



UNIVERSIDAD LASALLISTA BENAVENTE

ESCUELA DE INGENIERÍA EN COMPUTACIÓN
Con estudios incorporados a la
Universidad Nacional Autónoma de México
CLAVE: 8793-16



“SISTEMA DE INFORMACIÓN DE PLANTAS PROCESADORAS DE AVES MÓDULO (POLLO VIVO)”

TESIS

QUE PARA OBTENER EL TÍTULO DE:
INGENIERO EN COMPUTACIÓN

PRESENTA:
JORGE JAVIER PATIÑO CASTAÑÓN

ASESOR: ING. ALEJANDRO GUZMÁN ZAZUETA

Celaya, Gto.

Enero de 2007.



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Agradecimientos Personales

A DIOS por iluminarme y guiarme en cada momento de mi vida y mi trabajo

A mi esposa, por el gran amor que me has dado y ser un apoyo cada día de mi vida.

A mis hijos, por ser la causa de superación día por día y el motor que impulsa mi vida.

A mis padres, por enseñarme lo que se de la vida y seguir enseñándome, apoyándome en todo momento.

A mi hermano, por ser el mejor compañero y amigo que se pueda tener.

A mis maestros, por compartirme sus conocimientos de los que ahora gozo.

A mi asesor, por el apoyo y confianza con la que siempre me ha tratado.

Índice

Introducción

Capítulo I La empresa Bachoco S.A de C.V.	1
1.1. Antecedentes de la empresa Bachoco.....	2
1.2. Distribución geográfica.....	3
1.3. Negocio y participación del mercado	4
1.4. Ciclos vitales productivos	5
1.5. Estructura organización de la Dirección General	7
1.6. Infraestructura de Plantas Procesadoras de Aves	8
1.7. Objetivo del Proyecto	11
1.8. Situación actual.....	11
1.9. Decisiones.....	16
1.10. Beneficios.....	17
Capítulo II Fundamentación	20
2.1. Aplicaciones Windows.....	21
2.2. Power Builder Versión 4.0.....	22
2.3. Orientado a objetos, dirigido a eventos	24
2.4. Una interfaz gráfica	25
2.5. El modelo de desarrollo.....	27
2.6. Programación orientada a objetos	28
2.7. Modelo Cliente – Servidor	34
Capítulo III Metodología para la <i>Implementación de un SI</i>	38
3.1. <i>Metodología usada</i>	39
3.2. <i>Herramientas de análisis y diseño de sistemas</i>	41
3.2.1. Planteamiento de objetivos	41
3.2.2. Entrevista con los usuarios	41
3.2.3. Preguntas en la entrevista	44
3.2.4. Rentabilidad de los objetivos	47
3.2.5. Elaboración de diagramas de flujo.....	48
3.2.5.1. Diagramas de flujo de datos	49
3.2.6. Determinación de requerimientos	51
3.2.7. Diccionario de datos	54
3.2.8. Modelo entidad-relación.....	59
3.2.8.1. Componentes modelo entidad-relación.....	59

3.2.8.2.	Modelación gráfica de datos	60
3.2.8.3.	Comentarios del modelo entidad - relación....	66
Capítulo IV	Implementación Módulo de Pollo Vivo	67
4.1.	Implementación.....	68
4.2.	Orden de embarque	69
4.3.	Término de recepción.....	71
4.4.	Pollo vivo a venta.	73
4.5.	Prenómina.....	75
4.5.1.	Captura otras comisiones	75
4.5.2.	Cálculo de Prenómina.....	76
4.6.	Diesel.....	77
4.6.1.	Captura de diesel.....	77
4.6.2.	Reporte de rendimiento de unidades.....	78
4.7.	Reportes.....	79
4.7.1.	Pollo de granjas	79
4.7.2.	Pollo vivo a venta.....	81
4.7.3.	Producto a proceso.....	83
4.7.4.	Pollo en espera	85
4.7.5.	Registro diario de viajes.....	87
4.7.6.	Resultados por supervisor	89
4.7.7.	Conciliación pollo vivo por embarque	91
4.7.8.	Conciliación pollo vivo por granja	93
4.7.9.	Trasposos en línea	95
4.7.10.	Pollo decomisado	97
4.7.11.	Pollo muerto por causa	99
4.7.12.	Viajes realizados por tipo de transporte y zona	101
4.7.13.	Flujo de transporte	102
4.7.14.	Km. Por dolly	103
4.7.15.	Km. Por plataforma.....	105
4.7.16.	Aves por supervisor	107

Conclusiones

Bibliografía

Introducción

En la actualidad, el mundo experimenta cambios fundamentales, los continuos avances en la tecnología computacional así como en las comunicaciones tienen un efecto profundo sobre la forma en que las personas trabajan e inclusive se divierten, por lo que estos factores influyen en la forma en que se diseñan los sistemas.

El análisis y el diseño de sistemas, dentro de las empresas y organizaciones, se refiere al proceso de examinar la situación de una empresa con la intención de aplicar las mejores prácticas.

El desarrollo de sistemas se puede considerar en dos grandes componentes: el análisis de sistemas y el diseño del sistema, donde, el análisis especifica que es lo que el sistema debe de hacer, y el diseño establece el como alcanzar este objetivo.

Para el presente trabajo se hará uso del Ciclo de vida clásico de desarrollo de sistemas

1. Investigación preliminar
2. Definición de requerimientos.
3. Diseño del sistema
4. Desarrollo.
5. Prueba de los sistemas.
6. Implantación.

Para este trabajo solo se incluye el análisis y diseño del módulo de Pollo Vivo, el cual se muestran sus ventanas de captura y sus reportes.

Se desarrollan cuatro capítulos, los cuales se describen a continuación:

En el Capítulo I, se describe la empresa en la cual se basa este análisis, comentando los procesos que se manejan, algo de historia de la misma y los controles implementados para ello.

En el capítulo II, se presenta la fundamentación de los sistemas basados en Windows, se comenta lo que el lenguaje de programación usada, la estructura cliente - servidor, la programación orientada a objetos.

En el capítulo III, se explica la forma en que se lleva a cabo el ciclo de vida de desarrollo de los sistemas, explicando cada fase.

En el capítulo IV, se presenta la implementación del módulo de pollo vivo, las ventanas de captura y los reportes que conforman este modulo.

Por último, se encuentran las conclusiones a las que se llegó con el desarrollo del presente trabajo.

CAPÍTULO I

La empresa Bachoco S.A. de C.V.

1.1. Antecedentes de la empresa Bachoco.

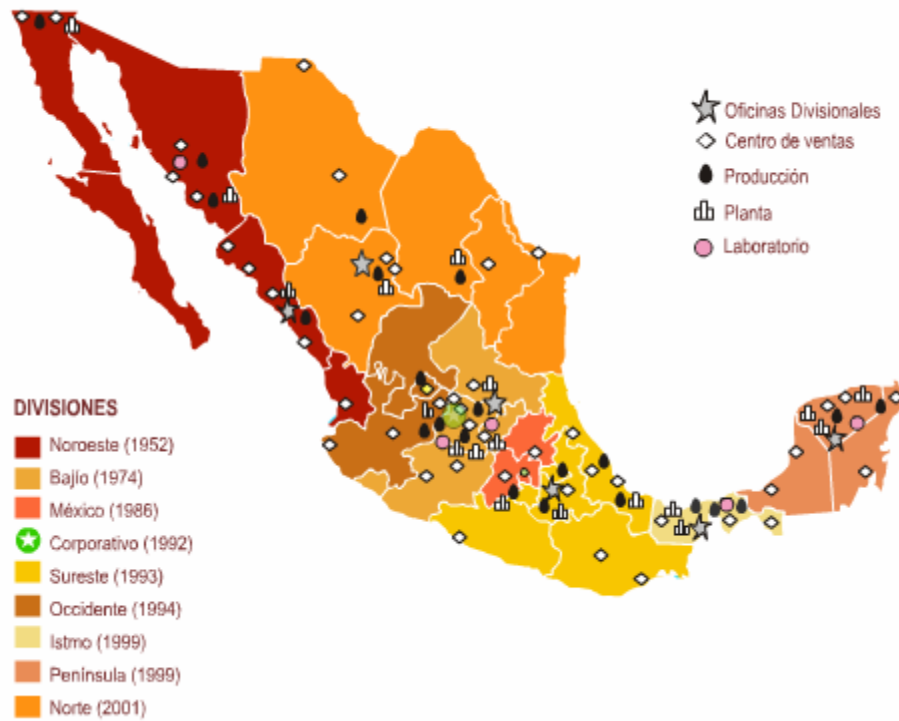
BACHOCO es un vocablo yaqui que significa “por donde pasa el agua”. Fundada el año de 1951, Bachoco, empresa mexicana, es hoy líder nacional en la producción y venta de pollo y huevo. Considerado el conjunto de operaciones en que participa es, asimismo, líder en su ramo.

El propósito central de la empresa es el de consolidarse como líder, con productos y servicios de la más alta calidad, para satisfacer las necesidades humanas de alimento de origen animal. Ello se respalda con un moderno sistema de producción, dotado de procesos eficientes y competitivos, mediante sistemas intensivos y controlables.

A la consolidación de su liderazgo coadyuvan, asimismo, la comercialización de los productos elaborados por la empresa, así como de otros de carácter perecedero, aprovechando su estructura; una rentabilidad que cumpla las expectativas de los accionistas y del desarrollo del negocio; así como el uso eficiente de los recursos y el desarrollo, colaboración e integración del personal.

1.2. Distribución geográfica.

Para una mejor atención a clientes y buscando cubrir la mayor parte del mercado en el país, Bachoco está distribuido geográficamente como se muestra en la siguiente gráfica:



1.3. Negocio y participación de mercado.

Bachoco elabora, mejora y vende los siguientes productos:

- **Huevo** de gallina para consumo humano. Éste se vende empacado en caja docenera y dieciochonera en tiendas de autoservicio y de abarrotes, o en cajas de 360 unidades a mayoristas y detallistas.
- **Pollo procesado** en varias presentaciones, dependiendo del tipo de mercado al que se dirige (mercado público, supermercado, rosticero y piezas de pollo, ya sea a granel o encharolado), las cuales se venden directamente a mayoristas, detallistas y tiendas de autoservicio. El procesamiento de las aves se realiza en plantas procesadoras propiedad de la empresa.
- **Alimento balanceado** para cerdo, ave y vacuno en venta comercial y para autoconsumo en plantas procesadoras de alimento.
- **Cerdo** en pie, en canal y en diferentes cortes.
- **Pie de cría.** Aves y cerdos para su utilización como reproductores o pías de cría.
- **Laboratorios pecuarios.** Vacunas bacterianas y productos farmacéuticos para aves y cerdos que se producen para autoconsumo y venta a terceros.

1.4. Ciclos vitales productivos.

Bachoco inicia, desarrolla y culmina distintos ciclos vitales, según el ramo productivo de que se trate:

Pollo de engorda: La producción de pollo de engorda se lleva a cabo teniendo por punto de partida aves progenitoras de raza Hybro que se importan de Holanda al primer día de nacidas. Las pollitas progenitoras gestan a las reproductoras, cuyos huevos, tras su estancia en cinco amplias y modernas incubadoras con capacidad para 2,497,000 huevos semanales, dan origen al pollo de engorda comercial. La población avícola permanece en 88 granjas propias en las cuales se obtiene el 90% de la producción total de la empresa y el 10% restante se obtiene en granjas rentadas.

Cada ciclo productivo se completa en un lapso de 7 a 8 semanas. La capacidad de producción anual es, en promedio, de 102 millones de pollos para consumo, los cuales se procesan en los cuatro rastros con que cuenta Bachoco.

La planta procesadora de aves de Celaya, Guanajuato, como ejemplo, tiene una capacidad instalada para 1,392,000 pollos semanales tipo mercado público, que es la presentación de mayor demanda, así como eviscerados tipo supermercado fresco, supermercado empacado al alto vacío, rosticero y partes de pollo en charolas.

En el mercado de la ciudad de México, Bachoco participa con un 30% del total. Siempre a la vanguardia, Bachoco ha introducido en el mercado el pollo con marca empacado al alto vacío, el cual es apoyado con publicidad específica para este producto.

Para satisfacer los requerimientos de insumos de estas dos líneas productivas, Bachoco cuenta con catorce plantas de alimento balanceado en todo el país, las cuales producen mensualmente 57 mil toneladas. Además, se cuenta con una capacidad de almacenamiento de 133 mil toneladas de grano, 26 toneladas de pasta vegetal y 71 unidades de transporte.

Estas plantas, a su vez, reciben insumos de dos empresas-soporte para fines de investigación y optimización nutricional:

- **Pecuarius Laboratorios:** Empresa en la cual Bachoco es el socio mayoritario, está localizada en Celaya, Guanajuato. Cuenta con una unidad moderna de diagnóstico y produce vacunas bacterianas y farmacéuticos veterinarios.
- **Vimision:** En la cual Bachoco tiene una participación importante. Se localiza en Ciudad Obregón, Sonora. Produce y comercializa pre-mezclas vitamínicas y de minerales para uso en alimentos balanceados en la planta de Ciudad Obregón. Es también el principal proveedor a nivel nacional de fosfato dicálcico, ingrediente importante en las raciones de los animales. Para ello, cuenta con una planta productora de ácido fosfórico en Empalme, Sonora y una planta productora de fosfato dicálcico en Guadalajara, Jalisco.

Huevo: La línea avícola ligera, cuyo producto terminal es el huevo, se produce en 64 granjas propiedad de Bachoco, con una capacidad total instalada de 4.5 millones de aves, y una producción anual promedio de 60 mil toneladas de huevo.

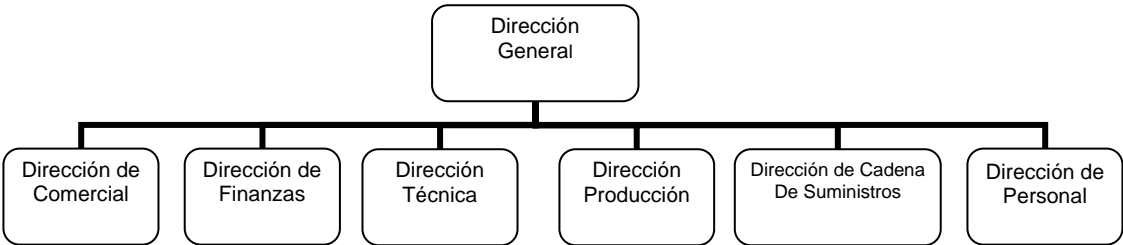
El ciclo de desarrollo con base en aves reproductoras y se lleva a cabo en una planta incubadora de Bachoco, con una capacidad de incubación de 1, 420,000 huevos al mes. Las aves así nacidas, son enviadas a las granjas, donde culminará el ciclo con la postura de huevo para plato o industrial.

El huevo para consumo humano se comercializa de tres maneras: en corrugados, con 360 huevos, en docenera y dieciochonera, siendo Bachoco la primera empresa mexicana en manejar este paquete para huevo de plato. Este novedoso concepto de comercialización en el mercado nacional, crea la pauta en el consumidor para la adquisición de huevo con marca.

Cerdo: La producción de cerdo de engorda, por su parte, también se lleva a cabo con animales híbridos, los cuales dan a luz a los cerdos de engorda comercial que se venden “en canal” y procesados a mayoristas y detallistas. Este ciclo se inicia en las granjas situadas en Mexicali, Ciudad Obregón, Navojoa, Los Mochis y Celaya. En su conjunto, se tiene una capacidad instalada de 6,600 vientres, con una producción anual de 100 mil cabezas.

1.5. Estructura organizacional de la dirección general.

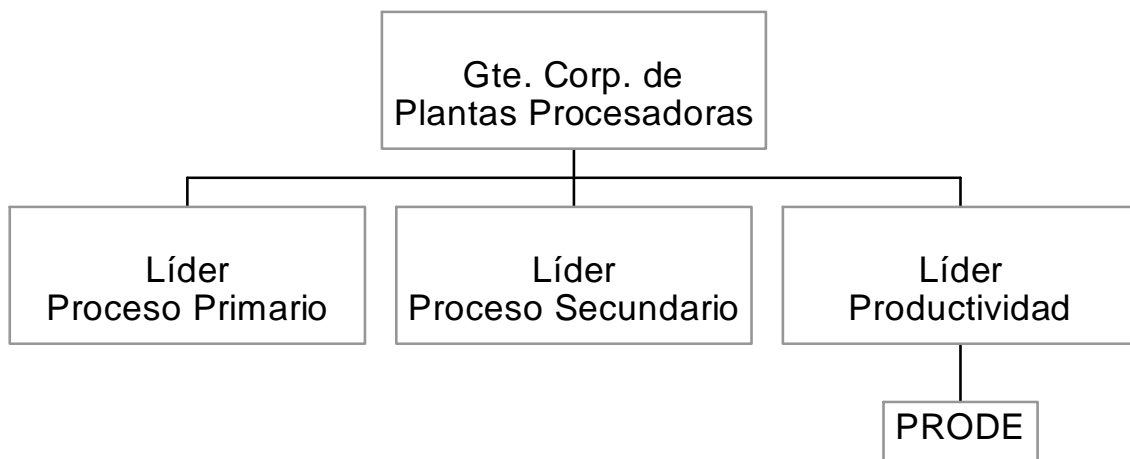
El organigrama de la dirección general de Bachoco está estructurado de la siguiente manera:



1.6. Infraestructura de Plantas Procesadoras de Aves.

Dada la importancia que Bachoco tiene del pollo procesado, se estructuró una gerencia corporativa de Plantas Procesadoras encargada de administrar el uso eficiente del recurso, tratando de dar un servicio eficiente a usuarios, cumpliendo con los estándares establecidos por cada uno de ellos, en lo que se refiere principalmente a la calidad del producto, esta gerencia depende de la Dirección Técnica.

El organigrama de la gerencia corporativa de Plantas Procesadoras de Aves esta estructurado de la siguiente manera:



Debido a la diversidad de productos que maneja Bachoco, se ha dividido las líneas de producción en cinco grandes familias de línea de Producción:

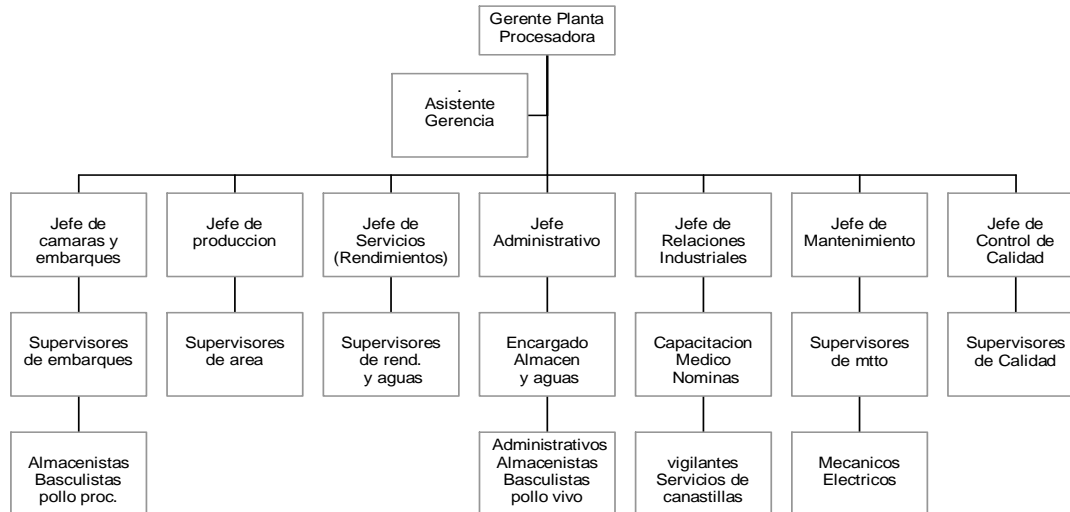
1. Mercado público
2. Mercado público eviscerado
3. New york
4. Eviscerado pesado
5. Eviscerado ligero

Para cubrir las necesidades de consumo de pollo Bachoco cuenta con siete Plantas Procesadoras de Aves distribuidas en las siguientes ciudades:



Dada la gran importancia que tienen las Plantas Procesadoras de Aves para Bachoco se ha implementado una organización funcional dentro de ellas.

La estructura organizacional de las Plantas Procesadoras de Bachoco, está formada como se ilustra en la gráfica siguiente:



No obstante los controles establecidos para mejorar la administración de las plantas procesadoras de aves, el crecimiento explosivo de la empresa y su volumen, han provocado que actualmente existan sistemas automatizados aislados, los cuales, debido a su falta de integridad dificulta y hace lento, poco confiable e inoportuno el procesamiento de la información requerida, ésto ha obligado a tratar de homogeneizar sistemas y controles en forma automatizada e integrada, buscando dar una respuesta completa a las exigencias de la alta gerencia.

Por la razón antes mencionada, se realizó el estudio de una propuesta, la cual consistía en desarrollar un sistema de información de plantas procesadoras de aves, para controlar la producción y administración de los diferentes procesos de la plantas procesadoras de aves, el proyecto fue bautizado con el nombre de “Sistema de Información de Plantas Procesadoras de Aves” (SIPPA).

1.7. Objetivo del Proyecto.

Desarrollar el “**SISTEMA DE INFORMACIÓN DE PLANTAS PROCESADORAS DE AVES (S.I.P.P.A.) MÓDULO POLLO VIVO**”, con el fin de agilizar las funciones operativas y administrativas de las Plantas procesadoras de aves de BACHOCO.

1.8. Situación Actual.

DEL PROCESO DE AVES

El procesamiento del ave es similar en las Plantas salvo por los distintos productos que salen de ellas.

En las Plantas Celaya y Coatzacoalcos se trabajan las líneas de Mercado Público y la de Eviscerado. En la línea de Eviscerado se producen el pollo tipo Rosticero, el pollo tipo Supermercado y del área de cortes se obtienen los cortes de tipo Bachoco, Deshuese, Americano, Shakey’s, Vampiro y Vip’s. Del sobrante de los cortes se obtiene Pasta y el sobrante de esta pasa a la Planta de Rendimientos.

En las Plantas de Culiacán y Merida solo se trabaja la línea de Eviscerado. Se producen el pollo tipo Rosticero, el tipo Supermercado y en el área de Cortes se obtienen los cortes tipo Bachoco, Deshuese, Americano y Kentucky Fried Chicken. De los sobrantes del área de cortes se obtiene Pasta y el sobrante de esta se pasa a la Planta de Rendimientos.

En las Plantas de Tecamachalco, Gómez Palacio y Lagos de Moreno se trabaja las líneas de eviscerado obteniéndose el pollo tipo Rosticero y la víscera por separado.

En todos los casos se recibe el pollo vivo de las granjas de engorda de la compañía, y en algunos casos de externas; antes de salir el camión para las granjas se pesa para obtener el peso tara del equipo que sale a la granja, una vez que el equipo regresa se vuelve a pesar con lo que se obtiene los kilos de entrada a la planta, en alguna operaciones se tiene el dato de peso de salida de granja, obteniendo la diferencia de estos se obtiene la merma en kilos por el tránsito; así también, si el pollo dura mas de una hora en espera en el anden se vuelve a pesar antes de entrar a la línea de proceso, con lo que se obtiene la merma en kilos por espera en andén. Al momento de colgarlo se separan los pollos muertos, con lo que se obtiene la mortandad

en tránsito en piezas y kilos. Una vez que fue sacrificado y dejado sangrar, la inspección de Control de Calidad baja de la línea al pollo de decomiso, que es aquel que fue mal matado, muy golpeado, con bajo peso o enfermo, este pollo no pasa a proceso por no ser óptimo para consumo. La última causa es dada a conocer directamente a los médicos de las granjas para controlar posibles epidemias y llevar el registro de los diferentes tipos de enfermedades.

El proceso anterior se aplica a toda el ave que se procesa, excepto a aquella que se vende viva, la cual solo se pesa a la entrada de la Planta; la disminución en el peso en cada una de las fases es importante dado que el rendimiento del producto se saca en base al peso que se metió realmente a la línea, no aquél que salió de la granja, de lo anterior es importante registrar el historial de viaje del ave para determinar las causas principales de pérdida para la compañía.

A continuación, se separa el pollo por línea de producción, el de Mercado Público y el de Eviscerado (en el caso de Celaya existe una línea para cada producto), más sucede que una vez separado se traspasan entre líneas de producción piezas por causas como son: alta demanda en una línea con poco pollo recibido o pollo de diferente peso al esperado del cual se puede obtener más provecho en la otra línea.

El pollo Mercado Público simplemente se mata, se desangra, se despluma, se descuelga de la línea y se mete en cajas a granel en forma de capas o camas, alternándolas con iguales capas o camas de hielo para su conservación. Se pesa en básculas de piso y se cuenta manualmente en todas las plantas que lo producen, y la remisión se elabora con dichos datos.

La línea de Eviscerado comienza el desangrado del ave, su desplume, el cortarle la cabeza al ave, luego las patas para después realizar el proceso de eviscerarlo (sacarle el paquete intestinal junto con pulmones, corazón, hígado y molleja), y por último se corta el pescuezo. Después se introduce en un equipo llamado "Chiller" que es un recipiente alargado de agua clorada donde el pollo permanece como hora y media para su lavado; la víscera comestible se lava de igual forma en un chiller especial mas pequeño y la no comestible pasa a la Planta de Rendimientos. En las cuatro plantas se trabaja igual.

Existe un producto de la familia Mercado Público que es el New York, que conserva patas, cuello y cabeza, pero se procesa como eviscerado.

A la salida del chiller, el pollo se selecciona de acuerdo a su peso con un sistema de básculas calibradas con rangos que se encuentran instaladas en la misma línea de la que cuelga el pollo, éste se empaqueta en cajas de cartón o rejillas de plástico cerradas y comúnmente pasa a las cámaras de conservación para reposarlo (escurrirlo). El pollo de esta forma se conoce como tipo Rosticero. Es en este pesaje que se considera la entrada inicial del producto al inventario de la Planta.

El pollo Supermercado es un pollo tipo Rosticero al que se le introduce un paquete de víscera comestible envuelto en papel encerado y un par de patas. Una vez conformado, de esta forma se empaqueta también en caja de cartón para su distribución. En la Planta de Culiacán se llega a distribuir producto en rejilla de plástico, observándose que ésta será la tendencia, salvo por su cliente de ellos, México que lo solicita todavía en caja de cartón.

El proceso de Cortes recibe el pollo tipo Rosticero para cortarlo y obtener sus partes (piernas, muslos, pechugas, alas, etc.), en diferentes combinaciones y empaquetarlo de esta forma. Esta área es un poco complicada dado que no solo se recibe entero y se corta, sino que algunos productos reciben otro proceso o corte diferente y se convierten en otros productos, lo cual implica entradas y salidas de productos en la misma área, en cuanto a manejo de inventarios se refiere; añadiéndose también que del sobrante o desperdicio de los cortes se obtiene pasta y el sobrante de esta pasa a la Planta de Rendimientos, el control de las entradas y salidas en esta parte es crítico, debido a que se pueden generar muchas pérdidas.

Al momento de embarcar el producto, se tienen en los andenes de embarque cajas refrigeradas para cada destino y comúnmente se están embarcando más de dos cajas al mismo tiempo en las cuatro plantas. El problema que se presenta aquí, aunque no es exorbitante, es el de introducir productos en las cajas erróneas, lo que genera falta de producto en unas bodegas y sobrante en otras. Pero existe un problema más preocupante, que aún minimizando este error operativo, las bodegas dicen recibir producto en cantidades y pesos, distintos a los que salen de las Plantas, lo que habla

de un deficiente control de la información con respecto a la distribución, siendo no posible muchas veces un registro confiable de lo que se embarca.

Para efectos de los rendimientos de productos de las Plantas, es vital registrar fielmente las pérdidas de kilos entre las diferentes fases del proceso, y esto se logra ubicando estratégicamente equipos de registro en los puntos claves de las transiciones.

Las Plantas Procesadoras de Aves enfrentan el problema de que no tienen información oportuna y confiable acerca de la producción y de los movimientos y situación de los inventarios de productos.

Haciendo un poco de historia, este problema tuvo los siguientes orígenes: se hicieron pruebas e intentos por costear el producto en las Plantas Procesadoras (Celaya y Culiacán en ese entonces). Éstas enviaban al área de Contabilidad del Corporativo un formato bastante extenso, donde resumían el pollo que habían recibido de granjas y el que habían procesado y en el íter, la sangre y plumas que habían quitado y la hidratación y escurrimiento que habían tenido. La producción se manejaba en piezas como unidad de medida base, aunque también registraban y trabajaban con pesos.

Por otro lado, la producción en las Plantas se ha ido incrementando con el tiempo, más no así su infraestructura para registrar la información en el mismo ritmo.

Todas las Plantas comenzaron, en un inicio, con registrar en formatos de papel la producción que obtenían, para luego sumarizar. La Planta de Celaya remisionaba sus productos por computadora, a la cual le capturaban la relación de cajas de eviscerado que subían al camión o la de carritos de Mercado Público que eran pesadas en la báscula de piso. El resto de la producción la registraba en formatos y luego la concentraban con hojas de cálculo electrónicas.

A fines de 1996 en la planta de Celaya se desarrolló un sistema que registrara las pesadas de las básculas del proceso automáticamente y que usara etiquetas con código de barras para el control de los inventarios y control de escurrimiento de productos. Este sistema fue desarrollado en Clipper 5.0 y se le adecuaron pantallas para la explotación de la información en Fox Pro para Windows; en la actualidad, todavía se siguen haciendo mejoras al sistema ya que ha tenido errores de programación que han alterado la integridad de los datos, la cual se ha tenido que

recuperar con accesos directos a la base de datos. Este desarrollo fue realizado fuera del control de Sistemas Corporativo y fue un intento de la planta de Celaya para llevar un registro más confiable de información.

La Planta de Culiacán tiene un sistema desarrollado en SISINF desde hace poco más de dos años, y este también registra los datos de las básculas automáticamente. Desde entonces, se le han hecho muy pocas actualizaciones y ahora el sistema está quedando obsoleto por no poder proporcionar la información de la manera que los usuarios lo requieren.

La Planta de Tecamachalco no tiene registro automático de pesadas de las básculas del proceso, como tampoco la Planta de Lagos de Moreno, y en ambas existe un sistema desarrollado en Fox Pro para DOS que les permite capturar la relación de productos que se embarcan para luego imprimir la remisión por computadora; también cuentan con un módulo para registrar las cantidades de pollo recibido de granjas, junto con sus características de viaje (flujo de tiempos), y sus características de mortandad, decomisos, pigmentación etc. Por último, tienen un módulo donde capturan las pruebas de control de calidad más usadas que aplican al producto.

El tener sistemas distintos para administrar la información, sobre todo hablando en un plano donde se tiende a la consolidación corporativa de datos entre las operaciones de la empresa, lo anterior también representa otro problema y es el no tener estandarizado el manejo administrativo de dicha información.

DE LA COMUNICACIÓN ENTRE LOS SISTEMAS

Debido a que los sistemas existentes en las Plantas, a excepción del de Culiacán, fueron desarrollados internamente en base a necesidades específicas de cada una, de acuerdo a sus capacidades e inquietudes personales por contar con un sistema y desarrollado por personal de sistemas contratado por la Planta, estos sistemas son prácticamente particulares.

Si bien la operación en las Plantas es la misma en el sentido de que su fin es similar, los controles llevados en cada una, sobre todo los administrativos, son diferentes ya que ha sido tarea de los últimos meses promover y establecer una estandarización en la operación a petición del Director de la misma, lo que en un inicio

causó, en este caso en particular, que cada Planta se promoviera por sus necesidades particulares, más no grupales.

Ahora bien, como resultado de lo anterior y de acuerdo a que con las capacidades de cada Planta los sistemas fueron desarrollados en diferentes plataformas, la comunicación entre ambos no es tan directa, presentándose la siguiente situación.

Como los sistemas de las Plantas Celaya, Tecamachalco y Lagos de Moreno fueron desarrollados en plataformas X-Base, la compartición de las bases de datos son a nivel de tablas sin problemas técnicos de comunicación, lo que reduce el problema a fijarse en qué estándares se encuentran dichas bases de datos, más con Culiacán, la compartición de la información solo es posible al nivel que en SISINF se acostumbra, a nivel ASCII, aparte de revisar la compatibilidad de las estructuras de datos.

Por todo lo anterior, no se puede decir que en la actualidad se cuente con información en línea proporcionada por las Plantas simultáneamente, como no sea a través de concentrados en hojas de cálculo, lo cual es prioritario cambiar.

1.9. Decisiones.

Actualmente cada Planta ha destinado sus recursos humanos y materiales para subsanar las deficiencias que de sistemas de información tienen, estableciendo gente para conteo manual de cajas, contratando capturistas para la generación de remisiones y especializándose ellos mismos en la configuración de sus equipos electrónicos con los del proceso al no contar con el apoyo de los proveedores.

El sistema de la Planta Celaya es actualmente el más completo dado que ha habido programadores dedicados al mismo desde su inicio, aunque, como se comentó anteriormente, ha sido detallado de acuerdo a las necesidades específicas de la Planta. El de Culiacán lleva tanto tiempo de no ser actualizado, que hay veces que no lo consideran confiable y lo complementan con trabajo manual. El de Tecamachalco y Lagos de Moreno les sirve para capturar lo que se remisiona y obtener concentrados, más el control de los inventarios es puramente manual. En el caso de las plantas de

Coatzacoalcos y Mérida se tienen aplicaciones de producción de terceros. En la planta de Gómez Palacio no se tiene ningún Hardware instalado.

Otro factor que se desprende de tener sistemas distintos en las Plantas es el de tener igualmente catalogados los productos que cada una obtiene con distintos códigos y a veces con distintas descripciones.

Debido a las situaciones antes mencionadas, se propone el desarrollo de un sistema que funcione para las cuatro Plantas, teniendo en cuenta los puntos de cercanía en el manejo de información de cada una. Haciendo caso de las disposiciones emitidas por el área comercial, se incluirán los códigos de productos únicos que debe haber en todas las Plantas.

Este sistema surge después de un análisis que involucró dos visitas a cada Planta, la primera para recabar las necesidades completas de control de información, partiendo desde considerar que no hubiere actualmente un sistema, conjuntar dichas necesidades de cada Planta comparándolas y obteniendo los puntos en común y sobre los puntos diferentes, promover cambios en el modo de manejo de información entre el personal interesado de cada Planta; obteniendo como producto final una base de estructuras de datos sobre las cuales se almacenaría la información del sistema para que el personal de las Plantas validara que dichas estructuras les sirvan, lo anterior en la segunda visita.

1.10. Beneficios.

TRÁFICO Y RECOLECCIÓN DE POLLO VIVO.

- Control de tiempos de viaje por destino.
- Estadísticas de viajes por tipo de producto, por destino, por distancias cubiertas, etc.
- Conciliación de producto recibido de granjas y control de mermas en tránsito.
- Concentración de información para obtener reportes por periodos de tiempo que eviten trabajo manual.

INDICADORES DEL PROCESO.

- Control y monitoreo inmediato de los diferentes parámetros de Control de Calidad como son pigmentación, golpeo, mal desplume, etc.

- Toma de decisiones oportunamente debido a la información de decomisos, pollo de segunda y variaciones del proceso.
- Reportes concentrados de los diferentes procesos por tipos de prueba y periodos de tiempo.
- Ganancia en tiempo que redunde en hacer análisis más detallados de la información.

CONTROL DE LA PRODUCCIÓN

- Apoyo en la generación de la programación de producción.
- La colección de datos automática proporciona control de inventarios más eficiente y movimientos de productos a reproceso más confiable.
- Reducción de mermas de productos entre los traspasos paulatinamente con tendencia a desaparecer.
- Reducción de tiempo en la elaboración de reportes que se usen en implementar nuevos y más eficaces controles de producción.
- Cierres de información de producción en cualquier día que permitan obtener rendimientos de productos cuando se requieran.
- Reportes de la producción al momento que se originan.
- Remisionado de productos automático con el uso de pistola lectora de código de barras acumulando salidas por lote y tipo de producto y eliminando tiempos perdidos ocupados actualmente en el proceso de generación de la remisión.
- Emisión de reportes de productos remisionados por bodega.
- Conciliación del producto remisionado a bodegas contra el recibido.

DISTRIBUCION

- Apoyo en la generación de la programación de los embarques.
- Control de tiempos de distribución de los productos.
- Estadísticas de viajes por destino, por producto y por periodos de tiempo.

ADMINISTRACIÓN

- Herramienta que permitirá monitorear la información necesaria y corregir desde su origen.

- Control de la información desde pollo en pie hasta remisiones.
- Reducción considerable de las diferencias en los diferentes puntos de emisión de la información.
- Reducción de tiempo en la elaboración en cierres de mes por la rapidez en la emisión de los reportes.
- Reportes de rendimientos para cierres mensuales

CAPÍTULO II

Fundamentación

2.1. Aplicaciones Windows.

El desarrollo de aplicaciones Windows es aun muy difundido a nivel mundial, por su enorme potencial de validación de entrada de datos a nivel local, así como la explotación de las funcionalidades de las estaciones de trabajo, aun así las aplicaciones Web están ganando cada vez más terreno. A continuación se muestra una comparación entre aplicaciones Windows y Web:

Característica	Aplicaciones Windows	Aplicaciones para Web
Localización y ejecución	La aplicación se contiene dentro de una o más carpetas, y requiere solamente del soporte de la infraestructura .NET .	La aplicación se contiene dentro de uno o más directorios virtuales, y utilizará los servicios de Internet Information Server (IIS) y la infraestructura .NET para ejecutarse. Para utilizar la aplicación se requiere de un explorador capaz de entender HTML.
Instalación	La instalación se realiza típicamente mediante un instalador. La mayor parte de las veces no requiere una conexión a una red interna o externa, y en el caso de que se deseen actualizar varios equipos se debe ejecutar el instalador tantas veces como los ordenadores lo requieran.	No requiere instalación en cada ordenador que desee hacer uso del mismo, aunque sí se necesita un explorador y conexión a la red. En el caso de que desee actualizar la aplicación, la tarea debe efectuarse solamente en el servidor.
Interfaz	La interfaz creada invoca funciones internas de Windows. Se cuenta con cientos de controles para realizar la misma. Para situar los mismos dentro de la pantalla, se utilizan coordenadas X e Y expresadas en píxeles. El resultado final puede verse igual en los diferentes sistemas operativos de Windows.	La interfaz es creada mediante etiquetas HTML. Se cuenta con una decena de controles para confeccionar la misma. Para situarlos dentro de la pantalla, se debe emplear el sistema de coordenadas de HTML o DHTML en el caso de que el explorador lo soporte. El resultado final puede verse parecido en diferentes exploradores pero no igual.
Gráficos	Los gráficos pueden ser creados mediante la utilización de GDI+	Los gráficos pueden ser creados mediante la utilización de GDI+ pero éste debe ser ejecutado del lado del servidor.
Ejecución de Eventos	La ejecución del código asociado a cada evento se produce tan pronto el mismo es iniciado, lo que ofrece una respuesta interactiva.	Los eventos no siempre son iniciados en forma automática, aunque si el explorador es Internet Explorer 5.0 o superior ASP .NET puede emplear en algunos casos DHTML, a los efectos de efectuar el evento en forma local (esto está sólo disponible en los controles de validación).
Acceso a recursos locales (memoria, archivos, red, etc.)	Las aplicaciones para Windows pueden tener acceso a los recursos, aunque existe la posibilidad de restringir los mismos.	El explorador restringe los recursos a los cuales la aplicación tendrá acceso.
Escalabilidad	Es medianamente escalable, y se deben tener en cuenta los costos de instalación y/o actualización.	Es sumamente escalable, y el costo de instalación no es fuerte, ya que se debe efectuar una sola vez por versión de aplicación.
Mercado	Clientes que utilizan sistema operativo Windows.	Clientes que utilizan cualquier sistema operativo.

Para el desarrollo de las aplicaciones de las plantas procesadoras de aves se uso el lenguaje de programación PowerBuilder 4.0 (Aplicaciones Windows), el cual es un poderoso lenguaje de programación orientado a objetos. A lo largo de esta sección describiremos algunas de las principales características de este software.

2.2. PowerBuilder Versión 4.0.

La versión 4.0 incluye nuevas características tales como:

- Soporte multiplataforma.
- Mejoras del modelo orientado a objetos.
- Mejoras en la edición de informes.
- Mejoras en el acceso a bases de datos.
- Mejoras en la conectividad.
- Nueva tecnología de compilación.
- Mejoras en el control de versiones.

Con la versión 4.0 de PowerBuilder se pueden realizar aplicaciones que pueden ser ejecutadas en entornos diferentes al de Microsoft Windows. Se prevé aumentar el número de plataformas, y eventualmente se podría incluir la HP/UX de Hewlett Packard, AIX de IBM, Solaris de Sun Microsystems, Macintosh System 7, Microsoft Windows NT y OS/2 de IBM. Se puede realizar una aplicación en un entorno y pasarla a cualquier otro, simplemente recompilándola en el nuevo entorno¹.

La versión 4.0 incluye nuevas funcionalidades para los objetos, entre las cuales se encuentran las que permiten crear o ampliar las clases no visuales. También se ha mejorado el acceso a objetos y clases externas, incluyendo herramientas KLL (Microsoft Dynamic Link Library).

1

Paúl Mahler, "PowerBuilder: Desarrollo de aplicaciones cliente servidor", Prentice Hall, 1995 , p.4

Ahora soporta la anidación de informes. Se puede incluir un objeto DataWindow dentro de otro objeto DataWindow. Ésto es muy útil en gran cantidad de aplicaciones, por ejemplo, para crear cabeceras de informes, secciones detalladas o para la realización de diversos formatos en un único informe.

Las tuberías de datos permiten conectar una fuente de datos con un destino simplemente definiendo la conexión. Pueden moverse decenas de miles de filas por minuto entre aplicaciones a través de estas tuberías de datos. También ha mejorado el soporte para los procedimientos de almacenamiento de Oracle 7.

La versión 4.0 soporta la arquitectura OLE 2.0 (Object Embedding and Linking) de Microsoft. También incluye la nueva versión Object Management Group (OMG) y Common Object Request Broker (CORBA), que se utilizan para la especificación de objetos.

Se ha incorporado una nueva tecnología de compilación Watcom. Incluye el generador de clases de objetos C++ Watcom. Con dicho generador se pueden construir librerías compatibles con C y C++.

La versión 4.0 mejora el control del código fuente y la gestión de la configuración. Puede gestionar diversas aplicaciones y desarrollos en paralelo.

Las versiones de las aplicaciones incluyen etiquetas de todos los objetos PowerBuilder. Una vez que los objetos se etiquetan es fácil realizar su seguimiento o devolverlo a cualquier versión anterior. Estas nuevas funcionalidades se incluyen en el producto Intersov PCVS, que se vende por separado.

2.3. Orientado a Objetos, Dirigido a Eventos.

En el pasado, con los lenguajes procedurales tradicionales como Basic, COBOL o C, el programador controlaba la operativa de ejecución de una aplicación. Un programa controlaba cuándo se abrían y cerraban las ventanas y el movimiento del cursor de un campo a otro. El programa en ejecución determinaba la secuencia de eventos. El programador determinaba dicha secuencia codificándola en el programa.

En las aplicaciones modernas con interfaces gráficas, es el usuario el que gestiona la secuencia de eventos que se producen en un programa en ejecución. El usuario determina cuándo comienza la aplicación, cuándo se abre y se cierra una ventana, cuándo se introduce el siguiente campo o cuando finaliza la aplicación. En lugar del programa o el programador, es el usuario el que controla los eventos.

En un entorno en el que el usuario es el que provoca los eventos a través de una interfaz gráfica de usuario, es difícil realizar aplicaciones con un lenguaje procedural. Es complicado escribir un programa para un entorno orientado a eventos cuando el programador es el que debe determinar la secuencia de eventos. Sin embargo, es fácil escribir programas para este entorno con los nuevos lenguajes orientados a objetos tal como PowerBuilder 4.0. Este está diseñado para trabajar con objetos y eventos. Tiene gran cantidad de facilidades predefinidas para la gestión de ventanas, eventos y objetos.

La realización de aplicaciones con PowerBuilder es fácil. Tiene herramientas para la creación de las aplicaciones y sus objetos, para la conexión a las fuentes de datos y para poder responder a los eventos que van ocurriendo mientras la aplicación se está ejecutando. Todo esto permite construir aplicaciones industriales fiables, de manera rápida, fácil y eficiente.

2.4. Una Interfaz Gráfica.

Una interfaz gráfica de usuario es un entorno orientado a objeto, dirigido a eventos. Las ventanas, así como los controles y otros objetos que aparecen en ellas, son objetos. Una imagen, un ícono o un gráfico son todos objetos.

Los usuarios pueden provocar varios eventos. Un usuario puede *hacer clic* con el ratón sobre una ventana. Los programas pueden también provocar eventos, por ejemplo, el evento cerrar de una ventana.

Los programas realizados con PowerBuilder responden a estos eventos, tales como *hacer clic* con el ratón. El comienzo de la respuesta al evento controla la operativa de la aplicación.

Los objetos tales como ventanas y controles tienen cada uno una lista de eventos asociados. El objeto aplicación de PowerBuilder tiene cuatro eventos asociados:

- close
- open
- idle
- system error

El objeto ventana tiene muchos eventos asociados, entre otros:

- open
- close
- move
- re-size

Algunos objetos de PowerBuilder, tales como las estructuras de datos y las funciones definidas de usuario, no tienen eventos asociados. Para poder realizar programas con PowerBuilder, quizá lo más difícil sea conocer los eventos y cómo gestionarlos. Se deben conocer los diferentes eventos de cada objeto y cómo realizar procedimientos y funciones para dichos eventos.

Por ejemplo, todas las ventanas tienen el evento *open*. Al abrirse una ventana se provoca el evento *open*. Se puede escribir un programa PowerScript para dicho evento. Cuando se abre la ventana, el procedimiento se ejecuta. Por ejemplo, se puede incluir en el procedimiento del evento *open* de una ventana la conexión a una fuente de datos.

Cualquier interfaz gráfica de usuario es un entorno con gran cantidad de eventos. Debido a esto, PowerBuilder tiene muchas funciones para realizar la gestión de dichos eventos.

2.5. El Modelo de Desarrollo.

Si se está acostumbrado a programar con un lenguaje procedural tradicional, se debe aprender una nueva forma de programar. PowerBuilder es un lenguaje orientado a objeto.

Se puede ver lo que hace un programa realizado en un lenguaje tradicional leyendo el programa. Por ejemplo, se puede tener un programa que presenta un formato de datos de entrada que permite al usuario introducir datos. Todas las operaciones relativas a la presentación y uso del formato se encuentran en el programa. Las sentencias, que se pueden imprimir y leer, describen todas las operaciones.

Una aplicación Power Builder no tiene un único lugar donde se pueda leer. No existe un listado que se pueda imprimir y leer para comprender la operativa de la misma.

Con Power Builder normalmente se realiza poca programación. Lo que se programa son los procedimientos de los eventos asociados a un objeto. Por ejemplo, cuando una aplicación se abre se provoca un evento. Es el evento open de la aplicación. Se puede escribir un procedimiento que se ejecute cuando se provoca el evento open, esto es, cuando se ejecuta por primera vez la aplicación. Se puede incluir en dicho procedimiento, por ejemplo, que el usuario dé el nombre de la fuente de datos de la aplicación.

La mayor parte del trabajo consistirá en crear objetos para la aplicación, no en programar cómo reaccionarán los objetos. Las herramientas del entorno de desarrollo PowerBuilder permiten crear fácilmente los objetos y escribir los procedimientos.

2.6. Programación Orientada a Objetos.

La programación orientada a objetos incorpora elementos que facilitan la realización de interfaces gráficas de usuario. PowerBuilder tiene herramientas que facilitan la realización de programas orientados a objetos. Estas herramientas se denominan painters.

Con los lenguajes de programación tradicionales se escriben programas. Estos están formados por módulos que se llaman en una determinada secuencia, determinada normalmente por el programa. Los datos van de acá para allá entre las rutinas del programa. El modelo de programación difiere en un entorno orientado a objeto, en el que se crean objetos que responden a eventos y que se comunican entre sí por medio de mensajes.

Objetos y atributos

Una aplicación Power Builder utiliza objetos tales como una ventana, un control, un dibujo o un icono. Los objetos tienen atributos. Por ejemplo, el objeto ventana tiene atributos tales como el tamaño, la forma o el color. Los objetos pueden incluir a su vez otros objetos. Por ejemplo, el objeto ventana de una captura de datos puede combinar una ventana principal, diversos objetos control y un objeto menú. Algunos de los objetos que se pueden utilizar, tales como menús y ventanas, los da PowerBuilder; otros, sin embargo, se tienen que crear.

Métodos

Una operación que requiere información sobre un objeto o que lo cambia de alguna manera se denomina *método*. En PowerBuilder los métodos se crean escribiendo los procedimientos. Por ejemplo, un procedimiento que abre una ventana o un procedimiento que la cierra son métodos de dicha ventana. En PowerBuilder los métodos son los procedimientos asociados con los diferentes objetos.

Eventos

Los procedimientos pueden incluir funciones. Una función es un procedimiento con nombre, que acepta argumentos de entrada y devuelve uno o más valores. Estas funciones pueden ser reutilizadas. Las funciones PowerBuilder son utilizadas en los métodos y pueden estar ligadas a un objeto determinado o ser globales a toda la aplicación.

En una aplicación Power Builder es el usuario más que el programa el que determina la secuencia de eventos de la misma. El usuario comienza la aplicación, abre y cierra ventanas, y provoca eventos, por ejemplo, haciendo clic sobre un botón de abrir o de cerrar. La aplicación puede también por sí misma provocar eventos. Por ejemplo, un programa que se está ejecutando puede provocar el evento idle después de un período predefinido de inactividad por parte del usuario. Se provoca dicho evento sin que intervenga el usuario.

Con Power Builder se pueden realizar procedimientos que determinan la respuesta a un evento. Un objeto ventana puede contener un procedimiento que responda al evento open o un procedimiento que responda al evento close. Cuando una aplicación utiliza un objeto, están a su disposición los eventos y los correspondientes procedimientos del objeto.

Mensajes

Cuando una aplicación se ejecuta, los objetos se comunican entre sí por medio de mensajes. Los mensajes enviados entre los objeto de una aplicación controlan a ésta. Supongamos un objeto ventana que contiene un objeto botón de finalización. Cuando la aplicación se ejecuta y se presenta la pantalla, aparece el botón finalizar. Al hacer clic sobre dicho botón, se cierra la ventana que está abierta. Al hacer el clic

sobre el botón, se provoca el evento clicked del botón finalizar. El procedimiento de este evento envía un mensaje de cierre a la ventana que contiene el botón. La ventana lo recibe provocando el evento close de la ventana, ejecutándose su correspondiente procedimiento.

Clases

Un lenguaje de programación tradicional contiene diferentes tipos de datos predefinidos, tales como *integres*, *real numbers* y *strings*. Con estos lenguajes tradicionales se pueden crear nuevas variables; estas nuevas variables deben ser de uno de los tipos predefinidos. Se puede utilizar una sentencia tal como

```
int a
```

Para crear una nueva variables denominada *a* del tipo integer. En este ejemplo, integer es un tipo que está predefinido en el lenguaje. En algunos lenguajes se pueden crear tipos de datos nuevos y crear variables pertenecientes a dichos tipos. Un lenguaje de programación orientado a objetos tiene clases en lugar de tipos de datos. Una clase en la programación orientada a objetos es similar aun tipo de datos en la programación tradicional.

Una definición de clase es un conjunto de atributos y métodos combinados para definir un objeto. Se puede tener una clase denominada ventana. La definición de la clase incluye los atributos de una ventana y los métodos que operan sobre ella. Los atributos incluyen la posición *x*, la posición *y*, la altura, la anchura y el tipo de ventana. Los métodos pueden incluir procedimientos que controlan la operativa de la ventana cuando ésta se abre o se cierra. Una clase es un marco para un objeto. En un lenguaje de programación orientada a objetos se crean clases. Una vez creadas éstas, se crean sus correspondientes objetos.

Con un lenguaje de programación tradicional se crean nuevos objetos a partir de los tipos de datos predefinidos. En la programación orientada a objetos se crean nuevos objetos a partir de cualquier clase. Una librería de clases es un conjunto de clases incluida en una librería Power Builder.

Instancias

Cada objeto pertenece a una clase. Cuando un programa crea un objeto, está *instanciando* el objeto. Una instancia es una manifestación de una clase. Por ejemplo, California es una instancia de la clases *estado*. Una ventana denominada *w_entrada* es una instancia de la clase ventana.

Herencia

Con la programación orientada a objetos, una nueva clase hereda todas las características de su padre. Cualquier cambio realizado sobre una clase se propaga descendentemente a toda la jerarquía de clase. Un painter de un objeto PowerBuilder permite crear un nuevo objeto que será miembro de una determinada clase. Por ejemplo, se puede utilizar el window painter para crear una nueva ventana que sea miembro de la clase *w_main*. Se puede utilizar un painter para crear una nueva clase. La nueva clase heredaría todos los atributos y métodos de su padre. Un objeto que es miembro de una nueva clase hereda todos los atributos de la clase padre. Se pueden incluir en la clase creada nuevos métodos y atributos, y utilizar los procedimientos tal y como se han heredado, ampliarlos o reescribirlos. Con PowerBuilder sólo se pueden crear nuevas clases a partir de las clases ventana, menú y objeto de usuario.

Crear un nuevo objeto a partir de la herencia de otro provoca que se cree una nueva clase. La selección en el correspondiente menú painter, ventana u objeto de usuario permite crear un nuevo objeto que hereda sus características de una clase existente, creándose a su vez una nueva clase.

Encapsulación y métodos

La encapsulación independiza y protege los datos. Un objeto encapsula sus datos y sus métodos. Por ejemplo, el tamaño de una ventana es un atributo encapsulado del objeto ventana. Para conocer el tamaño de una ventana se debe enviar un mensaje a la ventana que pida información sobre su tamaño, y solamente el objeto puede modificar sus datos encapsulados. Otros objetos pueden enviar mensajes a un determinado objeto solicitándole cambios, pero sólo el propio objeto puede realizar los cambios.

Polimorfismo

Polimorfismo significa que objetos pertenecientes a clases diferentes pueden aceptar el mismo mensaje. Se puede enviar el mensaje *print* a diferentes objetos. El mismo mensaje provocará que cada uno de los objetos imprima algo. Supongamos un pequeño negocio que envía diferentes tipos de paquetes. Los paquetes pequeños van por correo, los paquetes medianos por medio de un courier, los grandes por camión y los urgentes por avión. El mismo mensaje, *enviar*, sirve para cualquiera de los objetos. El efecto, sin embargo, es diferente para cada uno de los objetos. Cada objeto es enviado, aunque el método de envío es diferente.

En los lenguajes de programación tradicionales, las funciones que son diferentes deben tener nombres diferentes. Si se crea una función con el nombre de una que ya existe, se provocará un error al compilar o al linkar el programa. En un lenguaje de programación orientado a objetos, funciones diferentes pueden tener el mismo nombre. Como ejemplo de programación, imaginemos una aplicación con diversas ventanas correspondientes a diferentes clases. Cada una de las ventanas presenta diferentes datos a los usuarios. El mismo mensaje, *print*, puede dirigirse a cualquiera de las ventanas. Cada ventana puede utilizar una función que imprimirá los datos, independientemente de que cada ventana pertenezca a una clase diferente.

Sobrecarga de operadores

Normalmente un mensaje incluye uno o más argumentos. En PowerBuilder, funciones diferentes pueden tener nombres diferentes. Para que la función opere correctamente, los argumentos deben tener la capacidad de poder tomar diferentes tipos. Esta capacidad para usar la misma interfaz con diferentes propósitos es un ejemplo de sobrecarga de operadores. A continuación se presentan dos diferentes funciones utilizadas por dos objetos drawing correspondientes a dos clases diferentes. Las ventanas son una ventana de captura de datos y una ventana de gráficos.

```
w_captu.print (w_captur, 4, 3)  
w_grafi.print (w_grafi, "x título ", "y título")
```

En ambos casos el mensaje es el mismo, *print*. Ambas funciones tienen el mismo nombre, *print*. Los formatos de las funciones son los mismos, ambos mensajes tienen dos argumentos. Sin embargo, los argumentos son de diferentes tipos. La primera función utiliza dos argumentos enteros, la segunda dos cadenas. Los operadores están sobrecargados. En el primer caso, cada operador acepta un número; en el segundo caso, los operadores aceptan una cadena. La sobrecarga es posible ya que los atributos se analizan en tiempo de ejecución. El objeto que recibe el mensaje acepta los argumentos del tipo correspondiente. No hay errores de compilación, ya que no se chequea el tipo de los datos. El polimorfismo no se podría dar sin la facilidad de la sobrecarga. Por supuesto, PowerBuilder lo soporta.

2.7. Modelo Cliente - Servidor.

Con el aumento de la eficiencia, con las redes de computadoras más económicas y con un costo de personal más bajo, está surgiendo un nuevo modelo de computación. Este es el modelo cliente-servidor.

Un modelo cliente-servidor tiene tres componentes: un **servidor**, una **red** y **clientes**.

Un servidor es un computador que almacena y gestiona datos. Por ejemplo, en el servidor se puede ejecutar un sistema de gestión de bases de datos. Este contiene datos de alguna empresa, por ejemplo, puede contener todas las facturas de la misma.

El servidor es responsable de suministrar cualquier dato requerido por el usuario. El servidor obtiene los datos en función de la petición realizada y los envía al cliente. Es así mismo responsable del control de la concurrencia, del mantenimiento de la integridad de la base de datos y de la gestión de las transacciones. El servidor se responsabiliza de la seguridad de los datos y del control de accesos. Finalmente, también suministra el espacio de almacenamiento necesario para los datos.

Una máquina cliente solicita datos del servidor. El computador cliente ejecuta un programa localmente. Este programa está preparado para solicitar adecuadamente los datos al servidor. Así mismo, es responsable de la gestión de la presentación de los datos, de la interacción con el usuario y de la operativa de recuperación de datos. El computador cliente valida los datos introducidos por el usuario y genera las solicitudes de datos necesarias al servidor.

Una red conecta los clientes al servidor. Dos de las redes más comunes son Novell y TCP/IP. Una red Novel o TCP/IP puede conectar todas las estaciones de trabajo de una oficina al servidor de la base de datos.

Un cliente envía una solicitud de datos a un servidor. El servidor devuelve los datos seleccionados en respuesta a la solicitud. El cliente puede realizar cambios en los datos y devolverlos al servidor para incluirlos en la base de datos.

El computador cliente puede realizar la mayor parte de la manipulación o procesamiento de los datos. Por ejemplo, si el usuario está introduciendo datos, todas las validaciones y correcciones tienen lugar en el cliente. Cuando los datos son correctos, se envían a la correspondiente base de datos del servidor. El cliente puede solicitar también cambios que se realizan directamente en los datos almacenados en el servidor. Por ejemplo, el cliente puede enviar una solicitud al servidor para modificar o borrar todos los registros que están ligados a un determinado padre.

En el sistema de facturación de una compañía, el computador central (un servidor de red) mantiene una base de datos con toda la información relativa al sistema. Este servidor de base de datos está conectado a una red. Varios computadores personales y estaciones de trabajo repartidos por toda la compañía están también conectados a la misma red. Cada una de las estaciones de trabajo o computadores personales son un cliente.

La máquina que contiene la base de datos actúa ahora como un servidor de base de datos. Las máquinas cliente envían las solicitudes de datos a través de la red. El servidor obtiene los datos solicitados y los devuelve al cliente que los solicitó a través de la red. Una máquina cliente puede gestionar todas las interacciones con el usuario. Esto descargará la mayor parte del proceso desde el servidor al cliente.

Un programa Power Builder que se está ejecutando en una máquina cliente puede presentar un determinado formato de pantalla. El usuario puede introducir datos por medio de ese formato y el programa los utiliza para hacer una solicitud de datos al servidor.

Cualquier dato utilizado por la aplicación cliente está almacenado en la base de datos del servidor. Si el empleado que introduce los pedidos quiere información sobre ellos, dicha información proviene de la base de datos del servidor y va al correspondiente cliente a través de la red. Cualquier cambio realizado en las órdenes se reenvía de nuevo utilizando la red a la base de datos del servidor.

La aplicación cliente puede presentar los datos recuperados, por ejemplo, con un determinado formato de pantalla. También puede transformar los datos y presentarlos por pantalla. La aplicación podrá utilizar los datos recuperados del servidor para realizar un informe, u obtener una información resumen y presentarla con un formato normal o utilizando gráficos.

Datos Distribuidos.

En un entorno cliente-servidor, los datos pueden estar distribuidos en diferentes máquinas de una red. Estas, que actúan como servidores de bases de datos, pueden conectarse a una red. Los datos pueden estar distribuidos en diferentes servidores y en diferentes bases de datos. La facturación, las ventas y los envíos podrían estar en diferentes servidores, conteniendo cada uno de ellos la correspondiente porción de información de la empresa. Esta información puede encontrarse toda en un servidor y en una base de datos, en un servidor y en más de una base de datos, o en diferentes servidores y en diferentes bases de datos.

Power Builder permite construir aplicaciones que acceden a los datos desde cualquier lugar de la empresa. Múltiples conexiones concurrentes a diversos sistemas de gestión de bases de datos pueden acceder a datos en cualquier máquina. Se pueden desarrollar aplicaciones que acceden a datos remotos almacenados en mainframes, mini-computadoras o estaciones de trabajo, o acceder a datos almacenados en bases de datos locales.

Datos locales

Por supuesto, una aplicación Power Builder también puede ejecutarse en la misma máquina que contiene la base de datos. En el caso más simple, el cliente y el servidor se encuentran en la misma máquina, y por lo tanto no se necesita ninguna red. Power Builder se suministra con el sistema de gestión de base de datos relacional Watcom. Esta puede ejecutarse en una computadora personal con Microsoft Windows. Una aplicación que se ejecuta en Microsoft Windows puede acceder a la base de datos Watcom que también se está ejecutando en Microsoft Windows.

CAPÍTULO III

Metodología para la Implementación de Sistemas de Información

3.1. Metodología Usada.

Ciclo de vida clásico de desarrollo de sistemas.

La metodología empleada para el desarrollo del presente proyecto fue la del ciclo de vida clásico de desarrollo de sistemas (SDLC), la cual establece y define con claridad los siguientes aspectos:

- Las etapas y fases del proyecto, las actividades y tareas que las componen, así como los puntos de control necesarios para evaluar el proyecto.
- Políticas y lineamientos necesarios para la dirección del proyecto a implantar.
- La especificación de los resultados y productos esperados que deberán ser entregados al usuario.

El SDLC como metodología o práctica profesional presenta, un enfoque donde se hace imprescindible la activa y decidida participación del usuario, ya que de este, depende en gran medida la calidad de los resultados obtenidos.

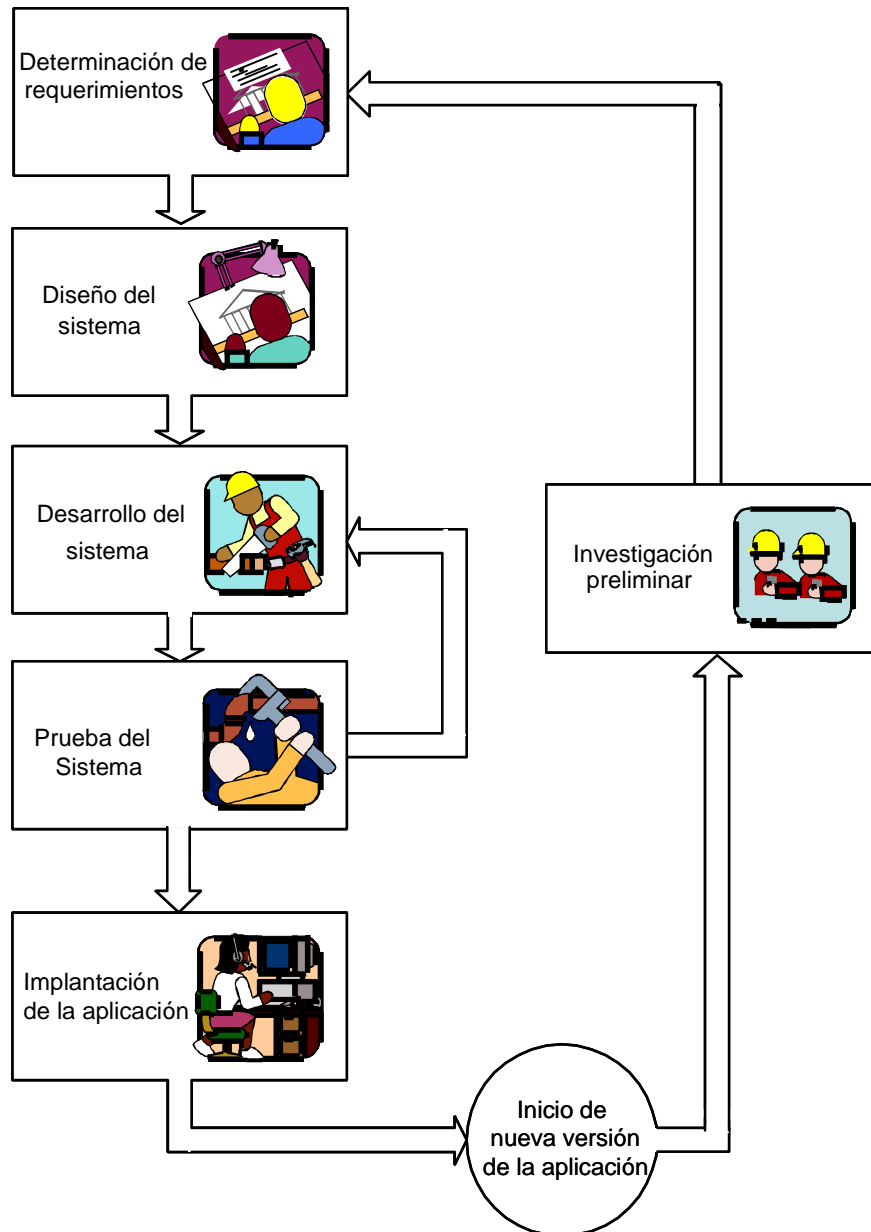
Es también gracias a este involucramiento que el usuario conoce permanentemente el estado que guarda el proyecto, fase, tarea o actividad, acumulando conocimiento y dominando las capacidades funcionales del producto mucho antes de ser implantado.

El SDLC al igual que el resto de metodologías similares, considera un ciclo de vida para el desarrollo e implantación de la solución integrando por fases, resumidas a continuación en forma breve:

1. Investigación preliminar
2. Definición de requerimientos.
 1. Diseño del sistema
 2. Desarrollo.
 3. Prueba de los sistemas.
 4. Implantación.

SDLC como se verá en el proyecto, se trató como base hacia el análisis orientado a objetos, cada uno de sus pasos será descrito a detalle en el siguiente capítulo.

Se muestra a continuación un diagrama del ciclo de vida clásico de desarrollo de sistemas:



Fuente: Aplicación creada por el autor de esta tesis.

3.2. Herramientas de Análisis y Diseño de Sistemas.

En el desarrollo de sistemas de información el uso de herramientas de análisis y diseño de sistemas es de usos cotidiano, a continuación se detallan algunas de las más usadas:

3.2.1. Planteamiento de objetivos.

El analista de sistemas debe determinar cuáles son los objetivos que se persiguen, para lo cual es preciso identificar las **necesidades de los usuarios**.

Dichos objetivos deben ser **claros, precisos y concisos**, pues de éstos dependerá en gran medida el éxito del producto final, ya que si éstos no son los adecuados o se prestan a malas interpretaciones el producto final no cubrirá las necesidades de los usuarios y en vano será todo el trabajo.

3.2.2. Entrevista con los usuarios.

La herramienta más importante en la etapa de planteamiento de objetivos es la entrevista con los usuarios, pues por medio de ésta, el analista de sistemas se da cuenta de cómo operan las cosas actualmente y como les gustaría a los usuarios que operaran en el futuro.

La forma más atinada para la realización de entrevistas es aplicarlas de arriba hacia abajo, es decir, comenzar por los niveles gerenciales y terminar con los trabajadores operacionales que participen en el sistema que se está estudiando.

A continuación se nombran una serie de aspectos que hay que tomar en cuenta si se desea tener éxito en una entrevista:

a) Antes de realizar cualquier entrevista es necesario solicitar autorización para la realización de la misma, ésto se debe hacer ante el jefe inmediato superior de la persona a ser entrevistada, y de ser posible que sea éste quien presente al analista con su subordinado.

b) Toda entrevista debe ser planeada con anterioridad con el fin de tener bien claro cuáles son los objetivos que se persiguen con ésta, es decir, realizar un esbozo de las preguntas que se van a hacer al entrevistado, así como el enfoque que se le va a dar a cada una de ellas; sin embargo el analista se debe permitir ciertos desvíos del plan, pues en ocasiones las respuestas obtenidas indican que es necesario ahondar en ciertos temas o quizá haya algunos temas que se tornen de poco o nulo interés, e incluso, hay ocasiones en las que hay que dar completamente un giro a lo que se tenía planeado.

En realidad, no siempre es necesario determinar con anterioridad las preguntas que se van a plantear, pues a veces el analista decide que es preferible realizar una entrevista menos formal en la que solamente se vaya dirigiendo al entrevistado hacia el tema o los temas que se desean conocer.

La decisión de realizar una entrevista bien planeada o una menos formal se toma basándose en la labor que el entrevistado realiza dentro de la organización y la información que se desea obtener.

c) Un punto importante para el éxito de la entrevista es buscar un horario oportuno para que el entrevistado no se esté distraendo, conteste todas las preguntas rápidamente sin analizarlas por tener trabajo pendiente, o se presente cualquier otra situación por la que no preste la atención que se necesita. Por lo anterior se recomienda hablar antes con el entrevistado y darle la oportunidad de determinar el día y la hora en que se realizará la entrevista, aclarándole que necesitamos de su completa atención.

d) El analista de sistemas debe causar buena impresión al entrevistado, debe ser cortés, nunca prepotente, debe transmitir confianza al entrevistado para que éste no sienta que se le está juzgando o fiscalizando; el analista debe presentarse vestido de una manera adecuada pues es difícil que alguien acceda a cooperar y poner su confianza en una persona sucia y desaliñada.

e) Se deben evitar las entrevistas largas, pues llega un momento en que el entrevistado se cansa y empieza a distraerse; por lo cual es preferible tener varias entrevistas cortas en lugar de una muy larga ².

f) Propiciar un ambiente adecuado de tal manera que la persona a entrevistar se sienta bien, para ello es necesario considerar ciertos aspectos tales como el dejar hablar al entrevistado, no interrumpirlo a menos de que se esté desviando demasiado del tema. El analista no debe suponer nada ni dar algo por hecho, pues el entrevistado se puede confundir o sentir mal. Hay que respetar a cualquier persona sea cual fuere su puesto dentro de la organización. Es imprescindible también tratar de hablarle a cada entrevistado en su lenguaje, pues no podemos utilizar el mismo con un gerente que con el trabajador de menor nivel.

g) El analista debe concretarse al tema, no empezar a tratar asuntos que no tienen nada que ver con el estudio del sistema.

2

Jerry Fitzgerald, Stallings Warren. "Fundamentos de Análisis de Sistemas". México, Continental 1989 p.110.

h) El analista debe tener despiertos sus cinco sentidos, debe estar pendiente de lo que pasa alrededor y del lenguaje corporal del entrevistado, pues en muchas ocasiones estos aspectos revelan cosas importantes.

i) Inmediatamente después de terminada la entrevista ésta se debe transcribir y documentar, incluso se pueden elaborar diagramas que contribuyan a una mejor comprensión de lo investigado.

Una grabación de la entrevista es de gran ayuda para la documentación, pero ésta se debe hacer sólo con el consentimiento del entrevistado. Un resumen y auto evaluación podría servir de gran ayuda para entrevistas posteriores.

3.2.3. Preguntas en la entrevista.

Las preguntas de la entrevista deben estar encaminadas a conocer perfectamente todas las actividades que se realizan en el sistema, así como las entradas requeridas y las salidas que éste produce; y por supuesto determinar cuáles son los cambios que necesita el actual sistema.

A continuación se presentan algunas preguntas que no deben faltar en una entrevista bien planeada.

El analista deber agregar aquéllas que considere necesarias. Primero se aplican las preguntas que ayudan al conocimiento del sistema actual.

Generales

- ¿Qué actividades se realizan?
- ¿Quién lo hace?
- ¿Cómo lo hace?
- ¿Cuándo, dónde, cómo, por qué, para qué lo hace?
- ¿Cuánto dura cada una de estas actividades?
- ¿Qué políticas de decisión se siguen?
- ¿Qué costumbres se tienen en el sistema?
- ¿Cuáles son los controles con los que se cuenta?

Entradas

- ¿Qué entradas hay al sistema?
- ¿De dónde vienen?
- ¿Cuándo vienen?
- ¿En qué formato?
- ¿Cómo se procesan las entradas?
- ¿Qué control de entradas se tiene?

Salidas

- ¿Qué salidas hay?
- ¿De dónde salen?
- ¿Cuándo salen?
- ¿En qué formato salen?
- ¿A dónde van?
- ¿Qué control de salidas se tiene?

Equipo de procesamiento

- ¿Cuáles son las características del equipo?
- ¿Qué aplicaciones se corren en el equipo?
- ¿Es seguro el equipo?
- ¿Qué controles se tiene?
- ¿Hay integridad?
- ¿En qué ambiente se trabaja - red, independiente, etc.?
- ¿Qué capacidad tiene el equipo?

Una vez que se conoce cómo trabaja el sistema actual, se procede a investigar los posibles cambios al sistema, para lo cual proponemos las siguientes preguntas:

- ¿Cuáles son los problemas más comunes dentro del sistema?
- ¿Qué agregaría a todo el proceso?

- ¿Qué eliminaría?
- ¿Qué cambios le haría?

3.2.4. Rentabilidad de los objetivos.

Cuando se han terminado las entrevistas con los usuarios y se conocen las necesidades y opiniones de los usuarios, el analista de sistemas está en condiciones de plantear ya en forma escrita los objetivos del "nuevo sistema".

Una vez redactados los objetivos se debe llevar a cabo un estudio de su rentabilidad, el cual debe ser revisado y aprobado por los altos mandos de la empresa, quienes decidirán si se realizará o no el desarrollo del sistema. Ésto con el objeto de tener la seguridad de lo que se está haciendo y por qué se está haciendo, pues sería frustrante que en etapas posteriores, o incluso una vez terminado el desarrollo, nos diéramos cuenta de que dicho desarrollo no proporciona ningún beneficio económico a la empresa, pues bien sabemos que el objetivo principal de cualquier empresa es "ganar dinero"³.

Si se tuvieran subprogramas que dependan de uno de los programas del segundo nivel, se anotarían abajo de éste en el siguiente nivel, y así sucesivamente.

Cuando el analista de sistemas ha terminado la elaboración de árboles Modulares ya está en condiciones de realizar la "organización de Módulos".

3.2.5. Elaboración de Diagramas de Flujo.

Los diagramas de flujo son una herramienta muy útil en la etapa del Análisis. Para un sistema proponemos dos tipos de diagramas de flujo:

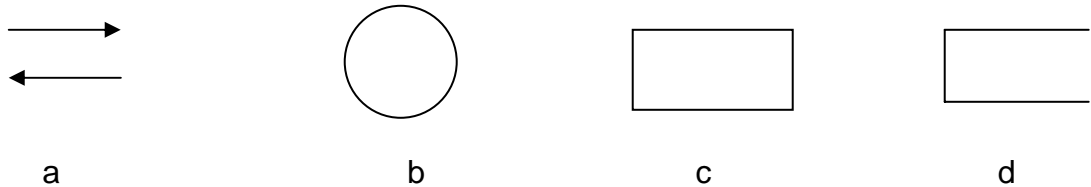
- Diagramas de Flujo de Datos.
- Diagramas de Flujo del Programa.

3.2.5.1. Diagramas de Flujo de Datos.

Este tipo de diagramas muestran cuáles son las entradas, salidas y procesos del sistema; así como los orígenes y destinos de los datos.

Es recomendable no hacer muy detallado ese tipo de diagrama, pues el objetivo principal es tener una visión general del sistema sin profundizar mucho en cada una de las operaciones que este realiza. Se puede elaborar un diagrama en el que se observe cómo está ubicado el sistema dentro de la organización y las relaciones que tiene con los demás sistemas en caso de que existan y otro en el que se muestre cómo intercalan los diferentes módulos que forman parte del sistema⁴.

Los diagramas de Flujo de Datos cuentan con cuatro tipos de gráficos que son explicados a continuación:



a) Flujo de datos. Son los datos que fluyen en las diferentes operaciones del sistema. Sobre la flecha debe tener el nombre con que se identifican los datos.

b) Procesos. Representan la conversión de datos de entrada en datos de salida. Cada proceso debe tener un número y un nombre.

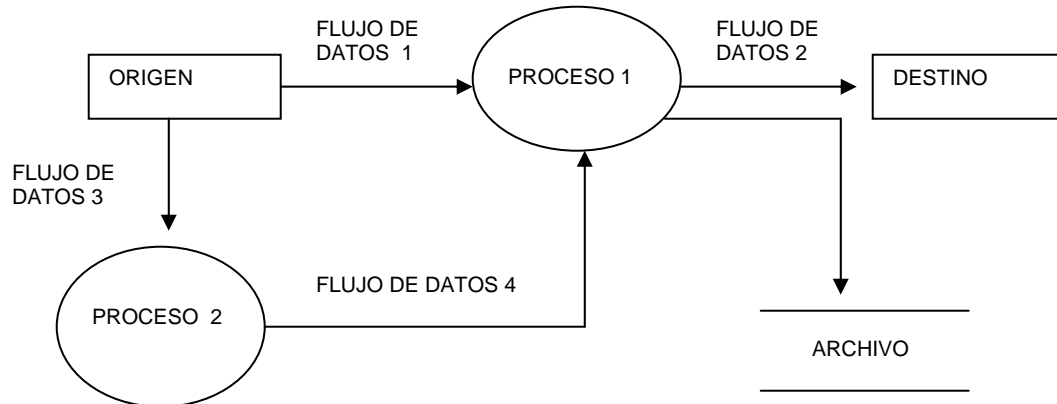
c) Origen o destino externo de datos. Representan personas u organismos que no interesan en el estudio del sistema, pero operan como un origen o destino de los datos.

4

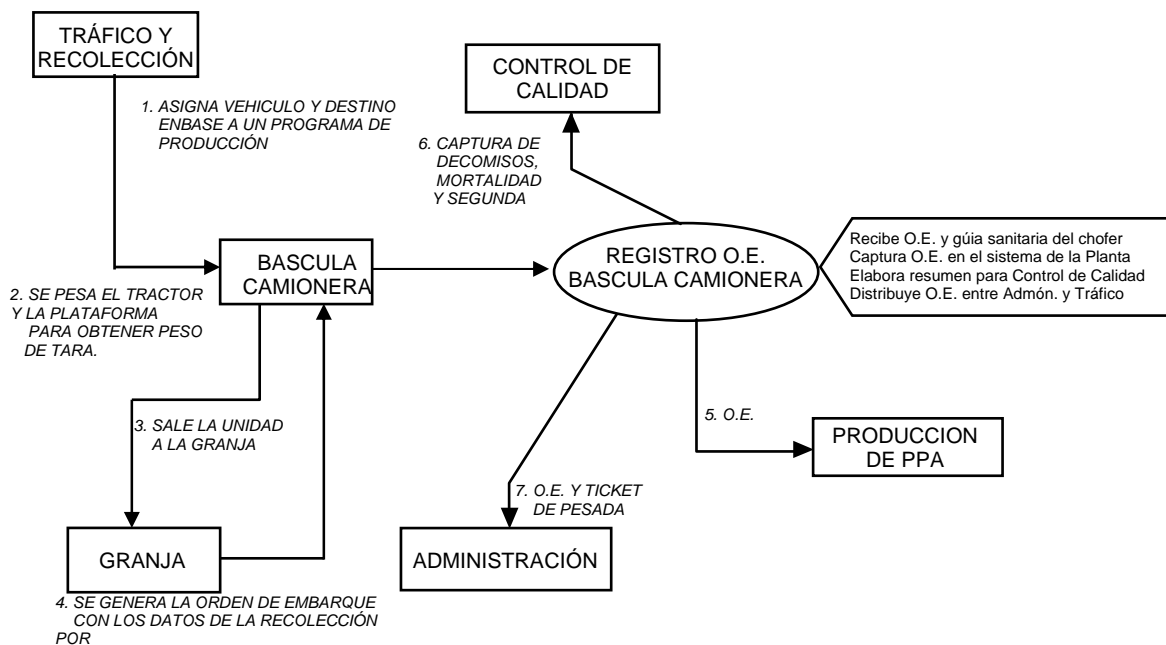
Senn James A., "Análisis y Diseño de Sistemas de Información", México, Mc Graw Hill, 1990, p. 118

d) archivos.- Son almacenes de datos. Cuando una flecha de flujo de datos apunta hacia el símbolo de archivo, se indica que se esta almacenando información en el archivo, pero si la flecha sale de él, es indicio de que se está obteniendo información del usuario.

A continuación se muestra un ejemplo de un Diagrama de Flujo de Datos.



A continuación se muestra el diagrama de flujo del pollo vivo que viene de granjas hasta el proceso:



3.2.6. Determinación de Requerimientos.

El objeto de la etapa de determinación de requerimientos es definir lo que el sistema debe ser capaz de realizar, para ello es necesario determinar cuáles serán las entradas, salidas, operaciones y recursos que necesitará el sistema para operar adecuadamente y cubrir las necesidades de la organización. No obstante debe quedar bien claro que en esta etapa aún no estamos rediseñando el sistema, sino determinando cuáles son los criterios generales de funcionamiento, mismos que nos ayudarán en el rediseño del sistema⁵.

Para la determinación de requerimientos el analista debe basarse principalmente en los datos obtenidos en las etapas de Planteamiento de Objetivos y Análisis del Sistema, pues éstas nos dicen cuáles son las necesidades de los usuarios. Pero también se deben tomar muy en cuenta los planes futuros de la organización para lograr que el "nuevo sistema" se ajuste a dichos planes. El analista de sistemas debe determinar los requerimientos del "nuevo sistema" en el siguiente orden:

- 1.- Salidas que debe producir el sistema, como son reportes, documentos, desplegados, etc.
- 2.- Entradas necesarias para producir las salidas esperadas, las cuales pueden ser tomadas de lentos fuentes o ser introducidas directamente al sistema.
- 3.- Todas las operaciones que debe realizar el sistema para producir las salidas esperadas.
- 4.- Los recursos que se necesitan para la operación del sistema, como son: hardware, software, recursos humanos, materiales y técnicos.

5

Victor Gerez, "Desarrollo y Administración de Programas de Computadora", México, Continental 1984, p.51

Es necesario que el analista determine primero cuáles serán las salidas que debe producir el sistema, y basándose en éstas podrá determinar cuáles son las entradas que se requerirán, qué operaciones deben llevarse a cabo y con qué recursos se debe contar para producir las salidas deseadas. Así como definir los controles con los que se debe contar.

Existe una nomenclatura utilizada dentro del análisis de sistemas la cual consiste en especificar los documentos ya sean de entrada o salida con una letra antecedido al requerimiento que sea para indicar el tipo de documento que se está utilizando. A continuación se especifican los valores y su significado.

O : Original

N : Nuevo

C : Cambios

B : Baja

Para poder tener una visión de lo anterior se muestra la forma en que es utilizada:

Requerimiento 1: O : Captura de Orden de Embarque

Se requiere que en el sistema se capturen los valores de la orden de embarque del pollo vivo que llega de la granja, esta captura se hará en el sistema y se complementará con los pesos obtenidos en la báscula camionera de la Planta.

El formato de la orden de embarque es el que se muestra a continuación:

BACHOCO, S.A. DE C.V.
Orden de Embarques de Pollo en Pie



BACHOCO^{M.R.}

No. de Ordenes	
Producción R/3:	
Compra R/3 :	

Fecha de Embarque		
Día	Mes	Año
22	12	2014

No. de Folio
Nº 33818

ORIGEN	
Operación:	LAGOS DE MORENO
No. operación:	0021

DESTINO	
Bachoco (PPA):	
Operación:	L20
Terceros (CTE):	
No. de Centro en R/3:	
Dirección:	

CARACTERISTICAS DEL POLLO EMBARCADO			
Granja: Pollo #	Caseta No.:	Caseta con tubos?:	Lote No.:
Comederos levantados ?:	Hr. de dietado:	Hr. retiro de agua:	
Tipo de Producto:	Aves consignadas:		
Temperatura:	Edad:	Sexo:	

DATOS DEL TRANSPORTE			
No. de Unidad:	No. de Plataforma:	No. de Dolly:	
No. de cuadrilla:	No. de espacios:	Aves por espacio:	
Operador:			
Montacarguista:			

RECEPCION DEL EMBARQUE			
HORARIO DE MANIOBRA		PESOS DEL EMBARQUE	
		Peso en Granja	Peso en anden PPA
Salida de PPA:		Bruto:	
Llegada de Granja:		Tara:	
Inicio de Carga:		Neto:	
Término de Carga:		Promedio:	
Salida de Granja:			
Llegada a PPA:			

Pollos muertos al arribo a PPA y/o Cliente: _____ Kgs. de los Pollos Muertos: _____

Observaciones: _____

Supervisor de Embarques

Encargado de Granja

Operador

Supervisor de Anden

EMBARQUES

3.2.7. Diccionario de Datos.

El diccionario de datos contiene las características de las entidades y atributos, que definen la estructura de la Base de Datos (Meta - Base de Datos). El objetivo del diccionario de datos es:

- Apoyar el diseño de la Base de Datos
- Facilitar el control de cada una de las entidades y atributos que forman parte de la estructura de Base de Datos del Sistema.
- Controlar dinámicamente. la estructura de la interfase al usuario, para las diferentes pantallas del sistema.

Ejemplo de diccionario de datos por entidad haciendo referencia a la Orden de Embarque.

SISTEMA	S.I.P.P.A.	SUBSISTEMA:	POLLO VIVO		
TABLA	ORDENES_EMBARQUE	DESCRIPCION	Ordenes de Embarque de pollo vivo.		
PROPOSITO :	Este catálogo contiene información referente a la orden de embarque que acompaña al pollo vivo que llega a la Planta para ser procesado.				
			ATRIBUTOS	TAMAÑO	
			44	255	

ATRIBUTO	DESCRIPCION	TIPO DE DATO	LONGITUD (bytes)	DOCUMENTO ORIGEN
orden_embarque	Número que identifica el embarque.	int	4	Cualquier valor numérico.
lote_int	Lote del día	smallint	2	PPA.dbo.c_lotes_produccion.lote_int
turno	Turno de proceso	smallint	2	Cualquier valor alfanumérico.
cve_linea_pollo	Clave que identifica la línea de producción.	smallint	2	PPA.dbo.c_lineas_pollo.cve_linea
cve_ppa	Clave que identifica a la PPA.	char	8	PPA.dbo.c_ppa.cve_ppa
cve_linea_fletera	Clave que identifica a la línea fletera.	smallint	2	Cualquier valor numérico.
cve_operador	Clave que identifica al operador.	smallint	2	Cualquier valor numérico.
cve_remolque	Clave que identifica al remolque.	smallint	2	Cualquier valor numérico.
cve_plataforma	Clave que identifica la plataforma.	smallint	2	Cualquier valor numérico.
cve_dolly	Clave que identifica al dolly	smallint	2	Cualquier valor numérico.

aves_solicitadas	Cantidad de aves que fueron solicitadas	int	4	Cualquier valor numérico.
cve_granja	Clave que identifica la granja.	int	4	PPA.dbo.c_granjas.cve_granja
cve_caseta	Clave que identifica la caseta.	varchar	20	Cualquier valor alfanumérico.
edad	Edad del producto.	real	4	Cualquier valor numérico real.
sexo	Sexo del producto.	varchar	1	Un único carácter.
h_salida_ppa	Hora de salida de la PPA.	datetime	8	Datos de tipo Fecha y Tiempo.
h_llegada_granja	Hora de llegada a la granja.	datetime	8	Datos de tipo Fecha y Tiempo.
h_inicio_carga	Hora de inicio de Carga.	datetime	8	Datos de tipo Fecha y Tiempo.
h_termino_carga	Hora de finalización de Carga.	datetime	8	Datos de tipo Fecha y Tiempo.
h_salida_granja	Hora de Salida a granja.	datetime	8	Datos de tipo Fecha y Tiempo.
cve_cuadrilla	Clave que identifica a la cuadrilla.	smallint	2	PPA.dbo.c_cuadrillas.cve_cuadrilla
aves_consignadas	Cantidad de aves consignadas.	int	4	Cualquier valor numérico.
h_ultimo_dietado	Hora de suministro de la última comida.	datetime	8	Datos de tipo Fecha y Tiempo.
f_llegada_ppa	Fecha de llegada a la PPA.	datetime	8	Datos de tipo Fecha y Tiempo.
peso_llegada_bruto	Peso de producto en bruto a su llegada.	real	4	Cualquier valor numérico real.
peso_llegada_tara	Peso del transporte sin producto	real	4	Cualquier valor numérico real.
peso_llegada_netto	Peso neto del producto a su llegada.	real	4	Cualquier valor numérico real.
peso_llegada_prom	Peso promedio del producto a su llegada.	real	4	Cualquier valor numérico real.
h_entrada_proceso	Hora de entrada a proceso.	datetime	8	Datos de tipo Fecha y Tiempo.
peso_proceso_bruto	Peso de producto en bruto al pasar a proceso	real	4	Cualquier valor numérico real.

peso_proceso_tara	Peso del transporte sin producto	real	4	Cualquier valor numérico real.
peso_proceso_netto	Peso neto del producto al pasar al proceso	real	4	Cualquier valor numérico real.
peso_proceso_prom	Peso prom. del producto al pasar al proceso	real	4	Cualquier valor numérico real.
aves_recibidas	Cantidad de aves recibidas.	int	4	Cualquier valor numérico.
estado	Identifica el estado de la remisión	smallint	2	Cualquier valor numérico.
cve_cliente	Clave que identifica al cliente.	smallint	2	PPA.dbo.c_clientes.cve_cliente
observaciones	Observaciones referentes al embarque.	varchar	50	Cualquier valor alfanumérico.
orden_full	Referencia a la orden de embarque que hace full	int	4	Cualquier valor numérico.
peso_granja_bruto	Peso de producto en bruto en granja	real	4	Cualquier valor numérico real.
peso_granja_tara	Peso del transporte sin producto	real	4	Cualquier valor numérico real.
peso_granja_netto	Peso neto del producto de granja	real	4	Cualquier valor numérico real.
peso_granja_prom	Peso promedio del producto de granja	real	4	Cualquier valor numérico real.
consecutivo	Referencia de entrada a proceso de la OE	smallint	2	Cualquier valor numérico.
orden_produccion_granja	Orden de producción de granja de SAP/R3	varchar	12	Cualquier valor alfanumérico.

RELACION DE INDICES

LLAVE PRINCIPAL	REFERENCIADA POR	ALTERNA 1	ALTERNA 2	ALTERNA 3
orden_embarque	PPA.dbo.pigmentacion	lote_int	cve_line	cve_ppa
	PPA.dbo.pollo_a_proceso			
	PPA.dbo.producto_decomisado			
	PPA.dbo.producto_muerto			
	PPA.dbo.producto_segunda			
	PPA.dbo.producto_seleccion	ALTERNA 4	ALTERNA 5	ALTERNA 6
	PPA.dbo.pruebas_producto	cve_granja	cve_cuadrilla	cve_cliente
	PPA.dbo.traspasos_pollo_vivo			
	PPA.dbo.acumulado_mp			

Nota: Los elementos descritos en el diccionario de datos deben provenir estrictamente de los formatos en los requerimientos establecidos en el proyecto.

3.2.8. Modelo Entidad – Relación.

El modelo **Entidad – Relación** fue propuesto por Peter Chen, en 1976, usado como el modelo sobre el cual se soporta el diseño de una base de datos.

Permite crear un modelo de datos en términos de: Entidades, sus Atributos y Relaciones entre las entidades.

3.2.8.1. Componentes Modelo Entidad Relación

- A) Entidades
- B) Atributos
- C) Relaciones

A) Entidades

Son los objetos principales acerca de los cuales se almacena información.

Son cosas de importancia o interés para un área de negocios o para un sistema que requiere del almacenamiento de datos.

Ejemplos: Personas, Lugares, Cosas o Eventos de interés.

Nombre Entidad

B) Atributos.

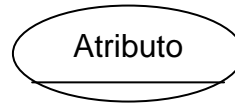
Son las características de las entidades.

Describen a las entidades.

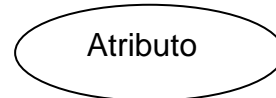
Representan características o cualidades de una entidad.

Ejemplos: Nombre de una persona, Nombres de ciudades, Número de empleado, Fecha de contratación, Monto a pagar.

Atributo identificador:
Identifican de manera única a cada
Ocurrencia de la entidad.



Descriptor:
Describen una característica o
cualidad de la Entidad.



C) Relaciones

Permiten representar diferentes tipos de relaciones entre entidades.

Tienen “semántica “, es decir, almacenan información acerca de la forma en que se asocian las entidades.

*Se dibujan así: _____
RELACIÓN.

3.2.8.2. Modelación gráfica de datos.

El propósito de los diagramas es mostrar las Entidades de datos y cómo éstas se relacionan.

El DER se concentra sólo en las entidades de datos.

Para construir un DER.

Pregunta inicial:

¿Cuáles son las entidades de interés acerca de las cuales se desea almacenar datos?

Para un negocio comercial podrían ser: PRODUCTOS, INVENTARIO, PROVEEDORES, FACTURAS, ORDENES DE COMPRA.

Construcción:

Dibujar un bloque para entidad identificada.

Para el nombre de las entidades, se recomienda que sean en singular. Ejemplo: CLIENTE en lugar de CLIENTES, ARTICULO en lugar de ARTICULOS.

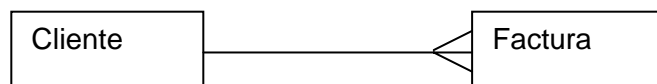
Siguiente pregunta:

¿Cuál relación existe entre cada par de entidades?

Para identificar las relaciones entre dos entidades se realizan dos preguntas:

- Primer pregunta de izquierda a derecha.
- Segunda pregunta de derecha a izquierda.

Ejemplo de Cliente - Factura



¿Un cliente cuantas facturas puede tener? Muchas.

¿Una factura cuantos clientes tiene? Uno.

Después se tiene que analizar si la relación es obligatoria y se representa así: —+

Se revisa en ambos sentidos a través de dos preguntas.

¿Una factura puede existir sin un cliente? No, ya que es obligatorio tener un cliente.

¿Puede un cliente existir sin factura? Si ya que no es obligatorio tener una factura.

Ejemplo de Producto - Factura



¿Un producto puede estar involucrado en muchas facturas? Sí.

¿Una factura cuantos productos puede tener? Muchos.

¿Una factura puede existir sin producto? Sí ya que no existe obligatoriedad.

¿Un producto puede existir sin una factura? Sí ya que no existe obligatoriedad.

En una relación de muchos a muchos se le debe asignar un nombre a la relación.

Tomando PRODUCTO Y PROVEEDOR



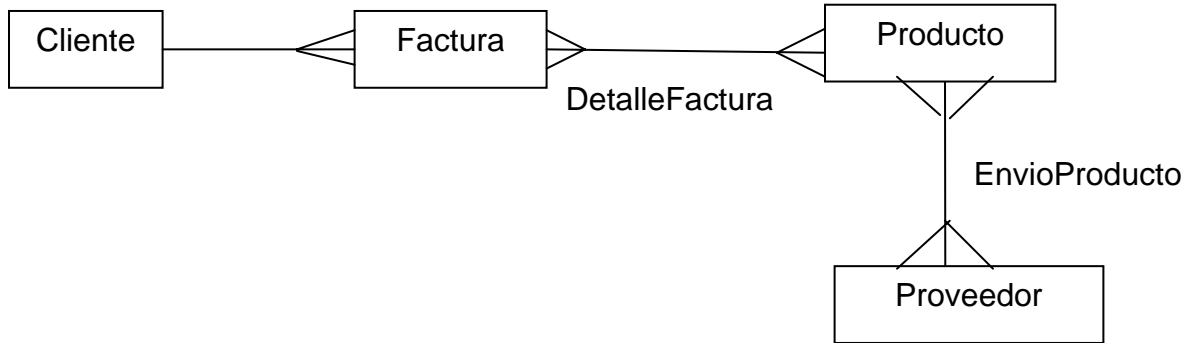
¿Un producto cuantos proveedores puede tener? Varios

¿Un proveedor cuantos productos puede tener? Muchos

¿Un proveedor puede existir sin producto? Sí ya que no tiene obligatoriedad

¿Un producto puede existir sin un proveedor? Sí ya que no tiene obligatoriedad

Entidades en un solo diagrama:



Para romper una relación de muchos a muchos se crea una tercera entidad con el nombre de la relación, esta tendrá una relación de uno a muchos con las entidades originales (el lado de muchos al lado de la nueva entidad y el lado de uno tendrá que ser obligatorio con las entidades originales).

Ejemplo:

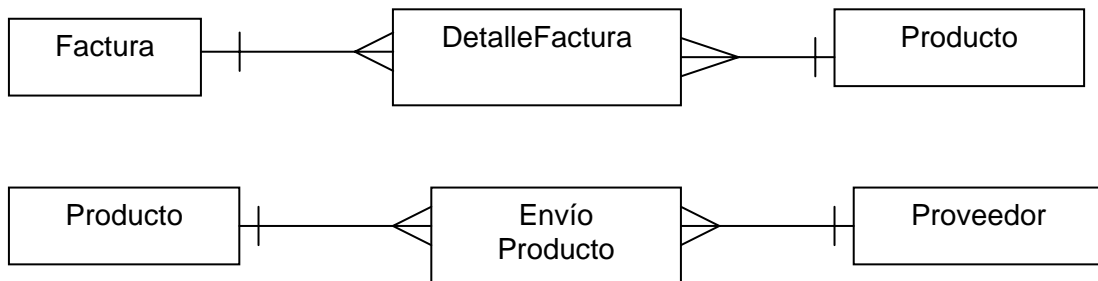
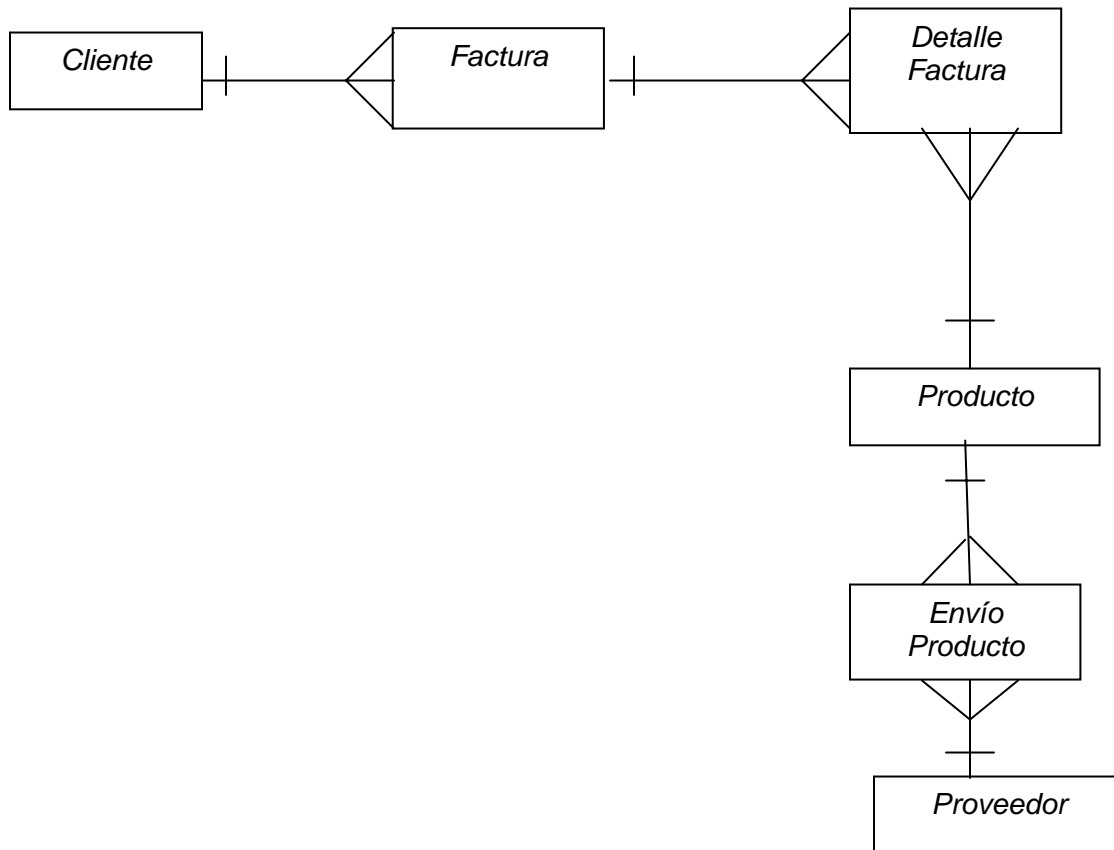


Diagrama completo



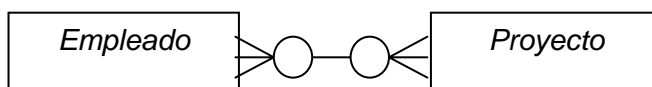
- **Relación Opcional:**

Dos entidades pueden estar relacionadas, pero no en todas sus ocurrencias.

Ejemplo: EMPLEADO y PROYECTO.

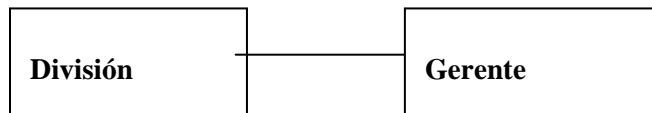
Un empleado puede estar asignado a un proyecto, a algunos proyectos o a ninguno. Un proyecto puede estar autorizado, y no tener empleados asignados

Representación:



- Relación Uno a Uno

Es cuando el atributo identificador de una entidad es el mismo de otra, a este tipo de relación se le llama de uno a uno.

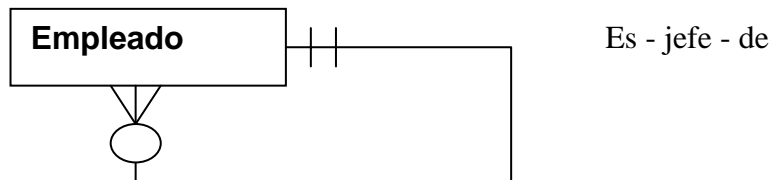


Una División es atendida por un solo Gerente y un Gerente atiende a una solo División.

- Relación Unitaria

Una entidad puede estar relacionada consigo misma.

Ejemplo:



El DER puede ser leído así:

“Un empleado puede ser jefe de cero o más empleados” o “Un empleado siempre reporta a otro empleado.”

3.2.8.3. Comentarios del Modelo Entidad - Relación

El modelo Entidad – Relación apoya el análisis.

- Definir los requerimientos de la empresa.
- escribir la información acerca de las entidades y sus relaciones, requeridas para modelar esos requerimientos.
- Determinar los tipos de transacciones que se busca ejecutar sobre la Base de Datos.

El modelo Entidad - Relación apoya el diseño.

- “Mejora” la habilidad del diseñador de Base de Datos.
- Permite definir los requerimientos de información del mundo real de la manera precisa.
- Define la semántica de las relaciones entre los datos.
- Especifica lineamientos para definir reglas de integridad.

CAPÍTULO IV

Implementación Módulo de Pollo Vivo

4.1. Implementación.

En este capítulo se muestran las ventanas terminadas del módulo de pollo vivo. Estas ventanas se implementaron en el lenguaje de programación PowerBuilder y usando SQL Server de Microsoft como manejador de base de datos.

Este sistema se encuentra trabajando al 100% en todas las plantas de Bachoco.

4.2. ORDEN DE EMBARQUE.

ALCANCE:

Aquí se captura toda la información del pollo vivo desde su salida de la granja hasta que llega a la Planta para ser procesado.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona ahora la opción de "Orden de embarque", donde aparecerá la ventana siguiente:

NUM. DE ORDEN DE EMBARQUE

TIPO DE LINEA DE CRIANZA EN GRANJAS

RESPONSABLE DE CUADRILLA

NOMBRE DE LA GRANJA

Orden de Embarque : 25917 Orden Full Orden Full : 0 Lote prod : 839

Orden de Prod/Pedido : 9175 Turno : 3

Granja: SN CARLOS 5 Linea P.V. : POLLO VIVO

Aves Solicitadas: 6,156 Aves Consignadas: 6,156 Sexo: 3.-Mixto Edad: 6.6 Caseta (s): 5 Cuadrilla: OMAR DE LA CRUZ

Linea Fletera: 0 Remolque: AZTECA Plataforma: UTILITY Dolly: 0

Operador: PROPIO

HORARIOS		Aves Recibidas	PESOS :			
Salida Rastro :	30/05/05 19:00	6,156	GRAIJA	PESADA 1	PESADA 2	
Llegada Granja :	30/05/05 20:00	Hora Ultima Dieta: 29/05/05 05:00	Peso Bruto	36,970.00	28,295.00	28,295.00
Inicio Carga :	30/05/05 20:30	Observaciones:	Tara	22,970.00	14,355.00	14,355.00
Termino Carga :	30/05/05 21:45		Peso Neto	13,950.000	13,940.00	13,940.00
Salida Granja :	30/05/05 23:20		Peso Prom.	2.266	2.264	2.264
Llegada Ppa.:	30/05/05 05:55					

Hora Proceso: 30/05/05 07:16

CHOFER

PESO DE GRANJA

PESO AL LLEGAR A LA PLANTA

PESO PARA DETERMINAR MERMA

Fuente: Aplicación creada por el autor de esta tesis.

La información que se registra de la orden de embarque tiene dos fines:

- 1.- Conocer la cantidad de aves y kilos de pollo que llegan a la planta para efecto de calcular rendimientos del proceso y para conciliar con granjas sus envíos.
- 2.- Conocer los flujos de los transportes de y hacia las granjas para determinar los tiempos perdidos en los traslados y prever de manera más efectiva los mantenimientos preventivos.

4.3. TÉRMINO DE RECEPCIÓN.

ALCANCE:

Una vez que la Orden de Embarque ha sido capturada, y el pollo al que esta amparada se le envía a proceso, se captura en este módulo su entrada al proceso, así como los traspasos que tenga en el andén a otras líneas.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:
PASO 1. Se selecciona la opción de "Pollo vivo" del menú principal,
PASO 2. Se selecciona ahora de "Término de recepción", donde aparecerá la ventana siguiente:

O. Emb.	Plat.	Consecutivo	Tno	Línea de Pollo original	Piezas	Peso Neto	Estado
25917	2863	4	3	POLLO VIVO	6,156	13,940.00	Procesada
25918	9999	30	1	POLLO VIVO	5,472	12,120.00	Procesada
67044	2864	28	2	EVISCERADO LIGERO	6,760	11,820.00	Procesada
67100	4694	27	2	EVISCERADO LIGERO	4,690	8,455.00	Procesada

Línea Producción	Aves Traspasadas		Aves Muertas		Piezas Selección		Tipo de Traspaso	Tiempo Proceso	Supervisor
	Piezas	Kilos	Piezas	Kilos	Piezas	Kilos			
ROSTICERO PESADO	6,120	16,546.30	22	49.82		0.00	Anden	00:47	SANCHEZ ESPINOZA, JL
TOTALES:	6,120	16,546.30	22	49.82	0	0.00			

LINEA A
TRASPASAR

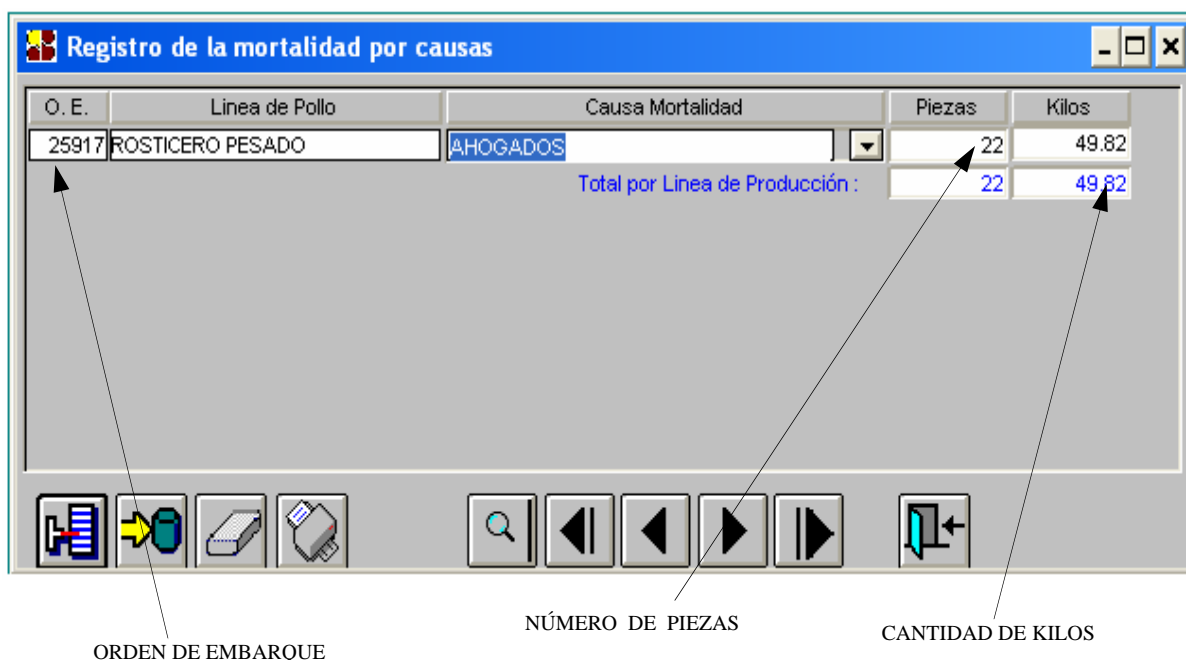
Fuente: Aplicación creada por el autor de esta tesis.

En la pantalla de término de recepción, se puede apreciar que en la primera ventana se encuentran las órdenes de embarque con su línea de origen del pollo vivo; las órdenes de embarque muestran el estado en el que se encuentran (anden, procesada o vendida), al seleccionar una orden de embarque que su estado muestre que aún esta en el anden, el sistema preguntará si desea pasar esta orden de embarque al proceso, de ser afirmativa la respuesta, se podrá capturar el traspaso a la línea de producción con sus piezas y kilos, pudiendo una orden ser traspasada a una o más líneas de producción.

Si usted selecciona una orden de embarque en la que su estado muestra vendida, quiere decir que esta orden fue vendida a un cliente directamente.

Después de capturar los traspasos de línea de producción, se podrá ahora capturar lo que fueron los decomisos, causas de mortalidad, segunda y selección de dicha orden de embarque, esto se realiza con los botones que se encuentran a un lado de la ventana de traspasos.

Al dar un click a cualquiera de estos botones, por ejemplo de mortalidad, se desplegará una pantalla como la siguiente:



Fuente: Aplicación creada por el autor de esta tesis.

En ella, se podrá capturar las aves muertas para la orden de embarque, en piezas, kilos y la causa de su mortalidad de la Orden de Embarque seleccionada.

4.4. POLLO VIVO A VENTA.

ALCANCE:

Algunas Plantas Procesadoras de Aves manejan lo que es la venta de pollo vivo a clientes, del cual también se lleva un registro.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona ahora la opción de "P. Vivo a Venta", donde mostrará la ventana siguiente:

ORDEN DE EMBARQUE TURNO FECHA DE VENTA ESTADO EN QUE SE ENCUENTRA

O. Emb.	Plat.	Consecutivo	Tno	Linea de Pollo original	Piezas	Peso Neto	Estado
25917	2863	0	3	POLLO VIVO	6,156	13,940.00	Procesada
25918	9999	30	1	POLLO VIVO	5,472	12,120.00	Vendida
67044	2864	28	2	EVISCERADO LIGERO	6,760	11,820.00	Procesada
67100	4694	27	2	EVISCERADO LIGERO	4,690	8,455.00	Procesada

Tarjeton: 4565443 Cliente: BOD MERCADO PUBLICO

Remolque: Plataforma: Dolly:

Operator: MONTERROSAS ROSALES PEDRO 1372 36 0

Salida de PPA: 30/05/05 14:13

Datos de la PPA		Datos del Cliente				
Llegada con Cliente:	30/05/05 15:30	Aves Enviadas:	5,472	Aves Recibidas:	5,472	Mortalidad
Inicio Descarga:	30/05/05 15:35	Peso Bruto:	22,350.00	Peso Bruto:	22,350.00	12
Termino Descarga:	30/05/05 16:25	Tara:	10,230.00	Tara:	10,225.00	
Salida del Cliente:	30/05/05 16:35	Peso Neto:	12,120.00	Peso Neto:	12,125.00	Remolachas
Llegada a PPA:	30/05/05 17:45	Peso Prom:	2.215	Peso Prom:	2.216	15

Regresar Orden

Fuente: Aplicación creada por el autor de esta tesis.

Esta ventana permite capturar los tiempos de trayecto entre la planta y el cliente final, los pesos con los que recibe el cliente el producto y la remolacha que recibió para conciliaciones sobre reclamos posteriores.

Para que una orden de embarque pueda ser capturada como venta de pollo vivo, su estado no debe de mostrar procesada, por que de tener el estado procesada quiere decir que el pollo que venia en esa orden de embarque fue pasada al proceso.

Esta captura solo aplica en algunas plantas, y es usada por el personal del área comercial para la venta de pollo vivo a clientes.

4.5. PRENÓMINA.

4.5.1. CAPTURA OTRAS COMISIONES.

ALCANCE:

En este módulo se captura la comisión de los empleados por otras actividades que no están dentro de sus responsabilidades y poder anexarla a su nómina.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Prenómina", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Captura otras comisiones",

PASO 3. Se selecciona y aparecerá la siguiente ventana:

ACTIVIDAD QUE REALIZO
EL EMPLEADO

Empleado	Concepto	Importe
MONTERROSAS ROSALES PE	Movimiento en patio de Maniobras	250.00
RODRIGUEZ NOTARIO IGNAC	Traslado de plataforma a patio de maniobras	175.00

Fuente: Aplicación creada por el autor de esta tesis.

Este módulo sólo captura los datos del empleado y el monto a pagar por cada actividad que se realiza extra a su trabajo normal de carga de pollo de la granja a la Planta Procesadora de Aves.

4.5.2. CALCULO DE PRENÓMINA.

ALCANCE:

Este módulo calcula la nómina de los empleados anexando las comisiones que se capturaron en "Captura Otras Comisiones".

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona ahora la opción de "Prenómina", que mostrará un menú dentro del cuales se encuentra "Calculo de Prenómina", se selecciona y se abrirá la siguiente ventana:

Prenomina choferes Pollo Vivo

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
Reporte del pago por salida de los choferes de pollo vivo

Fecha : 11/Oct/06
Página: 1/1
Versión:

Fecha inicial : Fecha final :

Centro de Costos : Tipo de contrato : **Eventual**

Fecha inicial : 30/05/05
Fecha final : 30/05/05

Seleccione el Centro de Costos

0	PERSONAL BAJA
402030	TRAFICO BODEGAS
403030	TRAFICO POLLO VIVO
404030	EMBARQUES

Tipo de Contrato

Semanal

Quincenal

Eventual

Fuente: Aplicación creada por el autor de esta tesis

Para generar el reporte se siguen los siguientes pasos:

- Se selecciona el rango de las fechas las cuales quiere el cálculo de la Prenómina.
- Ahora se selecciona el centro de costos y el tipo de contrato que tiene el empleado.

Ya que se han introducido los datos en el módulo anterior se procede a calcular nómina.

4.6. DIESEL.

4.6.1. CAPTURA DE DIESEL.

ALCANCE:

En este módulo se captura el consumo de diesel de las unidades de acuerdo al kilometraje recorrido.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Catálogos" del menú principal,

PASO 2. Se selecciona el módulo de "Diesel", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Captura de Diesel",

PASO 3. Se selecciona y aparecerá la siguiente ventana:

Fecha Carga	Operador	Kilometraje	Litros Diesel	Km recorridos desde ult. cap.	Rendimiento	Total Km recorridos	Total Diesel cargado
22/05/2005	32	125,873	250.00		Km/lts	0	250
24/05/2006	811	126,834	264.00	961	3.64 Km/lts	961	514
26/05/2006	841	127,865	275.00	1,031	3.75 Km/lts	1,992	789
30/05/2006	32	128,863	267.00	998	3.74 Km/lts	2,990	1,056

Fuente: Aplicación creada por el autor de esta tesis.

En esta ventana se capturan los datos de las unidades, en base al tipo, la clave y el centro de costos de la unidad, llevando un historial de los consumos de cada unidad.

4.6.2. REPORTE DE RENDIMIENTOS DE UNIDADES.

ALCANCE:

Este módulo reporta lo que rindieron las unidades de acuerdo a los viajes realizados y la cantidad de diesel que consumieron.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Catálogos" del menú principal,

PASO 2. Se selecciona el módulo de "Pollo Vivo", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Reporte de Rendimientos de Unidades",

PASO 3. Se selecciona y abrirá la siguiente ventana:

Reporte de consumo de Diesel

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves

Reporte de consumo de combustible

Fecha inicial: 01/05/05 Fecha final: 30/05/05

Centro de Costos: 403050 TRAFICO

Tipo de Unidad: 1 TRACTOCAMION

Unidad	Kilometros Recorridos	Litros Diesel	Rendimiento Unidad	Rendimiento estandar
1372		250.00 lts	Km / lts	0.00 Km / lts

Centro de Costos

9999	DESCONOCIDO
403050	TRAFICO
404050	EMBARQUES POLLO VIVO

Fecha inicial: 01/05/05

Fecha final: 30/05/05

Fuente: Aplicación creada por el autor de esta tesis.

Para crear el reporte es muy sencilla y se explica a continuación:

- Se selecciona el centro de costos de la(s) unidad(es).
- Se selecciona el rango de fechas en que se desea el reporte.
- Se presiona el botón para reportar rendimientos.

4.7. REPORTE.

En la Planta Procesadora de Aves toda la información que se introduce al SIPPA es utilizada para la toma de decisiones, es por esto que el sistema tiene un módulo de reportes como son:

4.7.1. POLLO DE GRANJAS.

ALCANCE:

La información que se genera en las Órdenes de Embarque es utilizada para saber todo lo relacionado a los pollos que mandaron de cada granja y poder sacar su respectivo reporte.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Pollo de Granjas",

PASO 3. Se selecciona y aparecerá la siguiente ventana:

Selección de la granja inicial		Selección de la granja final	
13801	PEDREGAL 1	13805	GERTRUDIS 2
13802	PEDREGAL 2	13806	GERTRUDIS 3
13803	RETIRO 3	13807	GERTRUDIS 4
13804	GERTRUDIS 1	13808	SN CARLOS 1
13805	GERTRUDIS 2	13809	SAN CARLOS 2
13806	GERTRUDIS 3	13810	SN CARLOS 3
13807	GERTRUDIS 4	13812	SN CARLOS 5

Todas las granjas

Fecha inicial: 30/05/05 Fecha final: 30/05/05

Fuente: Aplicación creada por el autor de esta tesis.

Después que se seleccionaron las granjas y el rango de las fechas que se necesite el reporte, se selecciona el botón para introducir los datos.

A continuación aparecerá en la pantalla la siguiente ventana:

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte del pollo vivo recibido de granjas

Fecha inicial :30/05/2005 **Fecha final :30/05/2005**

Orden	Linea de Pollo	Granja	Cta.	E	S	Cuad.	Aves consig.	Hora Dieta	Peso Llegada		
									Bruto	Tara	Ne
25917	POLLO VIVO	13812 SN CARLOS 5	5	6.6	X	1	6,156	05:00	28,295	14,355	1
67137	POLLO VIVO	13812 SN CARLOS 5	7	6.6	X	1	1,620	05:00	22,136	18,830	
67134	POLLO VIVO	13812 SN CARLOS 5	6	6.6	X	1	3,852	05:00	26,694	18,830	
67135	POLLO VIVO	13812 SN CARLOS 5	6	6.6	X	1	648	05:00	21,208	19,805	
67136	POLLO VIVO	13812 SN CARLOS 5	5	6.3	X	1	5,464	05:00	31,622	19,805	1
25918	POLLO VIVO	13812 SN CARLOS 5	5	6.6	X	1	5,472	05:00	25,585	13,465	1
Total por Granja :							23,202		155,540	105,090	5
Totales :							23,202		155,540	105,090	5

Fuente: Aplicación creada por el autor de esta tesis.

4.7.2. POLLO VIVO A VENTA.

ALCANCE:

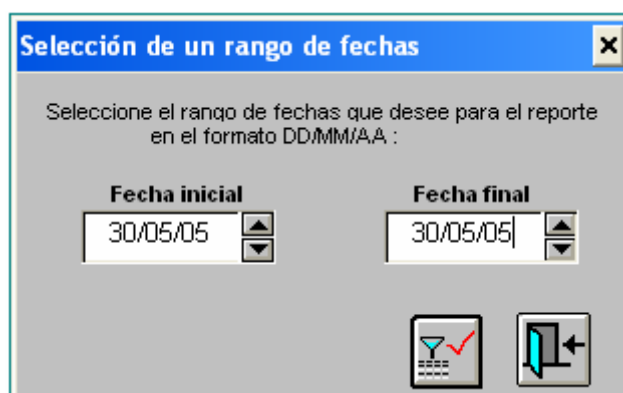
Este módulo nos genera un reporte del concentrado de la cantidad de pollos que fueron vendidos basados en los datos capturados en el módulo de "Pollo vivo a venta".

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona ahora el módulo de "Reportes", donde se desplegarán los módulos que este contiene, y se encuentra "Pollo vivo a venta", una vez seleccionado alguno se mostrará la siguiente ventana:



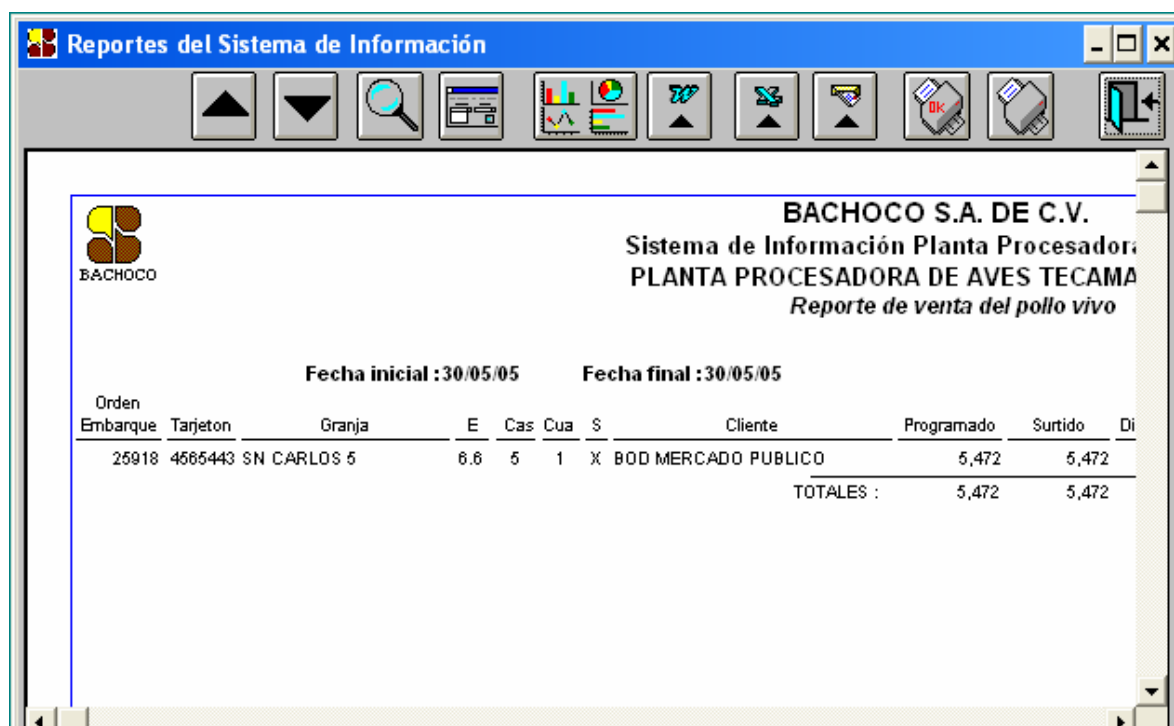
The screenshot shows a dialog box with a blue title bar that reads "Selección de un rango de fechas". Below the title bar, the text says "Seleccione el rango de fechas que desee para el reporte en el formato DD/MM/AA :". There are two input fields, one labeled "Fecha inicial" and one labeled "Fecha final", both containing the date "30/05/05". Each field has small up and down arrow buttons to its right. At the bottom of the dialog, there are two icons: the first is a bar chart with a red checkmark, and the second is a blue arrow pointing left.

Fuente: Aplicación creada por el autor de esta tesis.

La forma de generar el reporte es muy sencilla y se explica a continuación:

- Se selecciona el rango de fechas de donde abarcará el reporte, recordando que la fecha inicial debe de ser menor o igual que la fecha final.
- Por último se presiona el botón de generar el reporte.

El reporte generado será el siguiente:



BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora
PLANTA PROCESADORA DE AVES TECAMA
Reporte de venta del pollo vivo

Fecha inicial : 30/05/05 Fecha final : 30/05/05

Orden	Embarque	Tarjeton	Granja	E	Cas	Cua	S	Cliente	Programado	Surtido	Di
25918	4565443	SN CARLOS 5		6.6	5	1	X	BOD MERCADO PUBLICO	5,472	5,472	
TOTALES :									5,472	5,472	

Fuente: Aplicación creada por el autor de esta tesis.

4.7.3. PRODUCTO A PROCESO.

ALCANCE:

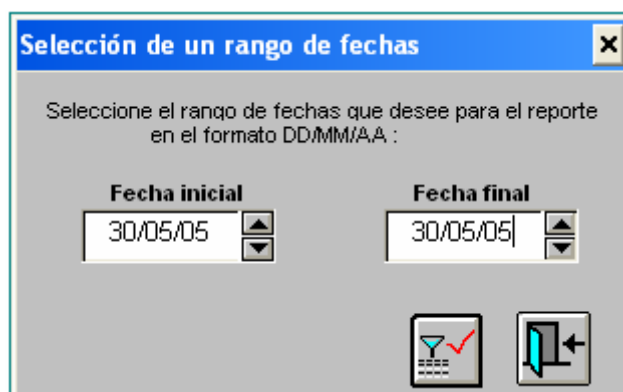
Este módulo nos genera un reporte del concentrado de la cantidad de pollos recibidos de las granjas, con los datos de los pollos vivos como las aves muertas por granja.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona ahora el módulo de "Reportes", donde se desplegarán los módulos que este contiene, y se encuentra "Producto a proceso", una vez seleccionado alguno se mostrará la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

La forma de generar el reporte es muy sencilla y se explica a continuación:

- Se selecciona el rango de fechas de donde abarcará el reporte, recordando que la fecha inicial debe de ser menor o igual que la fecha final.
- Por último se presiona el botón de generar el reporte.

El reporte generado será el siguiente:

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Av
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte del pollo vivo que entra a proceso

Fecha inicial : 30/05/2005 Fecha Final : 30/05/2005

Fecha: 30/05/05

Orden Embar.	L. Crianza en Granja	Aves	Consignadas Kilos	Aves Planta	Kilos 1er Pesada	Merma Transito	Kilos 2da Pesada	Merma Anden
25917	POLLO VIVO	6,156	13,950.00	6,156	13,940.00	10.0	13,940.00	0.0
67134	POLLO VIVO	3,852	7,939.00	3,852	7,864.00	75.0	7,864.00	0.0
67135	POLLO VIVO	648	1,416.00	648	1,403.00	13.0	1,403.00	0.0
67136	POLLO VIVO	5,454	11,924.00	5,454	11,817.00	107.0	11,817.00	0.0
67137	POLLO VIVO	1,620	3,341.00	1,620	3,306.00	35.0	3,306.00	0.0
67154	POLLO VIVO	4,014	0.00	4,014	7,815.00	-7,815.0	7,815.00	0.0
67158	POLLO VIVO	3,564	0.00	3,564	9,080.00	-9,080.0	9,080.00	0.0

Fuente: Aplicación creada por el autor de esta tesis.

4.7.4. POLLO EN ESPERA.

ALCANCE:

El pollo al llegar a la Planta no siempre pasa directamente a la línea de producción y se va a Andén de espera, donde este módulo se encarga de sacar el reporte.

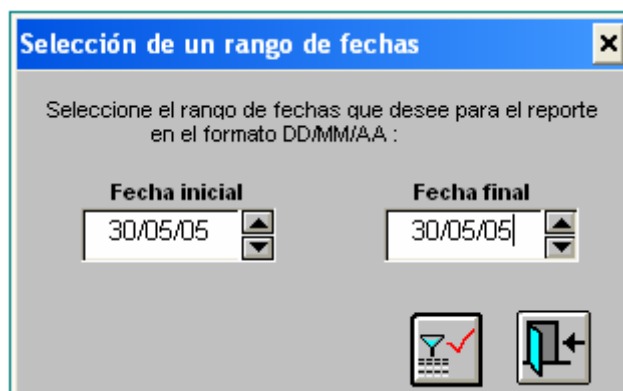
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reporte", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Pollo en espera",

PASO 3. Se selecciona y aparecerá la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para crear el reporte de realizan los siguientes pasos:

- Se seleccionan el rango de fechas en que se desea el reporte.
- Se presiona el botón de generar reporte.

El reporte generado será el siguiente:

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte del pollo vivo mantenido en anden

Fecha inicial : 30/05/2005 Fecha final : 30/05/2005

Orden Embarque	Producto	Aves Anden		Peso Llegada		Neto	Prom
		Aves Anden	Granja	Bruto	Tara		
67118	EVISCERADO PESADO	6,084	3442	25,875	14,270	11,605	.000
67119	EVISCERADO PESADO	324	3442	14,056	13,400	656	.000
67100	EVISCERADO LIGERO	4,690	3427	27,295	18,840	8,455	.000
67044	EVISCERADO LIGERO	6,760	3427	31,635	19,815	11,820	.000
Totales :				98,861	66,325	32,536	.000

Fuente: Aplicación creada por el autor de esta tesis.

4.7.5. REGISTRO DIARIO DE VIAJES.

ALCANCE:

Diariamente se llevan a cabo viajes de pollo de las diferentes granjas a la Planta Procesadora de Aves, el sistema registra toda esta información en las Órdenes de Embarque y las genera en el siguiente reporte.

PROCEDIMIENTO:

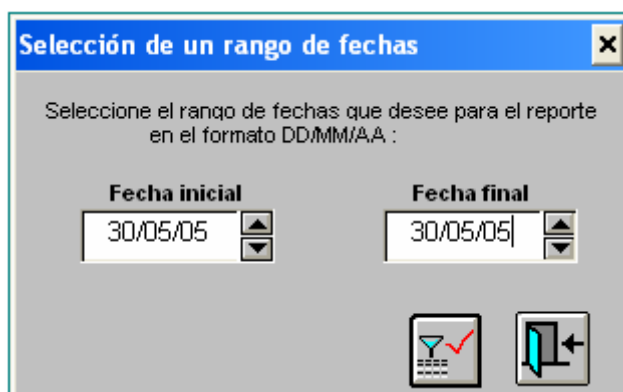
La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus

diferentes opciones, dentro de las cuales se encuentra "Registro diario de viajes",

PASO 3. Se selecciona y aparecerá la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte.
- Presionar botón para generar el reporte.

Después de seleccionar los rangos de las fechas para el reporte, aparecerá la siguiente ventana con la siguiente información.

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora
Registro diario de viajes de pollo vivo

Fecha inicial : 30/05/2005 Fecha final : 30/05/2005

Tipo de Producto : ROSTICERO LIGERO

Orden Emb.	Plat.	Aprov.	Granja	Caseta	Sexo	Prod.	Aves	P. Granja	Peso Proc.	P.P. Granja	P.P. Proc.	% Mo
67121	9999		3427	2	H	5	1,949	3,515.00	3,498.84	1.80	1.80	0.26
67158	4561	61%	3429	3	M	2	1,316	9,080.00	3,510.00	6.90	2.67	0.00
67158	4561	61%	3429	3	M	4	2,226	9,080.00	5,513.95	4.08	2.48	0.41
67151	4693	70%	3442	1	H	4	5,160	10,514.00	10,457.30	2.04	2.03	0.29
25917	2863	139%	13812	5	X	4	6,120	13,940.00	16,546.30	2.28	2.70	0.36
67154	4483	52%	3442	9	M	5	3,989	7,815.00	7,766.34	1.96	1.95	0.40
67164	2919	131%	3442	4	H	5	6,762	13,095.00	13,021.90	1.94	1.93	0.31

Fuente: Aplicación creada por el autor de esta tesis

4.7.6. RESULTADOS POR SUPERVISOR.

ALCANCE:

El sistema reporta cuantas aves embarcaron por supervisor de acuerdo a las órdenes de embarque y se realiza de la siguiente manera.

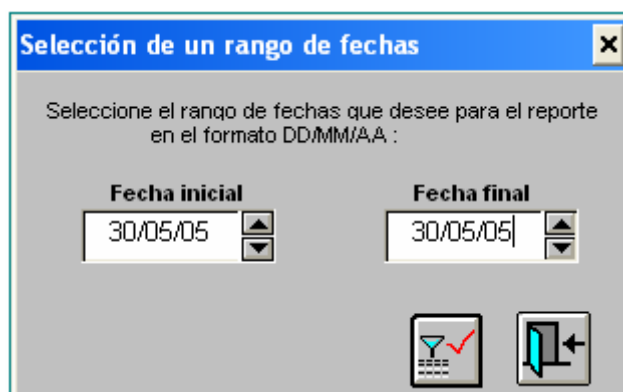
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reporte", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Resultado por supervisor",

PASO 3. Se selecciona y mostrará la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte.
- Presionar botón para realizar reporte.

Después de seleccionar el rango de fechas para el reporte, aparecerá la siguiente ventana con la siguiente información.

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte de resultados por supervisor

Fecha : 31/May/0
 Página: 1/1
 Versión: 4.05.03

Fecha inicial : 30/05/2005 Fecha final : 30/05/2005
Supervisor : CARLOS SANCHEZ

Orden	Granja	Embarque	Cuadrilla	Sexo	Cargadas		Recibidas		Mortalidad	
					Aves	Kilos	Aves	Kilos	Aves	Kilos
3442	67151	4	H	5,188	10,514.00	5,188	10,514.00	15	30.40	
3442	67154	4	M	4,014	7,815.00	4,014	7,815.00	16	31.15	
Totales de la Cuadrilla :					9,202	18,329.00	9,202	18,329.00	31	61.55
Totales por Supervisor :					9,202	18,329.00	9,202	18,329.00	31	61.55

Supervisor : FILEMON LOPEZ

Orden	Granja	Embarque	Cuadrilla	Sexo	Cargadas		Recibidas		Mortalidad	
					Aves	Kilos	Aves	Kilos	Aves	Kilos

Fuente: Aplicación creada por el autor de esta tesis.

4.7.7. CONCILIACION POLLO VIVO POR EMBARQUE.

ALCANCE:

Este módulo nos muestra un reporte detallado del pollo vivo que llega a la planta de acuerdo a lo que se registró en la orden de embarque. Y nos ayuda a corroborar las cantidades enviadas por las que se recibieron, viene por orden de embarque, granja, caseta y sexo.

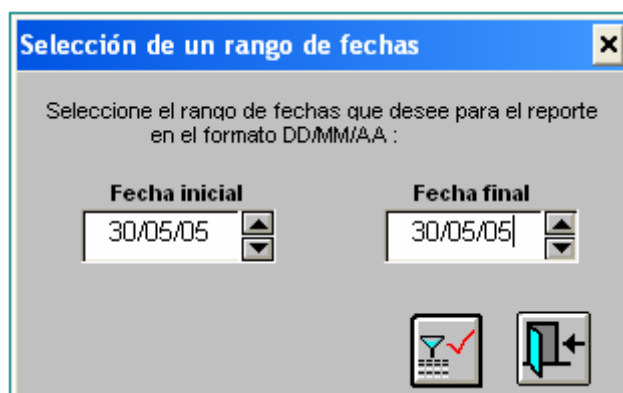
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reporte", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Conciliación pollo vivo por embarque",

PASO 3. Se selecciona y aparecerá la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte.
- Presionar botón para generar reporte.

Después de seleccionar el rango de fechas en que se realizará el reporte aparecerá la ventana donde se presenta la siguiente información:

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte de Conciliación de Pollo Vivo por Orden de Embarque

Fecha inicial : 30/May/05 Fecha final : 30/May/05

Fecha : 30/05/05

Línea de Pollo : 2 POLLO VIVO

Orden Embarque	Granja	Caseta	Sexo	Aves				Aves Muerl	
				Consignadas	Aves	Kilos	Peso Prom.	Aves	Kilos
67158	3429	3	M	3,564	3,564	9,080.00	2.55	14	37
Total por Granja :				3,564	3,564	9,080.00	2.55	14	37
67154	3442	9	M	4,014	4,014	7,815.00	1.95	16	31
Total por Granja :				4,014	4,014	7,815.00	1.95	16	31
25917	13812	5	X	6,156	6,156	13,940.00	2.26	22	48
67137	13812	7	X	1,620	1,620	3,306.00	2.04	7	14

Fuente: Aplicación creada por el autor de esta tesis.

4.7.8. CONCILIACIÓN POLLO VIVO POR GRANJA.

ALCANCE:

Este reporte nos muestra en forma detallada lo que se mandó por granja y nos dice el nombre de la granja, el número de aves, kilos y peso promedio de las aves tanto recibidas como muertas y nos sirve para corroborar la cantidad de pollos que enviaron de la granja y las que llegaron a la planta.

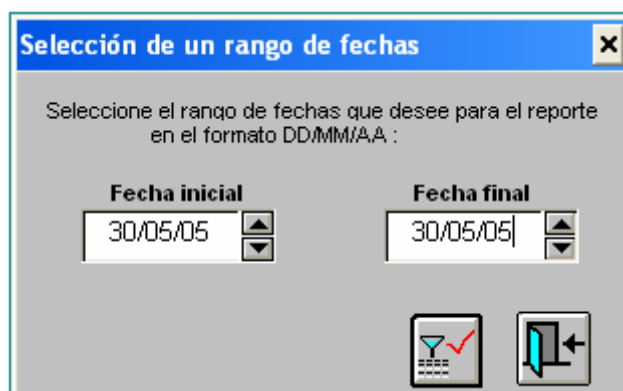
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reporte", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Conciliación pollo vivo por granja",

PASO 3. Se selecciona y mostrará la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte
- Presionar botón para realizar reporte.

Después de seleccionar el rango de fechas que se desee el reporte aparecerá en la pantalla la siguiente ventana donde se presenta la siguiente información.

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte de Conciliación de Pollo Vivo por Granja

Fecha inicial : 30/May/05 Fecha final : 30/May/05

Fecha : 30/05/05

Linea de Pollo : 2 POLLO VIVO

Granja	Aves		Aves Recibidas			Aves Muert	
	Consignadas	Aves	Kilos	Peso Prom.	Aves	Kilos	
3 STA. CLARA 1	3,564	3,564	9,080.00	2.55	14	37	
9 SAN PEDRO 4	4,014	4,014	7,815.00	1.95	16	31	
5 SN CARLOS 5	17,730	17,730	38,330.00	2.16	338	731	
Total por Linea de Pollo :	25,308	25,308	55,225.00	2.18	368	800	

Linea de Pollo : 3 EVISCERADO PESADO

Fuente: Aplicación creada por el autor de esta tesis.

4.7.9. TRASPASOS EN LÍNEA.

ALCANCE:

En la planta procesadora de aves es muy común los traspasos y este módulo saca en forma detallada los traspasos que hubo en un lapso de tiempo determinado.

Después de realizar la captura de datos en el módulo de término de recepción, se procede a realizar el reporte de acuerdo al siguiente procedimiento.

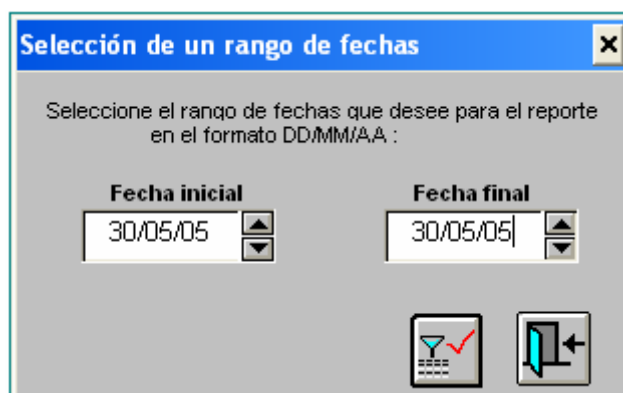
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" de menú principal

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Traspasos en la línea",

PASO 3. Se selecciona y mostrará la siguiente ventana:

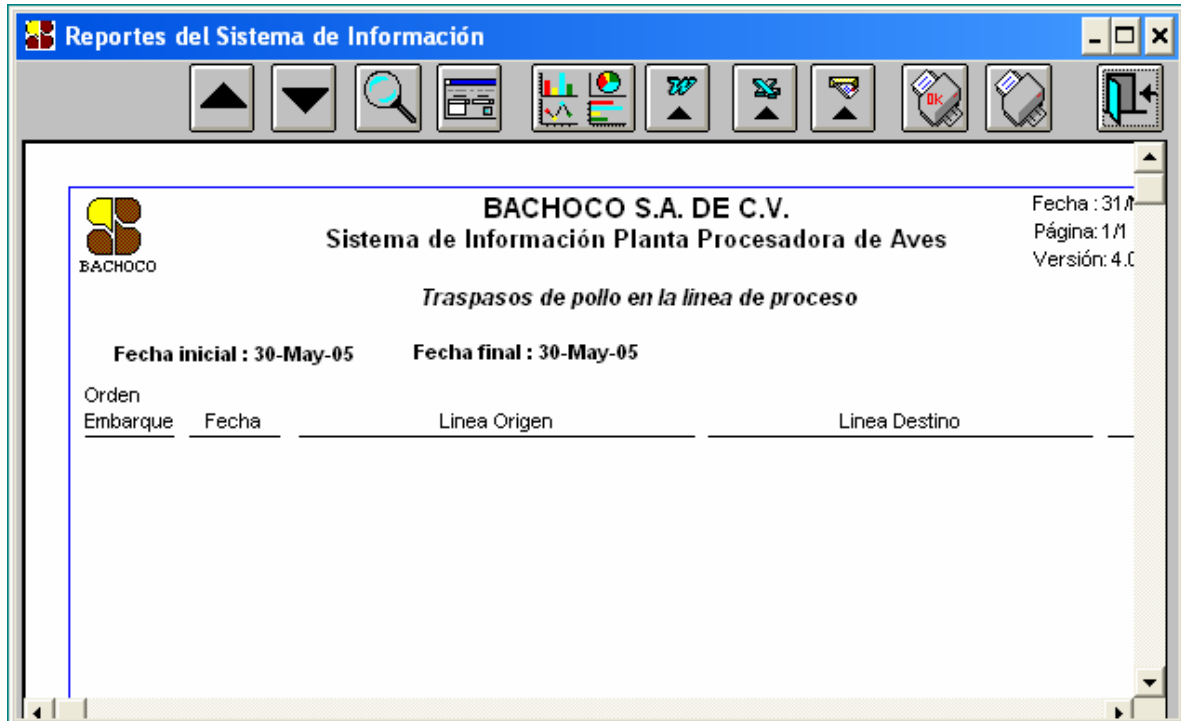


Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte
- Presionar botón para realizar reporte.

Después de seleccionar el rango de fechas en que se desee el reporte aparecerá en la pantalla la ventana con la información siguiente:



Fuente: Aplicación creada por el autor de esta tesis.

4.7.10. POLLO DECOMISADO.

ALCANCE:

Después de haber capturado la información en el módulo término de recepción se procederá a realizar el reporte de pollo decomisado mediante el siguiente procedimiento.

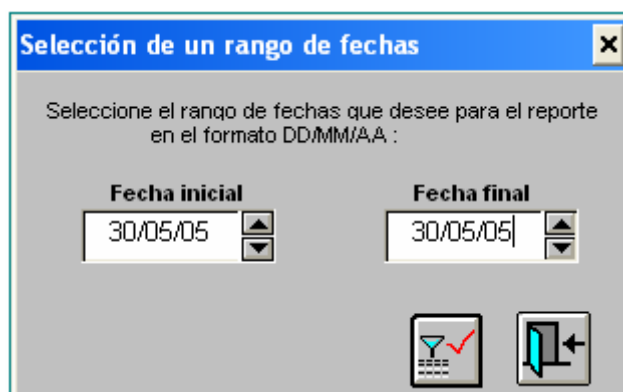
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Pollo Decomisado",

PASO 3. Se selecciona y mostrará la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte.
- Se presiona botón para realizar reporte.

Después de haber seleccionado el rango de fechas de las que se desea el reporte aparecerá en la pantalla la siguiente información:

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte del decomiso por orden de embarque

Fecha inicial : 30/05/05 Fecha final : 30/05/05

Orden Embarque	Fecha	Granja	Linea Produccion	Causa del decomiso
67158	30/05/05	STA. CLARA 1	MERCADO PUBLICO EVIS.	DECOMISO EN ANI
Total por Li				
67121	30/05/05	ANIMAS	ROSTICERO LIGERO	DECOMISO EN ANI
67132	30/05/05	ANIMAS	ROSTICERO LIGERO	DECOMISO EN ANI
67132	30/05/05	ANIMAS	ROSTICERO LIGERO	DECOMISO POR TI
67133	30/05/05	ANIMAS	ROSTICERO LIGERO	DECOMISO EN ANI
67133	30/05/05	ANIMAS	ROSTICERO LIGERO	DECOMISO POR TI

Fuente: Aplicación creada por el autor de esta tesis.

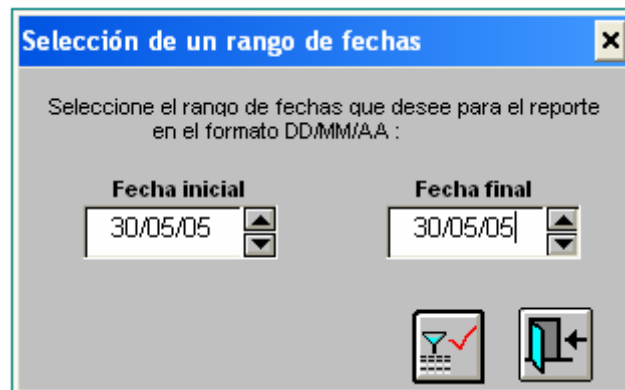
4.7.11. POLLO MUERTO POR CAUSA.

ALCANCE:

Este reporte muestra en detalle la cantidad de los pollos muertos por diferentes causas de acuerdo a la orden de embarque.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:
PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,
PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Pollo muerto por causa",
PASO 3. Se selecciona y mostrará la siguiente ventana:



Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona el rango de fechas en que se desea el reporte
- Se presiona botón para realizar reporte.

Después de haber seleccionado el rango de fechas que se desee el reporte aparecerá en la pantalla la siguiente información.

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PLANTA PROCESADORA DE AVES TECAMACHALCO
Reporte de mortalidad por orden de embarque

Fecha inicial : 30/05/05 Fecha final : 30/05/05

Orden Embarque	Fecha	Granja	Linea Produccion	Causa de
67121	30/05/05	ANIMAS	ROSTICERO LIGERO	AHOGADOS
67132	30/05/05	ANIMAS	ROSTICERO LIGERO	AHOGADOS
67133	30/05/05	ANIMAS	ROSTICERO LIGERO	AHOGADOS
67153	30/05/05	ANIMAS	ROSTICERO LIGERO	AHOGADOS
				Total por Li
67154	30/05/05	SAN PEDRO 4	ROSTICERO LIGERO	AHOGADOS
				Total por Li

Fuente: Aplicación creada por el autor de esta tesis.

4.7.12. VIAJES REALIZADOS POR TIPO DE TRANSPORTE Y ZONA.

ALCANCE:

El sistema reporta los viajes realizados por transporte y zona según la orden de embarque.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:
PASO 1. Se selecciona el módulo de "Pollo Vivo",
PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Viajes realizados por tipo de transporte y zona",
PASO 3. Se selecciona y aparecerá la siguiente ventana:

Estadística de Viaje de Pollo Vivo

BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
Viajes realizados por Tipo de Transporte y Zona

Fecha Inicial: 30/05/05 Fecha Final: 30/05/05

Linea de Pollo : EVISCERADO LIGERO

Tipo Cam.	ZONA "A"		ZONA "B"		ZONA "C"		ZONA "D"		ZONA "E"	
	Sencillos	Full	Sencillos	Full	Sencillos	Full	Sencillos	Full	Sencillos	Full
1	0	1	0	0	0	0	0	0	0	0
	0	11,450	0	0	0	0	0	0	0	0

	Viajes Sencillos		Viajes Full		Total Viajes	
	Viajes	Aves transportadas	Viajes	Aves transportadas	Viajes	Aves transportadas
Zona A :	0	0	1	11,450	1	11,450
Zona B :	0	0	0	0	0	0

Lineas de Pollo

- 1 MERCADO PUBLICO
- 2 POLLO VIVO
- 3 EVISCERADO PESADO
- 4 EVISCERADO LIGERO

Fecha inicial: 30/05/05 Fecha final: 30/05/05

Todas las Lineas

Fuente: Aplicación creada por el autor de esta tesis.

Para generar un reporte se realiza el siguiente procedimiento:

- Se selecciona la línea de pollo para el reporte que desea, si desea un reporte de todas las líneas se selecciona "Todas las líneas".
- Se selecciona el rango de fechas en que se desea el reporte
- Presionar botón para realizar reporte.

4.7.13. FLUJO DE TRASPORTE.

ALCANCE:

Este módulo reporta en forma detallada el flujo de unidades de pollo vivo.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Flujo del transporte",

PASO 3. Se selecciona y aparecerá la siguiente ventana:

Orden Embarque	Linea Fletera	Trabajador	Tipo Maje	Granja	Unidad	Plat.	Dolly	Cor
25917		PROPIO PROPIO PROPIO	S	13812	0	2863	0	6
25918		PROPIO PROPIO PROPIO	S	13812	0	9999	0	5
67118		PROPIO PROPIO PROPIO	S	3442	0	2861	0	6
67119		PROPIO PROPIO PROPIO	S	3442	0	4693	0	
67121		PROPIO PROPIO PROPIO	S	3427	0	9999	0	1
67132		PROPIO PROPIO PROPIO	S	3427	0	4693	0	6
67134		PROPIO PROPIO PROPIO	S	13812	0	4560	0	3
67135		PROPIO PROPIO PROPIO	S	13812	0	2864	0	
67136		PROPIO PROPIO PROPIO	S	13812	0	2864	0	5
67137		PROPIO PROPIO PROPIO	S	13812	0	4560	0	1

Fuente: Aplicación creada por el autor de esta tesis.

La forma para hacer un reporte es como a continuación se describe:

- Se selecciona un rango de fechas en que se desea el reporte.
- Se presiona el botón para hacer introducir las fechas.
- Si se desea exportar los datos del reporte a una hoja de cálculo de Excel de Micros se presiona el primer botón.
- Se presiona botón para generar reporte.

4.7.14. KM. POR DOLLY.

ALCANCE:

De acuerdo a la información de la Orden de Embarque se obtiene el recorrido de kilometraje por Dolly, este módulo crea un reporte proporcionando esta información.

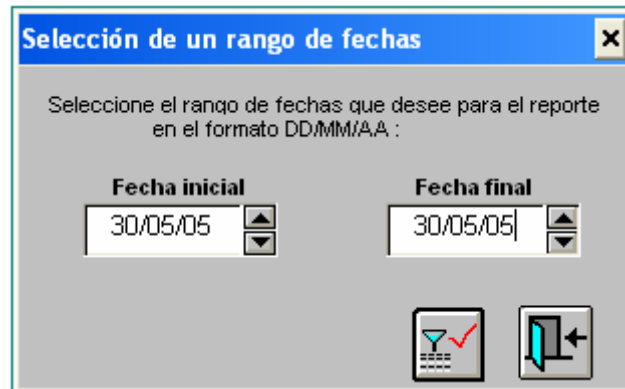
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Km. por Dolly",

PASO 3. Se selecciona y aparecerá la siguiente ventana:

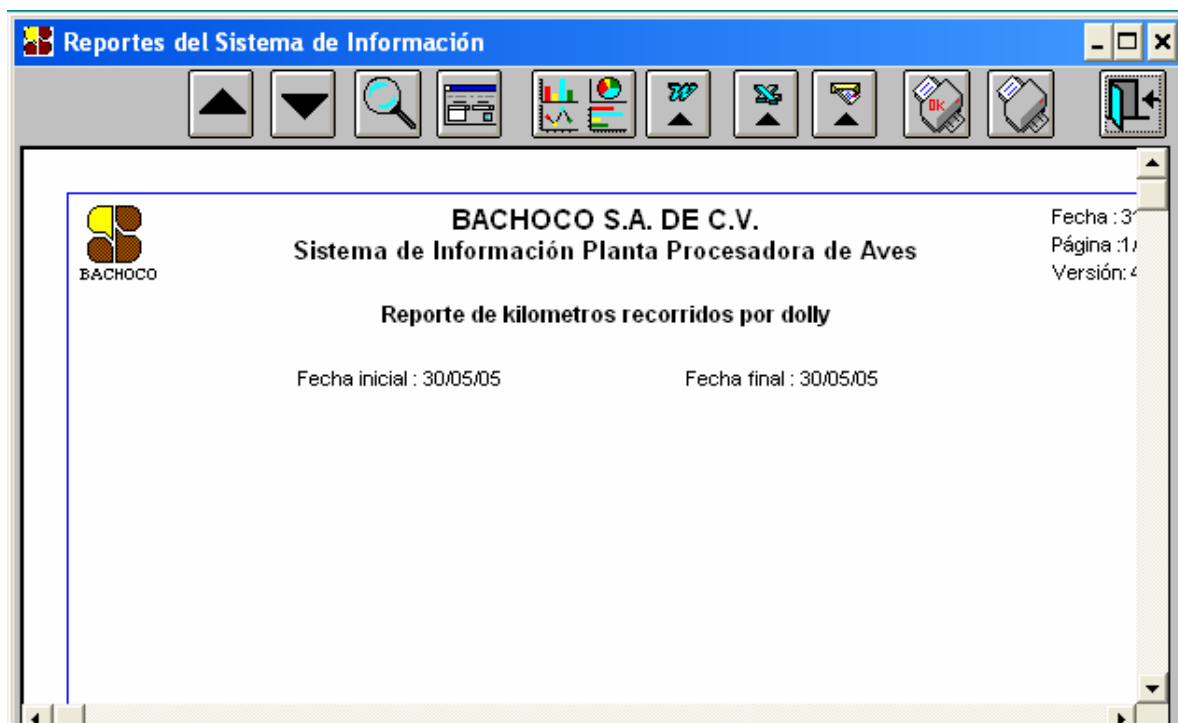


Fuente: Aplicación creada por el autor de esta tesis.

La forma para hacer un reporte es como a continuación se describe:

- Se selecciona un rango de fechas en que se desea el reporte.
- Se presiona el botón de generar reporte.

Después de seleccionar el rango de fechas en que se desee el reporte aparecerá en la pantalla la ventana siguiente:



Fuente: Aplicación creada por el autor de esta tesis.

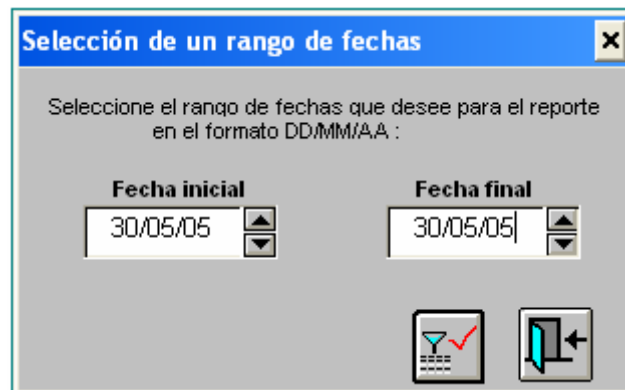
4.7.15. KM. POR PLATAFORMA.

ALCANCE:

De acuerdo a la Orden de Embarque éste módulo realiza un reporte del kilometraje recorrido por cada plataforma.

PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:
PASO 1. Se selecciona el módulo de "Pollo Vivo" del menú principal,
PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Km. por plataforma",
PASO 3. Se selecciona y aparecerá la siguiente ventana:

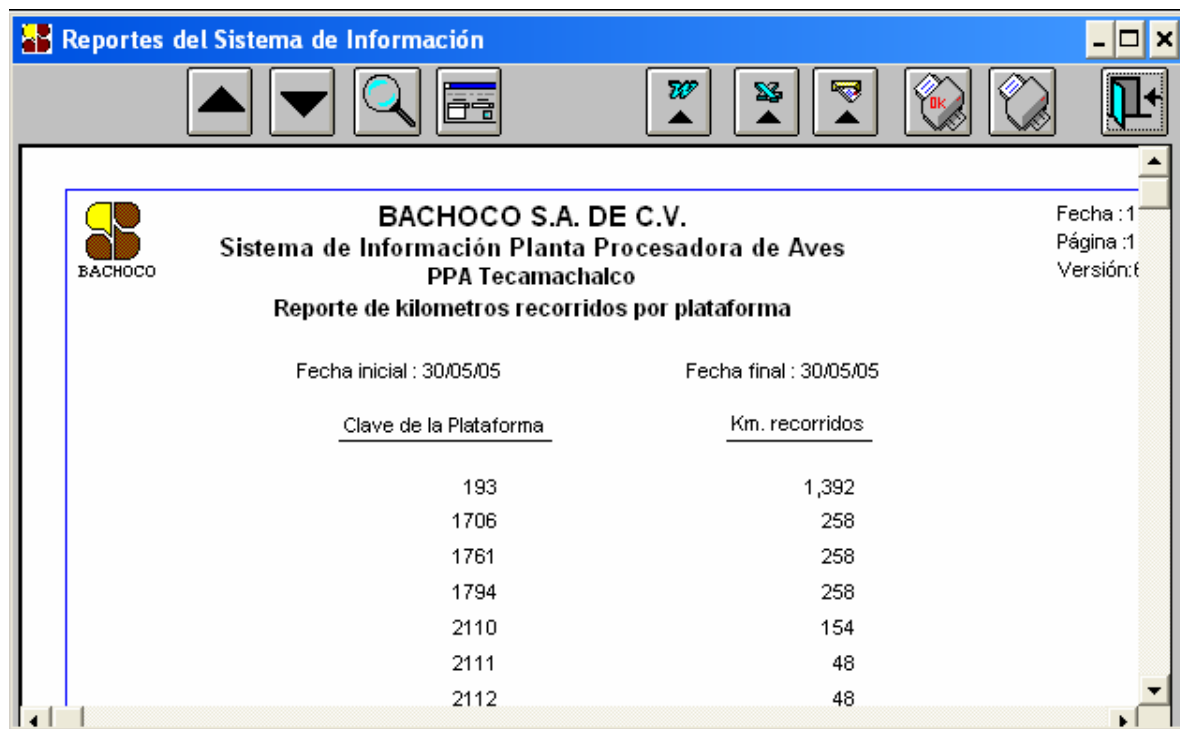


Fuente: Aplicación creada por el autor de esta tesis.

La forma para hacer un reporte es como a continuación se describe:

- Se selecciona un rango de fechas en que se desea el reporte.
- Se presiona el botón de generar reporte.

Después de seleccionar el rango de fechas para el reporte, aparecerá la ventana con la siguiente información.



BACHOCO S.A. DE C.V.
Sistema de Información Planta Procesadora de Aves
PPA Tecamachalco

Reporte de kilometros recorridos por plataforma

Fecha inicial : 30/05/05 Fecha final : 30/05/05

<u>Clave de la Plataforma</u>	<u>Km. recorridos</u>
193	1,392
1706	258
1761	258
1794	258
2110	154
2111	48
2112	48

Fecha : 1
Página : 1
Versión: t

Fuente: Aplicación creada por el autor de esta tesis.

4.7.16. AVES POR SUPERVISOR.

ALCANCE:

Este reporte nos muestra en forma detallada las aves cargadas y muertas por cada supervisor.

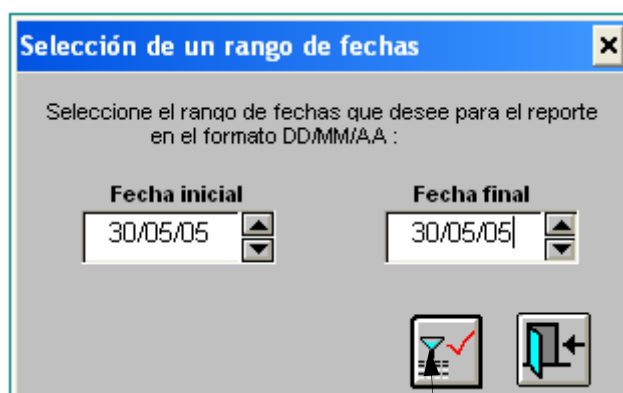
PROCEDIMIENTO:

La forma de acceder a este módulo desde el menú principal, es la siguiente:

PASO 1. Se selecciona el módulo de "Pollo Vivo",

PASO 2. Se selecciona el módulo de "Reportes", el cual se desplegarán sus diferentes opciones, dentro de las cuales se encuentra "Aves por supervisor",

PASO 3. Se selecciona y aparecerá la siguiente ventana:



BOTON PARA GENERAR
REPORTE

Fuente: Aplicación creada por el autor de esta tesis.

La forma para hacer un reporte es como a continuación se describe:

- Se selecciona un rango de fechas en que se desea el reporte.
- Se presiona el botón de generar reporte.

Después de seleccionar el rango de fechas para el reporte, aparecerá en la pantalla una ventana con la siguiente información.

BACHOCO S.A. DE C.V.				
Sistema de Información Planta Procesadora de Aves				
Reporte de aves cargadas por supervisor				
Fecha inicial: 30/05/05		Fecha final: 30/05/05		
30/5/2005				
Supervisor de Cuadrilla	Aves Cargadas	Aves Muertas	Total Aves Cargadas	Total Aves Muertas
CARLOS SANCHEZ	9,149	31	9,149	31
FILEMON LOPEZ	53,350	105	53,350	105
OMAR DE LA CRUZ	20,873	352	20,873	352
ROMULO JIMENEZ	39,577	109	39,577	109
Grand Total	122,949	597	122,949	597

Fuente: Aplicación creada por el autor de esta tesis.

CONCLUSIONES

Las herramientas de computación deben de ser capaces de facilitar la tarea del Analista y Diseñador de software. La mayoría de las empresas requieren aplicaciones de buena calidad en tiempos muy cortos, es por éso que no pueden esperar muchas semanas, ni meses para el desarrollo de una aplicación. Para un desarrollador de sistemas, la parte compleja debe estar en el diseño, no en la codificación.

Power Builder ofrece la capacidad de simplificar el desarrollo de sistemas, debido a que es uno de los mejores lenguajes orientados a objetos. Solamente debe de arrastrar el control y pegarlo en la forma, después implementar el código asociado con éste (que es muy poco debido a la herencia de ventanas).

Podría pensarse que el simplificar código solamente favorece al programador inexperto, pero no es así, los diseñadores avanzados están enfocados en el diseño del sistema, cuando ya tienen esto, entonces necesitan herramientas que les ofrezcan llevar a cabo su proyecto en la forma más rápida, sencilla y segura.

El valor de un diseñador, no se encuentra en la cantidad de código que pueda realizar, sino en la correcta definición, planeación, desarrollo e implementación de un sistema de información.

Quiero concluir con una reflexión, cuando el analista diseñador de sistemas inicia un proyecto, es importante seguir una metodología de análisis y diseño, para seleccionar la más adecuada, debe experimentar con diferentes soluciones para escoger la que mejor se adapte a sus necesidades y a las de su empresa.

Mi recomendación es que el desarrollador de software, se apegue lo más posible al seguimiento de una metodología de análisis y diseño de software, para así llegar a buen término con el desarrollo de sistemas de información.

Espero que el presente trabajo haya sido de su agrado, así como una gran ayuda para las siguientes generaciones.

Agradezco el interés por la lectura de esta tesis y les deseo mucha suerte en sus próximos desarrollos de sistemas de información.

BIBLIOGRAFÍA

Eliyahu M. Goldratt, Jeff Cox. "La Meta", Ediciones Castillo, Monterrey N.L. 1993, p.p. 408

Fitzgerald Jerry, "Fundamentos de Análisis de Sistemas", Editorial Continental, México 1989, p.p. 558

Paul Mahler, "Power Builder: Desarrollo de aplicaciones cliente servidor", Editorial Prentice Hall, México 1995, p.p. 410

Senn James A., "Análisis y Diseño de Sistemas de Información", Editorial Mc Graw Hill, México 1990, p.p. 558

Victor Gerez, "Desarrollo y Administración de Programas de Computadora", Editorial Continental, México 1984, p.p. 299

Consultas en Internet

Internet: <http://www.powerbuilder.org/>

Internet: <http://listas.rcp.net.pe/mailman/listinfo/powerbuilder>

Internet: <http://www.asiware.com/sistemasenpowerbuilder.php>

Internet: <http://www.programacion.com/direcciones/powerbuilder/>

Internet: http://www.solomanuales.org/manuales_powerbuilder-manuall214195.htm

Internet: http://www.mundotutoriales.com/tutoriales_powerbuilder-mdpal14195.htm

Internet: http://www.solomanuales.org/manuales_sql_server-manuall1119.htm