



UNIVERSIDAD AUSTRAL DE CHILE  
FACULTAD DE CIENCIAS DE LA INGENIERÍA  
INSTITUTO DE INFORMÁTICA  
ESCUELA DE INGENIERÍA CIVIL EN INFORMÁTICA

**PROPUESTA DE UN MODELO PARA  
ASEGURAR LA CALIDAD EN UN PROCESO DE  
MIGRACIÓN DE UN SOFTWARE. BASADO EN  
EL MODELO CMM.**

**Tesis de grado para optar al  
título de Ingeniero Civil en  
Informática**

**Profesor Patrocinante : Raimundo Vega**

**Profesor Co-Patrocinante : Nadja Starocelsky**

**Profesor Informante : Martín Solar**

**MARÍA LUZ OCHOA MANCILLA**

**VALDIVIA – CHILE  
2002**



VALDIVIA,

DE : NADJA ALEJANDRA STAROCELSKY VILLAVICENCIO

A : DIRECTORA ESCUELA INGENIERÍA CIVIL EN INFORMÁTICA

MOTIVO:

#### INFORME TRABAJO DE TITULACIÓN

Nombre Trabajo de Titulación: PROPUESTA DE UN MODELO PARA ASGURAR LA CALIDAD DE UN PROCESO DE MIGRACION DE UN SOFTWARE, BASADO EN EL MODELO CMM. Nombre del Alumno : MARÍA LUZ OCHOA MANSILLA

FUNDAMENTOS:

La tesis cumple con el objetivo propuesto, determinando un método exitoso de migración en una empresa que no cumple ninguna norma de aseguramiento de calidad. Cumple con las actividades y criterios propuestos.

Se utiliza un lenguaje claro y preciso en la redacción.

Por lo tanto califico la tesis con la siguiente nota:

Nota : 7.0  
(en números)

siete  
(en letras)



Nadja Starocelsky V.

Valdivia, 24 de Junio de 2002

**De :** Martín Gonzalo Solar Monsalves

**A :** Directora Escuela Ingeniería Civil en Informática

**Ref.:** Informe Calificación Trabajo de Titulación

**Nombre Trabajo de Titulación:**

"PROPUESTA DE UN MODELO PARA ASEGURAR LA CALIDAD EN UN PROCESO DE MIGRACION DE SOFTWARE BASADO EN EL MODELO CMM".

**Nombre Alumno:**

María Luz Ochoa Mancilla

**Evaluación:**

Cumplimiento del objetivo propuesto	6.5
Satisfacción de alguna necesidad.	7.0
Aplicación del método científico	7.0
Interpretación de los datos y obtención de conclusiones	6.0
Originalidad	6.0
Aplicación de criterios de análisis y diseño	6.5
Perspectivas del trabajo	6.0
Coherencia y rigurosidad lógica	7.0
Precisión del lenguaje técnico en la exposición, composición, redacción e ilustración	7.0
<b>Nota Final</b>	<b>6.6</b>

Sin otro particular, atte.:



Martín Solar Monsalves

*Esta tesis se la dedico a mi mejor amiga ... mi Mamita.*

*A mi Papito que no me ha fallado nunca y me sigue acompañando.*

*A mi hermanita que es mi redactora oficial y mi alegría.*

*A mi hermano Wiss, por demostrarme que no hay  
límite para los sueños.*

*A mi Chule por su amor incondicional.*

*A mi Nanita por toda su preocupación.*

*A toda mi familia..*

*Y a ti Nabil por entenderme, apoyarme y amarme.*

# INDICE

Resumen .....	4
Introducción .....	5
 <b>CAPÍTULO I</b>	
1. Estudio y comprensión del concepto de “Calidad” .....	7
 <b>CAPÍTULO II</b>	
2. Estudio y comprensión del modelo CMM. ....	13
2.1. Modelo de madurez de capacidades .....	18
2.2. Estructura del modelo de madurez de capacidades .....	21
2.2.1. Nivel 1: El caos y el heroísmo	
Nivel inicial o nivel de inmadurez .....	23
2.2.2. Nivel 2: El proceso repetible	
El proyecto planificado .....	25
2.2.3. Nivel 3: El proceso definido	
El proceso generalizado en todos los proyectos ...	27
2.2.4. Nivel 4: El proceso administrado	
La calidad planificada y confiable .....	29
2.2.5. Nivel 5: El mejoramiento permanente	
La calidad planificada y confiable .....	30
2.3. Áreas claves de proceso .....	32
2.4. Metas .....	35
2.5. Características comunes .....	38
2.6. Proyecciones .....	40
2.7. Críticas al CMM .....	43
2.8. Estado actual	
 <b>CAPÍTULO III</b>	
3. Formulación de la propuesta de normas para asegurar la calidad ....	45
3.1. Propuesta de un modelo para asegurar la calidad	
en un proceso de migración de un software .....	47

3.1.1. Análisis, revisión, comparación y especificación de los requerimientos .....	48
3.1.2. Revisión detallada del código del software a migrar ...	49
3.1.3. Estimación y plan de trabajo .....	50
3.1.4. Elaboración de un diccionario de términos. ....	60
3.1.5. Modificación de cada Query (o instrucción SQL) .....	61
3.1.6. Ejecución de cada una de las instrucciones SQL	
3.1.7. Comparación de los resultados con los requerimientos solicitados . ....	62
3.1.8. Aplicación de la migración.	
3.2. Aplicación del modelo CMM sobre el modelo propuesto para asegurar la calidad en un proceso de migración de un software .....	63

## **CAPÍTULO IV**

4. Aplicación de las normas propuestas en un ejemplo concreto de migración. ....	66
4.1. Análisis, revisión, comparación y especificación de los requerimientos	
4.2. Revisión detallada del código del software a migrar .....	67
4.3. Estimación y plan de trabajo .....	68
4.4. Elaboración de un diccionario de términos. ....	72
4.5. Modificación de cada Query (o instrucción SQL) .....	76
4.6. Ejecución de cada una de las instrucciones SQL .....	78
4.7. Comparación de los resultados con los requerimientos solicitados.	
4.8. Aplicación de la migración. ....	79

## **CAPÍTULO V**

5. Verificación y medición de resultados .....	80
Conclusiones .....	83
Referencias .....	85
Anexos .....	87

## **Resumen**

El proceso de migración de un software es un esfuerzo altamente complejo, en el cual se debe garantizar que el cambio cumplirá con los requisitos mínimos del usuario y que superará al software original.

Dicha problemática es la que llevó a plantear esta tesis como la búsqueda de ciertas normas para asegurar la calidad en el proceso de cambio de un software, obteniendo el mismo funcionamiento, respuesta y efectividad que el software original.

Aún cuando el software es un elemento intangible, sobre el cual es difícil realizar mediciones directas, se hace cada vez más necesario contar con las herramientas para asegurar la calidad obteniendo un mínimo de error.

Por tanto, esta tesis propone la concepción, desarrollo e investigación de un modelo metodológico que permita guiar hacia un proceso de aseguramiento de calidad en un proceso de migración de un software. Para ello, se tomará como base principalmente el Modelo de Madurez de Capacidades (Capability Maturity Model, CMM)

## **Summary**

The software's migration process is a highly complex effort, in the which it should be guaranteed that the change will fulfill the user's minimum requirements and that it will overcome to the original software.

This problem is the one that carried to outline this thesis like the search of certain norms to assure the quality in the process of change of a software obtaining the same operation, answer and effectiveness that the original software.

Still when the software is an intangible element, on which is difficult make direct mensurations. It becomes more and more necessary to have the tools to assure the quality obtaining a minimum of error.

Therefore, this thesis proposes the conception, development and investigation of a methodological model that it allows to guide toward a process of insurance of quality in a process of migration of a software. For that, I will take like base mainly the Capability Maturity Model (CMM).

## Introducción

La década del 90 se caracteriza por la palabra “*Calidad*”, sin importar a que elementos, entes o áreas se refiera. Gerencial, servicios, administración, gestión, hospitales, gobiernos, bancos, etc. La ciencia tampoco escapa a este fenómeno, medicina, economía, ingeniería, biología, etc. Todos dicen poseerla y dominarla, pero cómo poder conocer y dominar un concepto tan amplio, subjetivo y muchas veces ambiguo, y que ni siquiera la mayoría de las personas podrían dar una definición exacta de *¿qué es calidad?*.

La ingeniería de software no escapa a esta realidad, pero sin lugar a dudas se enfrenta con muchos más obstáculos para poder dominar la calidad en comparación con otras ciencias. Uno de estos obstáculos es el hecho que no existe una definición estándar y universal de que es calidad. Algunos organismos e instituciones como la ISO, IEEE y SEI, brindan definiciones aceptables, pero no homogéneas, dando como resultado que cada profesional utilice su propia definición de calidad.

El vocabulario de la International Standard Quality define calidad como : “*La totalidad de las propiedades y características de un producto o servicio que le confiere la aptitud para satisfacer necesidades expresadas o implícitas.*” ISO –1986]

Otro tema que trata esta tesis es el proceso de migración de software lo cuál conlleva un esfuerzo altamente complejo.

Una definición de migración es : “*Traslado de un PC a otro en condiciones de compatibilidad. Migrar es también elevar una versión de un producto de software a otra de más alto nivel, o bien el movimiento o traslado de una arquitectura en otra.*” Para el caso de esta tesis se utilizará el término de migración para referirse al cambio de un software que se ejecuta sobre una base de datos a otra.

El problema que generó el planteamiento de esta tesis fue el contar con un software desarrollado en Visual Basic 4.0 y sobre una base de datos Oracle 7.3 y la necesidad de cambiar la base de datos de esta aplicación a una Sybase 11.

Es por ello y para asegurar el correcto funcionamiento del software una vez finalizado el proceso de migración se hace necesario contar con un plan que permita asegurar la calidad. Estas normas pretenden guiar en algún proceso de migración basado en la experiencia y el desarrollo de la conversión antes mencionada. Además estos pasos pueden ser generalizados a otras bases de datos, no solo para el caso desarrollado en esta tesis.

Algunos de los principales alcances que motivaron a la realización de este proyecto de tesis fueron poder asegurar al final de un proceso migratorio que el software cumple con los requisitos solicitados y por tanto lograr la satisfacción del usuario. Aportando así con un método para obtener una mejora en la calidad de los procesos de migración de software, problemática que se encuentra actualmente presente en las empresas. Comprender y aplicar el tema del aseguramiento de calidad basado en el modelo CMM, tema que es de gran interés y aplicabilidad en los procesos productivos de software en la actualidad.

Los objetivos que persigue este trabajo de tesis son:

- Estudiar y definir el concepto de “Calidad” en el contexto de tema propuesto.
- Estudiar y comprender el modelo CMM.
- Desarrollar una propuesta de un modelo para asegurar la calidad en un proceso de migración de software.
- Aplicar el modelo propuesto en un proceso de migración desde una base de datos Oracle a una Sybase.

# CAPÍTULO I :

## **1. Estudio y Comprensión del Concepto de “Calidad”.**

Uno de los problemas que se enfrentan actualmente en el ámbito computacional es la calidad de software. Desde la década del 70 este tema ha sido motivo de preocupación para especialistas, ingenieros, investigadores y comercializadores de software, los cuales han realizado gran cantidad de investigaciones al respecto con dos objetivos fundamentales : *¿cómo obtener un software de calidad?* y *¿cómo evaluar la calidad de software?*

Todos concuerdan en que la calidad de software y el control de calidad del software son factores primordiales en la industria de desarrollo y mantención del software hoy en día. Sin embargo el término “calidad” es muy difícil de definir con precisión.

Existen varias corrientes de calidad desarrolladas por expertos en el tema como son : Deming, Juran, Crosby, Ishikawa y Taguchi.

Deming fue conocido como el principal gestor de la calidad en Japón. Según Deming un producto es de calidad cuando los clientes no sólo están conformes con lo que han recibido sino que alaban el producto o servicio. Algunos puntos que apoyan esta definición de calidad de Deming son :

- Se deben analizar todos los procesos constante y continuamente.
- No se debe depender de las inspecciones finales para lograr la calidad.

- Se debe persistir en el propósito de mejorar la calidad de productos y servicios.
- Todos deben participar en la transformación de la compañía.

Aún cuando los puntos anteriores fortalecen a una organización para implementar la calidad, Deming identifica siete “enfermedades” que pueden afectar la calidad de esa organización, éstas son :

- Falta de persistencia en los propósitos.
- El pensar a corto plazo.
- Evaluación anual del desempeño personal
- Cambio excesivo de empleados.

Otras de las principales corrientes de calidad es la de Joseph Juran. Este, junto a Deming iniciaron la revolución de la calidad en Japón a mediados de los cincuenta.

Según Juran, hay dos aspectos distintos pero relacionados con la calidad. El primero tiene que ver con los ingresos en forma directa y consiste en aquellas características y atributos que satisfacen las necesidades del cliente. En este caso producir mejor calidad significa costos más altos. El segundo aspecto está más bien relacionado con el costo de producción, y consiste en la ausencia de fallas y defectos. En este caso más calidad cuesta menos.

Otra corriente de calidad es la entregada por Phil Crosby. Crosby popularizo la idea que la calidad es “gratis”. Queriendo decir que el costo de la calidad es lo más bajo posible cuando hay cero defectos. Según Crosby, el objetivo de la administración de la calidad es la de prevenir defectos, no hay razón para permitir

fallas. No se debe permitir el uso de técnicas de depuración basadas en pruebas finales. Algunos pasos entregados por Crosby para obtener la calidad total son :

- Evaluar el costo de la mala calidad.
- Estimular a los empleados a comunicar los obstáculos que enfrentan para lograr las metas de calidad propuestas.
- Reconocer y apreciar a quienes contribuyen al programa de calidad.

Ishikawa fue el primero en usar el término “Control de Calidad Total” y es responsable del desarrollo de las siete herramientas para analizar problemas de calidad. Estas herramientas son : Diagramas de Pareto, Diagrama de Causa-Efecto, Histogramas, Planillas de Control, Diagramas de Dispersión, Diagramas de Flujo y Cartas de Control.

Algunos conceptos básicos de calidad entregados por Ishikawa son:

- La educación es la base de la calidad.
- El primer paso para implantar calidad es conocer los requisitos.
- Identificar y eliminar las causas de los problemas y no los síntomas.
- No confundir los medios con los objetivos.

Algunas otras definiciones de calidad encontradas en la literatura dicen que:

- Calidad es alcanzar altos niveles de satisfacción del usuario, portabilidad, robustez, mantenibilidad y ajuste para el uso.
- Altos niveles de satisfacción del usuario y adherencia a los requerimientos.
- Ausencia de defectos.

- Calidad debe ser medible cuando ocurre y predecible antes de que ocurra.

Existe una definición formal del término calidad, esta fue establecida por la Real Academia Española de la Lengua, que define este término como : “(*calidad es ...*) *propiedad o conjunto de propiedades inherentes a una cosa, que permite apreciarla como igual, mejor o peor que las restantes de su especie. En sentido absoluto, buena calidad, superioridad, excelencia.*” [RAE –1984]

Aún cuando la definición anterior permite normalizar el concepto entre las personas, no permite tratar cuantitativamente la calidad de un producto o proceso. Una definición más técnica es la obtenida del concepto de Calidad de Genechi Taguchi :

*“Calidad es la mínima pérdida ocasionada a la sociedad desde que el producto es embarcado. Sean :*

*Q : Calidad*

*L : Función de Pérdida*

*Y : Vector de Medición de Pérdida*

*Se define :*

$$Q = L^{-1} (Y)$$

*El objetivo es maximizar Q, que pasa por minimizar L.”* [Matar-1995]

Taguchi define la calidad en términos de la pérdida generada por ese producto a la sociedad. Esta pérdida es valorizada en moneda que permite comunicar la magnitud al personal no técnico. En una economía competitiva, la mejoría

continua de la calidad y la reducción de los costos son necesarios para la viabilidad de un negocio en el largo plazo.

Si bien esta definición es correcta puede presentar ciertas dificultades al ser tratada cotidianamente por los usuarios o incluso por los profesionales informáticos, debido a la variadas características socioculturales.

El vocabulario de la International Standard Quality define calidad como : *“La totalidad de las propiedades y características de un producto o servicio que le confiere la aptitud para satisfacer necesidades expresas o implícitas.”* [ISO-1986]

Por otro lado la define la calidad como : *“El grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario.”* [IEEE – 1990]

Otra acertada definición de calidad es la entregada por Pressman que dice que : *“Calidad es la concordancia del software producido con los requerimientos de explícitamente establecidos, con los standard de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente que desea el usuario.”* [Pressman-1993]

En general, clientes, usuarios y desarrolladores coinciden intuitivamente en el concepto de calidad. Aunque resulte difícil llegar a una definición en concreto, el concepto se encuentra en la mente de todos los involucrados en el proceso. Es así que se concuerda en que los principales factores que influyen o determinan de algún modo la calidad del software son : *la satisfacción del usuario, la correspondencia con los requerimientos y una baja tasa de defectos.*

Pero como llevar estas definiciones de calidad al proceso mismo de migración de un software.

- **Satisfacción del Usuario** : luego de efectuado el proceso de migración de un proceso o de un producto, el usuario espera el correcto funcionamiento de éste, con un desempeño igual o superior al producto original, y en un lapso de tiempo dentro de los márgenes establecidos previamente.
- **Correspondencia con los Requerimientos** : que el software migrado siga cumpliendo con los requerimientos originales bajo los cuales fue desarrollado.
- **Baja Tasa de Defectos** : que los errores encontrados luego de el proceso de cambio del producto tiendan a cero.

Todo esto nos lleva a que el proceso de cambio o migración debe ser un avance y un aporte para el sistema, que entregue mayores beneficios, como una mayor rapidez de respuesta en los procesos o menos errores.

## **CAPÍTULO II :**

### **2. Estudio y Comprensión del Modelo CMM.**

En los últimos quince años la industria mundial de desarrollo de software se ha preocupado por mejorar sus capacidades en el desarrollo de software de calidad. Las empresas están invirtiendo en la mejora de los procesos de desarrollo para poder distinguirse en el mercado. Se han definido varios modelos basados en las experiencias exitosas de la Ingeniería de Software que sirven de guía para las mejoras y unifican los criterios de evaluación de las empresas. Las normas ISO de serie 9000, el modelo estadounidense conocido como Capability Maturity Model (CMM), el BOOTSTRAP de la Comunidad Europea y la norma ISO 15504, conocida como SPICE, (que entrará en vigor en el año 2002) son los ejemplos más reconocidos de estos modelos. Las empresas hacen esfuerzos para implantar estos modelos y lograr la certificación o evaluación en alguno de ellos con el objetivo de obtener ventajas competitivas.

Pero para las empresas medianas y pequeñas los problemas principales son: la selección del modelo a seguir, su comprensión e interpretación y el costo de la implantación en la empresa. Esto se debe a que todavía no existen los servicios de consultoría nacionales confiables y que los servicios externos son muy caros. Otra razón es que los modelos de referencia mencionados son complejos, voluminosos y están definidos en términos que no son fácilmente entendibles. Además, fueron diseñados pensando en grandes empresas, lo que dificulta su adopción a las empresas pequeñas y medianas.

El proceso de desarrollo de software es el conjunto de actividades, métodos y prácticas utilizados para desarrollar y mantener software. Una organización o empresa debe definir claramente que proceso usará para desarrollar y mantener software. La calidad de este proceso determina la calidad del producto.

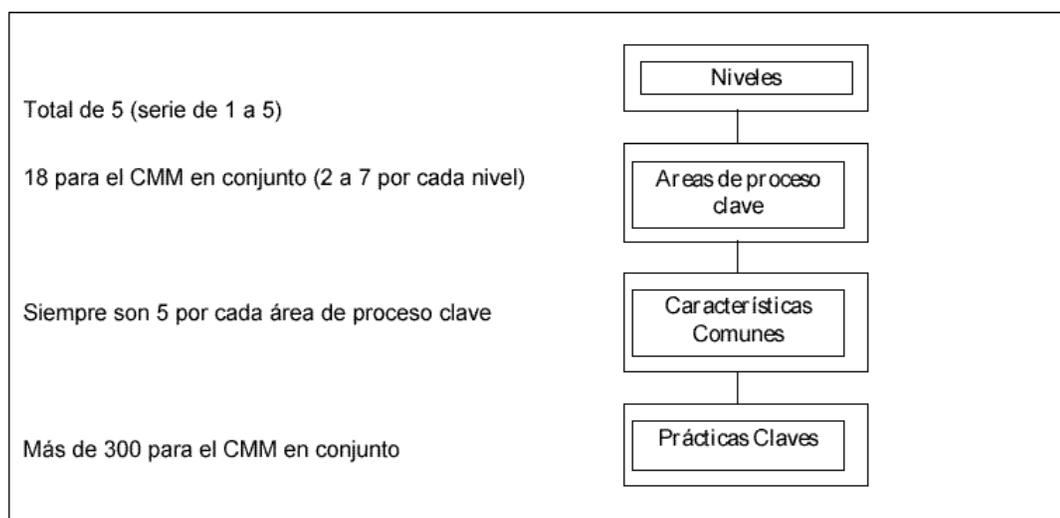
Es bien sabido que el negocio del software es el negocio actual y del futuro, y debido a la fuerte competencia se esperan importantes bajas en los precios. Por ello para que una empresa de desarrollo se mantenga competitiva en el mercado deberá lograr en corto tiempo reducciones importantes del costo de producción y aumento de la calidad. Es por este motivo que se ha buscado lograr los objetivos usando técnicas de procesos y calidad de ingeniería de software.

Uno de los modelos de procesos de software que está adquiriendo mayor interés es el CMM. La base fundamental del CMM engloba todos los enfoques actuales existentes sobre mejora y evaluación de procesos de desarrollo de software, incluyendo ISO, Bootstrap y SPICE, según la European Software Institute (ESI).

CMM entrega una guía a las organizaciones de software de cómo obtener control sobre los procesos de desarrollo y mantenimiento de software y como obtener un mejoramiento continuo de los procesos productivos de software, que es hoy la referencia más utilizada para sistemas de calidad para software. Este será el modelo que se analizará, profundizará y se seguirá en esta tesis. La elección de este modelo en particular se debe a que es el modelo orientado a software más antiguo, conocido y que mayormente ha sido aplicado por diversas organizaciones. Por consiguiente es uno de los modelos mayormente probado en forma real.

El modelo del CMM fue desarrollado inicialmente en EE.UU. para satisfacer las necesidades del Departamento de Defensa de ese país para mejorar las prácticas de desarrollo de software entre sus proveedores. Encargado de esa misión, el Software Engineering Institute (SEI de Carnegie Mellon) publicó la estructura del CMM - el trabajo de años de experiencia. Watts Humphrey (1989), a la cabeza del programa de procesos del SEI, propuso inicialmente el CMM como la base para organizar el desarrollo de software. Este modelo derivó del trabajo realizado por él y sus colegas en IBM y de los principios de calidad de Deming y Crosby.

Este modelo se compone por 5 niveles de madurez, cada nivel (menos el primero) tiene asociado un conjunto de KPAs (Key Process Areas) Areas Claves de Procesos, estas normas definen un conjunto de objetivos. Cada KPA incluye un número de prácticas claves que llevan a cumplir esos objetivos, no es necesario implementar todas las prácticas, pero sí es necesario cumplir con todos los objetivos del nivel acreditar y sus inferiores. (Ver fig. 1)



**Fig. 1.-** Arquitectura General CMM

La arquitectura del CMM toma la forma de una jerarquía, no diferente a un diagrama de alguna organización, consiste en cinco niveles de madurez. Cada nivel puede subdividirse en un número variable de áreas de proceso clave (2 a 7 por cada nivel y un total de 18 para todos los niveles) cada uno de los cuales comprende 5 categorías de prácticas llamadas características comunes. Las prácticas claves están en el fondo de la jerarquía y pueden encontrarse en todas las categorías de características comunes. Hay un conjunto de más de 300 prácticas claves en el modelo.

Los Cinco niveles de CMM son mostrados en la tabla 1.

NIVEL	DESCRIPCIÓN
<b>Nivel 5</b> <b>Optimizado</b>	Los procesos se mejoran a medida que se aplican. Toda la organización se beneficia en forma inmediata de las mejoras hechas por los distintos agentes del proceso. No hay muchas organizaciones en el mundo que hayan alcanzado este estado ideal.
<b>Nivel 4</b> <b>Administrado</b>	Los proyectos de software son capaces de planificar metas para la calidad de los productos en proceso de desarrollo.



<p><b>Nivel 3</b> <b>Definido</b></p>	<p>La empresa ha definido los procesos de gestión de ingeniería de manera estándar y completa. Los proyectos adaptan el proceso común de acuerdo a sus características particulares. El personal usa la biblioteca de procesos para realizar sus operaciones habituales.</p>
<p><b>Nivel 2</b> <b>Repetible</b></p>	<p>La empresa ha definido prácticas de gestión de proyectos mínimas. Algunos proyectos pueden repetir los éxitos pasados en nuevos desarrollos. Hay menos dependencia en los individuos. La gerencia es más proactiva y los planes de proyectos son más realistas.</p>
<p><b>Nivel 1</b> <b>Inicial</b> <b>o</b> <b>Ad-hoc</b></p>	<p>Los procesos varían según los individuos, quienes en forma habitual deben enfrentar crisis. El caos y los problemas predominan. El éxito es producto de la casualidad, mientras que los riesgos son elevados. El personal vive bajo stress y trabaja casi siempre más allá de lo normal.</p>

**Tabla 1.-** Los cinco niveles del modelo CMM.

## **2.1. Modelo de Madurez de Capacidades**

Según definición de la SEI (Software Engineering Institute) de la Universidad de Carnegie Mellon de Pittsburgh, la madurez de un proceso es determinada por el grado en el que un proceso es : definido, documentado, administrado, controlado y medido.

Existen varias propuestas de escalas para la evaluación de una organización. Una de ellas es el Modelo de Madurez de la Capacidades, más conocido como CMM (Capability Maturity Model) este modelo pueden ser usado por las organizaciones para determinar sus capacidades en el desarrollo o mantención de software basado en un proceso de mejoramiento continuo de la organización.

Algunas premisas de las escalas de madurez son que :

- Una organización madura tiende a producir con mayor calidad y productividad.
- Una organización sin madurez tiende a producir software pobre, con mayor costo y mayor riesgo de fracaso en los proyectos.

### **Características de un proceso maduro**

Un proceso puede considerarse maduro si cumple con los siguientes criterios:

- **Está definido** : El proceso es claro, sistemático y suficientemente detallado. Además existe acuerdo entre el personal, la gerencia y los proyectos respecto al proceso que se va a utilizar.

• **Esta documentado** : Esta escrito en un procedimiento publicado, aprobado y fácilmente accesible. Una de las mejores maneras es a través de una Intranet para apoyar los proyectos de desarrollo.

• **El personal ha sido entrenado en el proceso** : Los ingenieros de software y la gerencia han recibido cursos y entrenamiento en cada proceso que aplica a su trabajo.

• **Es practicado** : El proceso definido debe ser usado en las tareas habituales llevadas a cabo por los proyectos. El entrenamiento y la adaptación del proceso a la realidad de la empresa debieran garantizar su aplicación en la vida real.

• **Es apoyado** : La gerencia no sólo debe firmar y promover los procesos definidos, sino que además debe asignar responsabilidad al personal y al los jefes de proyecto por su cumplimiento.

• **Es mantenido** : El proceso es revisado regularmente, para asegurarse que está adaptado para satisfacer las necesidades reales de los proyectos.

• **Está controlado** : Los cambios y puestas al día del proceso son revisados, aprobados y comunicados oportunamente a todos los usuarios.

• **Se verifica** : La gerencia mantiene mecanismos para asegurarse que todos los proyectos siguen el proceso vigente.

• **Se valida** : Se asegura que el proceso mantiene concordancia con los requerimientos y estándares aplicables.

• **Se mide** :La utilización, los beneficios y el rendimiento resultante del proceso se miden regularmente.

• **Puede mejorarse** : Existen mecanismos y apoyo de la gerencia para revisar e introducir cambios en el proceso, de manera de mejorar su eficacia e incorporar nuevas metodologías.

**Características de un Proceso Inmaduro:**

Es fundamentalmente personal, no está documentado, es difícil compartirlo con otros miembros del equipo, no es fácil reproducirlo en nuevos proyectos, no hay entrenamiento, no todo el mundo lo conoce, no se mide, se aplica a veces solamente, es percibido como poco eficiente, es interpretado de manera distinta, etc.

Se considera que una organización ha alcanzado un nivel dado de madurez cuando ha cumplido con todos los objetivos asociados con cada una de las áreas claves desde el nivel 2 hasta el nivel identificado.

A medida que una organización progresa hacia una mayor madurez de su proceso de software, sus riesgos disminuyen en forma proporcional al incremento de la calidad de sus productos, a los tiempo estimados de los plazos y presupuestos previstos, así como a la satisfacción de los clientes.

Los principales cambios que ocurren durante este proceso, se resumen en la tabla 2.

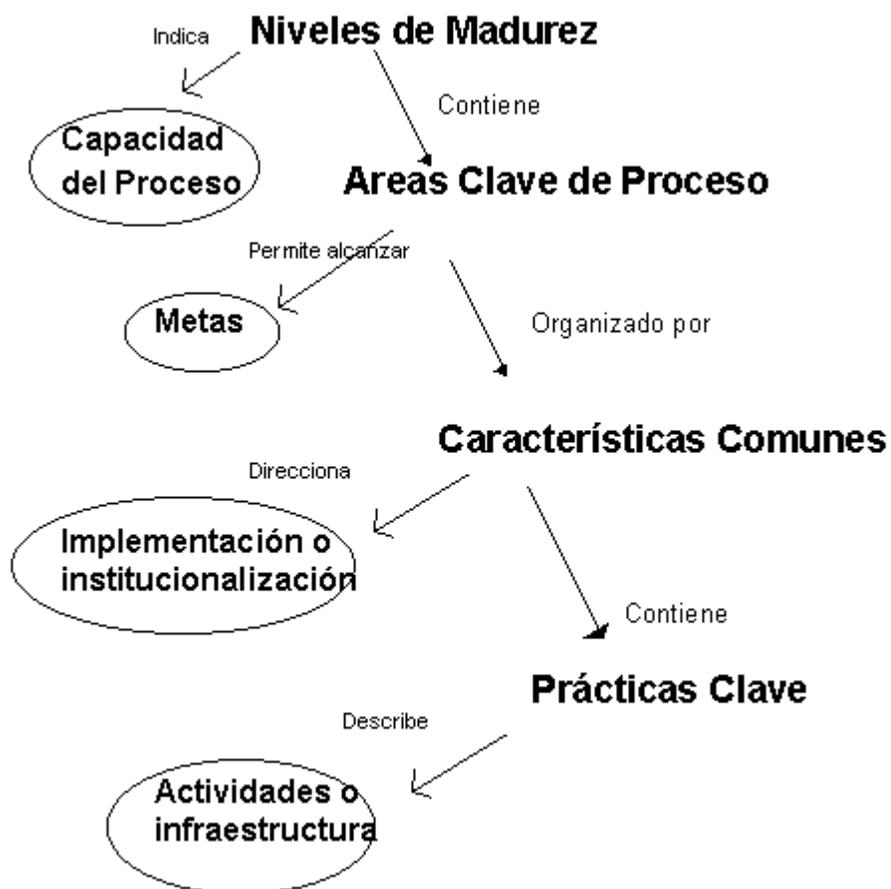
<b>Organización Inmadura</b>	<b>Organización Madura</b>
El proceso se improvisa y es caótico.	El proceso está definido, documentado, medido, controlado y es utilizado.

Tensión permanente.	Mejor calidad de vida.
Los costos, plazos y calidad son imprevisibles.	Los costos, plazos y calidad son previsibles.
Gestión en base a crisis.	Gestión proactiva.
El éxito está basado en esfuerzos individuales heroicos, a pesar del proceso existente.	El éxito se puede repetir y depende de la fuerza del equipo, gracias al proceso existente.
Las horas extraordinarias no programadas son crónicas.	Buena gestión con tiempo disponible.
La tecnología se usa pobremente, y se adapta en forma inadecuada	Uso juicioso de la tecnología.

**Tabla 2.-** Cambios producidos en un proceso inmaduro.

## **2.2. Estructura del Modelo de Madurez de Capacidades**

Los cinco niveles de madurez de proceso constituyen la estructura de más alto nivel del CMM. Deben ser interpretados como estados evolutivos, los cuales se sustentan en la implementación de los niveles más bajos. La figura 2 muestra la estructura del CMM.



**Fig. 2.-** Estructura de CMM.

Fuente : [Psulk-1993a]

Con excepción del Nivel 1, cada uno de estos **Niveles de Madurez** está compuesto por un cierto número de **Áreas Claves de Proceso**, conocidas a través de la documentación del CMM por su sigla inglesa: KPA. Cada KPA identifica una agrupación de actividades y prácticas relacionadas, las cuales cuando son realizadas en forma colectiva permiten lograr alcanzar las metas fundamentales del proceso. Las KPAs pueden clasificarse en 3 tipos de proceso: Administración, Organizacional e Ingeniería.

### **2.2.1. Nivel 1: El Caos y el Heroísmo**

#### **Nivel Inicial o Nivel de Inmadurez**

El nivel 1 o inicial es el estado primario en la evolución de las organizaciones que desarrollan software.

Una definición amplia es que en este nivel se encuentran todas las empresas que no han logrado implementar las prácticas básicas de gestión de proyectos e ingeniería de software definidas a partir del nivel 2 o superiores. Una empresa está en el nivel caótico cuando sus gerentes y personal afirmen que los proyectos no se pueden planear, que los requerimientos no se pueden tener bajo control, que no esté siempre en condiciones de controlar las versiones de producto, donde la calidad sea percibida como una burocracia innecesaria, cuando se acepte que los procesos son una cosa personal, cuando no se pueda verificar ni validar el producto, y sobre todo, cuando sus gerentes y personal vivan bajo condiciones de stress y frustración permanentes.

#### ***Heroísmo, caos y emergencia permanente***

En este tipo de empresas, el software es virtualmente producto del arte más que de la ingeniería. Cada "artista" crea su propio proceso personal, el cual es parte de su sello personal. La gerencia ocupa una parte significativa de su tiempo en paliar problemas y enfrentar clientes insatisfechos. Ante una situación de crisis permanente, se les hace difícil destinar recursos para definir o documentar procesos, lo que lleva a un círculo sin salida.

Cuando el proyecto se termina, la inversión hecha en desarrollar el proceso es raramente reutilizada en nuevos proyectos. Los desarrolladores de software generalmente tienen que trabajar largas horas y paliar problemas en forma cotidiana, lo cual les disminuye su creatividad y productividad netas. El éxito descansa en los hombros de estos héroes. Su nivel de frustración es

elevado y es muy frecuente que decidan explorar caminos en otras empresas con menor nivel de stress. El proceso, que no está documentado ni a sido compartido, se va con ellos, dentro de sus cerebros. Los reemplazantes heredan problemas y dificultades, pero son raramente capaces de recuperar los procesos de desarrollo. Esto obliga a reinventar la rueda, a un alto costo y retrasando los proyectos.

### ***Características Principales***

- Sin procesos definidos.
- Sin standards y si existen son ignorados.
- Poca habilidad para estimar.
- El éxito está basado en “héroes”.
- Poca visibilidad.
- Las fechas límite son más importantes que la calidad.
- Proyectos fuera de término y presupuesto.

## **2.2.2. Nivel 2: El Proceso Repetible**

### **El Proyecto Planificado**

El nivel 2 o Repetible hace posible la implementación de prácticas mínimas de administración de proyecto, de control de requerimientos, versiones de producto y de proyectos realizados por subcontratistas. El grupo o equipo humano que realizó el proyecto puede aprovechar su experiencia e inversión en procesos para aplicarla en un nuevo proyecto.

Este nivel no garantiza que todos los proyectos dentro de la empresa estén necesariamente al mismo nivel de madurez. Algunos pueden estar todavía en el nivel inicial.

### ***Beneficios de implantar el Nivel 2 del CMM***

El mayor beneficio obtenido de la implementación del nivel 2 es la planificación realista de los proyectos. Usualmente los cronogramas de proyecto expresan los deseos de la gerencia más que la realidad. Este principio conduce una situación de buscar culpables y generar excusas, produciendo al mismo tiempo frustración y desconfianza entre clientes y empleados.

Gracias a la adaptación del nivel 2 los cronogramas pueden ser cada día más confiables, y mejorar a medida que se acumula mayor información en las bases de datos de los proyectos pasados. El uso generalizado de métodos de estimación permite al personal del proyecto justificar plazos y recursos. Aún la experiencia personal juega un papel importante en la generación de planes de proyecto, pero ahora son decisiones informadas en lugar de adivinanzas.

### ***Características Principales***

- Administración y control de proyectos
- Se crean y documentan los procesos
- Se recolectan métricas y se realizan seguimientos al progreso
- Mejora la estimación
- Menos desvíos que en el Nivel 1

Este nivel todavía permite la proliferación y definición insuficiente de los procesos de ingeniería de software. Los proyectos comparten principalmente sus experiencias en materia de administración de proyectos, pero sus métodos técnicos pueden diferir. Aún existe incomunicación entre proyectos, grupos y entre personal y gerencia.

Este nivel identifica prácticas de sentido común que son aplicables en todo tipo de organizaciones de desarrollo de software, independientemente de su rubro, tamaño o ambiente de desarrollo. La ausencia de cualquiera de sus prácticas pone en peligro el éxito de la empresa.

### ***Las Áreas Clave de Proceso del Nivel 2***

Las áreas clave de proceso del nivel 2 o repetible son seis :

- Administración de Requerimientos.
- Planificación del proyecto de software.
- Seguimiento y supervisión del proyecto de software.
- Administración de subcontratos de software
- Aseguramiento de la Calidad del software
- Administración de la Configuración del software.

### **2.2.3. Nivel 3: El Proceso Definido**

#### **El Proceso Generalizado en Todos los Proyectos**

La empresa ha definido un conjunto de procesos, metodologías y herramientas comunes a todos los proyectos iniciados por la corporación. El proceso común está suficientemente documentado en una biblioteca accesible a todo los desarrolladores. Todo el personal ha recibido el entrenamiento necesario para entender el proceso estándar. Existen pautas y criterios definidos para adaptar dicho proceso a las necesidades y características propias de cada proyecto. El nivel de definición es detallado y completo. La dependencia (o el riesgo de depender) en individuos irremplazables es baja.

#### ***Beneficios de implantar el Nivel 3 del CMM***

La base de datos que reúne estadísticas de los proyectos pasados y en curso, permite planificar y comparar el rendimiento. Existen mecanismos de comunicación entre proyectos y departamentos, lo que garantiza una visión común del producto y una rápida acción para enfrentar los problemas.

A nivel 3 los empleados tienen una alta valoración de los procesos y entienden claramente la manera en que afectan su desempeño habitual. Los gerentes pueden realizar sus verdadera función, administrar.

El hecho de realizar revisiones tempranas en forma regular mejora visiblemente la calidad de los productos y minimiza las reiteraciones innecesarias. Curiosamente muchas organizaciones de nivel 1 realizan revisiones de pares, pero lo hacen de manera inconsistente y al primer signo de pánico las suspenden.

El nivel 3 ya es un estado avanzado y es percibido por algunos gerentes como un lujo.

### ***Características Principales***

- Implementación de standars, documentación de procesos para el desarrollo y mantenimiento.
- Guías para la adecuación de los procesos en un proyecto específico.
- Incremento de efectividad y reducción de costos y tiempos.
- Se crea el Software Engineering Process Group (SEPG).

### ***Las Áreas Claves de Proceso del Nivel 3***

Las áreas claves de proceso para el nivel 3 o definido son 7 :

- Enfoque en el proceso de la organización.
- Definición del proceso de la organización
- Programa de entrenamiento
- Gestión integrada del software
- Ingeniería del producto
- Coordinación entre grupos
- Revisión de pares

#### **2.2.4. Nivel 4: El Proceso Administrado**

##### **La Calidad Planificada y Confiable**

En este nivel la corporación mide la calidad del producto y del proceso de software. Ambos, producto y proceso, son seguidos en forma cuantitativa y se controlan mediante métricas detalladas. La capacidad de rendimiento del proceso es previsible. Las mediciones permiten detectar cuando las variaciones del rendimiento se salen de los rangos aceptables, de manera de poder tomar medidas correctivas para asegurar la calidad.

##### ***Beneficios de implantar el Nivel 4 del CMM***

La empresa es capaz de proponerse metas cuantitativas para la calidad de los producto y de los procesos de software. Es posible medir la productividad y calidad de los procesos de software a través de todo el proyecto.

Los proyectos pueden controlar la variación del rendimiento de sus productos y procesos para mantenerla dentro de fronteras cuantitativas aceptables. Es posible discriminar las variaciones significativas en el rendimiento del proceso de la variación (ruido) al azar, particularmente dentro de líneas de productos establecidas.

Es necesario aclarar que el hecho de contar con un sistema de métricas de software no significa que se esté en el nivel 4. Existen empresas de nivel 1 que miden cuidadosamente el número de defectos detectados durante las pruebas o tests. Es una virtual señal de alarma que les dice cuán graves son sus problemas, pero la inmadurez de sus procesos no les permite hacer nada efectivo, excepto tal vez abortar el producto para evitar un daño mayor que puede resultar de distribuirlo a los clientes.

### *Características Principales*

- El foco de este nivel es la medición. Implementación de un plan de métricas para evaluar el proceso de la organización.
- La calidad del producto debe ser alta y predecible.

Son muy raras las empresas que han decidido implementar este nivel. No son muchos especialistas de procesos que realmente tengan experiencia práctica, o incluso que entiendan bien las áreas claves de proceso del nivel 4. Son solamente 2 prácticas, pero imposibles de alcanzar si no se ha implementado firmemente los 2 niveles de madures anteriores.

### *Las Áreas Claves de Proceso del Nivel 4*

Las áreas claves de proceso del nivel 4 o Administrado son:

- Administración cuantitativa del proceso
- Administración de la calidad del software

## **2.2.5. Nivel 5: El Mejoramiento Permanente**

### **La Calidad Planificada y Confiable**

En el Nivel Optimizado, la característica principal es el mejoramiento continuo del proceso, en base a la realimentación cuantitativa y al ensayo de ideas y tecnologías innovadoras.

### ***Beneficios de implantar el Nivel 5 del CMM***

La organización entera se aboca al mejoramiento continuo del proceso. La corporación cuenta con los medios para identificar las debilidades y reforzar en forma proactiva el proceso, con objeto de prevenir la ocurrencia de defectos.

Los datos relativos a la eficacia del proceso de software se usan para analizar el costo y el beneficio de usar nuevas tecnologías y de implementar cambios al proceso de software.

Los proyectos de software analizan los defectos para determinar sus causas. Los procesos de software se evalúan para prevenir que los defectos conocidos vuelvan a ocurrir, asimismo las lecciones aprendidas son difundidas a otros proyectos.

### ***Características Principales***

- Prevención de los defectos. Seguimiento y análisis.
- Mejora continua de los procesos (calidad, productividad y tiempos de ciclos)
- Administración de la incorporación de nuevas tecnologías.

No existen más de 10 empresas en el mundo que estén a este nivel. Y las pocas que lo han logrado no divulgan sus secretos para mantener su ventaja competitiva. Este nivel es un estado ideal que probablemente nunca será alcanzado por la mayoría de las empresas productoras de software.

### *Las Áreas claves de proceso del Nivel 5*

Las áreas claves de proceso del nivel 5 o optimizado son :

- Prevención de defectos.
- Administración de cambio de tecnología
- Administración del cambio del proceso.

### **2.3. Áreas Clave de Proceso**

Las áreas clave de proceso son un subconjunto de CMM y están bajo el dominio del mantenimiento y desarrollo de software.

En el CMM hay en total 18 áreas claves de proceso y cada una de ellas está asociada a un solo nivel de madurez. No hay ninguna KPA asociada al nivel 1, el cuál es el nivel de partida y no tiene ningún requisito previo. Los otros niveles de madurez tienen de 2 a 7 KPAs cada uno.

Cada áreas clave de proceso identifica un conjunto de actividades relacionadas las que al ser desarrolladas en forma colectiva alcanzan el conjunto de metas consideradas importantes para mejorar la capacidad de un proceso. Entonces se dice que para alcanzar un cierto nivel de madurez deben ser satisfechas las áreas clave de proceso de ese nivel.

En la tabla 3 se detallan las áreas clave de cada nivel.

<b>Nivel de Madurez</b>	<b>Areas Claves de Proceso (KPA)</b>  (Key Process Areas)
-------------------------	---

<p>5 Optimizante</p>	<p><b>Prevención de defectos</b> (identificación de las causas de defectos y prevención de ellos de reincidencia por medio de su análisis y cambios en procesos definidos).</p> <p><b>Administración de cambios tecnológicos</b> (identificación de los beneficios de nuevas tecnologías {tales como herramientas, métodos y procesos} y transferencia de ellos a la organización en un modo ordenado. Se trata de lograr innovación permanente y eficiente).</p> <p><b>Administración del cambio del proceso</b></p>
<p>4 Administrado</p>	<p><b>Administración cuantitativa del proceso</b> (control de la performance del proceso del proyecto en forma cuantitativa). Esto abarca el registro de los resultados actuales del seguimiento de un proceso de software. Debe enfocarse en la identificación de causales específicas en la variación de un proceso estable y mensurable, también como proceder a corregir las circunstancias que crearon estas desviaciones.</p> <p><b>Administración de la calidad de software</b> (desarrollo de una comprensión cuantitativa de los productos del proyecto para lograr metas de calidad específicas).</p>

<p>3 Definido</p>	<p><b>Enfoque a procesos organizacionales</b> (establecimiento de responsabilidades organizacionales para las actividades, para mejorar la capacidad completa del proceso de la Organización). Definición de procesos organizacionales (desarrollo y mantención de un conjunto útil de activos de proceso que mejoran y pueden ser aplicados en todos los proyectos y, proveen una base para la definición de datos significativos para la administración del mismo en forma cuantitativa. Estos activos pueden ser institucionalizados a través de, por ejemplo, capacitación).</p> <p><b>Definición del proceso de la organización.</b></p> <p><b>Programas de capacitación</b> (desarrollo de habilidades y transferencia de conocimientos al personal para que sea eficiente y efectivo. La capacitación es una responsabilidad de la organización, pero los proyectos deben identificar las habilidades requeridas, y proveer la capacitación necesaria cuando sus necesidades son únicas).</p> <p><b>Administración de software integrado</b> (integración de actividades de administración con Ingeniería de Software. Es un proceso definido y coherente). Ingeniería de productos de software (ejecución consistente de un proceso correctamente definido que integra todas las actividades técnicas {análisis de requerimientos, diseño, código y prueba, entre otros} para producir productos de software consistentes y correctos en forma efectiva y eficiente).</p> <p><b>Ingeniería del producto de software.</b> En CMM se entiende por este término lo que se conoce como ingeniería de software.</p> <p><b>Coordinación intergrupala</b> (establecimiento de algún medio en que el equipo de ingeniería de software participe activamente con otros equipos de ingeniería {trabajo interdisciplinario} con el propósito de que el equipo completo del proyecto satisfaga en mejor forma los requerimientos del cliente).</p> <p><b>Revisiones entre pares</b> (remoción de defectos de los productos temprana y eficientemente. Las revisiones de a pares es un método efectivo e importante que puede implementarse, por ejemplo, a través de inspecciones).</p>
-----------------------	---

<p style="text-align: center;">2 Repetible</p>	<p><b>Administración de requerimientos</b> (comprensión común de los requerimientos del cliente/usuario entre éste y el equipo de trabajo del proyecto. Este acuerdo es la base efectiva para la planificación y administración de un proyecto).</p> <p><b>Planificación del proyecto de software</b> (establecimientos de planes razonables para la administración e ingeniería de un proyecto. Estos planes son la base de la administración del proyecto).</p> <p><b>Seguimiento y ajuste del proyecto de software</b> (visión adecuada del progreso actual, para que la administración tome acciones efectivas cuando la performance del proyecto se desvíe significativamente de los planes).</p> <p><b>Administración de los subcontratos de software</b> (selección de contratistas calificados y administración efectiva de ellos). Aseguramiento de calidad de software (provisión a administración de visión adecuada del proceso siendo usado y de la construcción de productos).</p> <p><b>Aseguramiento de la calidad del software.</b></p> <p><b>Administración de la configuración del software</b> (establecimiento y mantención de la integridad de los productos del proyecto, en todo su ciclo de vida).</p>
<p>1 Inicial</p>	<p>No tiene.</p>

**Tabla 3.-** Áreas claves de los niveles de CMM.

Fuente : [USM –1995]

## **2.4. Metas**

Las metas resumen las prácticas necesarias de implementar para cada área clave de proceso. Estas prácticas pueden ser usadas para determinar si una organización o proyecto ha implementado una determinada área clave de proceso. Las metas indican el alcance, los límites y el sentido de cada área clave de proceso. La satisfacción de una KPA se encuentra determinada por el logro de las metas.

La tabla 4 se muestran las metas de cada área clave de proceso en

cada uno de los niveles.

Nivel	Áreas Clave	Metas
<p style="text-align: center;"><b>2</b></p> <p style="text-align: center;"><b>Repetible</b></p>	<p style="text-align: center;"><b>Administración de Requerimientos</b></p>	<p>1: Los requerimientos del sistema asignados al software son controlados para establecer una referencia para el uso de la ingeniería de software y la administración.</p>
	<p style="text-align: center;"><b>Planeación de Proyectos de Software</b></p>	<p>1: Los estimados de software son documentados para su uso en la planificación y seguimiento del proyecto de desarrollo de software.</p> <p>2: Las actividades y compromisos del proyecto de desarrollo de software son planificados y documentados.</p>
	<p style="text-align: center;"><b>Seguimiento del Proyecto de Software</b></p>	<p>1: A los resultados actuales y el desempeño se les da seguimiento comparándolos contra los planes de desarrollo de software.</p> <p>2: Se toman – y administran hasta la terminación – acciones correctivas cuando los resultados actuales y el desempeño sufren desviaciones significativas de los planes de desarrollo de software.</p>
	<p style="text-align: center;"><b>Administración de Subcontratos de Software</b></p>	<p>1: El contratante selecciona subcontratistas de software calificados.</p> <p>2: El contratante y el subcontratista definen de común acuerdo los compromisos de uno con el otro.</p> <p>3: El contratante y el subcontratista mantienen comunicación a lo largo del proyecto.</p> <p>4: El contratante da seguimiento a los resultados actuales y al desempeño del subcontratista versus sus compromisos previamente contraídos.</p>
	<p style="text-align: center;"><b>Aseguramiento de la Calidad del Software</b></p>	<p>1: Las actividades de aseguramiento de la calidad del software (SQA) son planificadas.</p> <p>2: Se verifica de una manera objetiva la concordancia de los productos de software y de las actividades con los estándares, procedimientos, y requerimientos aplicables.</p>
	<p style="text-align: center;"><b>Administración de la Configuración del Software</b></p>	<p>1: Las actividades de administración de la configuración del software (SCM) son planificadas.</p> <p>2: Productos del trabajo de desarrollo de software son identificados, controlados, y se mantienen disponibles.</p>
		<p style="text-align: center;"><b>Enfoque de la Organización</b></p>

		software para la organización es desarrollado y es actualizado.
<b>3 Definido</b>	<b>Definición del Proceso De la Organización</b>	<b>2:</b> La información relacionada con el uso del Proceso estándar de desarrollo de software de la organización es reunida, revisada, y puesta a disposición de los interesados.
	<b>Programa de Capacitación</b>	<b>1:</b> Las actividades de entrenamiento son planificadas.
		<b>2:</b> Se proporciona el entrenamiento necesario para el desarrollo de las habilidades y los Conocimientos necesarios para la realización de los roles administrativos y técnicos.
	<b>Administración del Software Integrado</b>	<b>1:</b> El desarrollo de software actualizado es definido como una versión cortada a la medida del proceso estándar de desarrollo de software de la organización.
		<b>2:</b> El proyecto en particular es planeado y administrado de acuerdo al desarrollo de Software actualizado y definido para el proyecto.
	<b>Ingeniería de Software</b>	<b>1:</b> Las tareas de ingeniería de software son Definidas, integradas, y realizadas Consistentemente para producir el software.
<b>2:</b> Los productos del trabajo de desarrollo de Software se mantienen consistentes uno con el otro.		
<b>Coordinación Intergrupala</b>	<b>1:</b> Los requerimientos del cliente son acordados por todos los grupos afectados.	
	<b>2:</b> Los compromisos contraídos entre los grupos de ingeniería son acordados por los grupos Afectados.	
	<b>3:</b> Los grupos de ingeniería identifican, dan Seguimiento, y resuelven los asuntos Intergrupales.	
<b>Revisión Entre Pares</b>	<b>1:</b> Las revisiones entre pares son planificadas.	
	<b>2:</b> Los defectos en los productos del trabajo de Desarrollo son identificados y corregidos.	
<b>4 Administrado</b>	<b>Administración Cuantitativa del Proceso</b>	<b>1:</b> Las actividades de administración cuantitativa del proceso son planificadas.
		<b>2:</b> El desempeño del desarrollo de software actualizado y definido del proyecto es controlado cuantitativamente.
<b>3:</b> La capacidad del proceso estándar de desarrollo de software de la organización es conocida en términos cuantitativos.		
	<b>Administración de la Calidad del Software</b>	<b>1:</b> Las actividades de administración de la calidad del proyecto son planificadas.
		<b>2:</b> Metas medibles para la calidad del producto de software y sus prioridades son definidas.
		<b>3:</b> El progreso actual hacia el logro de las metas de calidad para los productos de software es cuantificado y administrado.
	<b>Prevención de Defectos</b>	<b>1:</b> Las actividades de prevención de defectos son planificadas.
		<b>2:</b> Las causas más comunes de defectos son señaladas e identificadas.
		<b>3:</b> Las causas más comunes de defectos son

<b>5 Optimizado</b>		priorizadas y sistemáticamente eliminadas.
	<b>Administración del Cambio Tecnológico</b>	<b>1:</b> La incorporación de cambios tecnológicos es planificada.
		<b>2:</b> Las nuevas tecnologías son evaluadas para determinar su efecto en la calidad y en la productividad.
		<b>3:</b> Las nuevas tecnologías que sean consideradas apropiadas son transferidas a las prácticas normales a través de la organización.
	<b>Administración del Cambio de Proceso</b>	<b>1:</b> La mejora continua del proceso es planificada.
		<b>2:</b> La participación en las actividades de mejora del proceso de desarrollo de software involucra a toda la organización.
<b>3:</b> El Proceso estándar de desarrollo de software de la organización y el desarrollo de software actualizado y definido para el proyecto son mejorados continuamente.		

**Tabla 4.-** Metas de cada área clave.

## **2.5. Características Comunes**

Las prácticas que deben ser realizadas por cada Area Clave de Proceso están organizadas en 5 características comunes, las cuales constituyen atributos que indican si la implementación y la institucionalización de un proceso clave es efectivo, repetible y duradero.

Las prácticas identificadas en la característica común de las Actividades Realizadas describen qué se debe implementar para establecer una capacidad de proceso. El resto de las otras prácticas, tomadas en conjunto, forman la base mediante la cual una organización puede institucionalizar las prácticas descritas en la característica común de las Actividades Realizadas.

### ***1. El compromiso para realización***

El compromiso para realización describe las acciones que la organización debe tomar para asegurar que el proceso se establezca y sea permanente. Normalmente abarcan el establecimiento de políticas y liderazgo a nivel de la organización. .

## ***2. La capacidad de realización***

La capacidad de realización describe las condiciones previas que debe existir en el proyecto o la organización para poder aplicar el proceso de software en forma efectiva. Normalmente enfocan los recursos, las estructuras de la organización, el entrenamiento y las condiciones previas que deben existir.

## ***3. Las actividades realizadas***

Las actividades realizadas describen los papeles y los procedimientos necesarios para implementar un área clave de proceso. Normalmente abarcan el establecimiento de planes y procedimientos, la ejecución del trabajo, el seguimiento del mismo, el control de las salidas del proceso, y las acciones correctivas que deben tomarse según sea necesario.

## ***4. Las mediciones y análisis***

Las Mediciones y Análisis describen la necesidad de medir el proceso y analizar las mediciones resultantes. Normalmente incluye ejemplos de las mediciones que se podrían tomar para determinar el estado y la eficacia de las actividades realizadas.

## ***5. La Verificación de la implementación***

La Verificación de la implementación describe los pasos que se deben seguir para asegurar que las actividades se llevan a cabo de acuerdo con el proceso establecido. Típicamente abarca las revisiones y las auditorías efectuadas por la alta gerencia, los jefes de proyecto y el grupo encargado de la garantía de calidad de software.

[Psulk-1993a]

## **2.6. Proyecciones**

El desarrollo de este modelo puede tomar de entre 2 a 3 años de nivel Inicial a el nivel Repetible, y entre 18 a 24 meses para los niveles siguientes. Para ello se debe adoptar una filosofía de “mejora continua”. Pero los riesgos siempre están presentes, algunos de ellos pueden ser :

- Intentar saltarse niveles o fijar tiempos arbitrarios
- Falta de compromiso político
- Falta de recursos
- Resistencia al cambio de cultura
- Expectativas de retorno de inversión no realistas

Según datos del Software Engineering Institute (SEI) organismo que regula toda la temática del CMM a escala mundial, a agosto del 2001, de un total de 964 organizaciones evaluadas según el modelo CMM alrededor del planeta, un 45,6% de ellas tenía menos de 100 empleados, y de estas, casi la mitad tenía menos de 50 empleados. Esto permite visualizar que el uso de esta herramienta es aplicable a cualquier tamaño de organización.

Por otra parte, las estadísticas del SEI son contundentes en cuanto a los beneficios que las empresas que han entrado en el proceso de mejoramiento utilizando como herramienta el CMM han obtenido:

- Los defectos del software se redujeron hasta en un 90%
- Los costos se redujeron hasta en un 50%.
- El tiempo de ciclo se redujeron en un 36%.
- La productividad mejoró en un 62%.

Toda la información anterior hace que sean naturales algunas conclusiones sobre la compatibilidad del uso de herramientas como el CMM y las empresas desarrolladoras de software jóvenes o pequeñas:

- a. El CMM es aplicable a cualquier tamaño de organización.
- b. Los beneficios de producir con calidad superan con creces los costos de la no calidad.
- c. El que una empresa sea joven o pequeña puede representar una ventaja en la implementación del modelo CMM pues la organización es más flexible en función de su tamaño, y la adopción de una cultura de calidad debería facilitarse.

Para iniciar un proceso de mejora primero que nada se debe tener claro en que nivel se encuentra la empresa u organización. El SEI desarrollo un método para evaluar el Nivel de Madurez de una organización, basado en el CMM. Este método se llama Software Process Assessment, y se basa en: la revisión de 4 a 6 proyectos, o respuestas a un cuestionario, o a través de entrevistas con líderes de proyectos y personal de desarrollo.

Algunos números obtenidos de empresas de software que optaron por utilizar CMM en los E.E.U.A. en 1999, nos muestran en que estado de madurez se encuentran sus procesos : 40% en inicial, 35% en repetible, 18% en definido, 5% en administrado y 2% (o menos) en optimizado. [Redi-2000]

Las empresas al implementar este modelo podrán obtener algunos beneficios como :

- Establecimiento de las bases para el mejoramiento continuo, al definir un plan de trabajo enfocado hacia las áreas claves de los procesos.
- Mejora de la productividad en el desarrollo de software al reducir la cantidad de re-trabajo, detectando errores en las etapas iniciales del desarrollo y mejorando la presupuestación de costos.
- Estandarización de las prácticas establecidas.
- Mejoramiento de la calidad y satisfacción del cliente, mediante una mejor administración de los requerimientos e identificación temprana de errores.
- Mejor cumplimiento de los tiempos de entrega estimados (y ofrecidos), a través de la planificación.
- Mejores condiciones de trabajo al incrementar el espíritu de trabajo en equipo, disminuir el estrés y las horas extra.

[CEGESTI-2001]

## **2.7. Críticas al CMM**

- Niveles mas altos imposibles de definir y casi imposibles de lograr.
- Es una escala, no se puede llegar al nivel 4 sin antes haber pasado por los niveles anteriores.
- Algunas KPAs son poco adecuadas para organizaciones de desarrollo cuyo desafío principal es salir rápidamente al mercado con un producto innovador, ya que implementarlas puede tomar mucho tiempo.
- El modelo de madurez favorece más al proceso de mantención que al proceso de desarrollo, ya que su objetivo principal es el alcanzar un proceso que pueda ser medido y controlado como base para un mejoramiento continuo.
- Problemas metodológicos y estadísticos en la determinación del nivel de madurez, esto debido a que el nivel se determina mediante el score o puntuación obtenida en el cuestionario de evaluación. Entonces el nivel de madurez se establece como aquel en que se satisfacen todas las áreas claves en forma continua.
- El nivel 1 puede ser alcanzado sin esfuerzo.

## **2.8. Estado Actual**

El estado actual en que se encuentra la empresa y por ende el proceso de migración en estudio es en nivel de madurez inicial. La empresa posee prácticas de gestión mínimas. Algunos de los proyectos pueden ser repetidos con éxito, pero solo en algunos casos. Hay una total dependencia en los individuos. Y los planes para los proyectos tienden a ser poco realistas.

Las áreas claves de procesos para el nivel 2 no son cumplidas, estas son :

- **Administración de Requerimientos** : No es cumplida.
  
- **Planificación de Proyectos** : No es cumplida.
  
- **Seguimiento y supervisión de proyecto** : No es cumplida.
  
- **Administración de Subcontratos de Software** : No es cumplida.
  
- **Aseguramiento de la Calidad del Software** : No es cumplida.
  
- **Administración de la Configuración del Software** : No es cumplida.

## CAPÍTULO III :

### **3. Formulación de la Propuesta de un Modelo para Asegurar la Calidad en un Proceso Migratorio de un Software**

En este punto se generará la propuesta de un modelo para asegurar la calidad en un proceso de migración de un software.

La migración desde una base de datos Oracle a una Sybase puede producir complicaciones en el código del programa, lo cual puede afectar el rendimiento de éste. Por ejemplo la función DECODE de Oracle no existe en Sybase, pero si existe la función CASE. Ambas expresiones pueden ejecutar una evaluación condicional dentro de una query.

Otro caso es el de las palabras similares en su funcionamiento, pero diferentes en la sintaxis, por ejemplo la función SUBSTR(char,m,n) de Oracle y SUBSTRING(char,m,n) de Sybase. Ambas funciones retornan la parte del char que comienza en la posición “m” con largo “n”. O la función NVL(exp1,exp2) de Oracle y a función ISNULL(exp1,exp2). Estas funciones sustituyen el valor especificado en la exp2 cuando la exp1 es NULL.

Los tipos de datos también son diferentes en ambas bases de datos, algunos ejemplos son : DATE en Oracle y DATETIME en Sybase, o VARCHAR2 en Oracle y VARCHAR en Sybase.

La conexión ODBC, también varia para ambas bases de datos.

Estos son solo algunos casos que diferencian a ambos lenguajes estructurados.

Es por ello y para asegurar el correcto funcionamiento del software una vez finalizado el proceso de migración que se hace necesario contar con un plan que permita asegurar la calidad. Estas normas pretenden guiar en algún proceso de migración basado en la experiencia personal y en como dado los problemas presentados se podrían obtener mejores resultados en un proceso migratorio. Además éstos pasos pueden ser generalizados a otras bases de datos, no solo para el caso desarrollado en esta tesis.

### 3.1. Propuesta de un Modelo para Asegurar la Calidad en un Proceso de Migración de un Software

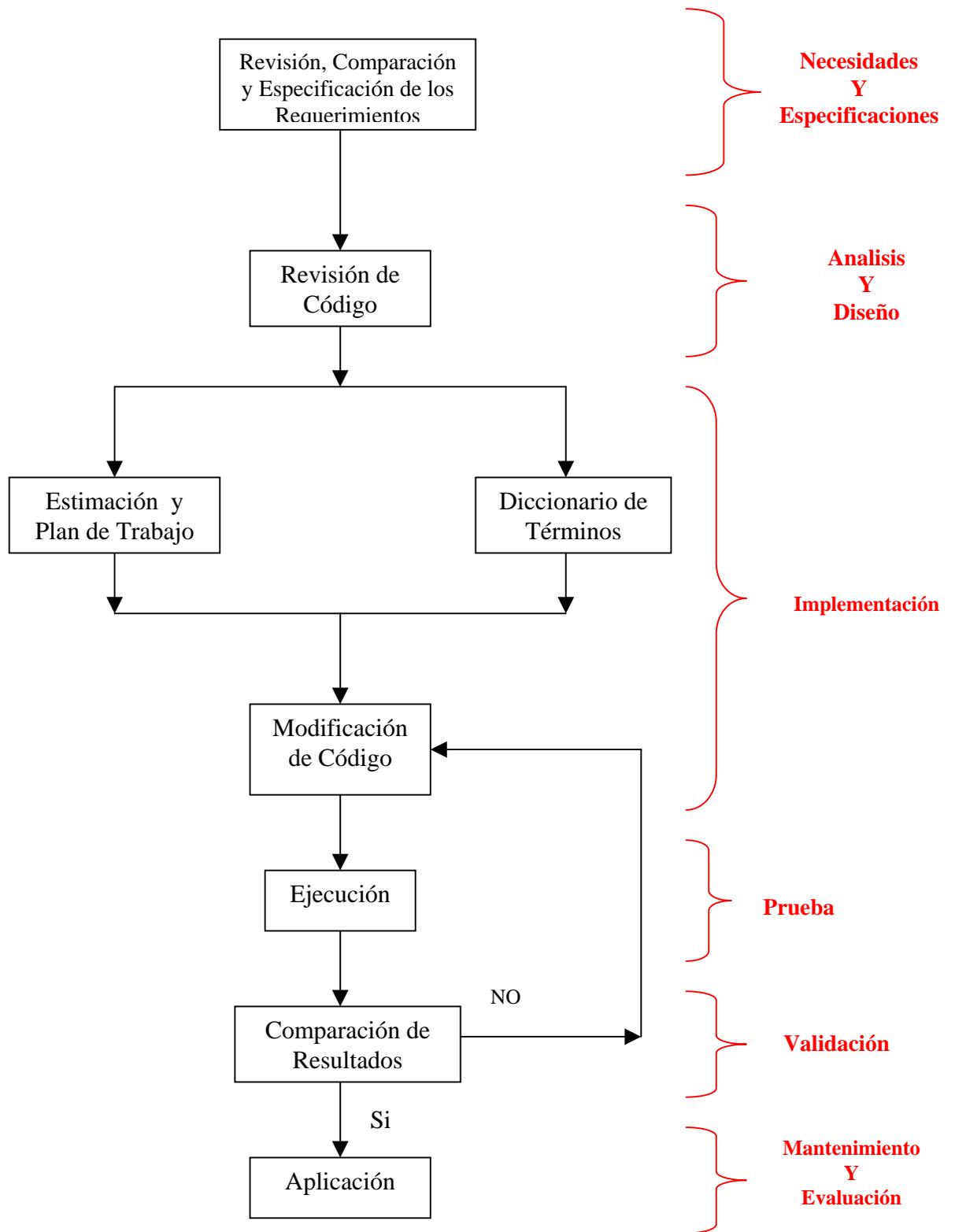


Fig. 3 .- Modelo propuesto para asegurar la calidad

### **3.1.1. Análisis, Revisión, Comparación y Especificación de los**

#### **Requerimientos**

El sistema a migrar posee sus funcionalidades ya establecidas y definidas al ser creado, pero puede que para este nuevo sistema, resultante de la migración, los requerimientos y las funcionalidades previas sean otras.

Se obtendrá así un nuevo documento que definirá con mayor precisión los atributos que debe tener el nuevo sistema requerido por el cliente .

Se debe describir el sistema en término de sus usos y usuarios, evitando consideraciones técnicas, mas bien en lenguaje simple de manera que sea claro.

Se determinan formalmente los requerimientos, lo que el sistema debe cumplir, y se determina la solución abstracta estableciendo modelos de comportamiento y de información.

Lo más probable será que no resulte posible obtener una buena especificación del sistema en una primera aproximación; serán necesarias sucesivas versiones del documento en que irán quedando reflejada la evolución de las necesidades del cliente (por una parte no siempre sabe en los primeros contactos todo lo que quiere realmente, y por otra parte pueden surgir cambios externos que supongan requerimientos nuevos o modificaciones de los ya contemplados).

- **Identificación del Problema** : en este proceso de debe obtener del cliente y futuros usuarios todos los antecedentes relacionados con el sistema. Interesa saber cuál es el problema a solucionar, que se desea cambiar del sistema antiguo y el por que?, además si requieren nuevas necesidades.
- **Requisitos del Sistema** : esta etapa es una descripción de alto nivel computacional del sistema como solución al problema identificado. Además la funcionalidad debe incluir tanto la original como la nueva.

- Especificación de Software :interpretación de los requisitos del sistema en términos de una descripción abstracta y de alto nivel del diseño del sistema. Los requisitos se expresan en lenguaje más formal y se acompañan de diagramas.
- Convalidación de la Especificación : esta etapa muchas veces ignorada puede redundar en importantes ahorros de tiempo y esfuerzos posteriores. Consiste en verificar la consistencia de los requerimientos, que no existan sinónimos, contradicciones y convalidar que sean realistas con lo que se necesita.

### **3.1.2. Revisión Detallada del Código del Software a Migrar**

Es necesario iniciar la tarea de migración del software con la revisión detallada del código fuente.

La revisión del código permite obtener una real magnitud del cambio que será necesario efectuar el proceso migratorio del sistema. A partir de esta revisión se podrá elaborar el plan de trabajo y el diccionario de términos para llevar a cabo la migración en forma exitosa.

Esta etapa se compara a las de Análisis y Diseño de la ingeniería de software.

En el análisis del sistema es necesario determinar que elementos intervienen en el sistema a migrar, así como su estructura, relaciones, evolución en el tiempo, detalle de sus funcionalidades, esto dará una descripción clara de qué sistema se va a construir, y qué nuevas funcionalidades va a aportar.

Mientras que en el análisis ya se tiene claro que debe hacer el sistema, el diseño permite determinar como se va a hacer. En este caso ya se tiene claro el lenguaje en el que se programará, ya que será el mismo del sistema original a migrar y la base de datos que es la requerida por el cliente.

### **3.1.3. Estimación y Plan de Trabajo**

A partir de la revisión del código fuente se puede elaborar el plan de trabajo para el proceso migratorio, y estimar el tiempo de duración del mismo.

- **Estimación** : mediante el modelo de Puntos de Función se puede obtener una estimación de la duración del sistema en base a los requerimientos anteriormente obtenidos.

#### **❖ Método de Análisis Puntos de Función MkII [Dolado - 2001]**

Los puntos de función MkII se obtienen a partir del producto de dos componentes, el "tamaño de procesamiento de la información" y el "ajuste de complejidad técnica"

En vez de utilizar cinco componentes: entradas, salidas, interfaces, consultas y archivos lógicos, el sistema se examina desde el punto de vista de las transacciones lógicas, consistiendo cada una de ellas en entrada, proceso y salida. Se define una transacción lógica como una combinación única de entrada/proceso/salida desencadenada por un único evento de interés para el usuario, o una necesidad de recuperar información.

La fórmula para obtener los puntos de función no ajustados es la siguiente :

$$UFP = W_1 * N_1 + W_E * NE + W_o * No \quad (1)$$

*Donde :*

$W_1$  : Peso para un dato de entrada

$N_1$  : N° de datos de entrada

$W_E$  : Peso para una entidad referenciada

$NE$  : N° de entidades referenciadas

$W_o$  : Peso para un dato de salida

$No$  : N° de datos de salida

Se pueden utilizar pesos calibrados en el propio entorno o bien utilizar pesos estándares, como los siguientes :

$$W_1 = 0.58 , \quad W_E = 1.66 , \quad W_o = 0.26$$

Es necesario también calcular el ajuste de la complejidad técnica (TCA), esto mediante la fórmula 2.

$$TCA = 0.65 + C * (\Sigma \text{ grado de influencia de las 19 características}) \quad (2)$$

*Donde :*

$C$  : valor actual medio industrial = 0.005

El ajuste de complejidad técnica refleja el grado de influencia de 19 factores, cada una de estas características tiene un grado de importancia de 0 a 5, y la suma de todas estas características se utiliza para calcular este factor, que puede variar de 0.65 a 1.35.

Las reglas de puntuación general de la influencia de cada característica se muestran en la siguiente figura :



**Fig. 4 .-** Valores de ajuste de complejidad

Estas reglas generales se sustituyen por las reglas específicas como se indica a continuación :

1) Comunicación de Datos

- 0** Aplicación es batch exclusivamente
- 1-2** Impresión o entrada de datos remota
- 3-5** Teleproceso (TP) interactivo
- 3** TP interface a un proceso batch
- 5** La aplicación es interactiva

2) Función Distribuída. "Distribuída" significa que los componentes de la aplicación están distribuídos en dos o más procesadores diferentes.

- 0** La aplicación no ayuda a la trasferencia de datos o a la función de procesamiento entre los componentes del sistema
- 1** La aplicación prepara datos para el usuario final de otro procesador

**2-3** Los datos se preparan para transferencia, se transfieren y se procesan en otro componente del sistema

**4** Igual que 2-3, pero con realimentación al sistema inicial

**5** Las funciones de procesamiento se realizan dinámicamente en el componente más apropiado del sistema.

3) Rendimiento (referido a la importancia de respuesta dentro de todo el sistema)

**0-3** Análisis y diseño de las consideraciones del rendimiento son estándar. No se requieren requerimientos especiales por parte del usuario

**4** En la fase de diseño se incluyen tareas del análisis del rendimiento para cumplir los requerimientos del usuario

**5** Además se utilizan herramientas de análisis del rendimiento en el diseño, desarrollo e instalación

4) Configuración utilizada masivamente (referente a la importancia del entorno)

**0-3** La aplicación corre en una máquina estándar sin restricciones de operación

**4** Restricciones de operación requieren características específicas de la aplicación en el procesador central

**5** Además hay restricciones específicas a la aplicación en los componentes distribuidos del sistema.

5) Tasas de Transacción (una alta llegada de transacciones provoca problemas más allá de los de la característica 3)

**0-3** Las tasas son tales que las consideraciones de análisis de rendimiento son estándares

**4** En la fase de diseño se incluyen tareas de análisis de rendimiento para verificar las altas tasas de transacciones

**5** Además se utilizan herramientas de análisis del rendimiento.

6) Entrada On-Line de datos

**0-2** Hasta el 15% de las transacciones tienen entrada interactiva

**3-4** 15% al 30% tienen entrada interactiva

**5** 30% al 50% tienen entrada interactiva.

7) Diseño para la eficiencia de usuario final

**0** Sistema batch

**1-3** No se especifican requerimientos especiales

**4** Se incluyen tareas de diseño para la consideración de factores humanos

**5** Además se utilizan herramientas especiales o de prototipado para promover la eficiencia.

8) Actualización On-Line

**0** Nada

**1-2** Actualización on line de los ficheros de control. El volumen de actualización es bajo y la recuperación fácil.

**3** Actualización on line de la mayoría de los ficheros internos lógicos

**4** Además es esencial la protección contra la pérdida de datos

**5** Además se considera el coste de recuperación de volúmenes elevados.

9) Complejidad del procesamiento (esto es complejidad interna)

¿Qué características tiene la aplicación?

- mucho procesamiento matemático y/o lógico
- muchas excepciones de procesamiento, muchas transacciones incompletas y mucho reprocesamiento de las transacciones
- procesamiento de seguridad y/o control sensitivo

**0** No se aplica nada de esto

**1-3** Se aplica alguna cosa

**4** Se aplican dos cosas

**5** Se aplica todo.

10) Utilizable en otras aplicaciones (el código se diseña para que sea compartido o utilizable por otras aplicaciones - no confundir con 13).

**0-1** Una aplicación local que responde a las necesidades de una organización usuaria

**2-3** La aplicación utiliza o produce módulos comunes que consideran más necesidades que las del usuario

**4-5** Además, la aplicación se "empaquetó" y documentó con el propósito de fácil reutilización

### 11) Facilidad de Instalación

**0-1** No se requieren por parte del usuario facilidades especiales de conversión e instalación

**2-3** Los requerimientos de conversión e instalación fueron descritos por el usuario y se proporcionaron guías de conversión e instalación

**4-5** Además se proporcionaron y probaron herramientas de conversión e instalación

### 12) Facilidad de Operación

**0** No se especifican por parte del usuario consideraciones específicas de operación

**1-2** Se requieren, proporcionan y prueban procesos específicos de arranque, backup y recuperación

**3-4** Además la aplicación minimiza la necesidad de actividades manuales, tales como instalación de cintas y papel

**5** La aplicación se diseña para operación sin atención

### 13) Puestos Múltiples. Añadir puntos por cada uno de los siguientes factores

**0** El usuario no requiere la consideración de más de un puesto

**1** De uno a cuatro puestos

**2** Cinco o más puestos

**1** Se proporciona documentación y plan de apoyo para soportar la aplicación en varios lugares

**2** Los puestos están en países diferentes

14) Facilidad de Cambio (esfuerzo específico de diseño para facilitar cambios futuros). Añadir puntos por cada uno de los siguientes factores

**0** No hay requerimientos especiales del usuario para minimizar o facilitar el cambio

**1-3** Se proporciona capacidad de consulta flexible

**1-2** Datos importantes de control se mantienen en tablas que son actualizadas por el usuario a través de procesos on-line interactivos.

15) Requerimientos de otras Aplicaciones

**0** El sistema es stand-alone

**1-4** Requerimientos del sistema para interfaces o compartición de datos deben ser sincronizados con otras aplicaciones

**5** Se deben sincronizar los requerimientos del sistema con cuatro o más aplicaciones

16) Seguridad, Privacidad y Auditoría. Añadir puntos por cada uno de los siguientes factores

**1** Si el sistema debe cumplir requerimientos personales (incluso legales) de privacidad

**1** Si el sistema debe cumplir requerimientos especiales de auditoría

**1-2** Si el sistema debe cumplir requerimientos excepcionales de seguridad para prevenir pérdidas de naturaleza financiera o militar

**1** Si se requiere el criptografiado de los datos de las comunicaciones

17) Necesidad de Adiestramiento al Usuario

**0** Si no es necesario material ni cursos

**1-4** Si se requiere entrenamiento especial o se proporcionan facilidades de ayuda on-line

- utilización de facilidades de "help" estándares

- desarrollo de " " especiales

- entrega de material especial de adiestramiento

**5** Se requiere un sistema separado de entrenamiento o simulador

18) Uso directo de otras empresas

**0** No existe otra empresa conectada al sistema

**1** Los datos se envían o reciben de empresas conocidas

**2** Empresas conocidas están conectadas al sistema en modo de lectura solamente

**3-4** Empresas conocidas están conectadas directamente al sistema con capacidad de actualización on-line

**5** Empresas desconocidas (público en general o un grupo demasiado extenso como para ser adiestrado) pueden acceder al sistema

19) Documentación. Contar 1 por cada tipo de documento que entrega al final del proyecto.

- Especificación Funcional del Sistema (datos y procesos)
- Especificación Técnica del Sistema
- Documentación del programa (al menos diagramas de flujo)
- Librería de Elementos de Datos
- Elemento de Datos/ Registro/ X-referencia del programa
- Manual de usuario
- Folleto de información general del sistema
- Librería de datos de prueba
- Material de curso de adiestramiento al usuario
- Documentos de coste/beneficio del sistema
- Informe de petición de cambios y errores

Se calcula el grado de influencia utilizando los valores :

- |          |                             |
|----------|-----------------------------|
| <b>0</b> | si 0-2 tipos de documento   |
| <b>1</b> | si 3-4                      |
| <b>2</b> | si 5-6                      |
| <b>3</b> | si 7-8                      |
| <b>4</b> | si 9-10                     |
| <b>5</b> | si 11-12 tipos de documento |

Para poder calcular el tiempo de entrega del proyecto necesitamos primero obtener :

- Tamaño del proyecto :

$$S = UFP * TCA \quad (3)$$

- Tasa de tiempo de entrega o puntos de función por semana :

$$D = 0.45 * \sqrt{S} \quad (4)$$

Así obteniendo estos valores se puede calcular el tiempo de entrega como :

$$\text{Tiempo de Entrega} = \frac{\text{Tamaño}}{\text{Tasa de Entrega}} = \frac{S}{D} \quad (5)$$

- **Plan de Trabajo** : debe incluir los recursos asignados al proyecto, niveles y tiempo de capacitaciones requerido, plan con fechas de entrega de versiones del producto o primeras pruebas y demostraciones.

#### **3.1.4. Elaboración de un Diccionario de Términos.**

Al igual que en la elaboración del plan de trabajo, el diccionario de términos se genera a partir de la revisión del código fuente del software a migrar.

Este diccionario debe incluir las palabras claves del código original y la palabra clave del nuevo lenguaje que la reemplazará.

De esta forma cada vez que se revise una instrucción SQL se consultará este diccionario, evitando demoras y posibles errores.

### **3.1.5. Modificación de Cada Query (o Instrucción SQL)**

Con las etapas anteriores ya realizadas se puede proceder a revisar y modificar cada una de las sentencias o instrucciones SQL las que se encuentran en el lenguaje original del software y que se cambiarán si es necesario al nuevo lenguaje.

Esta etapa se relaciona con la de implementación de ingeniería de software en donde se empieza a codificar algoritmos y estructuras de datos, definidos en las etapas anteriores, en el correspondiente lenguaje de programación y/o para un determinado sistema gestor de bases de datos.

### **3.1.6. Ejecución de Cada una de la Queries (o Instrucción SQL)**

Luego de modificar una instrucción SQL es necesario probar su correcto funcionamiento, es decir que no genere errores en la base de datos y que los resultados obtenidos sean los esperados.

Esta etapa se asemeja a la etapa de Prueba en la ingeniería de software, cuyo objetivo es garantizar que el sistema ha sido desarrollado correctamente, sin errores de diseño y/o programación. Es conveniente que sean planteadas al menos tanto a nivel de cada módulo (aislado del resto), como de integración del sistema (según sea la naturaleza del proyecto en cuestión se podrán tener en cuenta pruebas adicionales, p.ej. de rendimiento).

### **3.1.7. Comparación de los Resultados con los Requerimientos**

#### **Solicitados.**

Esta etapa tiene como objetivo la verificación de que el sistema desarrollado cumple con los requisitos expresados inicialmente por el cliente y que han dado lugar al presente proyecto.

De esta manera se puede asegurar que a partir del cambio y posterior comprobación de resultados de cada instrucción se llegará al correcto funcionamiento y cumplimiento de los requerimientos del software migrado.

### **3.1.8. Aplicación de la Migración.**

Finalmente la aplicación resultante se encuentra ya en fase de producción (en funcionamiento para el cliente, cumpliendo ya los objetivos para los que ha sido creada). A partir de este momento se entra en la etapa de mantenimiento, que supondrá ya pequeñas operaciones tanto de corrección como de mejora de la aplicación (p.ej. mejora del rendimiento), así como otras de mayor importancia, fruto de la propia evolución (p.ej. nuevas opciones para el usuario debidas a nuevas operaciones contempladas para el producto).

La mayoría de las veces en que se desarrolla una nueva aplicación, se piensa solamente en un ciclo de vida para su creación, olvidando la posibilidad de que esta deba sufrir modificaciones futuras (que tendrán que producirse con casi completa seguridad para la mayor parte de los casos).

### **3.2. Aplicación del Modelo CMM sobre el Modelo Propuesto para Asegurar la Calidad en un Proceso de Migración de un Software**

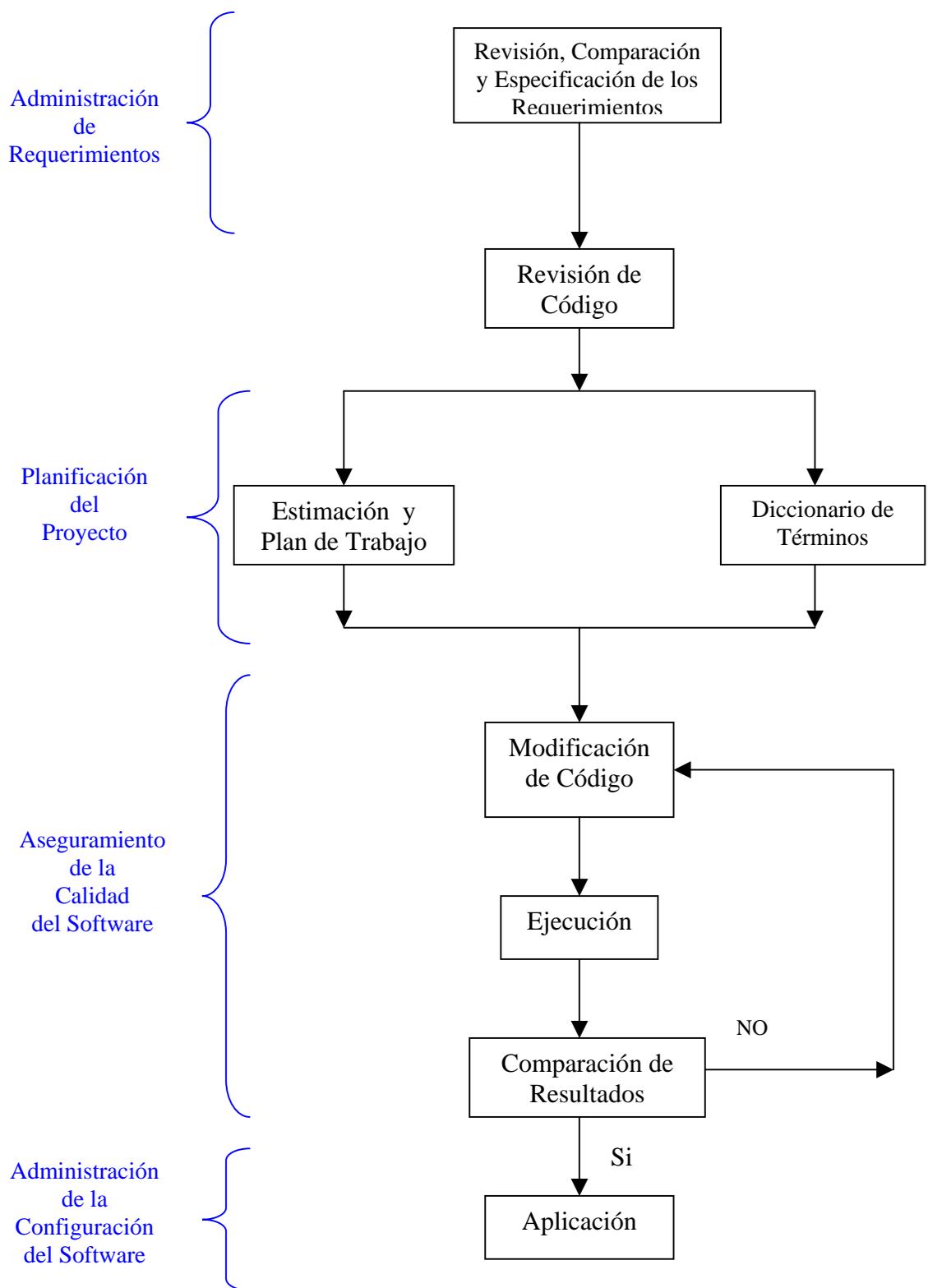
Como ya fue mencionado anteriormente, la empresa y por ende el proceso migratorio en estudio se encuentran en un nivel de madurez inicial. Esto debido a que la empresa posee prácticas de gestión mínimas, solo algunos de los proyectos pueden ser repetidos con éxito, existe una total dependencia sobre los individuos y los planes para los proyectos tienden a ser poco realistas.

Es por ello que el modelo propuesto para asegurar la calidad en un proceso migratorio de un software está basado sobre el nivel 2 del Modelo CMM, que es el nivel que se espera alcanzar.

Las áreas claves del nivel 2 son :

- Administración de Requerimientos
- Planificación del Proyecto
- Seguimiento y Supervisión del Proyecto
- Administración de Subcontratos de Software
- Aseguramiento de la Calidad del Software
- Administración de la Configuración del Software

La figura 5 muestra la aplicación de las áreas claves del nivel 2 del modelo CMM en el modelo propuesto..



**Fig. 5 .-** Aplicación de CMM sobre el modelo propuesto

- **Administración de Requerimientos** : los requerimientos del sistema asignados al software son controlados para establecer una referencia para el uso de la ingeniería de software y la administración.
  
- **Planificación del Proyecto** : los estimados del software son documentados para su uso en la planificación y seguimiento del proyecto de desarrollo de software. Las actividades y compromisos del proyecto de desarrollo de software también son planificados y documentados.
  
- **Seguimiento y Supervisión del Proyecto** : se toman y administran hasta que se termina, acciones correctivas cuando los resultados actuales y el desempeño sufren desviaciones significativas de los planes de desarrollo del software.
  
- **Aseguramiento de la Calidad del Software** : se verifica de manera objetiva la concordancia del producto de software y las actividades con los estándares, procedimientos y requerimientos aplicables.
  
- **Administración de la Configuración del Software** : los productos del trabajo desarrollado son controlados y se mantienen disponibles.

## CAPÍTULO IV :

### **4. Aplicación del Modelo Propuesto en un Ejemplo Concreto de Migración.**

El software a migrar fue desarrollado para una base de datos Oracle 7 y ahora se requiere que funcione sobre una base de datos Sybase 11.

Si bien los lenguajes de programación estructurados para ambas bases de datos son muy similares, poseen diferencias fundamentales en palabras claves, funciones, definición de variables, entre otras, las que pueden afectar el correcto funcionamiento del software pudiendo generar diferentes errores por sintaxis, esto no permite obtener los resultados deseados ni ejecutar el sistema.

Es así que para asegurar el correcto funcionamiento del software se ha planteado el modelo descrito en el punto 3.1, el que ahora será aplicado al ejemplo en estudio.

#### **4.1. Análisis, Revisión, Comparación y Especificación de los Requerimientos**

El objetivo es establecer los servicios que brindará el producto de software y las restricciones bajo las cuales deberá operar. Se distinguen :

- **Identificación del Problema** : en este proceso se debe obtener del cliente y futuros usuarios todos los antecedentes relacionados con el sistema. Interesa saber cuál es el problema a solucionar, que se desea cambiar del sistema antiguo y por que? , que nuevas necesidades son requeridas.

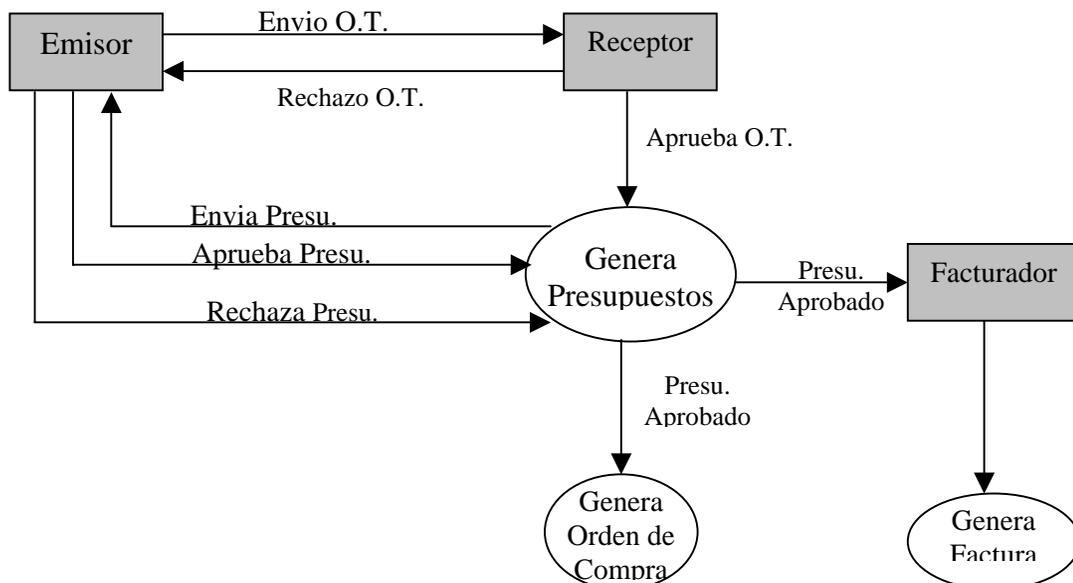
- Requisitos del Sistema : esta etapa es una descripción de alto nivel computacional del sistema como solución al problema identificado. Además la funcionalidad debe incluir tanto la original como la nueva
- Especificación de Software :interpretación de los requisitos del sistema en términos de una descripción abstracta y de alto nivel del diseño del sistema. Los requisitos se expresan en lenguaje más formal y se acompañan de diagramas.

#### **4.2. Revisión Detallada del Código del Software a Migrar**

Se debe revisar la totalidad del código fuente, con el objetivo de establecer la magnitud del cambio necesario para llevar a cabo el proceso migratorio.

Esta revisión se realizará módulo a módulo, permitiendo comprender la programación original, la forma de conexión a la base de datos y la ejecución de cada consulta.

##### **Análisis :**



**Fig. 6.-** Estructura del sistema a migrar.

**Diseño** : ya se tiene claro que el lenguaje de programación será el mismo utilizado en el programa original, Visual Basic 4.0 , y que la base de datos a utilizar será la requerida por el nuevo usuario Sybase 11.

### **4.3. Estimación y Plan de Trabajo**

**Estimación** : utilizando el método de análisis de puntos de función MkII se obtendrá una estimación del tiempo requerido para entregar el sistema.

De la ecuación (1):  $UFP = W_1 * N_1 + W_E * N_E + W_o * N_o$

Y utilizando los valores de los pesos estándares :

$$W_1 = 0.58 , W_E = 1.66 , W_o = 0.26.$$

Se tiene :

$$UFP = 0.58 * N_1 + 1.66 * N_E + 0.26 * N_o$$

Se debe calcular ahora los valores para  $N_1$  ,  $N_E$  y  $N_o$

- **Datos de Entrada ( $N_1$ ) :**

Los datos de entrada para emitir una orden de trabajo (O.T.) son :

- Nombre de usuario
- Password del usuario
- Nombre del cliente
- Marca
- Producto
- Campaña
- Departamento

- Usuario receptor
- Actividades solicitadas
- Propiedades de la actividad
- Fecha solicitada para el trabajo

Los datos de entrada para efectuar una consulta de ordenes como de presupuestos son :

- Estado en que se encuentra la orden o presupuesto
- Departamento
- Nombre del Cliente
- Marca
- Producto
- Campaña
- Fecha de inicio de la búsqueda
- Fecha de término de la búsqueda
- Número de la orden o del presupuesto buscado.

En total **N<sub>I</sub> = 21**

- **Entidades Referenciales (N<sub>E</sub>) :**

- Generar un número de orden de trabajo (al enviar la O.T.)
- Obtener O.T. y presupuestos a partir de las consultas

**N<sub>E</sub> = 2**

- **Datos de Salida (No) :**

- Número de O.T. o presupuesto
- Fecha de envío
- Nombre del cliente
- Marca
- Producto
- Campaña
- Descripción de la O.T.
- Estado
- Departamento
- Tipo de presupuesto (Interno o Externo)

**No = 10**

Por lo tanto reemplazando los valores anteriores en la ecuación (1), se tiene:

$$\text{UFP} = 0.58 * 21 + 1.66 * 2 + 0.26 * 10 = 18.1$$

Ya se ha obtenido el valor de los puntos de función. Ahora se debe encontrar el valor para el ajuste de complejidad técnica. Esto mediante la fórmula (2).

$$\text{TCA} = 0.65 + C * (\Sigma \text{ grado de influencia de las 19 características})$$

La suma de las 19 características evaluadas según el grado de influencia de cada una de ellas para este proyecto en particular es de 30.

Así reemplazando los valores se tiene que :

$$TCA = 0.65 + 0.005 (30) = 0.8$$

Para poder calcular el tiempo de entrega del proyecto se debe obtener además :

- Tamaño del proyecto :  $S = UFP * TCA$

$$S = 18.1 * 0.8 = 14.48$$

- Tasa de tiempo de entrega o puntos de función por semana :

$$D = 0.45 * \sqrt{S}$$

$$D = 0.45 * \sqrt{14.48} = 1.71 \text{ PF/sem}$$

Así obteniendo estos valores se puede calcular el tiempo de entrega como :

$$\begin{aligned} \text{Tiempo de Entrega} &= \frac{\text{Tamaño}}{\text{Tasa de Entrega}} = \frac{S}{D} \\ &= \frac{14.48}{1.71} \\ &= 8.5 \text{ semanas} \end{aligned}$$

### **Plan de Trabajo :**

- Recursos Asignados : 1 persona tiempo completo.
- Plan de trabajo : Se muestra en la tabla 5.

<b>Actividad</b>	<b>Tiempo</b>	<b>Fecha Inicio</b>
Inicio del trabajo		02/04/2001
Proceso de Cambio	1 mes	09/04/2001
Elaboración de Manuales	1 semana	30/04/2001
Entrega 1ª Versión		07/05/2001
Capacitación y Demostración	2 días	07/05/2001
Puesta en Marcha 1ª Versión	1 semana	07/05/2001
Modificaciones	1 semana	14/05/2001
Entrega 2ª Versión		21/05/2001
Puesta en Marcha 2ª Versión	1 semana	21/05/2001
Modificaciones	1 semana	28/05/2001
Capacitación Administrador	1 semana	04/06/2001
Futuras Versiones Garantía 1 año		11/06/2001

**Tabla 5.- Plan de Trabajo**

## **4.4. Elaboración de un Diccionario de Términos**

### **❖ Funciones**

Las funciones encontradas en el proceso migratorio y su definición son mostradas en la tabla 6.

Oracle	Sybase	Definición
<b>Substr</b> (char,m[,n])	<b>Substring</b> (char,m, n)	Retorna la parte de un carácter que comienza en m con largo n.
<b>chr</b> (n)	Char(n)	Convierte n al valor equivalente en ascii.
Sysdate	Getdate	Retorna la hora y fecha actual
Decode(exp,search1,return1,[search2,return2] ... [default])	No Existe	Si 'exp' es igual a cualquier 'search', retorna el valor indicado en el 'return'. Si no, retorna el valor del 'default'.
TO_CHAR(exp[,form])	No Existe	Convierte la exp que puede ser un NUMBER o de un DATE al valor de un CHAR, con el formato especificado en form.
TO_DATE(char[,form])	No Existe	Convierte la exp desde un valor CHAR a un valor DATE, con el formato especificado en form.
TO_NUMBER(char[,form] )	No Existe	Convierte a NUMBER cada carácter de char, ya que cada uno tiene asociado un número en el formato especificado por la opción form.
NVL(exp1,exp2)	ISNULL(exp1,exp2)	Reemplaza el valor especificado en 'exp2' cuando la 'exp1' es Null.

**Tabla 6 .-** Funciones.

Fuente : [Oracle -1992] [Sybase -1994]

## ❖ Tipos de Datos

Los tipos de datos definidos para ambas bases de datos se muestran en las tablas 7 y 8.

Tipo de Dato Oracle	Descripción
CHAR(tamaño)	String con caracteres de largo fijado por tamaño.
CHARACTER(tamaño)	Igual que CHAR.
DATE	Fechas con rango desde Enero 1, 4712 AC a Diciembre 31,4712 DC.
LONG	String con caracteres variable de largo variable hasta los 2 GB.
LONG RAW	Dato binario hasta los 2 GB.
MLSLABEL	Trusted ORACLE datatype
NUMBER(p,s)	Número con precisión p y escala s. p tiene un rango que va desde 1 a 38. Por omisión 38. S tiene un rango desde los – 84 a 127.
RAW(tamaño)	Dato binario hasta un tamaño de bytes. El máximo es de 20000 bytes. El tamaño debe ser especificado.
RAW MLSLABEL	Usado por Trusted Oracle.
ROWID	Valores de la pseudo columna ROWID.
VARCHAR2(tamaño)	String con caracteres de largo variable, con un largo máximo de tamaño. Máximo 2,000. Por omisión 1.
VARCHAR(tamaño)	Igual que VARCHAR2

**Tabla 7 .-** Tipos de datos definidos para la base de datos Oracle. Fuente : [Oracle -1992]

<b>Tipo de Dato Sybase</b>	<b>Descripción</b>
<b>Numérico exacto :</b>	
<b>Enteros</b>	
Tinyint	Entero de 0 a 255
Smallint	Entero de 32,767 a -32,768
Int	Entero de 2,147,483,647 a -2,147,483,648
<b>Numérico exacto :</b>	
<b>decimales</b>	
Numeric(p,s)	Decimal desde los $10^{38}-1$ a los $-10^{38}-1$
Decimal(p,s)	Decimal desde los $10^{38}-1$ a los $-10^{38}-1$
<b>Numéricos aproximados :</b>	
Float(presición)	Depende de la máquina
Double precisión	Depende de la máquina
Real	Depende de la máquina
<b>Moneda :</b>	
Smallmoney	214,748.3647 a -214,748.3648
Money	922,337,203,685,477.5807 a - 922,337,203,685,477.5808
<b>Date/time :</b>	
Smalldatetime	Enero 1, 1900 a Junio 6, 2079
Datetime	Enero 1, 1753 a Diciembre 31, 9999
<b>Caracteres :</b>	
Char(n)	255 caracteres o menos.
Varchar(n)	255 caracteres o menos.
Nchar(n)	255 caracteres o menos.
Nvarchar(n)	255 caracteres o menos.
Text	$2^{31}-1$ o menos.
<b>Binarios :</b>	
Binary(n)	255 bytes o menos.
Varbinary(n)	255 bytes o menos.
Image	$2^{31}-1$ bytes o menos.
<b>Bit :</b>	
Bit	0 o 1

**Tabla 8 .-** Tipos de datos definidos para la base de datos Sybase. Fuente : [Sybase -1994]

## 4.5. Modificación de cada Query

La siguiente consulta de SQL está extraída del software original que fue escrito para una base de datos Oracle.

Esta consulta debe entrega como resultado el número de la orden de trabajo consultada al igual que el número de versión, el estado en que se encuentra la O.T., la fecha en que fue modificada o en que se efectuó la última transacción, el nombre y apellido de la persona que realizó esta transacción y la observación de la misma transacción.

### - Query para Oracle :

```
SELECT
    T.ORTR_COD          ORDEN,
    T.ORTR_VERSION     VERSION,
    E.ESOT_DES         ESTADO,
    TO_CHAR( T.TRAN_FECHA_MODI, 'DD-MM-YYYY HH24:MM' ) FECHA,
    NVL(P.PERS_NOM, '' ) || ' ' || NVL( P.PERS_APE_PAT, '' ) NOMBRE,
    SUBSTR(NVL(TRAN_OBSERVACIONES,''), 1, 250) DESC
FROM
    TRAN T,
    FUPE F,
    PERS P,
    ESOT E
WHERE
    T.ORTR_COD = ' ' + Trim(Ortr_Cod) + ' ' AND
    T.TRAN_USUA_MODIFICACION = F.FUPE_COD AND
    T.TRAN_ESTADO = E.ESOT_COD AND
    F.PERS_COD = P.PERS_COD
ORDER BY ORDEN, VERSION, FECHA
```

**- Query Modificada para Sybase :**

```
SELECT
    T.ORTR_COD      ORDEN,
    T.ORTR_VERSION  VERSION,
    E.ESOT_DES      ESTADO,
    T.TRAN_FECHA_MODI  FECHA,
    ISNULL(P.PERS_NOM, '')  NOM,
    ISNULL( P.PERS_APE_PAT, '' )    APE_PAT,
    SUBSTRING(ISNULL(TRAN_OBSERVACIONES,''),1, 250)  DESC
FROM
    TRAN T,
    FUPE F,
    PERS P,
    ESOT E
WHERE
    T.ORTR_COD = '' + Trim(Ortr_Cod) + '' AND
    T.TRAN_USUA_MODIFICACION = F.FUPE_COD AND
    T.TRAN_ESTADO = E.ESOT_COD AND
    F.PERS_COD = P.PERS_COD
ORDER BY ORDEN, VERSION, FECHA
```

## **4.6. Ejecución de cada Query**

La instrucción modificada se ejecuta sobre la base de datos Sybase, para verificar si está correcta o presenta errores de compilación.

En este caso la consulta no contiene errores. Pero ahora se debe verificar que los resultados obtenidos son los correctos.

## **4.7. Comparación de los Resultados con los Requerimientos**

### **Solicitados**

La siguiente consulta debe entregar como resultado el número de la orden de trabajo consultada al igual que el número de versión, el estado en que se encuentra la O.T., la fecha en que fue modificada o en que se efectuó la transacción, el nombre y apellido de la persona que realizó esta transacción y una observación de ella. Esta información es necesaria para llenar el historial de la orden de trabajo, este historial indica todos los pasos que ha realizado la orden desde el momento de ser enviada a la fecha.

El resultado obtenido al ejecutar la instrucción antes mencionada se muestra en la tabla 9.

Donde la variable Ortr\_Cod será el número de la orden de trabajo seleccionada para el ejemplo. Ortr\_Cod = 115

<b>ORDEN</b>	<b>VERSION</b>	<b>ESTADO</b>	<b>FECHA</b>	<b>NOM</b>	<b>APE_PAT</b>	<b>DESC</b>
115	1	Enviada	17/12/2001	Barbara	Rios	Urgente!!
115	1	Recibida	17/12/2001	Manuel	Cajas	
115	1	Rechazada	17/12/2001	Manuel	Cajas	Por falta de información
115	1	Recibida	17/12/2001	Barbara	Rios	
115	2	Enviada	18/12/2001	Barbara	Rios	Urgente !!
115	2	Recibida	18/12/2001	Manuel	Cajas	

115	2	Aprobada	18/12/2001	Manuel	Cajas	OK
115	2	Presupuestada	21/12/2001	Manuel	Cajas	

**Tabla 9.-** Resultados obtenidos al ejecutar instrucción para Ortr\_Cod = 115.

Estos resultados son los esperados ya que se necesita establecer el historial de la orden de trabajo N° 115 y esta query entrega en forma rápida una completa visión de todos los pasos que esta orden ha seguido, quién la ha recibido en que fechas, porque fue rechazada y cuál es su estado actual.

#### **4.8. Aplicación de la Migración.**

Este proceso se repite hasta que la totalidad del código es revisado, modificado , ejecutado y comparado. Asegurando además que no presente errores al ser ejecutado en forma integrada.

## **CAPÍTULO V :**

### **5. Verificación y Medición de Resultados.**

El tiempo de entrega del proyecto calculado a partir de la ecuación (5) coincidió con el tiempo real tomado para completar el proceso de cambio. Es decir que se cumplió con el plan de trabajo calculado a partir de el tiempo de entrega.

Lo cuál no hubiese sido posible de lograr si no se hubiese seguido el plan de migración propuesto. Esto ya que los errores encontrados en cada instrucción eran inmediatamente corregidos, por lo que se evitaron errores posteriores al integrar el sistema y ejecutar en conjunto las instrucciones migradas.

El proceso migratorio propuesto está basado en el nivel 2 del Modelo CMM. Este modelo engloba todos los procesos existentes sobre mejora y evaluación de procesos de desarrollo de software. Algunas de las características mas importantes del nivel 2 son que se crean y documentan los proyectos, se realiza una administración y un control sobre los proyectos, se recolectan métricas y se realiza un seguimiento del progreso y se mejora la estimación.

Para acreditar el nivel 2 de calidad es necesario implementar las prácticas claves y cumplir con los objetivos de este nivel los que serán verificados a continuación :

- ❖ **Administración de Requerimientos** : los requerimientos del sistema fueron controlados a partir de la misma toma de requerimientos siguiendo las cuatro etapas definidas para un proceso de análisis y especificación de requerimientos. Es así como se identificó el problema a solucionar, se tomaron los requisitos del sistema, se realizó la especificación del software y finalmente se convalidaron las especificaciones verificando la consistencia de los requerimientos. Todo esto permite un importante ahorro de tiempo y esfuerzo para evitar posteriores errores.
  
- ❖ **Planificación del Proyecto** : se planifico y se realizó un seguimiento del proyecto en las etapas de desarrollo. Las actividades y compromisos del proyecto de desarrollo de software también fueron planificados y documentados. Mediante esto se puede establecer el cumplimiento de fechas de entrega y así finalmente cumplir con un importante factor de la calidad que es el cumplimiento de plazos.
  
- ❖ **Seguimiento y Supervisión del Proyecto** : se sigue y supervisa el proyecto con el fin de controlar el cumplimiento de los tiempos de entrega. Si el proyecto sufriera desviaciones significativas se debería tomar acciones correctivas, como incorporar personal en alguna tarea planificada, por ejemplo en la elaboración de manuales. Cosa que no fue necesaria en este caso, ya que los tiempos fueron cumplidos sin problemas.
  
- ❖ **Aseguramiento de la Calidad del Software** : a partir del modelo CMM se puede asegurar un nivel de calidad repetible y planificado que se obtiene al lograr el nivel 2. Este nivel hace posible la implementación de prácticas mínimas en la

administración de proyectos, asegurando mejorar en el proceso si son logradas las metas y áreas claves de un proyecto definidas para este nivel.

- ❖ **Administración de la Configuración del Software** : los productos del trabajo desarrollado son controlados y se mantienen disponibles para futuras modificaciones. Completamente documentadas y con un código altamente reutilizable.

## **Conclusión**

El modelo propuesto presenta un marco conceptual para mejorar la calidad de las actividades presentes en un proceso de migración de un software. El modelo es una guía para mejorar la calidad en un proceso de migración. Sin embargo, el modelo no garantiza el éxito de lograr procesos de gran calidad, ya que esto radica en cómo la empresa lo aplique y cómo se cumpla con los objetivos propuestos. Esto coincide con la utilización de CMM, la cuál permite y facilita un buen trabajo, pero no lo garantiza. La facilidad con que se adopten las prácticas propuestas en la norma depende fundamentalmente de la empresa.

La norma propuesta en esta tesis, permitió llevar a cabo una solicitud de un cliente en forma exitosa, cumpliendo con los tiempos y con lo esperado en cuanto a la funcionalidad del sistema. Esta norma permitió así enfrentar un problema nuevo, que era el de migrar un software y hacerlo de manera programada, de modo de poder obtener calidad en todo sentido : en la satisfacción del usuario y en el cumplimiento de los plazos.

El cumplimiento de los plazos se logró mediante la utilización de las normas propuestas, siendo posible establecer un plan de trabajo con tiempos reales, como se muestra en la tabla 5. Lo que permitió entregar el proyecto dentro de los plazos fijados. Esto no se hubiese conseguido sin seguir las normas, ya que no se contaba con experiencias anteriores en procesos migratorios.

Por tal motivo, el objetivo de esta tesis que propone desarrollar y establecer normas que permitan asegurar la calidad en un proceso migratorio de software, se ve reafirmada con el hecho de que, sin haber aplicado estas normas de seguro el proceso hubiese sido caótico.

### **Aportes de esta tesis :**

A continuación se resumen los principales aportes de esta tesis :

- Se ha propuesto un modelo para asegurar la calidad en un proceso migratorio de un software. Esta propuesta está basada en el modelo CMM, lo cual ha sido muy favorable para establecer un mejoramiento continuo del proceso migratorio, ya que permite definir un plan de trabajo enfocado hacia las áreas claves del proceso.
- Para mejorar la productividad en un proceso migratorio, se propone una mejor administración de los requerimientos como primer paso del modelo, lo que permite reducir el re-trabajo, detectando los errores en las etapas iniciales, mejorando así la calidad y la satisfacción del cliente.
- No se tiene conocimiento de ningún método desarrollado para controlar la calidad en un proceso migratorio. Más bien, todos los modelos están enfocados al desarrollo de software, que si bien es lo más utilizado, no es la única posibilidad, sobre todo hoy en día donde se requiere software en forma rápida. Esta norma propone 8 pasos a seguir en un proceso migratorio, pasos que no son del todo desconocidos ya que siguen la línea de las etapas de la ingeniería de software y a su vez las áreas claves del nivel 2 del modelo CMM. Por lo tanto esta propuesta de tesis aporta con un modelo orientado específicamente al proceso de migración de un software.

## Referencias

- **[Boll-1991]** “A Critical Look at software Capability Evaluations”. Bollinger, Terry B.. IEEE Software. 1991.
- **[Humphrey-1989]** “Managing the Software Process”. Humphrey, Watts. Addison-Wesley, 1989.
- **[IEEE-1990]** IEEE Std. 610 – 1990. Electrical and Electronics Engineers Institute.
- **[ISO-1986]** International Standard Quality ISO 8402. 1986.
- **[Matar-1995]** “Designing for Quality”. Joseph E. Matar, Robert H. Lochner.
- **[Paulk-1993a]** “Capability Maturity Model for Software version 1.1”. Paulk, Mark B., Curtis, B., Chrissis. SEI. 1993.
- **[RAE-1984]** “Diccionario de la Lengua Española”. Real Academia Española, Vigésima Edición, Madrid. 1984.
- **[Pressman-1986]** “Ingeniería del Software. Un Enfoque Práctico”. Pressman, Roger S. 3ª Edición. MacGraw-Hill, 1993.
- **[Oracle-1992]** “Oracle7 Server SQL Language Quick Reference”, Oracle Corporation, Part N° 5421-70-1292, U.S.A. Diciembre 1992.
- **[Sybase-1994]** “Sybase SQL Server Quick Reference Guide”, Sybase Inc., U.S.A. Junio 1994.
- **[Visconti-1998]** “Curso de Ingeniería de Software Avanzada”. Visconti, Marcello. Noviembre 1998.

## Internet

- [ USM - 199] “ CMM (Capability Maturity Models) Niveles de Madurez”  
[http://www.usm.edu.ec/informatica/sw\\_eng/calsoft/cmm/cmm2.htm](http://www.usm.edu.ec/informatica/sw_eng/calsoft/cmm/cmm2.htm)
- [CEGESTI –2001] “Servicios en "Capability Maturity Model" (CMM) “ Forbes, Roger . CEGESTI 2001.  
[http://www.cegesti.org/Espanol/servicios/calidad/CMM/ceg\\_serv\\_cmm.htm](http://www.cegesti.org/Espanol/servicios/calidad/CMM/ceg_serv_cmm.htm)
- [Dolado –2001] “Método de Análisis Puntos de Función MkII”. Dolado, José Javier. Universidad del País Vasco. España .Marzo del 2001.  
<http://www.sc.ehu.es/jiwdocoj/mmis/fpmkii.htm>
- [Fernández - 1999] “Necesidades de medición en la gestión y el aseguramiento de calidad del software”. Fernández, Luis. Universidad Europea de Madrid. Enero de 1999.
- [SET-2001] “Introduction to Capability Maturity Model”. Marcos Guerrero. 2001.  
<http://www.sets.cl/>
- [Redi-2000] “Resumen : Proyecto Interpretación CMM”.  
<http://redii.imas.unam.mx/redii/Ciclo2000/REDII2000/OktabaHana/resumen.htm>
- “Una comparación entre ISO 9001 y el Modelo de Capacidad y Madurez para Software”.  
[http://www.geocities.com/seminario\\_sqa/cmm\\_iso.html](http://www.geocities.com/seminario_sqa/cmm_iso.html)
- “Modelos y Métodos Para la Evaluación y Mejora del Proceso Software”.  
<http://lml.ls.fi.upm.es/doctorado/Asignaturas/MejProc.html>
- “Referenciales para Calidad en el Software”  
<http://www.orfo2000.es/sw1.htm>
- Software Engineering Institute (Carnegie-Mellon University)  
<http://www.sei.cmu.edu/>

## ANEXOS

Los siguientes anexos muestran las pantallas y el funcionamiento del sistema sobre el cuál se realizó el proceso migratorio desde una base de datos Oracle a una Sybase. Este sistema consta de dos partes participantes en el proceso de elaboración y producción de una orden de trabajo. Estas partes son el módulo Emisor o de Cuentas, que es el que elabora la O.T., es decir es quién solicita el trabajo a un área productiva.

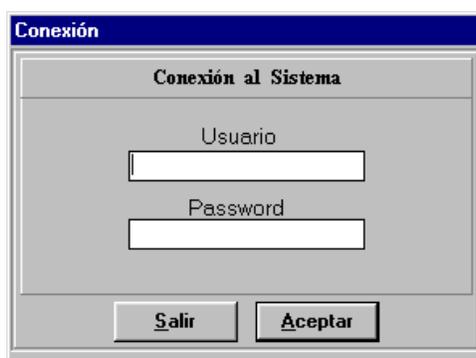
El otro módulo es el Receptor o productor del trabajo solicitado en la orden, es el responsable de aceptar o rechazar el trabajo.

El **Anexo 1** muestra el módulo Emisor, mientras que el **Anexo 2** detalla el funcionamiento y principales pantallas del módulo Receptor.

### **ANEXO 1 :**

#### **“ EMISOR DE ORDENES DE TRABAJO ”**

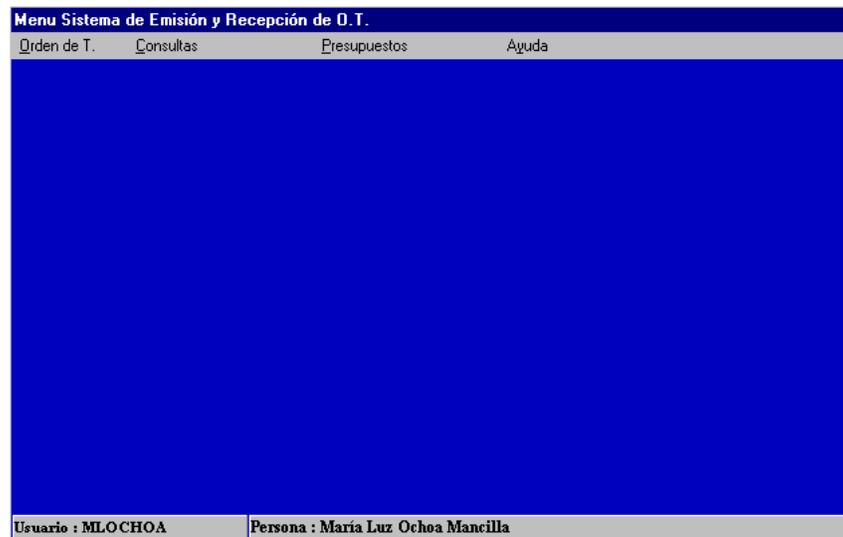
##### **1. Conexión al Sistema**



Mediante el ingreso de su *nombre de usuario* y su *password*, el sistema le permitirá la entrada, reconociendo los clientes y las actividades que le son permitidas efectuar.

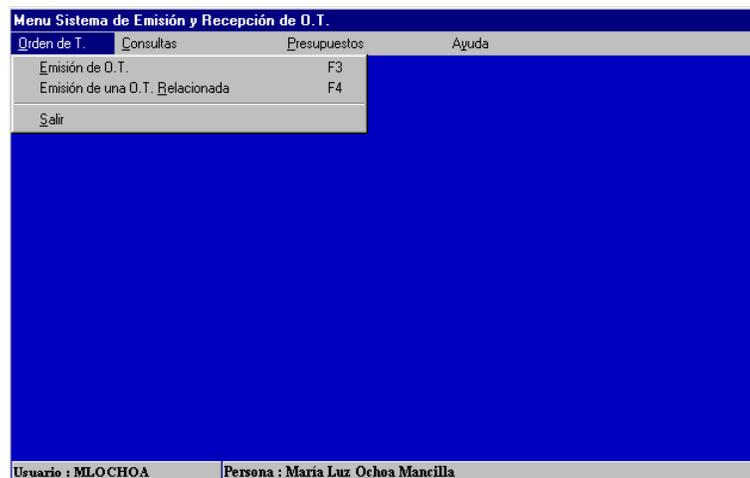
## 2. Menú Principal

Una vez dentro del sistema se muestra el *Menú principal* o *Menú Sistema de Emisión*.



## 3. Opción : Orden de Trabajo

Es la primera opción de menú principal.



Permite : - Emitir una Orden de Trabajo  
- Emitir una Orden de Trabajo Relacionada

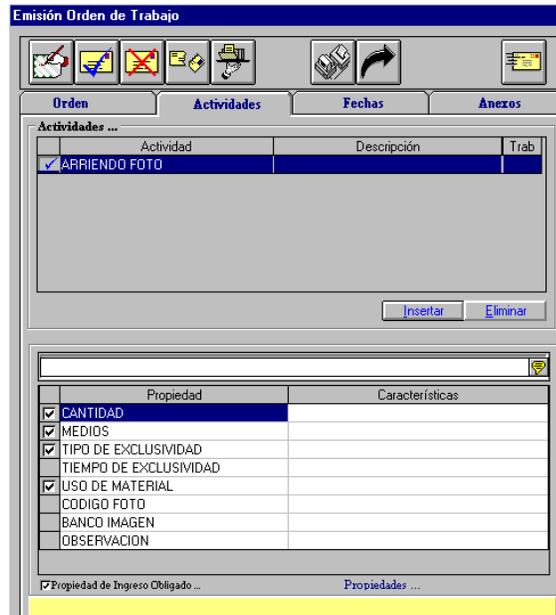
#### 4.1.- Emisión de una Orden de Trabajo

The screenshot shows the 'Emisión Orden de Trabajo' window with the 'Orden' tab selected. The window title is 'Emisión Orden de Trabajo'. The top toolbar contains icons for back, forward, cancel, and other actions. Below the tabs, the 'Numero' field is highlighted in yellow and contains '- 1'. The 'Cliente' dropdown is set to 'LEO BURNETT'. The 'Marca' dropdown is 'Institucional', 'Producto' is 'Institucional', 'Campaña' is 'Papelería', and 'Tipo O.T.' is 'Taller y Arte'. The 'Facturar A' section has radio buttons for 'Sin Costo' (selected), 'Cliente', and 'Holding'. The 'Destinatario' and 'C.C.' fields are empty. The 'BRIEF' section is empty. The 'Aplicación' section has checkboxes for 'Via Pública', 'Prensa', 'Revista', and 'P.O.P.', all of which are unchecked. A 'Crear Campaña' button is visible at the bottom left.

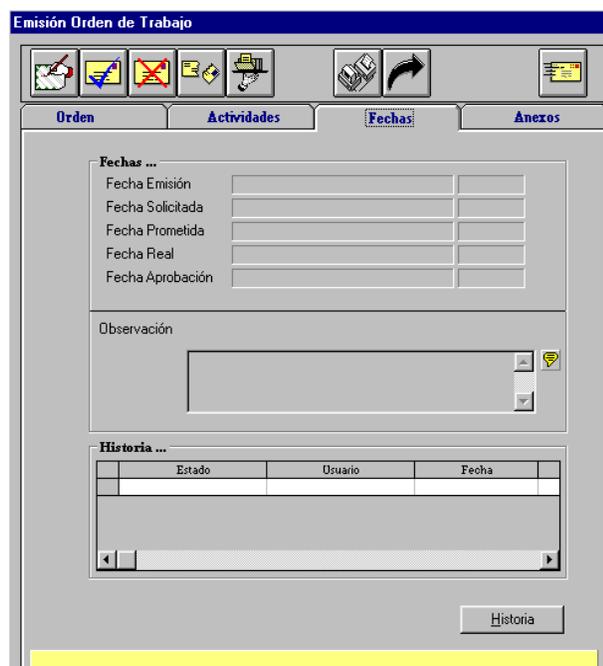
En la segunda lengüeta “Actividades”, se ingresan las actividades o trabajos que se desean realizar y sus propiedades asociadas.

The screenshot shows the 'Emisión Orden de Trabajo' window with the 'Actividades' tab selected. The window title is 'Emisión Orden de Trabajo'. The top toolbar contains icons for back, forward, cancel, and other actions. Below the tabs, the 'Actividades ...' section contains a table with columns 'Actividad', 'Descripción', and 'Trab'. The table is currently empty. Below the table are 'Insertar' and 'Eliminar' buttons. The 'Propiedades ...' section contains a table with columns 'Propiedad' and 'Características'. This table is also empty. At the bottom left, there is a checkbox for 'Propiedad de Ingreso Obligado ...' and a 'Propiedades ...' button.

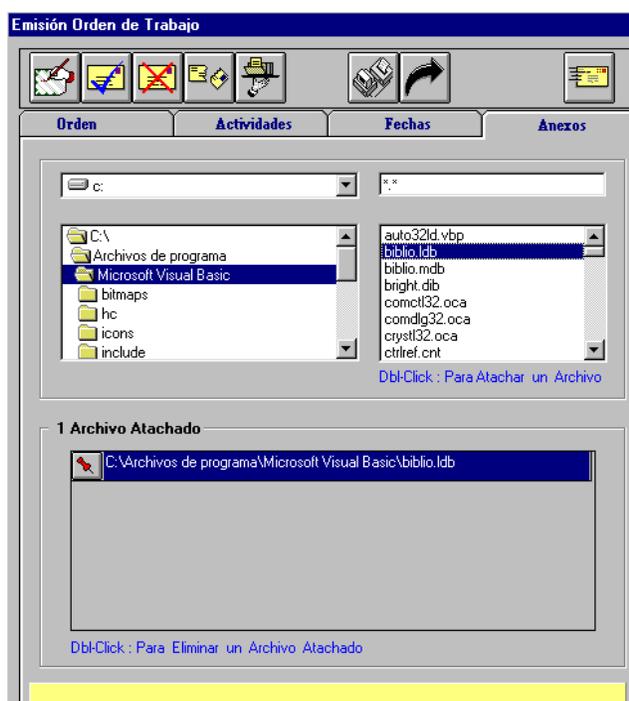
Para ingresar una actividad se debe pinchar el botón **Insertar** (“Insertar”), y seleccionar una de las actividades de la lista.



La tercera lengüeta “Fechas”, muestra las fechas importantes de una O.T., como por ejemplo, la fecha en que fue emitida, solicitada y prometida.



La lengüeta “Anexos” permite ubicar un archivo de tipo: Word, Excel, Power Point, .BMP, .JPG, .HTML, para poder ser enviado junto con la O.T.



Una vez que ha ingresados todos los datos obligatorios a la orden, esta puede ser **ENVIADA**, presionando el botón :  que se encuentra en la parte superior derecha de la O.T.

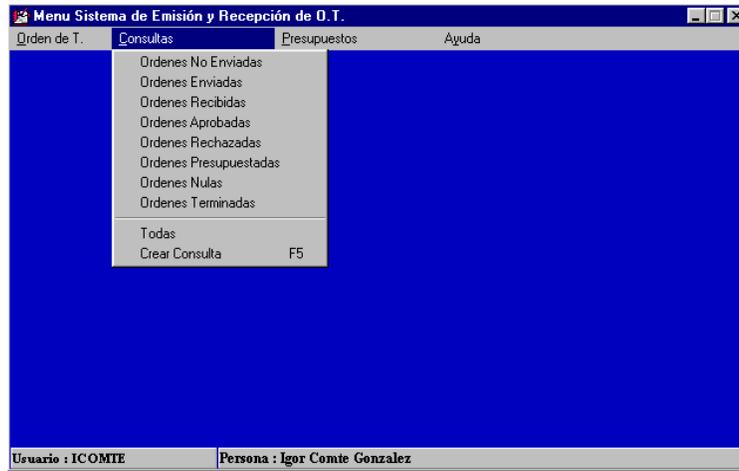
Hecho esto se solicitará ingresar la fecha para la cuál se necesita el trabajo, es decir la “Fecha Solicitada”.



Se genera un número único de la orden de trabajo.

#### 4. Opción : Consultas de O.T.

Es la segunda opción de menú principal.



Permite realizar consultas de las O.T. por :

- Estado :
  - Ordenes No Enviadas
  - Ordenes Enviadas
  - Ordenes Recibidas
  - Ordenes Aprobadas
  - Ordenes Rechazadas
  - Ordenes Presupuestadas
  - Ordenes Nulas
  - Ordenes Terminadas
- Todas :

Muestra todas las O.T. de todos los estados, tipos con y sin solicitud de cambio de comisión.
- Crear Consulta :

Esta opción permite crear su propia consulta de la ordenes.

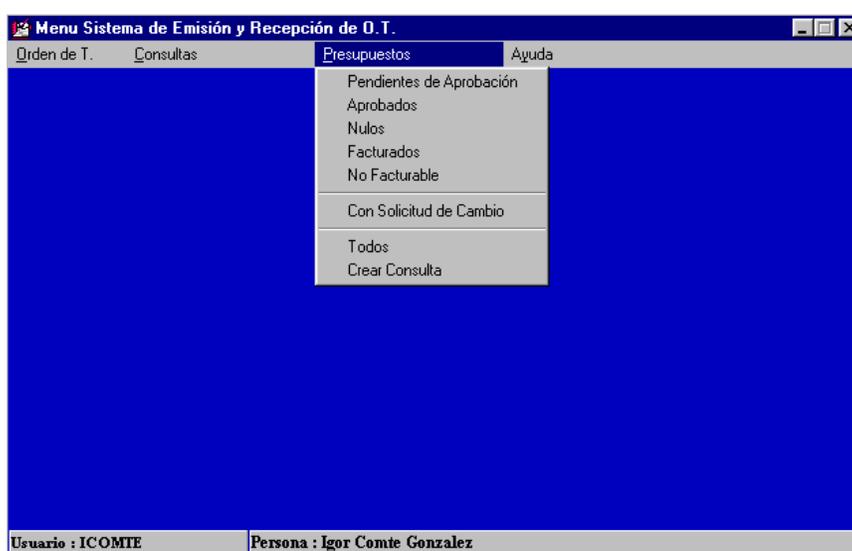
A screenshot of a dialog box titled "Defina su Consulta". It contains several sections for defining search parameters. The "Estado ..." section has checkboxes for "No Enviada", "Enviada", "Recibida", "Aprobada", "Rechazada", "Presupuestada", "Nula", and "Terminada". The "Departamento ..." section has checkboxes for "Creación", "Diseño", "Taller y Arte", "Audiovisual", "Fotografía", "Impresos", "Internet", and "Investigación". The "Fechas ..." section has "Desde :" and "Hasta :" labels followed by date input fields with the format "DD/MM/YYYY". At the bottom, there is a text box with the instruction "Seleccione los parámetros para realizar su consulta y luego presione el botón Aceptar" and two buttons: "Cancelar" and "Aceptar".

Una vez seleccionada la consulta a realizar, se mostrarán los resultados en la siguiente pantalla, desde donde se puede seleccionar una O.T. y abrirla.

N° O.T.	Cliente	Marca	Producto	Campaña
100 - 1	LEO BURNETT	Institucional	Institucional	Cambio
99 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
98 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
97 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
96 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
95 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
94 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
93 - 1	ENTEL	MULTICARRIER	CODIGO	MULTI CARRIER
92 - 1	LEO BURNETT	Institucional	Institucional	Cambio
91 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER
90 - 1	ENTEL	MULTICARRIER	CODIGO	CODIGO MULTICARRIER

## 5. Opción : Presupuestos

La tercera opción del menú principal es la de *Consulta de Presupuestos*. En ella se pueden consultar por el estado de los presupuestos.



Permite realizar consultas de los Presupuestos por :

- Estado :
  - Presupuestos Pendientes de Aprobación
  - Presupuestos Aprobados
  - Presupuestos Nulos
  - Presupuestos Facturados
  - Presupuestos No Facturables
  - Presupuestos con Solicitud de Cambio
- Todas :
 

Muestra todos los presupuestos en todos los estados, con y sin solicitud de cambio.

- Crear Consulta :

Esta opción permite crear una consulta particular de Presupuestos.

N° Presu	Fecha	N° O.T.	Cliente	Marca	Producto
907	22/03/02	1944 - 1	Falabella	Oferta Ahorro	Oferta Ahorro
905	22/03/02	1946 - 1	Falabella	Oferta Ahorro	Oferta Ahorro
844	21/03/02	1940 - 1	CMR	Institucional	Institucional
692	20/03/02	1876 - 1	Pillsbury	Häagen Dazs	Häagen Dazs
686	19/03/02	1909 - 1	Falabella	Belleza	Belleza
685	19/03/02	1900 - 1	LEO BURNETT	Institucional	Institucional
886	22/03/02	1942 - 1	CMR	Institucional	Institucional
789	18/03/02	1845 - 1	Falabella	Kermesse	Institucional
703	13/03/02	1708 - 1	Falabella	Basement	Basement
693	22/03/02	1929 - 1	Falabella	Belleza	Belleza
545	11/02/02	1073 - 1	Falabella	Argentina	Argentina
542	11/02/02	1069 - 1	Falabella	Argentina	Argentina

Al abrir un presupuesto, presionando el botón “Abrir” o haciendo Doble – Click sobre el presupuesto seleccionado, dependiendo de si el presupuesto es interno o externo se verá la siguientes pantallas.

**PRESUPUESTO DE TALLER Y ARTE**

O.T. N° : 793-1      Presupuesto N° : 380

Cliente : Administradora CMR Falabella Ltda.      Factura N° :

Marca : Tarieta      Fecha : 31 de Enero del 2002

Producto : Tarieta

Descripción : MODIFICACION DE COLOR DE FONDO AL ORIGINAL PENDON, TEMA: 10% DE DESCUENTO EN SU PRIMERA COMPRA

N° Act	Trabajo	Cantidad	Valor Unit. (\$)	Costo Neto (\$)	Total Neto (\$)	Total con IVA (\$)
1	Cambio de texto	1	11.000	11.000	11.000	12.980
	TOTAL			11.000	11.000	12.980



## 7. O.T. Rechazadas

Cuando una O.T. es rechazada por el departamento al que fue enviada, se debe abrir la orden para poder leer el motivo de rechazo.

Este se encuentra en la lengüeta “*Fechas*” de la orden.

Historia ...				
	Estado	Usuario	Fecha	
1	Enviada	María Luz Ochoa Mancilla	24/01/02 16:00:23	TrC
2	Recibida	Claudia Muñoz	24/01/02 16:01:11	Ord
3	Rechazada por Producción	Claudia Muñoz	24/01/02 16:02:11	MA
4	Recibida	María Luz Ochoa Mancilla	24/01/02 16:02:57	Ord

Esta O.T. puede ser reenviada realizando los cambios necesarios para que esta vez la orden sea aceptada por el departamento.

Una vez realizadas las modificaciones a la O.T. rechazada esta puede ser re-enviada, presionando el botón . Con esto la orden será enviada nuevamente al departamento seleccionado, manteniendo el mismo número pero variando la versión.

Numero : 54 - 3

Cliente: ENTEL  
Marca: MULTICARRIER PERSONAS  
Producto: Emisión de Ordenes de Trabajo  
Campana:  
Tipo O.T.:  
Fue Enviada la Orden con Número : < 54 - 3 >

Facturar A:  S  N

Destinatario: Claudia Muñoz  
C.C.:  
BRIEF:

Envia Orden de Trabajo a la Persona Destinatario ...

## ANEXO 2

### “RECEPTOR DE ORDENES DE TRABAJO”

#### 1. Conexión al Sistema



The image shows a dialog box titled "Conexión" with a sub-title "Conexión al Sistema". It contains two text input fields: "Usuario" and "Password". Below the fields are two buttons: "Salir" and "Aceptar".

Mediante el ingreso de su *nombre de usuario* y su *password*, el sistema le permitirá la entrada, reconociendo los clientes y las actividades que le son permitidas efectuar.

#### 2. Función Administrativa

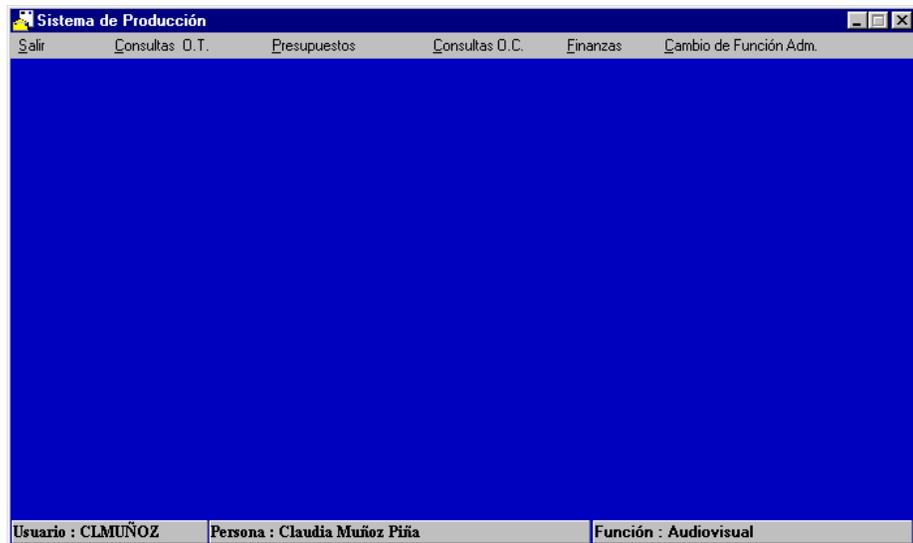
Si la persona que establece la conexión al sistema posee más de una función administrativa, el sistema solicita seleccionar con cuál de ellas efectuará la conexión al sistema.



The image shows a dialog box titled "Función Administrativa". It contains the text "Seleccione la Función Administrativa a Utilizar en esta sesión :". Below the text is a dropdown menu. At the bottom are two buttons: "Salir" and "Aceptar".

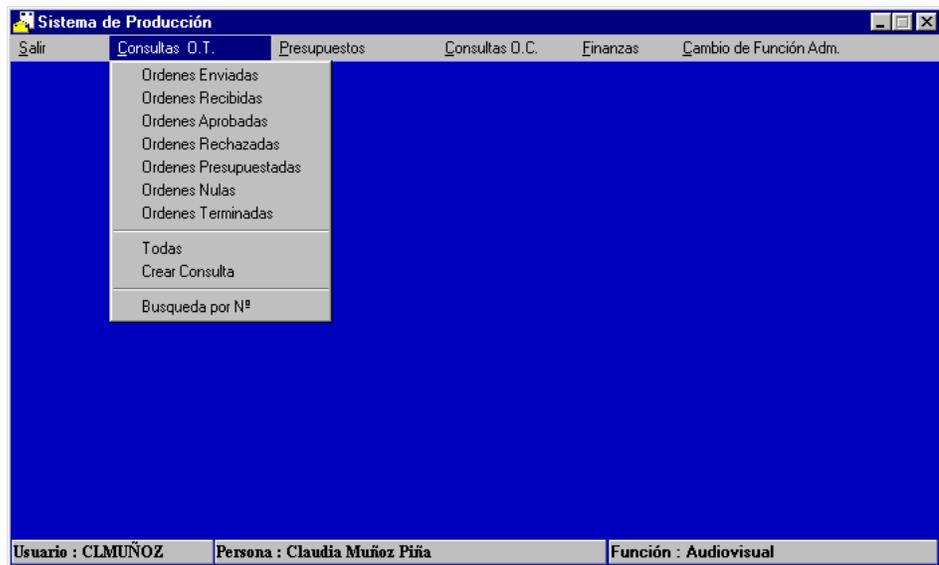
### 3. Menú Principal

Una vez dentro del sistema se muestra el *Menú principal* o *Menú Sistema de Producción*.



### 4. Opción : Consultas

Permite realizar consultas de las O.T. por :



- Estado :
  - Ordenes Enviadas
  - Ordenes Recibidas
  - Ordenes Aprobadas
  - Ordenes Rechazadas
  - Ordenes Presupuestadas
  - Ordenes Nulas
  - Ordenes Terminadas

- Todas :  
Muestra todas las O.T. de todos los estados.
- Crear Consulta :  
Esta opción permite crear su propia consulta o una consulta específica.

- Búsqueda por Número :  
Permite realizar una búsqueda de la orden de trabajo por número. Para ello se debe ingresar el número buscado en la siguiente pantalla.

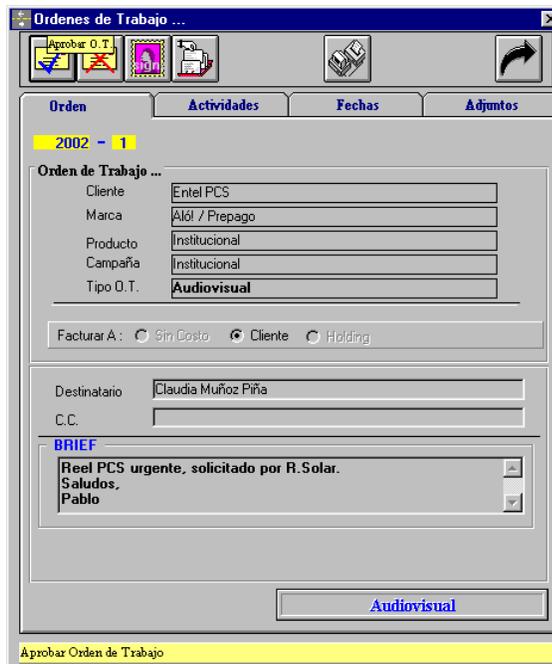
Una vez realizada la consulta se mostrarán los resultados encontrados en la siguiente pantalla.

N° O.T.	Cliente	Marca	Producto	Campaña
2002 - 1	Entel PCS	Alól / Prepago	Institucional	Institucional
2001 - 1	ENTEL	CORPORATIVO	Institucional	Rei varios
2000 - 1	Laboratorio Chile	Red Off	Red Off	Campaña Red Off verano 2002
1999 - 1	Helmanns	Mayonesa	Mayonesa	Campaña 2002
1998 - 1	Falabella	Internet	Internet	aviso Novios Revista Ofertahorro
1997 - 1	Copesa	QUE PASA	QUE PASA	Revista Catas y Cavas
1996 - 1	Copesa	QUE PASA	QUE PASA	Aviso revista especializada M2
1995 - 1	Falabella	Jeans	Jeans	moda juvenil jeans
1994 - 1	Falabella	Institucional	Institucional	Quilpué
1993 - 1	Falabella	Oferta Ahorro	Oferta Ahorro	Muebles

En esta pantalla se muestra mediante iconos el estado en que se

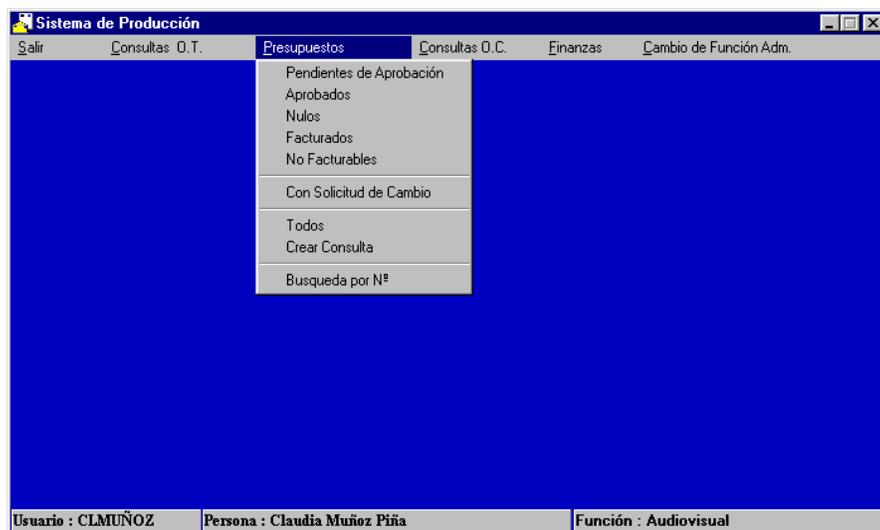
encuentra la O.T.

Desde esta, se puede seleccionar una O.T. y abrirla haciendo doble-click sobre ella, o presionando el botón  ubicado en la parte inferior derecha de la pantalla. Al hacer esto se desplegará la O.T. seleccionada.



## 5. Opción : Presupuestos

Permite realizar consultas sobre los presupuestos existentes los que pueden estar dentro de las siguientes categorías:



- Estado :
  - Presupuestos Pendientes de Aprobación
  - Presupuestos Aprobados
  - Presupuestos Nulos
  - Presupuestos Facturados
  - Presupuestos No Facturables
  - Presupuestos Con Solicitud de Cambio
- Todos :  
Muestra todos los Presupuestos en todos los estados.
- Crear Consulta :  
Esta opción permite crear su propia consulta o una consulta específica sobre los presupuestos.

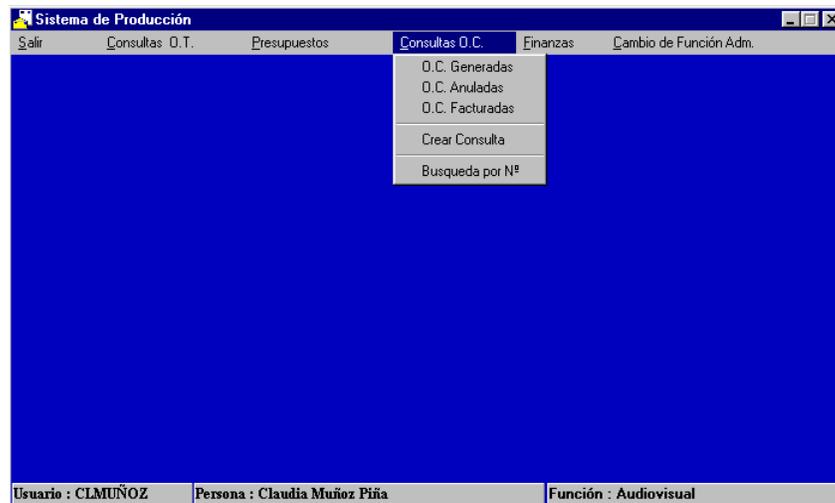
- Búsqueda por Número :  
Permite realizar una búsqueda de un número de presupuesto o un rango de presupuestos. Para ello se debe ingresar el número buscado en la siguiente pantalla.

Una vez seleccionada la consulta de presupuestos a realizar, los resultados encontrados se muestran en la siguiente pantalla :



## 6. Opción : Consulta Orden de Compra

Permite realizar consultas sobre las Ordenes de Compra existentes las que pueden estar dentro de las siguientes categorías:



- Estado :
  - Ordenes de Compra Generadas
  - Ordenes de Compra Anuladas
  - Ordenes de Compra Facturadas
- Crear Consulta :

Esta opción permite crear su propia consulta o una consulta específica sobre las ordenes de Compra.

The 'Defina su Consulta' dialog box contains several sections. The 'Estado Orden de Compra' section has checkboxes for 'Generada', 'Anulada', and 'Facturada'. The 'Cliente' and 'Proveedor' fields are dropdown menus. The 'Departamento' section has checkboxes for 'Creación', 'Fotografía', 'Diseño', 'Impresos', 'Taller y Arte', 'Internet', 'Audiovisual', and 'Investigación'. The 'Fechas' section has 'Desde' and 'Hasta' date pickers with 'DD/MM/YYYY' format. At the bottom, there is a blue instruction: 'Seleccione los parámetros para realizar su consulta y luego presione el botón Aceptar', and 'Cancelar' and 'Aceptar' buttons.

- Búsqueda por Número :

Permite realizar una búsqueda por número de orden de compra. Para ello se debe ingresar el número buscado en la siguiente pantalla.

The 'Busqueda' dialog box has a title bar 'Busqueda' and a subtitle 'Busqueda de O.C.'. It features a text input field labeled 'N° :'. Below the input field are two buttons: 'Buscar' and 'Salir'.

Una vez seleccionada la consulta de ordenes de compra a realizar, los resultados encontrados se muestran en la siguiente pantalla :

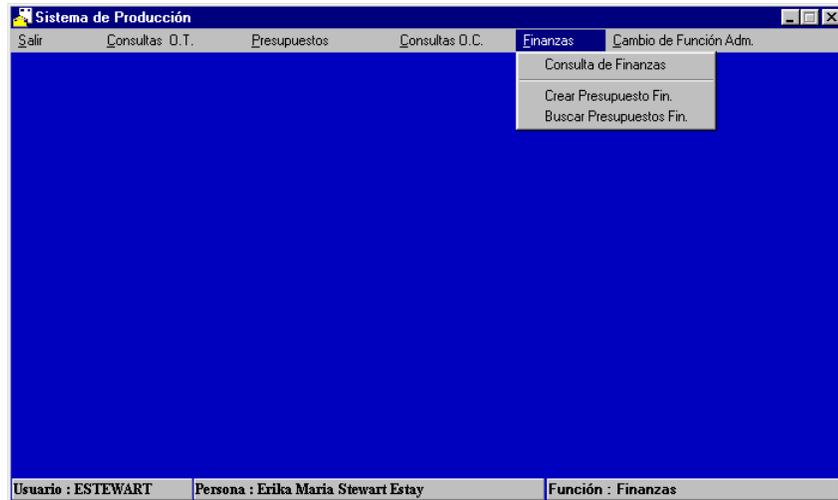
Fecha	Nº O.T.	Nº Presu.	Nº O.C.	Cliente	Monto (\$)	Proveedor
21/03/2002	1321 - 1	506	249	ING Seguros de Vida	2.430.800	Finish
21/03/2002	1621 - 1	591	247	Procter & Gamble	708.000	E.M. Audio
18/03/2002	1615 - 1	658	211	Helmanns	724.520	Finish
15/03/2002	1590 - 1	589	190	Falabella	2.218.400	Finish
14/03/2002	1673 - 1	609	188	Falabella	29.500.000	Del Carril
14/03/2002	1680 - 1	626	184	Marlboro	94.400	Guillotina
14/03/2002	1336 - 1	532	180	Copesa	1.121.000	Finish
11/03/2002	1573 - 1	588	158	LEO BURNETT	413.000	Finish
11/03/2002	1388 - 1	548	153	Asociación de AFP AG	1.652.000	W Producciones
07/03/2002	1489 - 1	562	141	Falabella	17.700	W Producciones
07/03/2002	1254 - 1	483	137	Falabella	1.014.800	M.C.M.

Desde esta pantalla se puede imprimir la lista de los resultados obtenidos y también se pueden abrir las ordenes de compra ya creadas. Esto haciendo doble-click sobre una de ellas.

Ordenes de Compra	
<b>Orden de Compra</b>	
Señores : <input type="text" value="Finish"/>	O. de C. Nº : <u>249</u>
Entregar a : <input type="text" value="Leo Burnett"/>	O.T. Nº : <u>1321-1</u>
Facturar a : <input type="text" value="ING Seguros de Vida S.A."/>	Presupuesto Nº : <u>506</u>
Dirección : <input type="text" value="Avda. Suecia 211 piso 7 Providencia"/>	Fecha : <u>21/03/2002</u>
Rut : <input type="text" value="96549050 - 7"/>	
<p>Detalle ...</p> <p>Realización de 4 Animatic Campaña Ahorro Previsional Voluntario de ING .</p> <p>Costo <input type="text" value="2.060.000"/></p> <p>Iva <input type="text" value="370.800"/></p> <p>10 % <input type="text"/></p> <p><b>Total</b> <input type="text" value="2.430.800"/></p>	
<input type="button" value="Anular"/> <input type="button" value="Guardar"/> <input type="button" value="Imprimir"/> <input type="button" value="Salir"/>	

## 7. Opción :Consulta Finanzas

Esta consulta es exclusiva para el departamento de Finanzas.



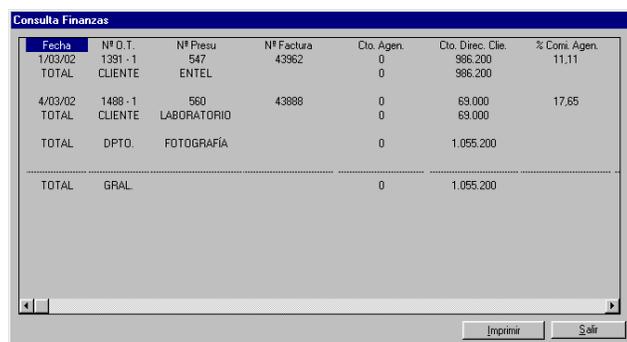
Esta opción permite :

- **Crear una Consulta de Finanzas**

Esta opción permite realizar consultas y extraer listados de las inversiones de un cliente por departamento o estado de los presupuestos.



El resultado de esta consulta se muestra en la siguiente pantalla :



Fecha	Nº O.T.	Nº Presu	Nº Factura	Cto. Agen.	Cto. Direc. Cte.	% Com. Agen.
1/03/02	1391 - 1	547	43952	0	996.200	11,11
TOTAL	CLIENTE	ENTEL		0	996.200	
4/03/02	1488 - 1	560	43988	0	69.000	17,65
TOTAL	CLIENTE	LABORATORIO		0	69.000	
TOTAL	DPTO.	FOTOGRAFÍA		0	1.055.200	
TOTAL	GRAL.			0	1.055.200	

- **Crear Presupuesto de Finanzas**

Esta opción permite crear presupuestos especiales de finanzas. Estos presupuestos no tienen O.T. que los genere.

**Borrador de Presupuesto de Finanzas**

Ciente:

Marca:

Producto:

Trabajo:

Documento de Pago:   IVA  US\$

Ingreso de Ctos.

Tema	Cantidad	Cto. Neto	Tot. Neto	Tot. c/IVA

Ingresar

Ctos. Ingresados ...

Nro.	Tema	Cantidad	Costo Neto (\$)	Total Neto (\$)	Total (\$)	Tot. US\$

Borrar

**Borrador de Presupuesto de Finanzas**

Ciente:

Marca:

Producto:

Trabajo:

Documento de Pago:   IVA  US\$

Ingreso de Ctos.

Tema	Cantidad	Cto. Neto	Tot. Neto	Tot. c/Reten.	Tot. US\$
Taxis	15	300.000	4.500.000	4.500.000	6.667

Ingresar

Ctos. Ingresados ...

Nro.	Tema	Cantidad	Costo Neto (\$)	Total Neto (\$)	Total (\$)	Tot. US\$
1	Se detalla un cobro	2	150.000	300.000	300.000	444

Borrar

**Borrador de Presupuesto de Finanzas**

Ciente:

Marca:

Producto:

Trabajo:

Documento de Pago:   IVA  US\$

Ingreso de Ctos.

Tema	Cantidad	Cto. Neto	Tot. Neto	Tot. c/Reten.	Tot. US\$
Taxis	15	300.000	4.500.000	4.500.000	6.667

Ingresar

Ctos. Ingresados ...

Nro.	Tema	Cantidad	Costo Neto (\$)	Total Neto (\$)	Total (\$)	Tot. US\$
1	Se detalla un cobro	2	150.000	300.000	300.000	444

Borrar

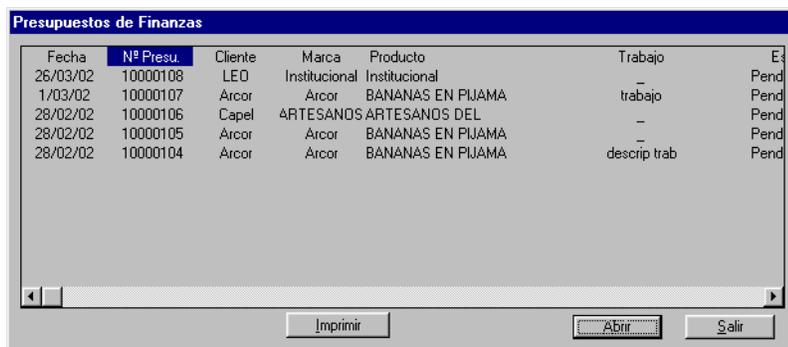
**Producción**

Presupuesto Generado.  
Nro. Presu = 10000108

Aceptar

- **Buscar Presupuestos de Finanzas**

Permite realizar una búsqueda de todos los presupuestos de finanzas y entrega el resultado de esta búsqueda.

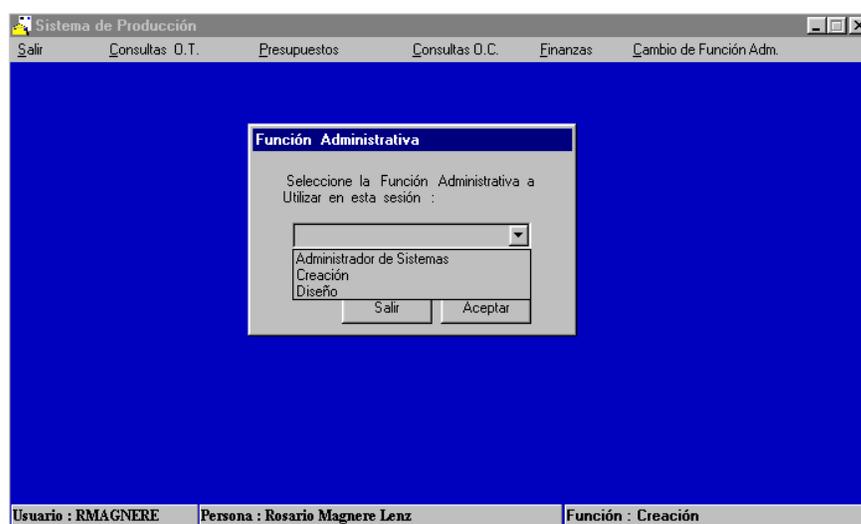


Fecha	Nº Presu.	Cliente	Marca	Producto	Trabajo	Est.
26/03/02	10000108	LED	Institucional	Institucional	-	Pend
1/03/02	10000107	Arcor	Arcor	BANANAS EN PIJAMA	trabajo	Pend
28/02/02	10000106	Capel	ARTESANOS	ARTESANOS DEL	-	Pend
28/02/02	10000105	Arcor	Arcor	BANANAS EN PIJAMA	-	Pend
28/02/02	10000104	Arcor	Arcor	BANANAS EN PIJAMA	descrip trab	Pend

## **8. Opción : Cambio de Función Administrativa**

Esta opción es válida para las personas que cumplen mas de una función administrativa en el sistema de producción.

En vez de salir del sistema solo cambia de función.



## **9. Alarma de Recepción de una Orden de Trabajo**

Esta pantalla aparece en forma de una alarma visual y sonora (beep), para avisar el envío de una O.T.

La alarma se anulará y dejará de sonar al presionar el botón “Anular Alarma” o al abrir la O.T.

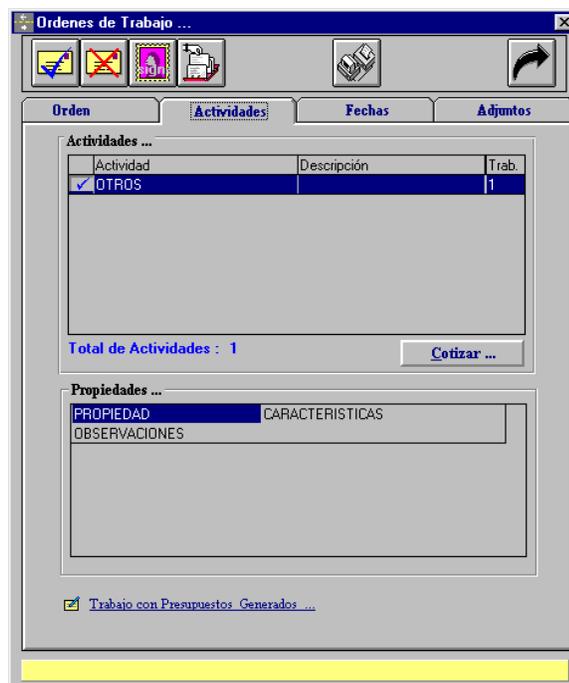


Una vez que la O.T. ha sido abierta, puede ocurrir que la O.T sea *Aceptada* o por el contrario sea *Rechazada*. Ambas situaciones se analizarán a continuación.

### Receptor

Luego de Aprobar la O.T. esta pasa a estado : “Aprobada” y recién entonces pueden realizarse las cotizaciones y presupuestos.

Para realizar las cotizaciones se debe presionar el botón :



Se puede realizar cotizaciones de dos tipos :

- Cotización Interna
- Cotización Externa

La cuál se selecciona en la pantalla “Tipo de Cotización”:



- **Cotización Interna**

Solo es permitida esta opción para los departamentos que posean “Lista de Precios Interna”.

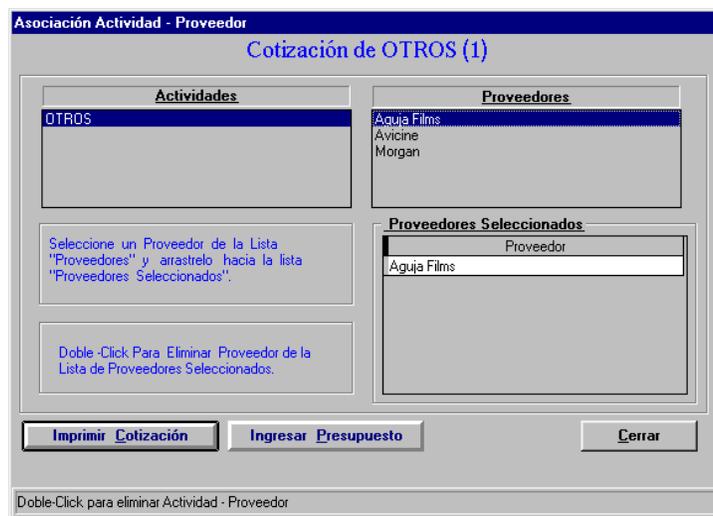
Si es este al caso se llega a la siguiente pantalla :



- **Cotización Externa**

Permite realizar una cotización externa para cada una de las actividades de una O.T.

Cada trabajo o actividad tiene asignado una lista de proveedores, se debe seleccionar al menos tres de ellos para realizar una cotización.



Cuando las cotizaciones han sido devueltas por los proveedores con el monto correspondientes, se debe pinchar el botón **Ingresar Presupuesto** con lo cuál se muestra la pantalla de Borrador de Presupuesto Externo :

**Borrador de Presupuesto Externo**

Proveedor: **Aguja Films**

Trabajo: Trabajo a realizar

Detalle: detalle del trabajo

Observación:

Recomendación Agencia:

Documento de Pago: Factura  US\$

Botones: Salir, Imprimir, Grabar

---

**Ingreso de Cotización**

	Cantidad	Cto. Neto	Comi.	Tot. Neto	Tot. c/IVA
OTROS	3	300000			

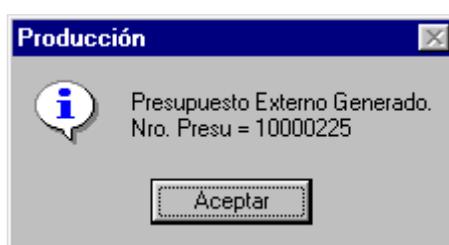
Botón: Ingresar

---

**Proveedores con Cotizaciones Ingresadas ...**

Proveedor	Nro. Coti.	Cantidad	Costo Neto (\$)	Comisión Agencia (\$)	Total Neto (\$)	Total con IVA o Retenciones (\$)
Aguja Films	1	2	250.000	27.775	277.775	327.774

Botones: Mover Coti., Borra Coti.



## 10. Generar una Orden de Compra

Para generar una Orden de Compra (O.C.) es necesario seleccionar el *Presupuesto Externo*. Y luego presionar el botón: **Orden de Compra**.

**Presupuesto Externo**

**PRESUPUESTO DE AUDIOVISUAL**

O.T. Nº : 85-1 Presupuesto Nº : 10000223  
 Cliente : ENTEL S.A. Factura Nº :  
 Marca : MULTICARRER PERSONAS  
 Producto : CODIGO Ingreso de Coti. para O.C. 19 de Marzo del 2002

Trabajo : Ingrese el Nro. de la Cotización  
 Si es más de una separamos con ',' (coma)  
 (Deben pertenecer al Mismo proveedor)

Detalle :  
 Observación :

Nro. Coti.	Producto	Cantidad	Costo Neto (\$)
1	Aviacion	2	2.000,00
2	Aviacion	3	3.000,00

Recomendación Agencia :

Botones: Orden de Compra, Imprimir, Anular, Salir

Se deben ingresar los números de las cotizaciones aprobadas por el cliente, si es más de una se deben separar por coma. Solo admite proveedores iguales para una O.C.

Ordenes de Compra

### Orden de Compra

Señores :	<input type="text" value="Ávicine"/>	O. de C. N° :	
Entregar a :	<input type="text" value="Leo Burnett"/>	O.T. N° :	<input type="text" value="85-1"/>
Facturar a :	<input type="text" value="ENTEL S.A."/>	Presupuesto N° :	<input type="text" value="10000223"/>
Dirección :	<input type="text" value="Av. Andrés Bello 2687 P#13"/>	Fecha :	<input type="text" value="03/26/2002"/>
Rut :	<input type="text" value="92580000-7"/>		

Detalle ...	
<input type="text" value="Auí se llena con una glosa o detalle del trabajo"/>	Costo <input type="text" value="5,000.000"/>
	Iva <input type="text" value="900.000"/>
	10 % <input type="text"/>
	<b>Total</b> <input type="text" value="5,900.000"/>

Una vez que está lista la O.C. y se ha ingresado un detalle o Glosa, puede ser guardada. Luego de lo cuál entregará el número de la O.C.