresa para la toma de decisiones, requiriendo también almacenar la elaboración de cada uno de los reportes y poder contar con los mismos en cualquier momento de manera fácil y rápida.

Dicho procedimiento actualmente se realiza manualmente y es realizado por una persona, dedicándole gran tiempo para su realización, por lo que se requiere de elaborar un programa hecho a la medida para solucionar este problema; que permita la elaboración y realización de dichos reportes, logrando con ello que los cálculos se realicen de manera rápida, confiable y automatizada, y poder contar con reportes veraces en el momento en que se requieren.

La empresa actualmente cuenta con una computadora Pentium a 233 Mhz de velocidad, con 24 Mb de memoria en Ram y 3.1 Gb de espacio en Disco Duro y diversa paquetería como Windows 95, Office 97, Corel 7.0, Norton Antivirus, entre otros.

#### **4.3.1 CARACTERÍSTICAS DEL PROGRAMA.**

Por lo expuesto anteriormente, se requiere de un programa que tenga las siguientes características:

- Debe contar con una interfaz amigable para el usuario.
- Contar con una Base de Datos para llevar un registro histórico.
- Contar con un buen nivel de seguridad.
- Trabajar en ambiente gráfico.
- Realizar consultas de los registros.
- Tener los formatos de los estados financieros más utilizados, con la alternativa de poder agregar o modificarlos.
- Contar con los métodos de análisis más utilizados por la empresa, a fin de que realice sus análisis financieros.

#### **4.4 JUSTIFICACIÓN.**

##### **4.4.1 LENGUAJE DE PROGRAMACIÓN.**

De acuerdo a los puntos mencionados anteriormente y a lo expuesto en el capítulo dos de esta investigación, el lenguaje de programación que se empleará en la realización del prototipo será el lenguaje Borland C++

en su versión 5.0, ya que es un lenguaje orientado a objetos y permite el manejo de los conceptos orientados a objetos, como los son las clases, los objetos, la herencia, el polimorfismo, los mensajes, las instancias, los métodos, la reutilización de código, etc., presentando también gran facilidad para crear interfaces amigables, crear y manejar bases de datos, diseñar y modificar formatos o pantallas acorde a las necesidades requeridas, además de ser un lenguaje de programación muy utilizado actualmente. También existe bibliografía disponible en donde apoyarse, permitiendo con ello abarcar las necesidades requeridas para el desarrollo del programa en cuestión.

#### **4.4.2 AUTOR (METODOLOGÍA).**

Para la realización de este trabajo se tomará como referencia la metodología propuesta por los autores Ann L. Winblad, Samuel D. Edwards y David R. King, de su libro *Software Orientado a Objetos*, así como la metodología de Grady Booch, de su libro *Análisis y Diseño Orientado a Objetos con Aplicaciones*, lo cual nos permitirá aprovechar las metodologías propuestas por ambos autores, tomando lo necesario de cada uno de ellos para realizar este trabajo, permitiendo con ello abarcar todos los conceptos vistos con anterioridad de la orientación a objetos, además de que ambos autores manejan una simbología fácil y sencilla de

comprender pero muy completa y poderosa para llevar a cabo el análisis y diseño, y finalmente ver reflejado el prototipo objeto de este trabajo de investigación.

## **4.5 ANÁLISIS Y DISEÑO**

### **4.5.1 DEFINICIÓN DEL PROBLEMA Y DESCRIPCIÓN DE LA SOLUCIÓN.**

Primeramente se debe comenzar por definir el problema y una descripción de la solución.

Definición del problema: elaborar un programa para el calculo y elaboración de análisis financieros.

Descripción de la solución: el sistema para la elaboración de análisis financieros permite a los usuarios crear reportes. Los reportes creados se pueden modificar, también se pueden visualizar o imprimir. El programa permite capturar estados financieros, modificar los valores de los estados financieros, aplicar sobre dichos valores, diversos métodos de análisis, también permite visualizar o imprimir los estados financieros.

#### 4.5.2 IDENTIFICACIÓN DE OBJETOS.

El siguiente paso consiste en identificar los posibles objetos, subrayando los sustantivos.

El sistema para la elaboración de análisis financieros permite a los usuarios crear reportes. Los reportes creados se pueden modificar, también se pueden visualizar o imprimir. El programas permite capturar los valores de los estados financieros, modificar los valores de los estados financieros, aplicar sobre dichos valores, diversos métodos de análisis, también permite visualizar o imprimir los estados financieros.

De acuerdo a lo anterior, reportes y estados financieros parecen ser objetos. Una vez identificados los objetos se deben de asociar los atributos con cada uno de ellos, como se muestra a continuación:

<b>Objeto</b>	<b>Atributos</b>
Reportes	se puede crear
	se puede modificar
	se puede visualizar
	se puede imprimir

Estado financiero	se pueden capturar
	se pueden modificar
	se pueden procesar
	se pueden visualizar
	se pueden imprimir

#### 4.5.3 IDENTIFICACIÓN DE MÉTODOS.

El paso siguiente consiste en identificar los posibles métodos, subrayando los verbos.

El sistema para la elaboración de análisis financieros permite a los usuarios crear reportes. Los reportes creados se pueden modificar, también se pueden visualizar o imprimir. El programas permite capturar estados financieros, modificar los valores de los estados financieros, aplicar sobre dichos valores, diversos métodos de análisis, también permite visualizar o imprimir los estados financieros.

<b>Objeto</b>	<b>Método</b>	
Reporte	crear	modificar
	visualizar	imprimir

Estado financiero	capturar	modificar
	procesar	visualizar
	imprimir	

#### **4.5.4 DEFINICIÓN DE INTERFACES ENTRE OBJETOS.**

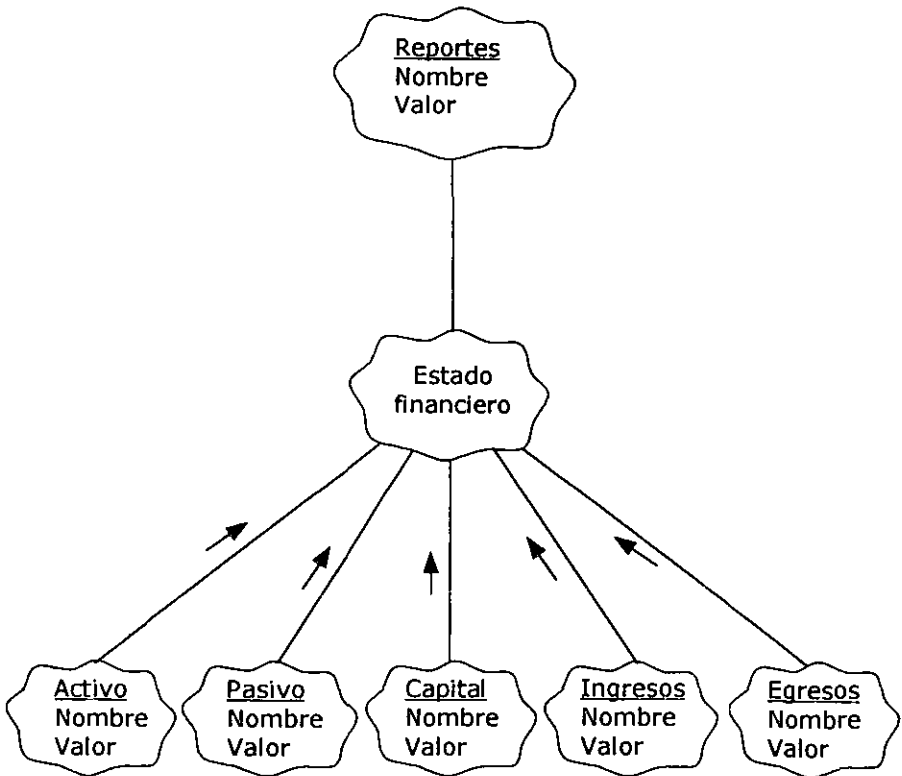
La definición de las interfaces entre objetos es el proceso final, y una vez más se crea una descripción escrita.

La clase reporte contiene una variable instancia llamada reporte (su mismo nombre) y los métodos siguientes: crear, modificar, visualizar e imprimir. La clase estado financiero contiene una variable instancia llamada estado financiero (su mismo nombre) y los métodos capturar, modificar, procesar, visualizar e imprimir.

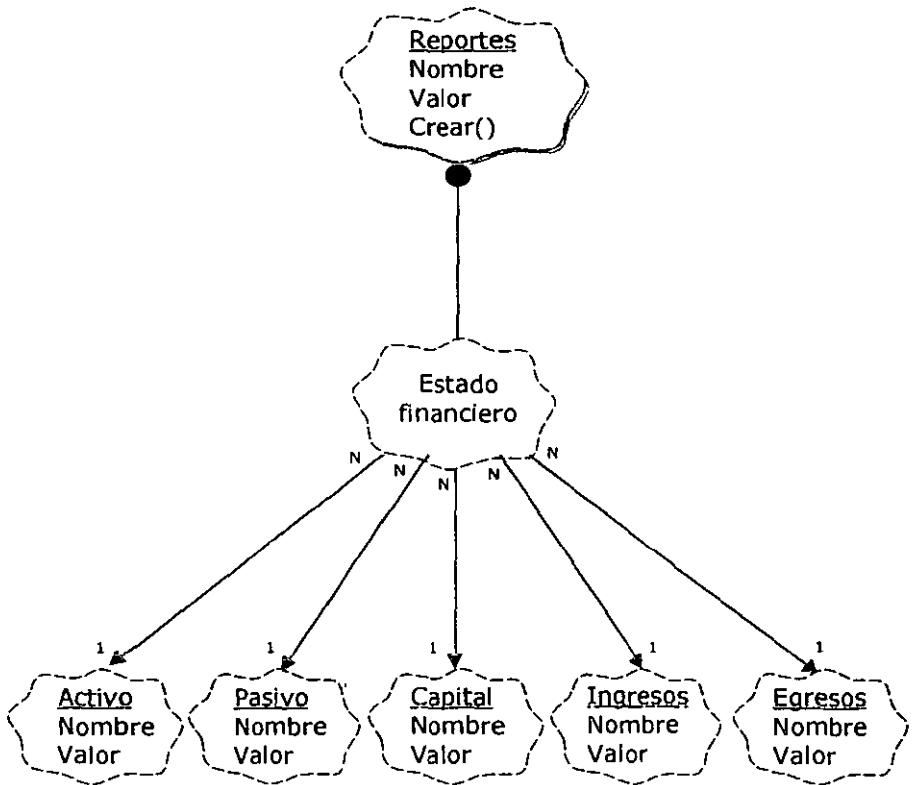
A continuación en base a la identificación y definición de clase y objetos, se procederá a construir los diferentes escenarios a través de diagramas que representen físicamente las relaciones de objetos, las relaciones de clases, el estado de transición y por último el diagrama de módulos.



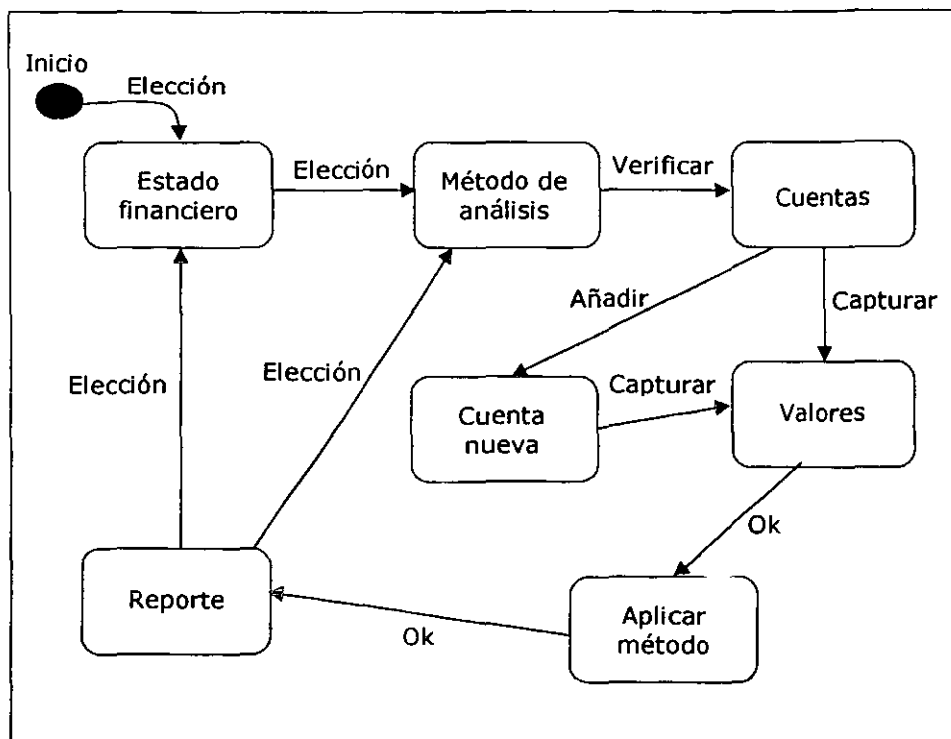
#### 4.5.5 DIAGRAMA DE OBJETOS.



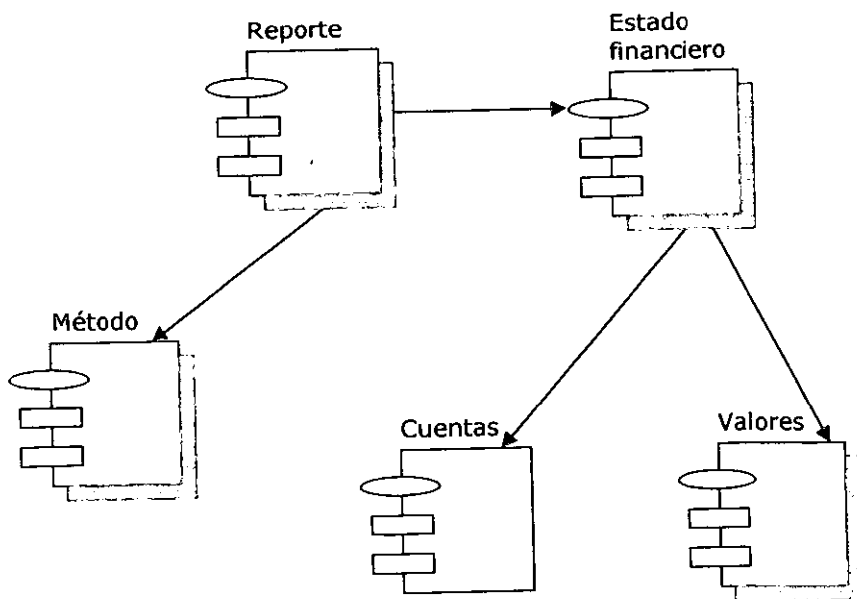
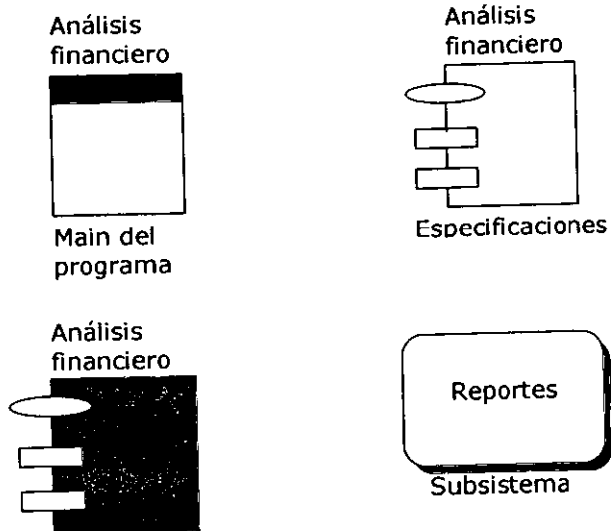
#### 4.5.6 DIAGRAMA DE CLASES.



#### 4.5.7 DIAGRAMA DE ESTADOS DE TRANSICIÓN.



#### 4.5.8 DIAGRAMA DE MÓDULOS.



#### 4.5.9 DICCIONARIO DE DATOS.

<b>Nombre</b>	Reportes	Estado Financiero	Activo
<b>Categoría</b>	Metaclase	Clase	Subclase
<b>Pertenece a</b>		Metaclase Reportes	Clase Estado financiero
<b>Atributos</b>	Nombre Valor	Ninguno	Nombre Valor
<b>Parámetros</b>	Ninguno	Ninguno	Ninguno
<b>Descripción</b>	Reportes elaborados en base al método elegido	Método aplicable a un estado financiero	Entrada de datos

<b>Nombre</b>	Activo	Pasivo	Capital
<b>Categoría</b>	Subclase	Subclase	Subclase
<b>Pertenece a</b>	Clase Estado financiero	Clase Estado financiero	Clase Estado financiero
<b>Atributos</b>	Nombre Valor	Nombre Valor	Nombre Valor
<b>Parámetros</b>	Ninguno	Ninguno	Ninguno
<b>Descripción</b>	Entrada de datos	Entrada de datos	Entrada de datos

<b>Nombre</b>	Ingresos	Egresos
<b>Categoría</b>	Subclase	Subclase
<b>Pertenece a</b>	Clase Estado financiero	Clase Estado financiero
<b>Atributos</b>	Nombre Valor	Nombre Valor
<b>Parámetros</b>	Ninguno	Ninguno
<b>Descripción</b>	Entrada de datos	Entrada de datos

Una vez analizados los conceptos anteriores y determinado el análisis y diseño orientado a objetos, vemos que definitivamente este paradigma ofrece herramientas más sencillas pero más poderosas en el desarrollo de sistemas.

Habiendo determinado los objetos y sus métodos, así como las clases y sus atributos resulta más sencillo plasmar estos conceptos en un lenguaje de programación.

## CONCLUSIONES

El análisis de estados financieros es una actividad sumamente importante que realizan las empresas, no sólo para que éstas conozcan su situación financiera, si no además para que las demás empresas a fines, conozcan también dicha situación financiera a fin de tomar decisiones acertadas sobre las medidas de acción que decidirán el rumbo de cada empresa en particular; una vez elaborada la presente investigación en donde primeramente se identificó el problema de la institución, documentando para ello toda la información que la empresa requería para elaborar sus análisis financieros, pasando posteriormente a hablar un poco sobre la historia de los lenguajes de programación, tanto estructurados como orientados a objetos, resaltando en cada uno de ellos, la finalidad para la cual fueron creados y algunas de las ventajas que presentan los lenguajes orientados a objetos en comparación con los estructurados, el siguiente paso consistió en hablar sobre los que es el análisis y diseño orientado a objetos, así como los conceptos que maneja este paradigma, así como también sobre la ingeniería de software, la cual es una herramienta que nos ayuda a elaborar un software eficiente y de mejor calidad, finalizando con un caso práctico en donde se trato de aplicar toda la información recabada en el presente trabajo y los conocimientos adquiridos en mi formación como estudiante.

El objetivo de la presente investigación era el de determinar cada una de las necesidades para realizar un análisis y diseño y desarrollar un programa a la medida para el departamento de contabilidad de la empresa Promotora y Constructora Zirahuen, definiendo un método y paradigma para llevar a cabo el desarrollo de dicho programa; logrando con esta investigación cubrir con los objetivos planteados en un principio, sin embargo se pudo observar que realmente sí era necesario haber desarrollado un programa a la medida ya que la velocidad con que se elaboraban anteriormente los reportes, eran muy lento y muy tedioso, trayendo como consecuencia pérdidas para la organización, ya que para la toma de decisiones se necesita contar con información confiable, veraz y oportuna, y no se cumplía con el sistema que manejaban actualmente, sin embargo con el desarrollo de este programa podemos asegurar una acertada y confiable toma de decisiones señalen el rumbo correcto de la empresa.

En la actualidad muchas de las empresas requieren de programas hechos a la medida y acorde a sus necesidades, debido a que la paquetería que se maneja comercialmente en ocasiones no cubre con dichas necesidades, por lo que, el desarrollo de sistemas todavía sigue siendo un campo no muy explotado en esta ciudad de Uruapan.



## BIBLIOGRAFÍA

Booch, Grady, Análisis y Diseño a Objetos con Aplicaciones, Addison-Wesley / Días de Santos, México, 1996.

Calvo Langarica, Cesar, Análisis e Interpretación de Estados Financieros, PAC, S.A. de C.V, México, 1993.

Galvez Azcanio, Ezequiel, Análisis de Estados Financieros e Interpretación de sus Resultados, Ecasa, 1992.

García Mendoza, Alberto, Análisis e Interpretación de la Información Financiera, Continental, S.A. de C.V., México, 1990.

Kennedy Ralph, Dale McMullen, Stewar Yarwood, Coaut, Estados Financieros: Forma, Análisis e Interpretación, Limusa S.A. de C.V., México, 1993.

Moreno Fernández, Joaquín, Las Finanzas en la Empresa: Información, Análisis, Recursos y Planeación, I.M.C.P.A.C, México, 1995.

Pressman, Roger S., Ingeniería de Software: Un Enfoque Práctico, Mc Graw-Hill, México, 1995.

Tucker, Allen B., Lenguajes de Programación, Mc Graw-Hill, México, 1990.

Winblad, Ann L., et.al., Software Orientado a Oobjetos, Addison-Wesley/Díaz de Santos, Wilmington, Delaware, E.U.A., 1993.

### **Revistas:**

De la More, Salvador, "Hablemos de Lenguajes Orientados a Objetos", en: Revista Personal Computing-México, Año 7, Febrero 1995, México.

López Michelone, Manuel, "Los Lenguajes de Programación a Través del Tiempo", en: Revista Personal Computing-México, Año 7, Febrero 1995, México.

Marin, Gerardo, "Los Lenguajes Estructurados Viven y Trabajan", en: Revista Personal Computing-México, Año 7, Febrero 1995, México.