

CAPÍTULO **II**: BASES TEORICAS

2.1 Antecedentes

- ✓ CASTILLO MARTÍNEZ, Cristian Patricio, 2007, Sistema Informático para la Administración de Consulta Externa de un Hospital Público, Ecuador: Universidad Tecnológica América.

☞ Debido a que en varios Hospitales Públicos de la ciudad, el historial clínico de los pacientes es llevado a cabo de manera manual, lo que genera pérdida de tiempo e información si es que no se cuenta con el orden debido, es que se plantea la elaboración de este sistema. Esta aplicación informática fue realizada bajo una arquitectura cliente / servidor, escrito en lenguaje Java y con funcionalidades de fácil entendimiento para el usuario. Se contó con el asesoramiento de Tutores especializados en la rama de Bases de Datos (MySQL), programación en Java y Diseño de Sistemas lográndose cumplir con los objetivos planteados del proyecto y con el cronograma establecido.

- ☞ Se concluyó que Java, aparte de sus bondades como software libre, facilita el trabajo al programador en cuanto a las funcionalidades que brinda, sobre todo en la realización de este tipo de sistemas.
- ✓ BAQUERIZO PALMA, Guillermo, 2007, Sistema de Información para Perfiles Estadísticos de Pacientes con Diabetes e Hipertensión. Escuela Superior Politécnica del Litoral de la ciudad de Guayaquil – Ecuador.
 - ☞ Tanto las clínicas más representativas como las menores fueron tomadas en cuenta como parte de la investigación encontrándose como problema principal muchos inconvenientes lógicos como resultado del manejo manual de las historias clínicas. La problemática radicaba bajo dos factores: una gran cantidad de tiempo y esfuerzo era realizado en la búsqueda de historias clínicas archivadas originando lentitud en el servicio así como un riesgo de deterioro del material en el que estas están plasmadas. Como solución ante tal problemática se planteó la automatización del proceso de registro de datos generados en las consultas médicas mediante un Sistema Web. El Sistema Web fue desarrollado bajo la plataforma Microsoft Visual Studio .NET 2003, con el lenguaje de programación ASP .NET y utilizando como base de datos Microsoft SQL Server 2000. Como resultado del desarrollo de este sistema una gama de recursos tanto en información, estadísticas y gráficos fueron brindados a los pacientes de las diversas clínicas a través de Internet.
 - ☞ Se concluyó que la utilización de la tecnología ASP y ASPX, facilita en gran medida la elaboración de los módulos de ingreso, actualización, eliminación y recuperación de la información y la elaboración de las gráficas, además de garantizar un buen rendimiento en comparación con otras tecnologías. El resultado del desarrollo del sistema fue exitoso, permitió un manejo de la información de la historia clínica de los pacientes, accediendo a estadísticas y gráficos que permitan a los doctores emitir análisis sustentados en la información que dispone el sistema.

- ✓ JARAMILLO FEIJOO, Leyda E., 2007, Diseño y Elaboración de un Sistema de Información para el Análisis Estadístico de Historias Clínicas de Pacientes con Enfermedades Cardiológicas, Ecuador: Escuela Superior Politécnica del Litoral.

☞ En este proyecto, propone el Diseño y Elaboración de un Sistema de Información para el Análisis Estadístico de Historias Clínicas de Pacientes con Enfermedades Cardiológicas aplicado al caso de un hospital de la ciudad. Debido a que en muchos de los hospitales de la ciudad la mayoría de procesos son manuales y tediosos, el proyecto tuvo como finalidad agilizar estos en beneficio de los pacientes, proporcionando reportes estadísticos de la población de pacientes para una eficiente toma de decisiones de los doctores. Sin embargo no todos los procesos fueron tomados, el objetivo principal fue la automatización del proceso de apertura de la historia clínica como los ingresos, consultas de pacientes y el análisis estadístico de los datos del paciente para mejorar el servicio de la salud en la sociedad, además de ofrecer información acerca de los factores de riesgo, sobre cómo prevenir enfermedades del corazón, llevar el control de las historias clínicas y preguntas frecuentes de los pacientes. El sistema fue desarrollado en ASP .NET ya que para el autor ofrecía varias ventajas importantes acerca de los modelos de programación Web. Se concluye que el sistema de información se encuentra actualmente funcionando en algunos hospitales de la ciudad y lleva un registro de datos personales y clínicos de pacientes en el área de cardiología, que podrá ser complementada en un futuro con otros módulos que puedan ser añadidos al sistema actual.

- ✓ CASSARÓ CASTILLO, Guillermo M., 2007, Desarrollo de un Sistema Web de Citas Online para mejorar el Proceso de Atención al Cliente en la Clínica Dental Gamarra de Trujillo, Perú: Universidad Privada del Norte

☞ En este proyecto, se muestra una problemática debido a la mala estructuración y manera de llevar a cabo el proceso. Las peticiones de citas son anotadas en una agenda de citas de manera manual,

en donde el tiempo de atención de cada paciente es un aproximado calculado arbitrariamente. Una ficha es creada y archivada de manera ordenada por cada paciente, sin embargo la búsqueda de estos documentos se volvió un trabajo tedioso debido al creciente número de pacientes con el que contaba la institución. Ante esta problemática se planteó el desarrollo de un sistema Web de citas Online que permita mejorar el proceso de atención al cliente en la clínica.

- ☞ Para el desarrollo del producto de software se utilizó la metodología RUP, bajo el modelado UML, con el lenguaje de programación ASP.NET y un motor de base de datos SQL Server 2000. Se concluyó en una elevada satisfacción del cliente, y en diversas ventajas ofrecidas por las herramientas utilizadas en este tipo de sistemas.

2.2 Marco Teórico

2.2.1 Visual Studio 2010

Microsoft Visual Studio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows. Soporta varios lenguajes de programación tales como Visual C++, Visual C#, Visual J#, ASP.NET y Visual Basic .NET, aunque actualmente se han desarrollado las extensiones necesarias para muchos otros.

Visual Studio permite a los desarrolladores crear aplicaciones, sitios y aplicaciones web, así como servicios web en cualquier entorno que soporte la plataforma .NET (a partir de la versión net 2002). Así se pueden crear aplicaciones que se intercomunican entre estaciones de trabajo, páginas web y dispositivos móviles.

Versiones

- ✓ Microsoft Visual Studio 6
- ✓ Microsoft Visual Studio .NET (2002)
- ✓ Microsoft Visual Studio .NET 2003
- ✓ Microsoft Visual Studio 2005

- ✓ Microsoft Visual Studio 2008
- ✓ Microsoft Visual Studio 2010
- ✓ Microsoft Visual Studio 2012
- ✓ Microsoft Visual Studio 2015

A partir de la versión 2005 Microsoft ofrece gratuitamente las Express Editions. Estas son varias ediciones básicas separadas por lenguajes de programación o plataforma enfocadas para novatos y entusiastas. Estas ediciones son iguales al entorno de desarrollo comercial pero sin características avanzadas. Las ediciones que hay son:

- ✓ Visual Basic Express Edition
- ✓ Visual C# Express Edition
- ✓ Visual C++ Express Edition
- ✓ Visual J# Express Edition (Desapareció en Visual Studio 2008)
- ✓ Visual Web Developer Express Edition (para programar en ASP.NET)
- ✓ Visual F# (Apareció en Visual Studio 2010, es parecido al J#)*

Adicionalmente, Microsoft ha puesto gratuitamente a disposición de todo el mundo una versión reducida de MS SQL Server llamada SQL Server Express Edition cuyas principales limitaciones son que no soporta bases de datos superiores a 4 GB de tamaño, únicamente utiliza un procesador y un Gb de RAM, y no cuenta con el Agente de SQL Server.

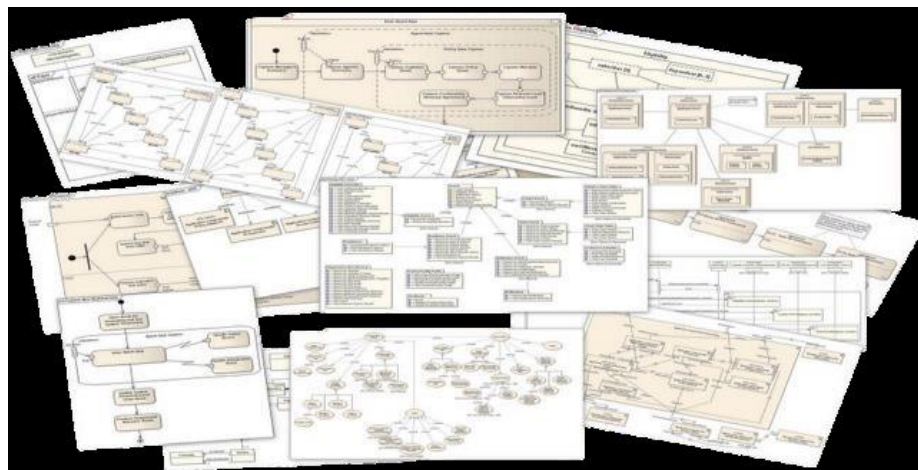
En el pasado se incluyeron los siguientes productos:

- ✓ Visual InterDev
- ✓ Visual J++
- ✓ Visual FoxPro
- ✓ Visual SourceSafe

2.2.2 Lenguaje de Modelamiento Unificado (UML).

El Lenguaje de Modelado Unificado (UML: Unified Modeling Language) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90s. UML es llamado un lenguaje de modelado, no un método. El lenguaje de modelado es la notación (principalmente gráfica) que usan los métodos para expresar un diseño.

Figura N° 01: Diagramas UML.



Fuente: Internet

2.2.3 Proceso Unificado Racional (RUP)

Rational Unified Process (RUP), basado en UML (Unified Modeling Language) para la ingeniería de sistemas y de software. Esta metodología proporciona al equipo del proyecto procedimientos y herramientas que promueven las siguientes prácticas:

- a) Gestión de los Requisitos.
- b) Desarrollos iterativos.
- c) Uso de arquitecturas basadas en componentes.
- d) Desarrollo Visual del Software (con UML).
- e) Verificación continua de la calidad del software.
- f) Gestión de los Cambios.

A. Características principales

- a) Guiado/Manejado por casos de uso:

La razón de ser de un sistema software es servir a usuarios ya sean humanos u otros sistemas. Un caso de uso es una facilidad que el software debe proveer a sus usuarios. Los casos de uso reemplazan la antigua especificación funcional tradicional y constituyen la guía fundamental establecida para las actividades a realizar durante todo el proceso de desarrollo incluyendo el diseño, la implementación y las pruebas del sistema.

b) Centrado en arquitectura:

La arquitectura involucra los elementos más significativos del sistema y está influenciada entre otros por plataformas software, sistemas operativos, manejadores de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados y requerimientos no funcionales. Los casos de uso guían el desarrollo de la arquitectura y la arquitectura se realimenta en los casos de uso, los dos juntos permiten conceptualizar, gestionar y desarrollar adecuadamente el software.

c) Iterativo e Incremental:

Para hacer más manejable un proyecto se recomienda dividirlo en ciclos. Para cada ciclo se establecen fases de referencia, cada una de las cuales debe ser considerada como un mini proyecto cuyo núcleo fundamental está constituido por una o más iteraciones de las actividades principales básicas de cualquier proceso de desarrollo.

d) Desarrollo basado en componentes:

La creación de sistemas intensivos en software requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema. Esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o que se desarrollen y maduran sus componentes.

e) Utilización de un único lenguaje de modelamiento:

UML es adoptado como único lenguaje de modelamiento para el desarrollo de todos los modelos.

f) Proceso Integrado:

Se establece una estructura que abarque los ciclos, fases, flujos de trabajo, mitigación de riesgos, control de calidad, gestión del proyecto y control de configuración. El proceso unificado establece una estructura que integra todas estas facetas. Además esta estructura cubre a los vendedores y desarrolladores de herramientas para soportar la automatización del proceso, soportar flujos individuales de trabajo, para construir los diferentes modelos e integrar el trabajo a través del ciclo de vida y a través de todos los modelos.

A.1 Fases del Ciclo de Desarrollo

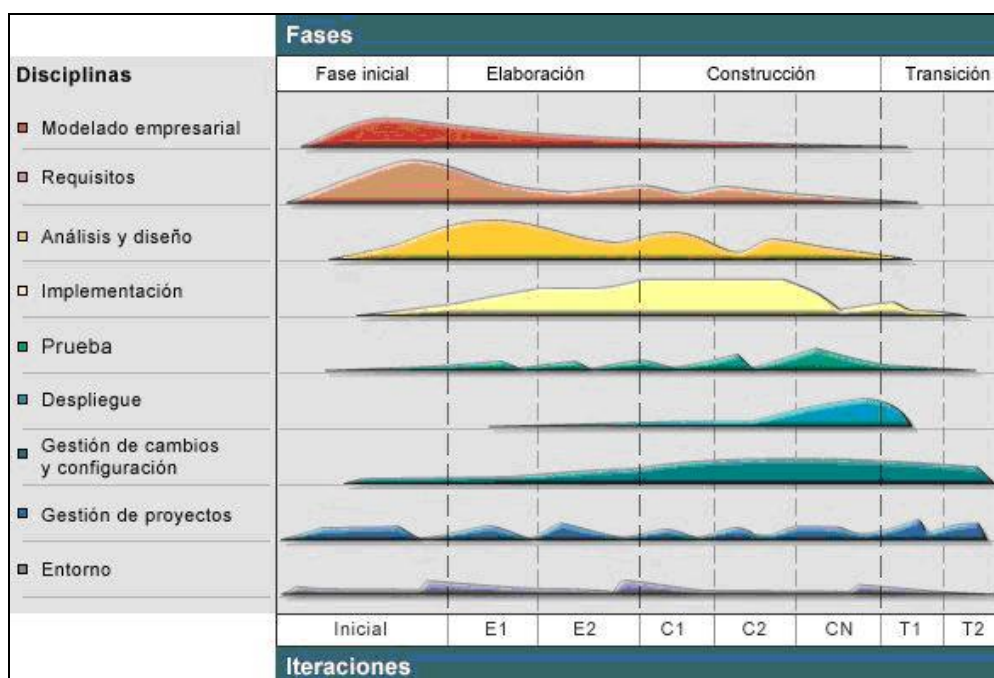
El RUP es un modelo en fases que identifica cuatro diferentes en el proceso del software. Estas fases son:

- a) Inicio: Abarca la comunicación con el cliente y las actividades de planeación. Al colaborar con los clientes y usuarios finales se identifican los requisitos de negocios para el software que se describen a través de un conjunto preliminar de casos de uso que describen cuales características y funciones son deseables para cada clase importante de usuarios.
- b) Elaboración: Refina y expande los casos de uso preliminares que se desarrollan como una parte de la fase de inicio. Además, expande la representación arquitectónica para incluir cinco visiones diferentes del software.
- c) Construcción: Desarrolla o adquiere los componentes del software que harán que cada caso de uso sea operativo

para los usuarios finales. Todas las características y funciones necesarias y requeridas del incremento del software se implementan en código fuente. Conforme los componentes están en proceso de implementación, se diseñan y ejecutan pruebas de unidad para cada uno de ellos.

d) Transición: Abarca las últimas etapas de la actividad genérica de construcción y la primera parte de la actividad genérica de despliegue. El software se entrega a los usuarios finales para realizar pruebas betas, y retroalimentación del usuario, reporta tanto defectos como cambios necesarios. Además, el equipo de software crea la información de soporte necesaria para el lanzamiento. Al final de la fase de transición, el incremento de software se convierte en un lanzamiento de software utilizable.

Figura N° 02 -Fases de RUP



Fuente: http:// analisis1daid.wikispaces.com/Metodologia_RUP_UML

A.2 Artefactos

RUP en cada una de sus fases realiza una serie de artefactos que sirven para comprender mejor tanto el análisis como el diseño del sistema. Estos artefactos son los siguientes:

Inicio:

- ✓ Especificación de Requerimientos.
- ✓ Modelo de Casos de Uso del Negocio.
- ✓ Modelos de Objetos del Negocio.
- ✓ Modelo del dominio del Problema.

Elaboración:

- ✓ Diagramas de caso de uso.

Construcción:

Vista Lógica:

- ✓ Diagrama de Clases.
- ✓ Diagrama de Capa de Datos.
- ✓ Diagrama de Base De Datos.

Vista de Implementación:

- ✓ Diagrama de Colaboración.
- ✓ Diagrama de Secuencia.
- ✓ Diagrama de estados.
- ✓ Diagrama de Actividades.

Vista Conceptual:

- ✓ Modelo del dominio.

Vista física:

- ✓ Diagrama de Componentes.
- ✓ Diagrama de despliegue.

2.2.4 Rational Rose.

Es una de las más poderosas herramientas de modelado visual para el análisis y diseño de sistemas basados en objetos. Se utiliza para modelar un sistema antes de proceder a construirlo.

Cubre todo el ciclo de vida de un proyecto:

- ✓ Concepción y formalización del modelo
- ✓ Construcción de los componentes

- ✓ Transición a los usuarios
- ✓ Certificación de las distintas fases.

Es una herramienta software para el Modelado Visual mediante UML de sistemas software.

Características

- ✓ Permite Especificar, Analizar, Diseñar el sistema antes de Codificarlo.
- ✓ Mantiene la consistencia de los modelos del sistema software.
- ✓ Chequeo de la sintaxis UML.
- ✓ Generación Documentación automáticamente.
- ✓ Generación de Código a partir de los Modelos.
- ✓ Ingeniería Inversa (crear modelo a partir código).

2.2.5 Microsoft SQL 2008

Los sistemas de Información se dividen en dos categorías (Casares, 2003):

Sistemas operacionales: Son aquellos que tienen como objetivos reflejar el estado y funcionamiento de la empresa registrando las transacciones u operaciones diarias, de aquí que los mismos se conozcan como sistemas de Procesamiento de Transacciones en Línea (OLTP).

Sistemas para el soporte de decisiones: Son aquellos que tienen como objetivos medir y controlar el desarrollo de las variables importantes del negocio, buscando identificar, proyectar y predecir tendencias a partir de los datos acumulados.

Desde que se inició la era de la computadora, las organizaciones han usado los datos desde sus sistemas operacionales para atender sus necesidades de información. Algunas proporcionan acceso directo a la información contenida dentro de las aplicaciones operacionales. Otras, han extraído los datos desde sus bases de datos operacionales para combinarlos de

varias formas no estructuradas, en su intento por atender a los usuarios en sus necesidades de información (Casares, 2003).

Bill Inmon fue uno de los primeros autores en escribir sobre el tema de almacenes de datos, define un almacén de datos en términos de las características del repositorio de datos (Inmon, 2007):

Orientado a temas: Los datos en la base de datos están organizados de manera que todos los elementos de datos relativos al mismo evento u objeto del mundo real queden unidos entre sí.

Variante en el tiempo: Los cambios producidos en los datos a lo largo del tiempo quedan registrados para que los informes que se puedan generar reflejen esas variaciones.

No volátil: La información no se modifica ni se elimina, una vez almacenado un dato, éste se convierte en información de sólo lectura y se mantiene para futuras consultas.

Integrado: La base de datos contiene los datos de todos los sistemas operacionales de la organización y dichos datos deben ser consistentes.

Los mercados de datos son subconjuntos de datos de un almacén de datos para áreas específicas. Desde el punto de vista de diseño se aplica en el mercado de datos todo lo que es aplicable a un almacén de datos (Inmon, 2007).

El Modelo Dimensional es el más usado en los sistemas de almacenes de datos, este es diferente al modelo relacional utilizado en los sistemas OLTP. Este modelo se basa en dimensiones, las cuales representan categorías de información, atributos que representan un único nivel dentro de una dimensión, pueden existir jerarquías de atributos las cuales expresan relaciones entre diferentes atributos y por último tablas de hechos, las cuales contienen datos de interés, que presentan un nivel de granularidad. La granularidad es el nivel más bajo de información que será almacenado en la tabla de hechos. El primer paso al diseñar una tabla de hechos es determinar la granularidad.

Esquemas del diseño dimensional

Esquema Estrella: Una tabla de hechos en el centro conectada con un conjunto de tablas de dimensiones.

Esquema Copo de Nieve: Un refinamiento del anterior donde algunas dimensiones se normalizan en tablas más pequeñas.

Constelación de Hechos: Múltiples tablas de hechos comparten tablas de dimensión que se visualizan como una constelación de hechos.

La gestión administrativa reconoce que una manera de elevar su eficiencia está en hacer el mejor uso de los recursos de información que ya existen dentro de la organización. El almacén de datos, es actualmente, el centro de atención de las grandes instituciones, porque provee un ambiente para que las organizaciones hagan un mejor uso de la información que está siendo administrada por diversas aplicaciones operacionales (Casares, 2003).

La arquitectura de un almacén de datos consta de tres niveles (Casares, 2003):

Bases de datos fuentes (producción e históricos).

Una base de datos con datos resumidos extraídos de las bases de producción (almacén de datos).

Interfaces orientadas a usuarios que extraen información para la toma de decisiones. Las clásicas son: consultas y reportes, análisis multidimensional y Minería de Datos.

Base de Datos Fuentes

Consiste en bases de datos de producción así como en históricos de dichas bases. Estas bases de datos pueden estar implementadas en diferentes tipos de sistemas: BD-Relacionales, BD-geográficas, BD-textos, archivos, etc. Una característica común es que almacenan ítems de datos atómicos, los cuales son relevantes como datos de producción, pero pueden ser demasiado finos para servir como base para la toma de decisiones. Además, la noción de calidad de los datos en estas bases se basa en la

consistencia de dichos registros, independientemente de la relevancia que estos tengan dentro del problema.

Una componente importante en el almacén de datos es el Diccionario de Datos (Meta-Data), en el mismo se describen los datos almacenados con el objetivo de facilitar el acceso a los mismos a través de las herramientas de explotación del almacén de datos. El Diccionario de Datos establece correspondencias entre los datos almacenados y los conceptos que estos representan para facilitar la extracción de información por parte del usuario final.

Interfaces orientadas a usuarios que extraen información para la toma de decisiones

Interfaces para consultas y reportes complejos: Permiten al usuario construir gráficas y reportes a partir de la información contenida en el almacén de datos y descrita a través del Diccionario de Datos. Algunas funcionalidades típicas de estas herramientas son: agrupamiento y des agrupamiento dinámico de datos en reportes, cambios en el orden de los campos del reporte, visualización del resultado de las consultas en forma gráfica (barras, torta, puntos, etc.). Estas herramientas generan las expresiones en el lenguaje de consulta que recupera los datos pedidos (típicamente SQL), se conectan al almacén de datos, recuperan el resultado y lo formatean según la especificación dada.

Productos de análisis de datos (OLAPs): Permiten representar los datos del problema en términos de *dimensiones*. Por ejemplo, si se trata de ventas de productos en diferentes zonas, una dimensión del problema son las zonas, otra los productos y otra el tiempo. De esta manera, las consultas de análisis de datos de una dimensión en función de la otra se realizan en forma inmediata.

Herramientas de Minería de Datos: Permiten explorar el almacén de datos en búsqueda de relaciones desconocidas o inesperadas entre los datos.

Las principales motivaciones para construir un almacén de datos son las siguientes (Casares, 2003):

Disponer de Sistemas de Información de apoyo a la decisión.

Disponer de bases de datos que permitan extraer conocimiento de la información histórica almacenada en la organización.

Diseñar una base de datos que permita ejecutar consultas desconocidas.

Microsoft SQL Server 2008

Microsoft SQL Server 2008 brinda una plataforma para la construcción y el mantenimiento de almacenes de datos, a continuación se muestran algunas de sus nuevas características y mejores prácticas asociadas a las mismas:

✓ **Compresión de Datos**

La compresión de datos reduce el espacio requerido para almacenar tablas e índices permitiendo un uso más eficiente de la capacidad de almacenamiento. Existe la posibilidad de la compresión por artículo o por página. La compresión por artículo almacena todos los campos en un formato de ancho variable, la compresión por página hace lo mismo pero se realiza entre los artículos de una misma página. Un diccionario a nivel de página es usado para almacenar los valores comunes, además prefijos comunes de valores de columnas son almacenados solo una vez en la página. Ambas formas de compresión pueden ser aplicadas a las tablas y a los índices.

✓ **Encriptación de Datos Transparente**

La encriptación de datos transparente permite que los datos sean almacenados de forma segura mediante la encriptación de los ficheros de la base de datos. El SQL Server realiza la encriptación y des encriptación directamente por lo que el proceso es transparente para la aplicación conectada. Si se utiliza a la vez compresión de datos y encriptación, se deben realizar las operaciones en este orden.

✓ **Gobernador de Recursos**

El Gobernador de Recursos les permite a los administradores el control y la asignación de recursos como Procesadores y Memoria a las aplicaciones de más alta prioridad.

✓ **Adición en Caliente de Procesadores y Memoria**

La edición de SQL Server Enterprise de 64 bits permite la adición en caliente de procesadores y memoria sin necesidad de apagar el servidor ni limitar las conexiones existentes.

✓ **Operador MERGE**

El nuevo operador MERGE simplifica el proceso de carga de un almacén de datos desde su fuente. Este nuevo operador distingue los artículos nuevos y actualizados en la base de datos fuente y realiza la acción apropiada en el almacén de datos.

✓ **Nuevos Tipos de Datos Espaciales**

Los nuevos tipos de datos espaciales GEOGRAPHY y GEOMETRY permite que datos espaciales puedan almacenarse directamente en SQL Server 2008. GEOGRAPHY permite representar datos geodésicos en tres dimensiones los cuales son utilizados por aplicaciones GPS y GEOMETRY permite representar puntos en planos de dos dimensiones. Además existe una integración con Virtual Earth lo cual permite representaciones gráficas de localizaciones físicas.

Todas estas nuevas características hacen de Microsoft SQL Server 2008 una herramienta de avanzada para la creación y el mantenimiento de almacenes de datos.

2.3 Definiciones de Términos

2.3.1 Aplicación

Programa informático diseñado como herramienta para permitir a un usuario realizar uno o diversos tipos de trabajos. Esto lo diferencia principalmente de otros tipos de programas como los sistemas operativos (que hacen funcionar al ordenador), las utilidades (que

realizan tareas de mantenimiento o de uso general), y los lenguajes de programación (con el cual se crean los programas informáticos).

2.3.2 Base de Datos

Una base de datos es un conjunto de datos organizados, entre los cuales existe una correlación y que además, están almacenados con criterios independientes de los programas que los utilizan. También puede definirse, como un conjunto de archivos interrelacionados que es creado y manejado por un Sistema de Gestión o de Administración de Base de Datos (Data Base Management System-DBMS).

2.3.3 Encapsulamiento

Consiste en ocultar los detalles de la implementación de un objeto, a la vez que se provee una interfaz pública por medio de sus modos permitidos.

2.3.4 Herramienta CASE

(Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software reduciendo el costo de las mismas en términos de tiempo y de dinero.

2.3.5 Información

Conjunto de datos capturados, organizados, clasificados e integrados con un significado y contexto.

2.3.6 Interfaz Gráfica de Usuario

Componente de una aplicación informática que el usuario visualiza y a través de la cual opera con ella. Está formada por ventanas, botones, menús e iconos, entre otros elementos.

2.3.7 Modelo de Datos Conceptuales

Son los orientados a la descripción de estructuras de datos y restricciones de integridad. Se usan fundamentalmente durante la etapa de Análisis de un problema dado y están orientados a representar los elementos que intervienen en ese problema y sus relaciones. El ejemplo más típico es el Modelo Entidad-Relación.

2.3.8 Modelos de Datos Lógicos

Son orientados a las operaciones más que a la descripción de una realidad. Usualmente están implementados en algún Manejador de Base de Datos. El ejemplo más típico es el Modelo Relacional, que cuenta con la particularidad de contar también con buenas características conceptuales (Normalización de bases de datos).

2.3.9 Modelos de Datos Físicos

Son estructuras de datos a bajo nivel implementadas dentro del propio manejador. Ejemplos típicos de estas estructuras son los Árboles B+, las estructuras de Hash, etc.

2.3.10 Modelo de Procesos

Debe ser entendido, a saber, por dos cuestiones importantes: el modelado y los procesos. Frecuentemente los sistemas (conjuntos de procesos y subprocesos integrados en una organización) son difíciles de comprender, amplios, complejos y confusos; con múltiples puntos de contacto entre sí y con un buen número de áreas funcionales, departamentos y puestos implicados. Un modelo puede dar la oportunidad de organizar y documentar la información sobre un sistema.

2.3.11 OLTP

Es una base de datos para soportar procesos transaccionales en línea, puede no ser adecuada para el Data Warehouse ya que ha sido diseñada para maximizar la capacidad transaccional de sus datos y típicamente tiene cientos de tablas la gran mayoría

normalizadas. Su diseño también ha sido condicionado por los procesos operacionales que deberá soportar para la óptima actualización de sus datos, normalmente muchas de sus tablas en constantes y continuos cambios. Los sistemas Data Warehouse están orientados a procesos de consultas en contraposición con los procesos transaccionales.

2.3.12 Sistema de Información

Conjunto formal de procesos que, operando sobre una colección de datos, recopila, elabora y distribuye la información necesaria para la operación de dicha empresa.

2.3.13 Software

Termino general que denomina diversos tipos de programas usados en la computación.