

UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA
FACULTAD DE GEOLOGIA, GEOFISICA Y MINAS
ESCUELA PROFESIONAL DE INGENIERIA GEOFISICA



**DESARROLLO DE UNA ESTRUCTURA PARA UNA BASE DE DATOS
SISMICOS EN EL SISTEMA ORACLE Y CONSULTA A PARTIR DE
APLICACIONES CGI
ESTIMACIÓN DE LA RELACIÓN ATENUACIÓN - INTENSIDAD PARA
SISMOS EN EL PERÚ**

Tesis presentada por el Bachiller en
Ciencias Geofísicas:

IGOR ALBERTO VALDIVIA POLANCO

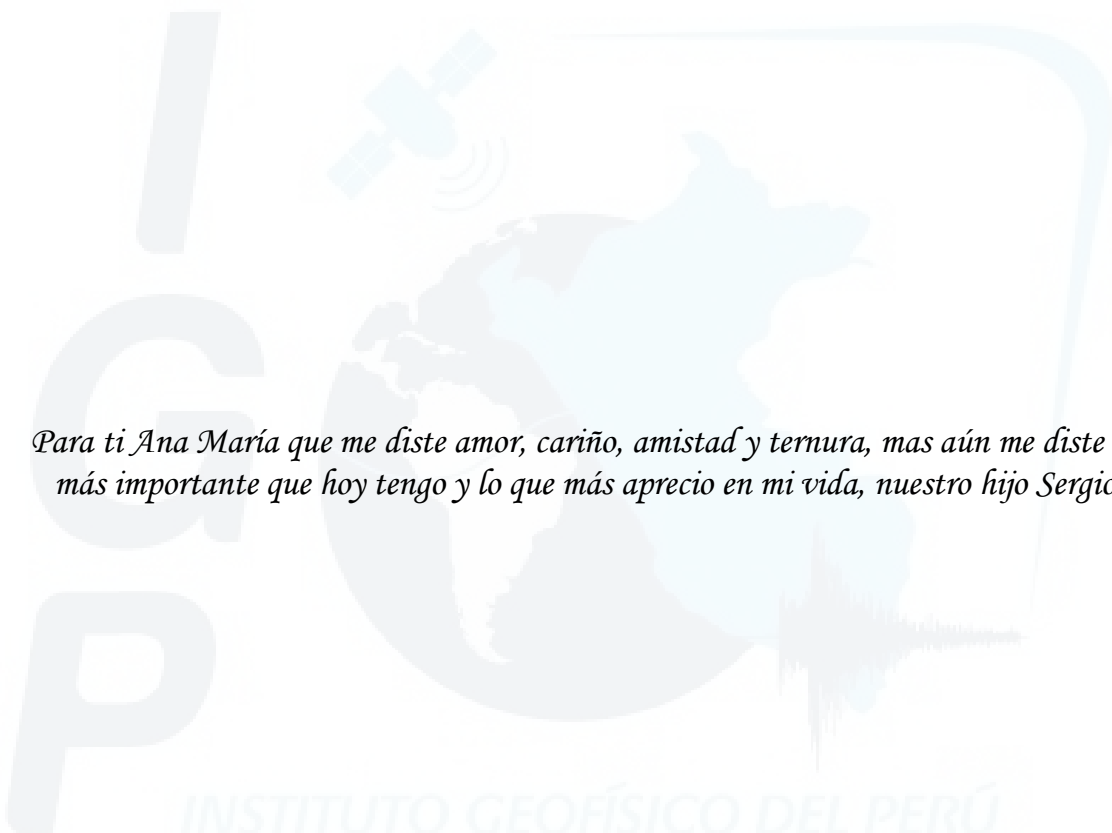
Para optar el Título Profesional de
Ingeniero Geofísico

AREQUIPA – PERÚ
2003

Dedico este trabajo a mi Padre Celestial que en todo momento me guía



Para ti papá y mamá, gracias por ayudarme a dar un pasito más en mi vida y hacer de mi un hombre cada día más humano.



Para ti Ana María que me diste amor, cariño, amistad y ternura, mas aún me diste lo más importante que hoy tengo y lo que más aprecio en mi vida, nuestro hijo Sergio.

Para todos ustedes hermanos míos, gracias por ser tan unidos y por ser la familia en donde aprendí a amar.

INDICE

AGRADECIMIENTOS

INDICE	I
RESUMEN	VII
CAPITULO 1. INTRODUCCION	
1.1.- Por qué una Base de Datos Sísmicos	3
1.2.- Area de Estudio	5
1.3.- Objetivos del Trabajo	5
CAPITULO 2. CARACTERÍSTICAS DEL BANCO DE DATOS SISMICOS	
2.1.- Introducción	8
2.2 .- Sismicidad en el Perú	8
2.2.1.- Sismicidad Histórica	9
2.2.2.- Sismicidad Instrumental	12
2.2.3.- Principales Sismos Destruyentes	15
2.3.- Información Sísmica	19
2.3.1.- Red Sísmica Nacional (RSN)	19
2.3.1.1.- Características Instrumentales de la RSN	21
2.3.1.2.- Archivos Generados por las Estaciones de Banda Ancha	22
2.3.1.2.- Archivos Generados por las Estaciones de Periodo Corto	23
2.3.2.- Archivos con Datos Sísmicos	24
2.3.2.1.- Archivos de Entrada	25
2.3.2.2.- Archivos de Salida	25
2.3.3.- Catálogos Sísmicos	27
2.3.4.- Soluciones Gráficas	28
2.3.3.- Informes y Estudios Sismológicos	28
CAPITULO 3. EL SOFTWARE DE BASE DE DATOS ORACLE	
3.1.- Introducción	30

3.2.- El Oracle	30
3.2.1.- Características de Oracle	32
3.3.- El servidor Oracle8i	34
3.3.1.- Introducción	34
3.3.1.1.- Areas Clave de Enfoque Oracle	34
3.3.1.2.- Capacidad de Usuarios y Tamaño	35
3.3.1.3.- Partición y Paralelismo	35
3.3.1.4.- Configuraciones del Servidor	36
3.3.2.- Características de Oracle8i Server	37
3.3.3.- El Oracle8i y la tecnología Objeto – Relacional	41
3.3.3.1.- El Enfoque Relacional	42
3.3.3.1.1.- Conceptos Básicos	42
3.3.3.1.2.- Claves Primarias	45
3.3.3.1.3.- Interrelaciones. Claves Foráneas	45
3.3.3.1.4.- Integridad Relacional	49
3.3.3.1.5.- Formas Normales	50
3.3.3.1.1.- Lenguajes Relacionales	51
3.3.3.2.- Base de Datos Orientadas a Objetos	52
3.4.- Arquitectura de Oracle	56
3.4.1.- Diccionario de Datos	56
3.4.2.- Archivos de Datos	57
3.4.2.1.- Datos de Usuario y datos de sistema	57
3.4.3.- Espacio de Tablas	57
3.4.3.1.- Nombres y Contenidos de los Espacios de Tablas	59
3.4.4.- Registros de Rehacer	60
3.4.4.1.- Funcionamiento de los Registros de Rehacer	61
3.4.5.- Archivos de Control	62

3.4.5.1.- Funcionamiento de los Archivos de Control	62
3.4.6.- Programas	63
3.4.7.- Procesos Auxiliares de la Base de Datos	64
3.4.8.- Estructura de Memoria	67
3.4.8.1.- Area Global del Sistema	67
3.4.8.2.- Area Global de Programa	70
3.4.9.- Instancias	70
3.5.- El Servidor de Base de Datos “Clima”	73
3.5.1.- Características del Hardware del Servidor de Base de Datos.	73
3.5.1.- Seguridad del Servidor de Base de Datos “Clima”	74
CAPITULO 4. ESTRUCTURA Y DISEÑO DE LA BASE DE DATOS SISMICOS	
4.1.- Introducción	75
4.2.- SQL/Plus	75
4.2.1.- El SQL	76
4.3.- Objetos de Base de Datos	79
4.3.1.- Tablas	79
4.3.2.- Vistas	82
4.3.3.- Indices	84
4.3.4.- Sinónimos	84
4.4.- Estructura de Tablas de la Base de Datos Sísmicos	85
4.4.1.- Tabla catalogo_1471_1982	87
4.4.2.- Tabla Catalogo_1983_act	92
4.4.3.- Tabla Program_files	97
4.4.4.- Tabla Inform_1471_1982	98
4.4.5.- Tabla inform_1983_act	99
4.4.6.- Tabla Graphics	100
4.4.7.- Tabla Estaciones_param	101

4.4.8.- Tabla Estaciones_imform	106
4.4.9.- Tabla Ondas_sis	107
4.4.10.- Tabla Ondas_sac	109
4.5.- Relaciones entre Tablas de la Base de Datos Sísmicos	111
4.5.1.- Diagrama Entidad – Relación	112
4.6.- Vistas de la Base de Datos Sísmicos	112
CAPITULO 5. MECANISMOS DE INSERCIÓN Y ALMACENAMIENTO DE DATOS	
5.1.- Introducción	115
5.2.- SQL*Loader	115
5.2.1.- Palabras Clave en Línea de Comando de SQL*Loader	117
5.3.- Archivo de Control	123
5.4.- Datos de Ingreso y Archivo de Datos	127
5.4.1.- Registros Físicos	127
5.4.2.- Registros Lógicos	127
5.4.3.- Campos de Datos	127
5.5.- Salida de SQL*Loader	132
5.5.1.- Archivo Log Carga completa	132
5.5.2.- Archivo Log Carga Incompleta	134
5.5.3.- Archivo de Fallos	135
5.5.4.- Archivo de Descarte	135
5.6.- Ingreso de los Archivos Binarios a la base de datos	135
5.6.1.- LOB Internos	136
5.6.1.1.- Cargando LOB Internos Usando LOBFILE	136
5.6.2.- LOB Externos	138
5.6.2.1.- Ingreso e LOB externos usando SQL*Loader	139
5.7.- Aplicaciones Diseñadas para la generación de Archivos de Datos	141

5.7.1.- Organización de Directorios dentro del Servidor	142
5.7.2.- Archivos de Datos Generados por Programas	144
5.7.3.- Archivos de Control	145
CAPITULO 6. CONSULTA A LA BASE DE DATOS SISMICOS:	
APLICACIONES CGI	
6.1.- Introducción	148
6.2.- Interfaz Común de Compuertas (CGI)	148
6.2.1.- Lenguaje de Programación del CGI	149
6.2.2.- Mecanismo de Trabajo del CGI	149
6.3.- Estructura del Esquema de Consulta de la Información Sísmica	152
6.3.1.- Estructura de los CGI del Esquema de Consulta	153
6.3.2.- Estructura de los documentos HTML	158
6.3.3.- Documentos HTML de salida	159
CAPÍTULO 7. CÁLCULO DE LA RELACIÓN DE ATENUACIÓN	
– INTENSIDAD PARA SISMOS EN PERÚ	
7.1.- Introducción	165
7.2.- Tectónica y Sismicidad	166
7.3.- La Intensidad	169
7.3.1.- Definición	169
7.3.2.- Escalas de Intensidad Sísmica	169
7.4.- Análisis de los Mapas de Intensidad	173
7.4.1.- Mapas de Intensidad de Sismos de Subducción	176
7.4.2.- Mapas de Intensidad de Sismos Continentales	176
7.5.- Modelos de Atenuación y cálculo de las Relaciones	
Atenuación – Intensidad	182
7.5.1.- Modelo Elipsoidal de Atenuación	183
7.5.1.1.- Relación Atenuación – Intensidad	185
7.5.2.- Modelo Circular de Atenuación	186

CAPITULO 8. CONCLUSIONES Y RECOMENDACIONES**BIBLIOGRAFIA****ANEXOS****A. Programas CGI y otras Aplicaciones****B. Escalas e Intensidad Sísmica****C. Participación en trabajos de Investigación**

- *Evaluación de las intensidades macrosísmicas en las provincias de Caravelí y Camaná (Arequipa) debidas al terremoto del 23 de junio del 2001. Informe Final, CNDG – IGP, p 305-314.*
- *La magnitud de sismos locales y regionales ocurridos en el Perú a partir de la onda LG y duración de su registro. Boletín de la Soc. Geológica del Perú. V. 94 (2002), p 61-70.*
- *Estructura de una Base de Datos Sísmicos en el Sistema Oracle y Consulta a partir de aplicaciones CGI. Boletín de la Soc. Geológica del Perú (en impresión Julio del 2003).*
- *Cálculo de la Relación Intensidad - Atenuación a partir de las isosistas de Sismos de Subducción ocurridos en el Perú. Compendio de trabajos de investigación. CNDG – Biblioteca IGP, p 47-58.*

RESUMEN

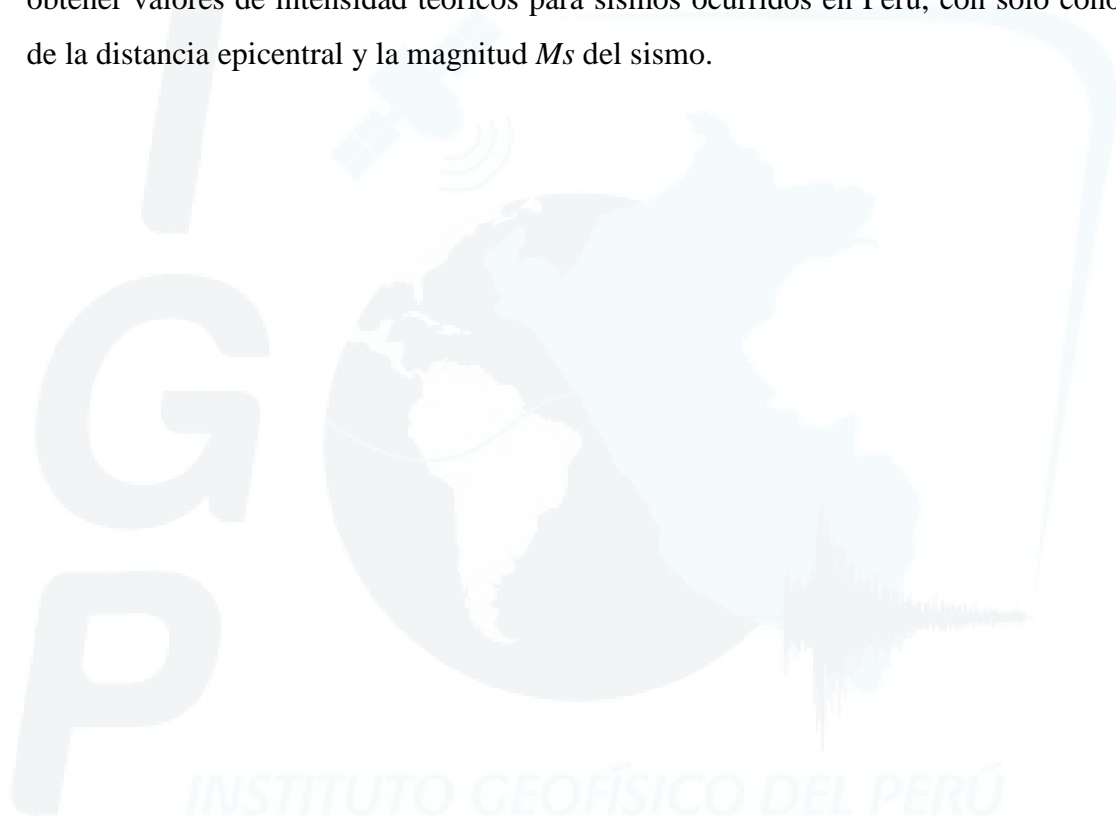
En este estudio se propone desarrollar la estructura para una Base de Datos Sísmicos haciendo uso del “Software de Base de Datos Oracle versión 8i”, a fin de almacenar y administrar la información sísmica del Centro Nacional de Datos Geofísicos del Instituto Geofísico del Perú (CNDG - IGP). Una vez creada la Base de Datos, esta podrá ser consultada a través de la Internet, empleando para ello aplicaciones CGI. Asimismo, como parte de este estudio se calculó relaciones Atenuación – Intensidad para los sismos ocurridos en Perú, para los cuales se ha establecido un modelo elipsoidal y circular para la atenuación de la intensidad.

El desarrollo de la sismometría ha permitido que el Instituto Geofísico de Perú cuente con una Red Sísmica Nacional compuesta por 31 estaciones sísmicas (20 de periodo corto y 11 de banda ancha) distribuidas en todo el país. Debido a este gran número de estaciones, el Centro Nacional de Datos Geofísicos (CNDG) cuenta con un gran volumen de información sísmica que considera archivos con formas de onda, lectura de fases, parámetros hipocentrales, catálogos sísmicos, informes técnicos así como, estudios de grandes sismos y de sismicidad. Para la organización y administración de este tipo de información se utiliza el software de base de datos Oracle y su tecnología Objeto – Relacional. Como resultado se ha estructurado un Banco de Datos Sísmicos en *tablas*, las mismas que se encuentran relacionadas una con otra formando un esquema Relacional y en las cuales se ha almacenado toda la información del CNDG tanto de tipo estructurado como no – estructurado; es decir, números, caracteres, datos alfanuméricos, archivos gráficos, de texto y archivos binarios.

Oracle es apoyado en una muy sofisticada arquitectura interna la cual esta organizada en un potente servidor de base de datos que paralelamente realiza labores Web; por lo tanto, es fácil de ser acoplada a Internet, esta característica ha permitido desarrollar consultas dinámicas a la base de datos con aplicaciones ejecutadas dentro del servidor mediante un conjunto de programas CGI. La estructura de la Base de Datos Sísmicos del CNDG – IGP desarrollada en este estudio, se encuentra operativa en el servidor “Clima” y puede ser libremente consultado por Internet a la siguiente dirección electrónica; <http://clima.igp.gob.pe/cndg/index.htm>. En esta dirección los usuarios

podrán disponer de toda la información sísmica existente en el CNDG, desde formas de onda hasta informes técnicos – científicos.

Asimismo, en este estudio se propone dos relaciones Atenuación – Intensidad para sismos ocurridos en Perú a partir de la metodología propuesta por Ambraseys (1985). Además, se ha definido que los sismos con origen en el proceso de subducción obedecen a un modelo de atenuación de la intensidad de tipo *elipsoidal* con eje mayor paralelo a la línea de costa; mientras que, los sismos con origen en la deformación continental desarrollan una atenuación de tipo *circular*. Estas relaciones permitirán obtener valores de intensidad teóricos para sismos ocurridos en Perú, con sólo conocer de la distancia epicentral y la magnitud M_s del sismo.



CAPITULO 1

INTRODUCCION

En la actualidad, haciendo uso de la Internet cualquier usuario que se encuentre en línea puede consultar y obtener cualquier tipo de información disponible en ella. Dicha facilidad con que la información se difunde mundialmente se da en el ámbito comercial, político, periodístico, pero principalmente en el campo de la investigación y la ciencia. En tal sentido, haciendo uso de un servidor de Internet se puede dar a conocer noticias, hacer de conocimiento cualquier tipo de servicio al público, poner anuncios de compra y venta de un producto o publicar y dar a conocer los resultados de la investigación hecha dentro de alguna especialidad, con la seguridad de que toda la información llegará a los usuarios que la soliciten con suma facilidad y en muy poco tiempo.

Así, el uso de la Internet como un mercado abierto requiere que el trámite en línea de la información que se pone a disposición se realice con cierto grado de transparencia y orden, pues al cliente o interesado en dicha información requiere un trato cómodo y simple al momento de consultar y extraer el producto o información que le interesa. Para tal objetivo, la tecnología ha permitido desarrollar sofisticadas Bases de Datos empleando las herramientas y aplicaciones proporcionadas por un *Software* de alta versatilidad que permitirá realizar dicha labor administrativa sin importar que tan grande sea el volumen de los datos.

Un Software de Base de Datos consiste en un conjunto de programas, aplicaciones y herramientas adicionales que permiten administrar un grupo importante de datos. Dichas programas, aplicaciones y herramientas van ha permitir a un *Administrador de Base de Datos* o *DBA* (Database Administrator) crear objetos o estructuras en los cuales se pueda almacenar dichos datos. Asimismo, dichas herramientas y aplicaciones permitirán manipular dichos datos con la intensidad de

realizar labores de selección, actualización, inserción, borrado u otras operaciones que se requiera hacer con los datos; además de generar mecanismos de consulta de fácil aplicación. Estas operaciones de consulta se realizan tanto en la PC (Personal Computer) donde radica la base de datos como desde una PC cualquiera conectada a la PC de la base de datos empleando los diferentes protocolos de comunicación (Telnet, TCP/IP, HTTP). Por otro lado, si para el usuario es importante la fácil adquisición de los datos dentro de un trámite de información por la Internet, para el Administrador de la base de datos la seguridad de estos durante dicho trámite es también importante a fin de evitar que los datos puedan ser distorsionados conscientemente o intencionalmente durante el cumplimiento de las solicitudes y las transferencias de datos. En tal caso, si llegara a ocurrir una alteración de la información en la base de datos, esta perdería la “Confiabilidad de la Información”, la cual se considera como requisito primario dentro de las características de una base de datos. Para cumplir tales objetivos, el módulo básico del “Software de Administración de Base de Datos Oracle8i” o simplemente Oracle8i desarrolla todas las funciones antes mencionadas. El Oracle8i es un software de base de datos cuya tecnología esta definida como “Objeto – Relacional”, esto quiere decir que sus bases de datos sacan provecho tanto de la tecnología “Relacional” como de la tecnología “Orientada a Objetos” en el manejo de la información. Esta capacidad le da muchas ventajas a Oracle para desarrollar bases de datos muy potentes y seguras; además, le da al DBA la posibilidad de almacenar y manejar cualquier tipo de información o de datos; es decir, esta hecha para trabajar con datos estructurados y para datos no - estructurados.

El objetivo de este trabajo principalmente radica en la implementación de una base de datos haciendo uso del Oracle y de su tecnología Objeto – Relacional, empleando la información sísmica generada por el Centro Nacional de Datos Geofísicos del Instituto Geofísico del Perú (CNDG - IGP), durante el proceso de investigación de los sismos en Perú. Además, en este trabajo se desarrolla un estudio que consiste en la obtención de la relación Atenuación – Intensidad para los sismos de Perú empleando la metodología de Ambraseys (1985) y a partir de dichas relaciones determinar modelos de atenuación para dichos sismos.

1.1.- Por qué una Base de Datos Sísmicos

El desarrollo de la investigación requiere de una buena administración de la información, la cual si no se logra, puede llegar a entorpecer y limitar la labor científica. Por tal razón, la tecnología a través de la informática ha permitido brindar formas y mecanismos de administración de cualquier tipo de información de manera adecuada y principalmente ordenada, sin importar que tanto sea el volumen ni el tipo de información que se manipule. De tal forma que, como la investigación sismológica requiere de toda la información posible que proveniente del monitoreo de los sismos, del procesamiento de los datos sísmicos, de la observación y análisis de los diferentes fenómenos producidos antes, durante y después de la ocurrencia de los sismos fuertes y destructores. Esta debería de estar almacenada y administrada de manera coherente. En conjunto, toda esta información supera fácilmente varios millares de datos para una determinada área de estudio. Asimismo, la información que se genera aquí no solo es de importancia para los investigadores locales, sino también para los investigadores de las diferentes partes del mundo donde se estudia y se lleva a cabo la investigación de estos fenómenos naturales. Por estas razones anteriormente descritas es necesario que dicha información sea almacenada de forma ordenada en un BANCO DE DATOS o Base de Datos utilizando un software especial como el Oracle. Este software representa uno de las más importantes opciones en la actualidad para el desarrollo de bases de datos seguras y versátiles.

Otro punto importante que justifique el desarrollar una base de datos sísmicos para el Perú, es la que se refiere al alto potencial sísmico de Perú. El Perú se ubica en una de las regiones de mayor potencial sísmico en el mundo (Tavera y Buforn, 2001 y Bernal et al, 2002) y su sismicidad es registrada por una red sísmica regional de manera permanente. Esta información registrada, al ser procesada, genera la información se encuentra en diferentes tipos de archivos con formatos de diferentes tipos. El Instituto Geofísico del Perú, a través de la Dirección del Centro Nacional de Datos Geofísicos - Sismología, (CNDG – Sismología) a venido realizando dicha labor, con el almacenamiento de la información sísmica en los diferentes módulos (Disco Duro, CD

– ROM, cintas magnéticas) pero sin una relación lógica; es decir, no presentan una estructura “Relacional” entre los objetos que contienen la información y esta propensa a tener errores de formato o estar repetidos. Dicha información estaba siendo presentada en la página Web del CNDG - IGP sin que esta presente medidas de seguridad en cuanto a la restricción de consulta provocando esto que se limite la difusión de la información existente en el área de Sismología. En el presente estudio se pretende solucionar estos problemas haciendo uso de la tecnología brindada por el Software de Base de Datos Oracle8i. Además, el modelo de base de datos desarrollado para la información sísmica podrá ser aplicado en otros campos de la ciencia en los cuales se maneje grandes volúmenes de datos y que tengan tanto información estructurada como también información no – estructurada.

El desarrollo de este estudio se encuentra organizado por capítulos los cuales siguen una secuencia que permitirá dar a conocer “paso a paso” la forma como se ha desarrollado la implementación de la Base de Datos Sísmicos del CNDG empleando para esto el Software de Base de Datos “Oracle”. Asimismo, se dará a conocer la metodología que permitió determinar la relación Atenuación – Intensidad para los sismos y definir sus modelos de atenuación.

Cada capítulo trata los siguientes aspectos: el Capítulo 2 trata la importancia de contar con un Banco de Datos, su elaboración y su aprovechamiento visto desde la óptica de la sismología; además, describe toda la información que produce el área de sismología del CNDG - IGP durante el desarrollo de los respectivos estudios sobre los sismos ocurridos en Perú. En el Capítulo 3, se introduce los conocimientos y conceptos del Software de Base de Datos Oracle, sus características, ventajas, herramientas y aplicaciones; así como, la tecnología Objeto – Relacional y las ventajas que brinda para el desarrollo de bases de datos. En el Capítulo 4, se detalla la estructura creada para la Base de Datos Sísmicos del CNDG - Sismología del Instituto Geofísico del Perú con la descripción de cada uno de los objetos que la conforma (tablas, listas, etc), no sin antes hacer una introducción de los conceptos del SQL (Structured Query Language) que van a permitir crear dichos objetos y manipular su información. En el Capítulo 5, se

describe y explica al detalle los métodos de almacenamiento de la información haciendo uso de las herramientas que brinda el “Oracle” y de algunos programas creados por el Administrador de la Base de Datos Sísmicos para optimizar dicha labor. En el Capítulo 6, se detalla todo lo referente a la estructura de la consulta a la base de datos por Internet utilizando herramientas del Oracle y programas ejecutados dentro del servidor de Base de datos. En Capítulo 7 se describe la forma como se ha determinado la relación de Atenuación – Intensidad para los sismos de Perú empleando para ello los parámetros sísmicos y los mapas de intensidades de algunos sismos importantes ocurridos en el Perú. En el Capítulo 8 se detallan las conclusiones y recomendaciones a los cuales se ha llegado al concluir el presente estudio..

1.2.- Área de Estudio

El área de estudio que comprende este trabajo se remite a toda la región en la cual se producen eventos sísmicos que afectan al territorio peruano y que sean registrados por la Red Sísmica Nacional (RSN). Pero además, como la sismología es una ciencia de estudio global y dada la sensibilidad de los equipos en la actualidad, los eventos que ocurren en el mundo y que sean registrados por la RSN, también forman parte de la información que contendrá la Base de Datos; por lo tanto, indirectamente el área de estudio estaría representada por todo el mundo.

1.3.- Objetivos del Presente Estudio

Dentro de los principales objetivos que se pretende alcanzar en este estudio se listan los siguientes:

- Desarrollar un procedimiento adecuado para la estructuración y organización de un Banco de Datos, que pueda ser utilizado con los datos producidos en cualquier área de investigación.
- Evaluar la importancia de estructurar y desarrollar un Banco de Datos Sísmicos para

el Perú y las ventajas que ofrece.

- Entender, comprender y manejar las herramientas y aplicaciones que ofrece el módulo básico del Software de Base de Datos Oracle8i y su tecnología “Objeto – Relacional”.
- Organizar la información sísmica del Instituto Geofísico del Perú en un Banco de Datos a fin de que esta pueda ser administrada y puesta a disposición de los usuarios, apoyándose en la tecnología y herramientas del módulo básico del Software de Base de Datos Oracle8i.
- Diseñar una estructura de almacenamiento de datos sobre un esquema “Relacional” entre los objetos que la conforman (tablas) empleando las herramientas que ofrece Oracle.
- Realizar la inserción de la información a la base de datos de manera fácil, rápida y segura empleando las herramientas que ofrece Oracle y aplicaciones creadas adicionalmente.
- Desarrollar un Sistema de Consulta Dinámica de la información haciendo uso de un diseño orientado sobre la Arquitectura de Internet empleando programas CGI creados y ejecutados dentro del servidor.
- Estructurar las pautas necesarias a seguir para la administración de la Base de Datos Sísmicos del CNDG - Sismología a fin de que pueda ser realizada posteriormente por un Administrador o personal a cargo.
- Obtener una relación Atenuación – Intensidad para sismos ocurridos en Perú con origen en el proceso de subducción y en los procesos de deformación de continental.

- Lograr mediante una sustentación oral del desarrollo y resultados de este estudio, el grado de Ingeniero Geofísico en la Universidad Nacional de San Agustín de Arequipa.



CAPITULO 2

CARACTERISTICAS DEL BANCO DE DATOS SÍSMICO

2.1.- Introducción

En este capítulo, se dará a conocer la importancia de estructurar un Banco de Datos Sísmicos para Perú, teniendo en cuenta el gran potencial sísmico que posee dicha región y por consiguiente la gran cantidad de información sísmica que se genera. Asimismo, se realiza un análisis de las principales fuentes de información sísmica y una descripción del tipo de información que se genera en cada una de ellas a partir del monitoreo sísmico empleando una red de estaciones sísmicas; el procesamiento de datos y la obtención de resultados en diferentes archivos con diferentes formatos (*.dat, *.jpg, *.sis, *.sac, etc.). La información sísmica utilizada es la generada en el área de sismología del Centro Nacional de Datos Geofísicos del Instituto Geofísico del Perú.

2.2.- Sismicidad de Perú

El Perú está situado en una de las regiones con mayor potencial sísmico en el mundo (Tavera y Bufo, 1998; Tavera y Bufo, 2001 y Bernal et al, 2002) y dicha sismicidad está relacionada principalmente con el proceso de interacción de las placas Nazca y Sudamericana, ambas presentes en el borde oeste del territorio peruano, siendo este el motor de origen y fuente principal de muchos de los grandes sismos que han ocurrido a lo largo de la historia sísmica de Perú. Una segunda fuente generadora de sismos se ubica al interior del continente y es atribuida a las deformaciones corticales presentes de Norte a Sur sobre la Cordillera Andina y a lo largo de la denominada Zona Subandina (Bernal, 2002).

La Sismicidad de Perú puede ser dividida en Sismicidad Histórica e Instrumental. La Sismicidad Histórica considera a los sismos ocurridos en Perú entre

1400 y 1959, mientras que la Sismicidad Instrumental considera los sismos suscitados a partir de 1960 hasta la actualidad, debido a que en el año de 1960 se instaló una red sísmica mundial de monitoreo permanente conocida como “*World Wide Standardized Seismograph Network*” (WWSSN), la cual contribuyó al cálculo de los parámetros que definen las características de los sismos (tamaño y ubicación). A fin de reconocer el gran volumen de sismos que se generan en Perú se realizará una descripción de cada uno de estos periodos de información, para luego describir brevemente las características de los principales sismos de magnitud elevada que han ocurrido en el territorio peruano. Este análisis de la sismicidad permitirá evaluar mejor el potencial sísmico que afecta al Perú.

2.2.1.- Sismicidad Histórica

La información histórica de la sismicidad de Perú corresponde al periodo comprendido entre 1500 y 1959 aproximadamente. Esta información data del tiempo de la conquista y colonización del país, y se encuentra esparcida en diferentes obras inéditas, manuscritos, crónicas, narraciones, informes administrativos y otros, los cuales han sido hechos por parte de clérigos y gobernantes de aquellos años. Gran parte de esta información ha sido recolectada por Polo (1904), Barriga (1939), Silgado (1978), y Dorbath et al, (1990). En general, estos autores indican que el sismo más antiguo para el cual se dispone de información, corresponde al año de 1513.

A continuación, se describe algunos puntos de vista de los investigadores más importantes que se ocuparon de la sismicidad histórica de Perú.

- T. Polo (1904). El historiador Toribio Polo remarca la existencia de mas de 2500 sismos desde la conquista hasta fines del siglo XIX y advirtió que por causas diversas no se anotaron muchos de los ocurridos entre los años de 1600 a 1700. Asimismo, indica que los grandes sismos produjeron cuantiosos daños materiales debido a que las construcciones de aquellos años eran inadecuadas para resistir los movimientos violentos del suelo.

- P. Barriga (1939). El Padre Barriga describe las características de todos los sismos ocurridos en la región del Sur del Perú hasta 1555. Esta información sirvió de base para que otros investigadores retomaran el trabajo de recopilar la información de los sismos de mayor magnitud.
- E. Silgado (1978). El trabajo realizado por este autor se constituye como la recopilación mas completa de los sismos importantes ocurridos en el Perú entre 1513 y 1974. La descripción de las características de los sismos es detallada y bastante clara, siendo muchos de ellos estudiados por el mismo autor consignando los valores de intensidad local y regional. Asimismo, el autor estima la magnitud de los sismos a partir del área de intensidad máxima a fin de compararlos con sismos mas recientes.
- Dorbath et al (1990). Estos autores realizan una revisión detallada de toda la información existente sobre sismos históricos a fin de correlacionar las áreas de intensidad máxima y las longitudes de ruptura de los grandes sismos a fin de estimar su magnitud. Los resultados obtenidos por estos autores, son comparados con los valores obtenidos en otras investigaciones aplicando diferentes métodos. Asimismo, los autores sugieren que los grandes sismos de Perú tienen un periodo de recurrencia del orden de una centuria (100 años).

Todos los sismos ocurridos durante el periodo histórico no tienen determinada con precisión la profundidad focal; sin embargo, debido a la ubicación geográfica (entre la fosa y la línea de costa) y daños observados en superficie, estos pueden ser considerados en su mayoría como superficiales (Bernal, 2002).

En la Figura 2.1 se presentan los sismos con intensidades iguales o mayores al valor de intensidad VIII en la escala de Mercalli Modificada y en ella se observa que todos se ubican a lo largo de la línea de costa, en el Centro y Sur del país (Silgado, 1978; Dorbath et al., 1990). La mayoría de estos sismos generaron maremotos y produjeron daños a lo largo de la costa Oeste de Sudamérica desde 9°S hasta 37°S y en

muchos lugares alrededor del Pacífico (Montessus de Ballore, 1911; Hatori, 1968). Para el interior del país, solo se dispone de información de tres sismos importantes, el del Cuzco de 1650, Huaraz de 1946 y el de Satipo de 1947 (Tavera y Bufo, 1998). Esto significa que existe la posibilidad de que hayan ocurrido sismos de gran magnitud que no han sido reportados debido a la falta de comunicación o de informes elaborados por los habitantes de dichas regiones. La región central da información sobre sismos correspondientes a las ciudades mas pobladas o importantes.

Según Silgado (1978) y Dorbath et al., (1990), en la costa central del Perú, los sismos mayores son los de 1586 (el primer gran sismo del que se obtuvo documentación histórica), 1687 y el de 1746 que destruyó completamente la ciudad de Lima y generó un maremoto con olas de 15 a 20 m de altitud. Durante el periodo 1513-1959, Lima fue destruida sucesivamente por un total de 15 sismos (Silgado, 1978). En la región Sur, los sismos mas importantes son los de 1604, 1784 y 1868, este ultimo es el mejor documentado y descrito en detalle por Montessus de Ballore (1911) y Vargas (1922). Estos sismos destruyeron las principales ciudades del Sur del Perú y Norte de Chile. El sismo de 1868 fue sentido desde Guayaquil (Ecuador) hasta Concepción (Chile) generando un importante maremoto con olas de 15 m de altitud (Silgado, 1978; Dorbath et al., 1990). En la región Andina, el único sismo histórico documentado es el ocurrido en 1650, que destruyó la ciudad del Cuzco y fue sentido en Lima, Arequipa y La Paz. En la zona Central, el sismo ocurrido en 1947 afectó un área máxima de 4000 km² y fue sentido casi en todo el Perú. Para este período (1513-1920), no existe información sobre sismos ocurridos en la zona Andina y Subandina del Norte y Centro del Perú, pero en la actualidad se ha comprobado que estas zonas son sísmicamente muy activas (Tavera y Bufo, 1998).



Figura 2.1. Mapa del Perú con los sismos históricos ocurridos durante el periodo 1513 – 1959 (Silgado, 1978; Dorbath et al., 1990).

2.2.2.- Sismicidad Instrumental

La sismicidad instrumental considera a los sismos ocurridos a nivel mundial a partir de 1960, debido a que en ese año surge la Red Sísmica Mundial (WWSSN). Un mapa de sismicidad de Perú, para el periodo comprendido entre 1960 a 1995 elaborado por el Instituto Geofísico del Perú, se aprecia en la Figura 2.2 y en la cual los sismos han sido diferenciados por su profundidad en sismos de foco superficial ($h < 60$ Km,

círculos rojos), intermedios ($60 < h < 350$ Km, círculos verdes) y profundos, ($h > 350$ Km, círculos azules). Los sismos con foco superficial se ubican en la zona oceánica y su distribución es paralela a la línea de costa (Stauder, 1975; Barazangi y Isacks, 1976), presentándose sismos de magnitud elevada con relativa frecuencia (magnitud ≥ 7.0); tal como los ocurridos en 1960 ($M_s = 7.5$), 1966 ($M_s = 8.0$), 1970 ($M_s = 7.8$), 1974 ($M_s = 7.6$), 1996 ($M_w = 7.7$) y 2001 ($M_w = 8.4$). De estos sismos, el mas importante es el que ocurrió el 31 de mayo de 1970, conocido como uno de los mas catastróficos en el mundo por haber producido la muerte de 67000 personas, 250000 desaparecidos y 180000 heridos (Plafker et al., 1971; Abe, 1972; Silgado, 1978). Otro grupo importante de sismos son los producidos por la subsidencia del escudo brasileño bajo la Cordillera Andina, estando la mayor parte de estos localizados en la zona de transición entre la Cordillera Oriental y el margen occidental de la zona Subandina (entre 3°S y 13°S). En la zona del Altiplano también existen sismos superficiales pero son menos numerosos y más dispersos. En general, los sismos que ocurren en esta zona presentan magnitudes moderadas (≤ 6.5) como los ocurridos en 1969 ($m_b = 5.6$ y $m_b = 5.8$) en Huancayo, 1990 ($m_b = 6.1$) en Moyobamba y 1991 ($m_b = 6.5$) en Rioja. Asimismo, se observa la presencia de zonas de bajo índice de sismicidad entre la línea de costa y la Cordillera Occidental (4°S a 13°S), mostrando que en estas zonas el régimen de deformación cortical probablemente es menor o que existen períodos mayores de recurrencia para sismos de magnitud elevada (Tavera y Buforn, 1998).

La sismicidad con foco intermedio ($60 < h < 350$) se distribuye de manera irregular, pudiendo definirse tres zonas. La primera se distribuye paralela a la línea de costa por debajo de los 9°S y se caracteriza por la frecuente presencia de sismos de magnitud elevada, siendo los más importantes los ocurridos en 1958 ($M_s = 6.5$), 1968 ($M_s = 6.0$), 1974 ($M_s = 6.4$), 1979 ($M_s = 6.9$), 1982 ($m_b = 6.1$) y 1993 ($m_b = 6.0$) (Tavera y Buforn, 1998). Un segundo grupo de sismos se localiza en el interior del continente (zona Norte y Centro) a lo largo de la Cordillera Oriental y la zona Subandina. Estos sismos siguen una línea N-S y raramente producen daños. El tercer y más importante grupo se localiza en la región Sur del Perú, siendo esta región la de mayor índice de sismicidad. En los dos primeros grupos los sismos alcanzan

profundidades del orden de 100 a 120 km; mientras que, en el tercer grupo la profundidad máxima es de 300 km. La actividad sísmica más profunda ($h > 350$) se localiza en la región Centro y Sur del Llano Amazónico. Esta actividad, es mayor en la región central (6°S a 11°S) y se alinea en dirección N-S cubriendo un área de aproximadamente 500 km^2 de longitud (borde Perú - Brasil); mientras que, en la región Sur es menos numerosa y más dispersa cerca del en el borde Perú - Bolivia (Tavera y Bufo, 1998).

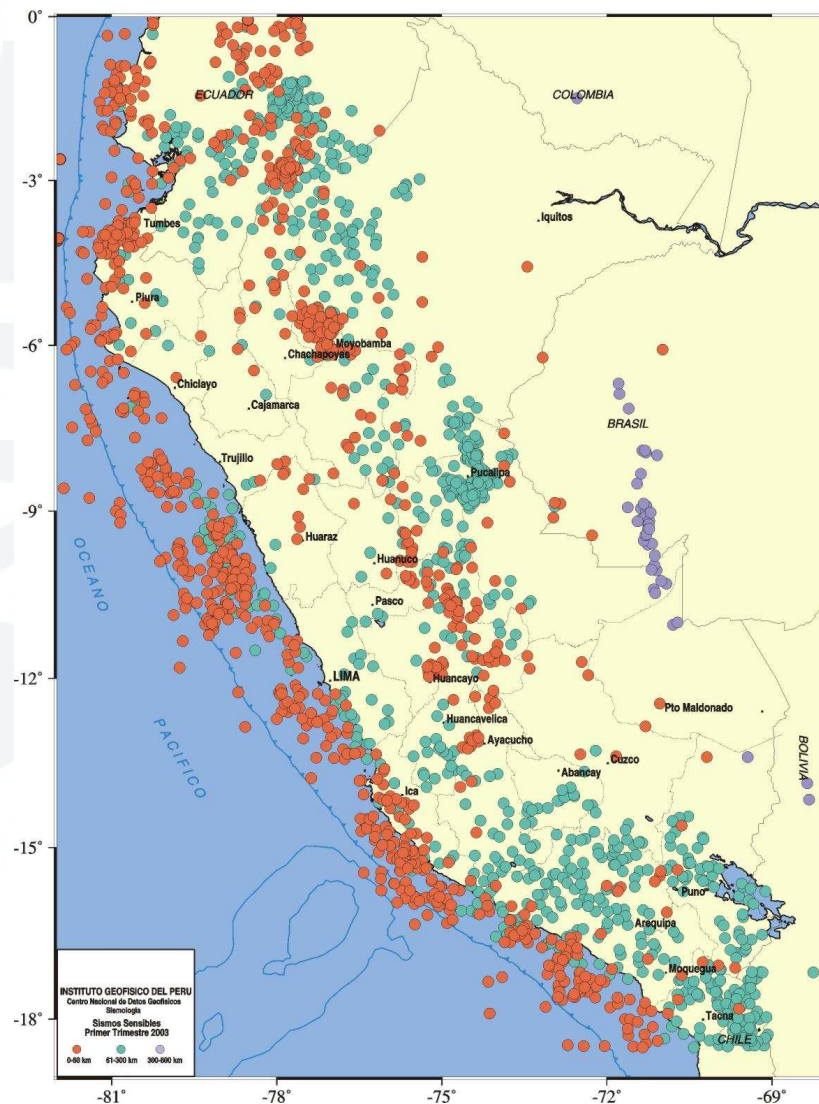


Figura 2.2. Sismicidad Instrumental de Perú para el periodo comprendido entre 1960 - 1995. a) sismos con foco superficial en rojo, con foco superficial en verde y de foco profundo en azul.

2.2.3.- Principales Sismos Destruyentes

La historia sísmica de Perú data del año de 1500, fecha a partir de la cual el mayor número de sismos de magnitud elevada han ocurrido frente a la línea de costa llegando a afectar considerablemente a las ciudades que se ubican de norte a sur en el borde oeste de Perú. En el interior del continente, el número de sismos de magnitud elevada es menor pero el grado de destrucción algunas veces es similar a lo observado para el grupo anterior. En la Tabla 2.1 se presenta a los sismos de mayor tamaño ocurridos en el Perú y que ocasionaron gran destrucción (Bernal, 2002). En esta Tabla se puede ver que desde 1582 hasta el año 2001 se tienen contabilizados 31 sismos con magnitud superior a 5,6 y con intensidad máxima mayor a VI en la escala de Mercalli Modificada. Así mismo se tienen sismos con valores de intensidad muy elevados (IX y X en MM), tanto en el interior del continente como en la línea de costa. De esta relación de sismos se puede destacar los sismos de 1604 y 1868 en el sur del país por ser muy destructivos y provocar tsunamis con olas mayores a 15 metros, el de Ancash (1970) el cual provocó el mayor número de muertos en la historia sísmica de Perú, el de Cuzco (1650) y el de Arequipa del 2001, el cual fue considerado como uno de los más fuertes ocurridos en el Perú. Esta frecuencia de sismos da una idea del elevado potencial sísmico que tiene el Perú y de la importancia de integrar toda esta información en una base de datos o Banco de Datos, al igual que la información sísmica registrada y procesada de manera continua en el CNDG – IGP.

Tabla 2.1. Relación de los sismos de mayor tamaño ocurridos en Perú entre 1500 hasta la actualidad. MM, escala Modificada de Mercalli y (*), sismos que produjeron Tsunami.

Fecha (dd/mm/aaaa)	Lat. (°)	Long. (°)	Prof. (km)	Mag. (Ms/mb)	I max MM	
22/01/1582	-16,6	-71,6	--	7,5	X*	La ciudad de Arequipa quedó en ruinas. Cayeron alrededor de 300 casas y perecieron
09/07/1586	-12,1	-77	--	7,7	IX*	Los principales edificios de la ciudad de Lima se vinieron al suelo. Perecieron mas de 22 personas.
24/11/1604	-17,8	-70,9	--	7	IX*	Las ciudades de Arequipa, Moquegua, Tacna y Arica quedaron en ruinas. Perecieron mas de 23 personas.
14/02/1619	-7,9	-79	--	7,7	IX	La ciudad de Trujillo quedo en ruinas. Perecieron 350 personas.
31/03/1650	-13,5	-71,7	--	7,2	X	En la ciudad del Cuzco cayeron templos y edificios, Se observó agrietamiento en el terreno.
13/11/1655	-12,3	-77,6	--	7,7	IX	El sismo derribó muchas casas y edificios en Lima.
12/05/1664	-14	-76,8	--	7,8	X	La ciudad de Ica quedo destruida y perecieron 300 personas.
16/06/1678	-12,3	-77,8	--	7,7	IX*	Muchas edificaciones de Lima quedaron en ruinas. En Lima, Callao y Chancay se contabilizó 10 muertos.
20/09/1687	-13	-77,5	--	8,2	IX*	Lima fue destruida por dos sismos. El primero desarticulo y sacudió los edificios y el segundo mas prolongado en duración, los acabó de arruinar ocasionando cerca de 100 muertos.
29/10/1746	-11,9	-77,2	--	8,4	X*	Lima y Callao quedaron destruidas. Un Tsunami terminó por destruir el Callao. De 3 000 casas existentes sólo quedaron 25 en pie, donde perecieron 1141 personas.
13/05/1784	-16,5	-72	--	7,8	X*	La ciudad de Arequipa sufrió la ruina de edificios y viviendas. Perecieron 54 personas.
10/07/1821	-16	-72,9	--	18,5	VIII	El sismo ocasiona grandes daños en la provincia de Arequipa donde perecen un total de 162 personas.
13/08/1868	-18,3	-70,6	--	8,6	X*	Sismo que causó destrucción en Arequipa, Tacna, Moquegua, Arica e Iquique. El Tsunami que siguió al sismo presentó olas de 16 m de altura. Perecieron aproximadamente 180 personas en todo el sur.

Fecha (dd/mm/aaaa)	Lat. (°)	Long. (°)	Prof. (km)	Mag. (Ms/mb)	I max MM	
24/05/1940	-11,2	-77,7	--	6,6	VIII*	Ruina parcial de la ciudad de Lima. Importantes pérdidas económicas. Perekieron 179 personas y 3 500 heridos.
24/08/1942	-15,5	-74,7	--	6,7	IX*	Sismo que destruyó los alrededores de Ica y Arequipa. Se produjo un Tsumani con olas que alcanzaron 3 m de altura. Perekieron 30 personas y 25 heridos por diversas causas.
10/11/1946	-8,4	-77,8	--	6,9	X	Ruina parcial en las localidades del departamento de Ancash. Perekieron 1396 personas a pesar de la escasa densidad de población en esta zona.
01/11/1947	-11,2	-75	--	6,2	IX	Sismo que afecto la zona central de Perú. En Satipo y alrededores se contabilizó 200 muertos.
21/05/1950	-14,4	-72,1	15	5,6	VII	La ciudad del Cuzco sufrió daños en el 50% de sus edificaciones. Perekieron 120 personas y hubo 275 heridos.
12/12/1953	-3,8	-80,5	--	6,7	VIII	Sismo que afecto seriamente a la ciudad de Tumbes.
17/10/1966	-10,72	-78,6	38	6,4	VIII	Sismo que destruyó paralelamente a Lima y Callao. Se contabilizó 100 muertos y daños materiales que ascendieron a mil millones de soles.
01/10/1969	-11,8	-75,2	14	5,8		Sismo que afecto a la ciudad de Huancayo en el departamento de Junín.
31/05/1970	-9,3	-78,8	64	6,4	VIII	Destrucción parcial en el departamento de Ancash. El número de víctimas fue de 50000, 20000 desaparecidos y 150 heridos. Mayor destrucción se produjo en Yungay debido al desprendimiento de bloques de hielo del nevado Huascaran.
10/12/1970	-4,1	-80,6	20	6,3	VIII	Este sismo ocasionó gran destrucción en la ciudad de Tumbes.
03/10/1974	-12,2	-77,5	21	6,2	VIII	El sismo ocasionó mayor destrucción en la ciudad de Lima. Perekieron 78 personas, 2500 heridos y la pérdida material fue estimada en 2700 millones de soles.
19/02/1979	-16,5	-72,5	52	6,2	VII	Sismo que afectó el extremo Oeste del departamento de Arequipa. Se contabilizó 215 heridos.
13/02/1981	-15,6	-74,4	62	5,6	VI	Sismo que afectó al departamento de Ayacucho.

<i>Fecha (dd/mm/aaaa)</i>	<i>Lat. (°)</i>	<i>Long. (°)</i>	<i>Prof. (km)</i>	<i>Mag. (Ms/mb)</i>	<i>I max MM</i>	
05/04/1986	-13,4	-71,9	57	5,4	VII	Dstrucción de la ciudad del Cuzco originando la muerte de 7 personas, 80 heridos y aproximadamente 13000 damnificados.
30/05/1990	-6,1	-77,2	24	6,1	VI	Sismo que afecto a la ciudad de Moyobamba en donde perecieron 135 personas, mas de 800 heridos. Mayor destrucción se produjo en Rioja y Soritor.
05/04/1991	-5,9	-77	19	6,4	VII	Nuevamente Moyobamba fue destruida por un sismo en el cual perecieron 135 personas, 252 heridos y daños de consideración en 8000 viviendas.
12/11/1996	-15,3	-76,44	14	6,5	VII	Dstrucción en la zona urbana de la ciudad de Ica. Perecieron 17 personas, 1500 heridos y 100000 damnificados.
23/06/2001	-16,2	-73,75	38	6,9/7,2	VIII*	Todo el sur del Perú fue afectado por este sismo. Mas de 217400 personas sufrieron sus efectos, 17580 casas fueron destruidas y perecieron 64 personas. El terremoto fue seguido por un tsunami con olas de 7 - 8 metros de altura en la ciudad de Camaná.

2.3.- Información Sísmica

Tal como se describió en el punto anterior, el Perú es un laboratorio natural de sismología del cual se obtiene información sísmica de manera continua. La obtención de esta información parte del monitoreo permanente de los sismos empleando una red sísmica regional. Toda la información proveniente de dicha red sísmica es analizada y procesada por programas especializados a fin de identificar principalmente los tiempos de arribo de las ondas sísmicas así como su amplitud periodo y duración del sismo. Esta información es almacenada en archivos, los cuales representan para la Base de Datos Sísmicos un grupo o fuente importante de información. Seguidamente, la información de dichos archivos es usada para calcular los parámetros finales de ubicación y tamaño de los sismos, para los cuales se emplea programas de procesamiento sísmico (EPI, EPIGRAF). La información resultante también es almacenada en archivos y representa otra fuente de información. Así, la información contenida en estos archivos es sintetizada en documentos conocidos como Catálogos Sísmicos, los cuales contiene la información resumida de toda la sismicidad, sea esta Histórica o Instrumental. Estos Catálogos Sísmicos representan otra fuente de información sísmica. Una vez que se tiene la información necesaria de los sismos, se empiezan a hacer los análisis y los estudios respectivos los cuales traen como resultado la elaboración de informes, gráficos, reportes y/o diferentes resultados que derivan de la interpretación de la información obtenida. Finalmente, esta información también es una fuente importante para la Base de Datos Sísmicos. A continuación se describirá y tratará estas fuentes de información y de los datos que proporcionan cada una de ellas.

2.3.1.- Red Sísmica Nacional

La principal y más importante herramienta de obtención de datos sísmicos en Perú y de donde deriva toda la información sísmica que se analiza y almacena, es la Red Sísmica Nacional del Instituto Geofísico del Perú (RSN - IGP). En general, una red sísmica esta constituida por un conjunto de estaciones, que de acuerdo con las dimensiones del área de estudio, puede ser denominada como local, regional o mundial.

Las redes sísmicas *locales* consideran una distribución de estaciones en áreas pequeñas en donde las distancias entre estaciones es mínima y la transmisión de datos en general se realiza de manera directa a un centro de adquisición de datos. La finalidad de estas redes es obtener información microsísmica. Las redes sísmicas *regionales*, consideran áreas mayores de monitoreo, y por lo tanto, la distancia entre estaciones es mayor y la transmisión de la data se realiza por telemetría, línea telefónica o satélite. La finalidad de estas redes es realizar un monitoreo sísmico regional siendo este el objetivo general de las redes sísmicas de cada país. Finalmente, una red *mundial* esta constituida por estaciones distribuidas en todo el mundo a grandes distancias y la transmisión de los datos se hace por satélite, siendo su principal objetivo el monitorear la sismicidad a escala global.

La Red Sísmica de Perú o Red Sísmica Nacional (RSN) es de tipo regional, siendo la primera estación instalada en 1907 en la ciudad de Lima. Posteriormente en 1931, se instaló una estación en Huancayo, la cual estaba equipada con 3 sismómetros de periodo corto y 3 de periodo largo. En 1962, estaciones similares se instalaron en Ñaña, en el departamento de Lima y en Arequipa (Tavera, 2001). Estas estaciones fueron integradas a la Red Sísmica Mundial “World Wide Standardized Seismograph Network”. Sin embargo, se puede considerar que la Red Sísmica Nacional (RSN) tuvo sus inicios en la década de los ochenta, fecha en la que estuvo constituida por 20 estaciones de periodo corto, instaladas cerca de la costa en la región Norte, Centro y Sur del Perú. A partir del año 1996, la RSN inicia sus implementación con estaciones de banda ancha y en la actualidad esta conformada por 31 estaciones, de las cuales, 20 son de periodo corto con transmisión de datos por telemetría y 11 de banda ancha con acceso remoto vía interrogación telefónica. En la Figura 2.3 se muestra un mapa con las estaciones que conforman la RSN.

La Red Sísmica Nacional genera un volumen importante de información de diferentes tipos y diversos formatos. Los archivos no solo contienen la información digital correspondiente a las formas de onda de los eventos registrados, si no que también brinda las características instrumentales de los equipos que conforman dicha

red de monitoreo, la que de hecho es muy importante. A continuación, se tratará los tipos de datos generados a partir de esta fuente de información y algunas características de estos.

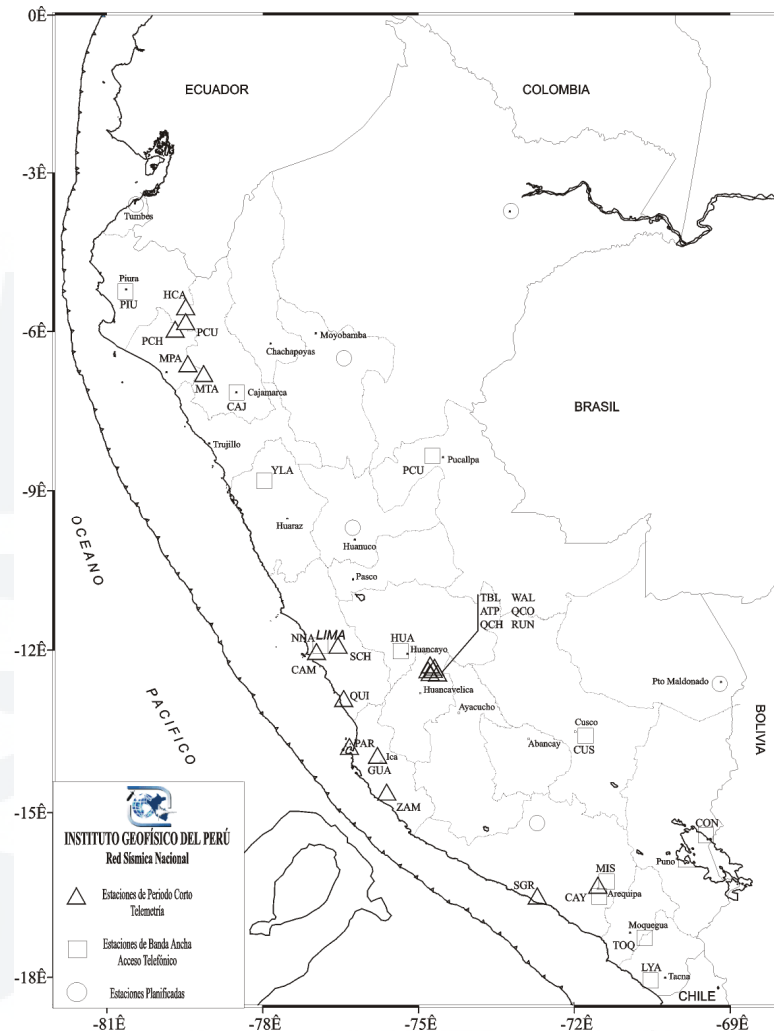


Figura 2.3. Mapa del Perú con las Estaciones que integran la Red Sísmica Nacional.

2.3.1.1.- Características Instrumentales de las Estaciones de la RSN

Al momento de realizar cualquier estudio de sismología que haga uso de formas de onda, es necesario conocer las características de los instrumentos que registran dichos sismos. La señal sísmica va a ser registrada por el sensor, es convertida a pulsos

eléctricos cuyos movimientos son amplificados, filtrados y modulados antes de ser almacenados en formato binario. A fin de reconstruir la señal original; es decir, proporcionar el movimiento del suelo, es necesario eliminar todo el efecto del instrumento, de ahí la necesidad de conocer las constantes instrumentales de cada estación sísmica. Dentro de las características instrumentales de las estaciones sísmicas se tiene: Nombre de la estación, Código Nacional de la estación, ubicación Geográfica de la estación (Latitud, Longitud, Altitud), Tipo de Sensor de la estación, ancho de banda del sensor de la estación, rango dinámico y ganancia de la estación.

2.3.1.2.- Archivos generados por las Estaciones de Banda Ancha

Las estaciones de banda ancha de la RSN generan archivos que contienen las formas de onda de los eventos sísmicos registrados por estas, los cuales están en formato binario. Esta información es grabada inicialmente en unidades magnéticas, y luego en CD - ROM en un formato de registro inicial denominado SUD (*.sud) y posteriormente son convertidos a un formato estándar que pueda ser reconocido por otros programas de procesamiento de sismos (ejemplo, formato SAC *.sac). Las estaciones de Banda Ancha registran los sismos en tres componentes: dos horizontales y una vertical, por tal razón, se generan tres archivos binarios. El nombre de estos archivos está dado por la fecha y hora en que ocurrió el sismo, el nombre de la estación y un código que determina la componente de registro. Un ejemplo del formato que tienen los nombres de estos archivos es: 993580527.con.bhn, 993580527.con.bhe, 993580527.con.bhz; para la componente Norte - Sur, Este - Oeste y Vertical respectivamente. En ellos, 993580527 representa la fecha y hora del sismo expresado por el año (99, correspondiente a 1999), el día del año o día juliano (358), la hora (05) y el minuto (27); además señala el código de la estación de donde proviene el dato (con, estación de Conima) y por último, un código que señala la componente de registro. Un ejemplo que muestra la señal de un sismo registrado por estaciones de banda ancha y visto a través de un programa especializado, se muestra en la Figura 2.4.

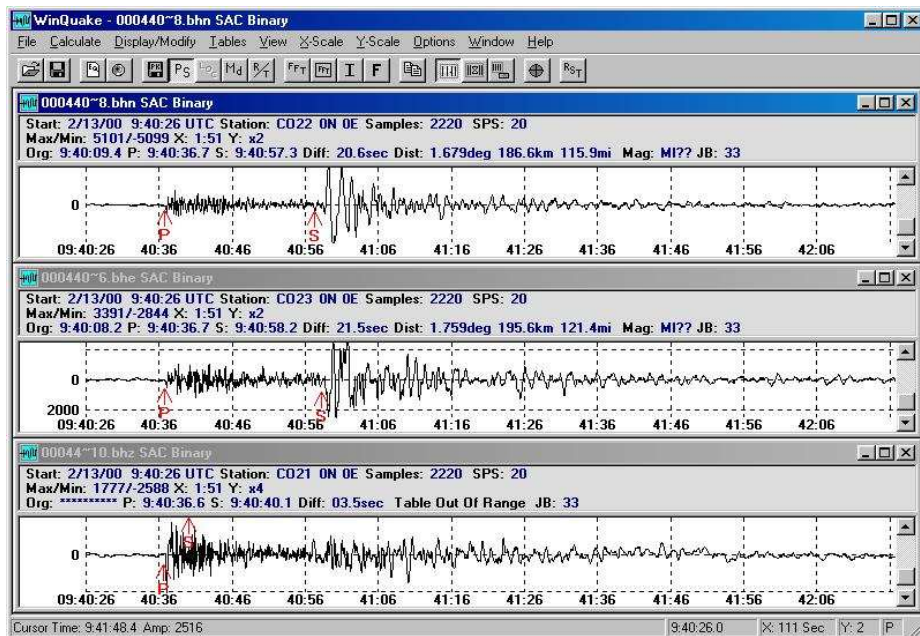


Figura 2.4. Formas de onda de un sismo registrado por la estación de Banda Ancha Conima. En la figura se muestra las tres componentes de registro: una vertical (Z) y dos horizontales (E – W y N – S).

2.3.1.3.- Archivos generados por las Estaciones de Periodo Corto

Las estaciones de periodo corto (SP, Short Period) están agrupadas en 4 grupos: La Red Telemetría del Norte, la RT de Lima, la RT de Huancayo y la RT del Sur. Cada una de estas redes está conformada por un grupo de estaciones de SP comunicadas a una sub - sede, que es donde llega la señal de cada estación en tiempo real (ver Figura 2.3). La adquisición de la información con estaciones de Periodo Corto se realiza mediante un sistema de adquisición denominado ACQ - Sismalp que permite el registro de eventos por STA/LTA (Bernal, 2002). Al suscitarse un evento sísmico, las formas de onda de cada estación se integran en un solo archivo de tipo binario cuya extensión es SIS (*.sis). Adicionalmente, se genera un archivo con el mismo nombre del archivo que contiene la forma de onda (*.sis), pero con extensión NDX (*.ndx), conocido como el “header”, el cual contiene información referida al archivo SIS. Para visualizar las señales contenidas en el archivo SIS se utiliza un programa de procesamiento (Ejemplo: Sismalp). El nombre de estos archivos está dado por la fecha en que ocurrió el sismo y el número de sismo ocurrido en el día; es decir, si es el primero en el día tomara el

numero “01”, si es el tercero tomará el número “03” o si es el vigésimo será el “20”. Un ejemplo del nombre que poseen dichos archivos es: 99121003.sis, 99121003.ndx, 99121003.dep; donde “991210” representa la fecha del evento en formato yymmdd (yy = año, mm = mes y dd = día), y las dos últimas cifras indican el número de sismo ocurrido en el día. Asimismo, el archivo binario DEP (*.dep), contiene información que resulta del procesamiento del archivo *.sis con el programa Sismalp. Un ejemplo que muestra la señal de un sismo registrado por la red de estaciones de periodo corto y visto a través de un software especializado, se muestra en la Figura 2.5.

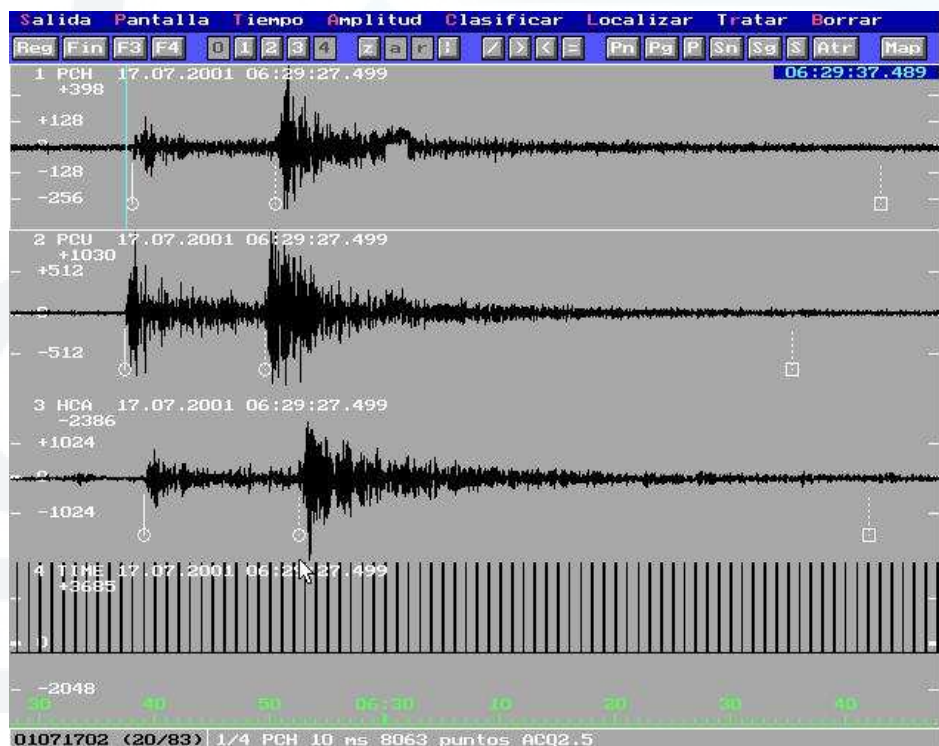


Figura 2.5. Formas de onda de un evento sísmico registrado por estaciones de periodo corto. Estos registros pertenecen a la Red Norte.

2.3.2.- Archivos con Datos Sísmicos

El análisis de las formas de onda de los sismos se realiza a fin de obtener las lecturas de los tiempos de arribo de las fases P y S, y el valor de los parámetros que permiten cuantificar la energía liberada por el sismo (duración y amplitud de fases) haciendo uso de programas especializados como el *Sismalp* o el *Winquake*. La información sísmica leída de cada forma de onda registradas por las diferentes

estaciones de la RSN, se almacenan en un solo archivo para cada sismo. Asimismo, la información de dichos archivos y los programas de procesamiento de datos sísmicos, como el EPI, permiten obtener los parámetros de ubicación y tamaño del sismo, que es la información final que los sismólogos requieren. Los datos que permiten obtener la ubicación del sismo están contenidos en el *Archivo de Entrada de los Programas de Procesamiento de Datos Sísmicos* y los resultados se encuentran en el *Archivo de Salida de los Programas de Procesamiento de Datos Sísmicos*.

2.3.2.1.- Archivos de Entrada

El contenido de los Archivos de Entrada a los Programas de Procesamiento de Datos Sísmicos esta conformado de la siguiente forma (ver Figura 2.6): la primera línea contiene la fecha, hora y minuto del sismo, y a partir de la segunda línea la información se organiza en filas y columnas. Cada una de estas filas contiene la información obtenida del registro sísmico de una estación determinada. La primera columna lleva el código de la estación. La segunda columna lleva un número que representa el peso que tiene la lectura de tiempo de arribo de la fase P y su valor oscila entre 0 y 4. La tercera columna en una nomenclatura que indica el tipo de fase P. La cuarta columna representa un valor en segundos que corresponde al tiempo de arribo de la fase P, donde los segundos se consideran desde el minuto indicado en la primera línea. La quinta columna, al igual que la segunda, indica el peso de la lectura del tiempo de arribo pero para la fase S. La sexta columna indica la nomenclatura del tipo de fase S. La séptima indica el valor en segundos del tiempo de arribo de la fase S, donde los segundos se toman en cuenta desde el minuto indicado en la primera columna. En una ultima columna y muy separada de las demás se encuentra el valor de la duración del sismo. Así también, en la primera línea, seguido del valor de minuto, se indica los valores de intensidad máxima sentida en diferentes localidades.

2.3.2.2.- Archivo de Salida

El contenido de los Archivos de Salida de los Programas de Procesamiento

Sísmico esta conformado de la siguiente forma (ver Figura 2.7): en la primera línea se especifica el número de sismo en el mes, la fecha y la hora origen (GMT) del sismo. La segunda, tercera y cuarta línea indican los valores de latitud (en grados), longitud (en grados) y profundidad (en kilómetros) del sismo respectivamente. En la quinta línea se indica el valor de magnitud *mb* calculado para el sismo. En la sexta línea se indica los valores de intensidad máxima del sismo y las localidades afectadas. Desde la séptima línea la información se organiza en filas y columnas. Cada fila o línea representa la información para cada estación que registró el sismo. La primera columna es el código de la estación. La segunda columna representa la distancia epicentral (Distan). La tercera columna indica el acimut comprendido entre el epicentro calculado para el sismo y la estación (Azm). La cuarta representa el ángulo de incidencia (Ain). La quinta representa el tiempo de arribo de la fase P calculado por el algoritmo, en segundos (TPCal). La sexta representa el valor del tiempo de arribo de la fase P leído (P-Seg). La séptima representa la residual para el tiempo de arribo de la fase P (P-Res). La octava representa el peso de la lectura de la fase P (W) en el procesamiento. La novena representa el valor del tiempo de arribo de la fase S leído (S-Seg). La décima representa el valor de la residual para el tiempo de arribo de la fase S (S-Seg). La onceava representa el peso de la lectura de la fase S (W) en el procesamiento. La columna doce y trece contiene valores de magnitud “mb” del sismo (mb1 y mb2).

27 Ene 2001	23h 57m	III-IV Rioja, III Moyobamba-Chachapoyas, II Bagua										
PAR	0EP	134.25	0ES	226.54							350	
GUA	0EP	137.28	0ES	232.15							260	
ZAM	0EP	147.25	0	0.00								
HLS	4EP	72.10	4ES	122.33								
CAJ	4EP	52.80	4ES	91.85								
PUC	0EP	69.78	0	0.00								
PCH	0EP	50.60	0	0.00								
PCU	4EP	48.50	0	0.00								
HCA	0IPC	47.10	0	0.00								

Figura 2.6. Contenido de un Archivo de Entrada. Esta información se genera de la lectura de tiempos de arribo de las fases P y S en cada una de las estaciones que registraron el sismo. Para la explicación de cada valor ver el texto

Sismo N°	92	Fecha:	27 Ene 2001	Hora Origen (GMT): 23h 57m 8.12s								
Latitud:	-5.1606	±	1.2 km.									
Longitud:	-77.1321	±	1.4 km.									
Profundidad:	96.6 km.	±	9.3 km.									
Magnitud:	5.1 mb2											
Intensidad:	III-IV Rioja, III Moyobamba-Chachapoyas, II Bagua											
Esta	Distan	Azm	Ain	TPCal	P-Seg	P-Res	W	S-Seg	S-Res	W	mb1	mb2
PAR	967.1	175	113	126.1	134.3	0.0	0	226.5	-0.0	0	0.0	5.3
GUA	992.8	172	113	129.3	137.3	-0.2	0	232.1	0.0	0	0.0	4.8
ZAM	1069.5	171	113	138.9	147.3	0.2	0	0.0	0.0	0	0.0	0.0
HLS	418.0	192	114	57.6	72.1	6.3	4	122.3	14.4	4	0.0	0.0
CAJ	266.9	215	115	39.0	52.8	5.7	4	91.8	16.2	4	0.0	0.0
PUC	450.7	143	114	61.7	69.8	-0.1	0	0.0	0.0	0	0.0	0.0
PCH	297.6	252	115	42.8	50.6	-0.3	0	0.0	0.0	0	0.0	0.0
PCU	271.8	253	115	39.6	48.5	0.8	4	0.0	0.0	0	0.0	0.0
HCA	264.6	260	115	38.7	47.1	0.3	0	0.0	0.0	0	0.0	0.0
rmc = 0.20												

Figura 2.7. Contenido de un Archivo de Salida. Esta información se genera después de procesar la información del Archivo de Entrada en los programas sísmicos. Para la explicación de los diferentes valores ver el texto.

2.3.3.- Catálogos Sísmicos

Un Catálogo Sísmico contiene los parámetros sísmicos de la Sismicidad de una región en forma sintetizada; es decir, los parámetros de ubicación y tamaño del sismo en un formato dado por filas y columnas. Cada fila de datos de un catálogo representa la información de un sismo. Los Catálogos Sísmicos son documentos cuya información sísmica representa muchas veces el primer recurso al iniciar una investigación y consulta de uno o varios sismos de una determinada región y para un periodo determinado. Los Catálogos Sísmicos contienen información concreta de cada uno de los sismos ocurridos en una región, tanto de la Sismicidad Histórica como de la Instrumental.

Básicamente, para el Perú existen 4 Catálogos: *Catálogo Sísmico República del Perú* (1471 - 1982) del Proyecto SISAN y Publicado en 1984; *Catálogo Sísmico del Perú* (1500 - 1984) desarrollado por el Instituto Geográfico Nacional de España, US Geological Survey, Pontificia Universidad Católica del Perú y Universidad Nacional de Ingeniería, fue Publicado en 1985 y el *Catálogo Sísmico del Perú* (1500 - 1982), desarrollado por el Instituto Geofísico del Perú - Proyecto SISRA, publicado en 1986.

Las características de los parámetros focales de los sismos contenidos en estos catálogos, han sido ampliamente analizados y evaluados por Tavera y Agüero (2000). Estos autores unificaron los datos de estos tres catálogos a fin de obtener uno homogéneo, constituyendo una “Versión Corregida y Actualizada” para el periodo 1471 - 1982. La información contenida en este Catálogo se integrará a la Base de Datos Sísmicos.

2.3.4.- Soluciones Gráficas

La interpretación de los resultados dentro de un análisis o investigación de los sismos se expresa mediante gráficos, mapas, esquemas u otro tipo de información gráfica. Este tipo de información ayuda a presentar e interpretar mejor los resultados obtenidos de los análisis sísmicos. Esta información se encuentra en los distintos formatos: *.jpeg, *.gif, *.bmp, *.tif y otros. Como ejemplo se puede citar a los mapas de intensidad, mapas sísmicos, perfiles sísmicos, mapas sismotectónicos, geológicos y de estructuras.

2.3.5.- Informes de Estudios Sismológicos

Los resultados de un análisis sísmico se encuentran documentados en Informes teóricos. El contenido de estos informes consta de la metodología empleada en un estudio o trabajo de investigación y los resultados obtenidos. En si, cada uno de estos informes representa el logro más importante en un centro de investigación. Estos informes son documentos o archivos elaborados en procesadores de texto por lo que están representados básicamente por archivos con extensión *.doc, *.txt entre otros. Existen también otros formatos en los cuales se puede encontrar esta información, por ejemplo el formato PDF (*.pdf) o el PostScript (*.ps).

Finalmente, en la Figura 2.8 se muestra un esquema de un diagrama general que señala la secuencia que se sigue durante el desarrollo de un estudio de Sismicidad en una determinada región. En el Instituto Geofísico del Perú, estos informes son elaborados por la dirección del Centro Nacional de Datos Geofísicos.

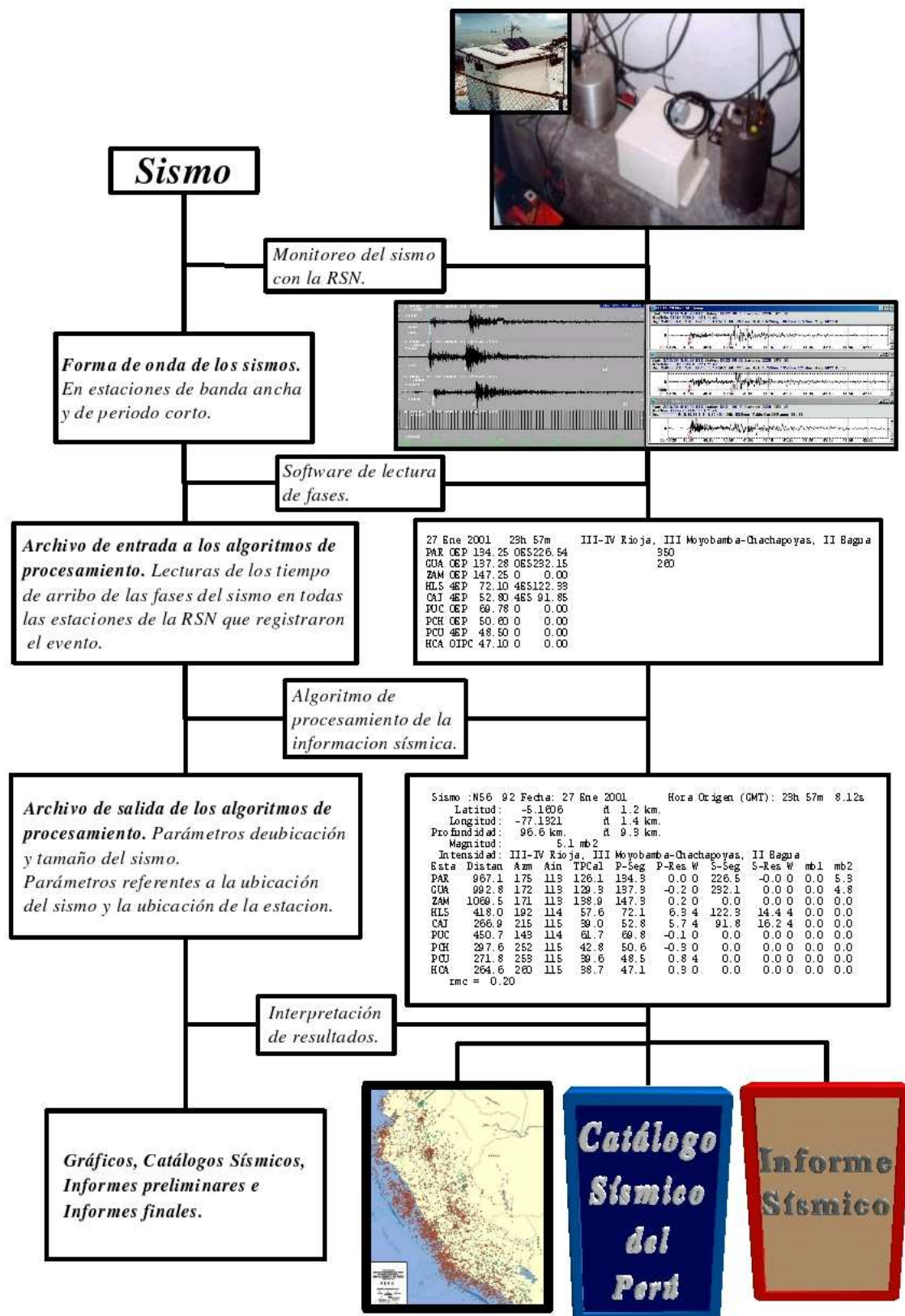


Figura 2.8. Esquema que muestra las etapas de obtención de la información sísmica y la información resultante en cada una de estas etapas.

CAPITULO 3

EL SOFTWARE DE BASE DE DATOS ORACLE

3.1.- Introducción

En este capítulo se introducirá al conocimiento y los conceptos aplicados en el manejo del Software de Base de Datos Oracle⁸ⁱ y su tecnología Objeto – Relacional. Para tal objetivo, se empezará tratando sobre el Oracle y sus características. Seguidamente se evaluará el uso de sus herramientas y las aplicaciones con las que cuenta el módulo básico del producto y del servidor de Oracle, el cual representa el motor del producto. Con la intención de entender mejor la tecnología Objeto – Relacional que brinda Oracle, se tratará conceptos sobre bases de datos relacionales y también se definirá que es una base de datos orientada a objetos y cuales son sus ventajas con respecto a las bases de datos tradicionales. Además, se tratará sobre la arquitectura que tiene una base de datos Oracle y los mecanismos de seguridad que ofrece Oracle para con los datos que almacena.

3.2.- El Oracle

El software de base de datos Oracle, es hoy en día una de las opciones más importantes para la administración de la información en cualquier empresa o institución que maneje y trabaje con grandes volúmenes de información y que desee que esta se haga de forma segura y transparente utilizando como medio de transmisión y consulta a la Internet.

Oracle System Corporation fabrica productos software y proporciona servicios para la gestión electrónica de la información. Además de esto, desempeña uno de los papeles más importantes en la adopción del modelo de computación de Internet, en aquellas áreas en donde las organizaciones se fijan en la World Wide Web (www) para

implantar aplicaciones distribuidas, tanto de consulta como de administración de la información, considerando que en estos días, Internet es la infraestructura ideal para crear aplicaciones que permitan realizar este tipo de tareas. Considerando la tendencia actual a utilizar componentes de consulta dinámica y gestión electrónica de datos, Internet es el sitio ideal para unir las piezas de las que constan estas aplicaciones. Internet, combinado con la arquitectura de computación *Cliente/Servidor* de tres niveles, es el lugar perfecto para estos sistemas de administración y gestión de datos. Los tres niveles de los que consta el enfoque *Cliente/Servidor* son los siguientes (ver Figura 3.1):

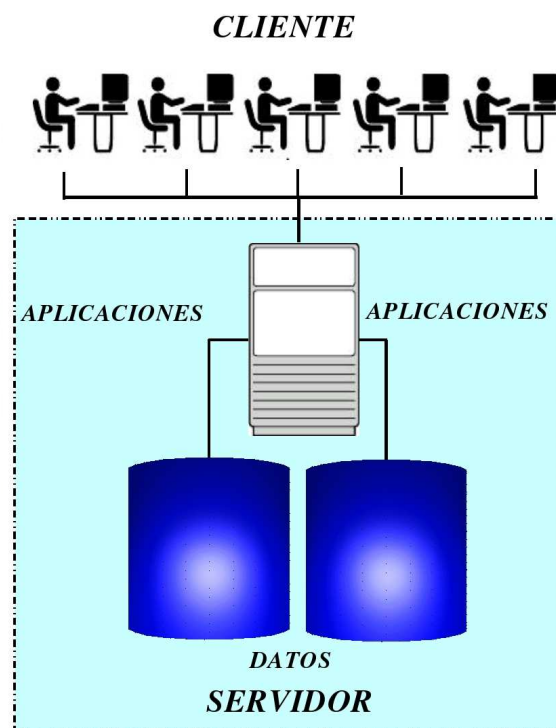


Figura 3.1. Arquitectura de Internet Cliente/servidor.

1.El cliente: En ocasiones se denomina nivel de presentación, es donde se muestra la información al usuario y en donde se aceptan los datos de entrada para su procesamiento.

2.El nivel de aplicaciones: En este nivel se lleva a cabo todo el procesamiento de

acuerdo con la lógica implementada por el sistema. Aquí es donde se comprueba la integridad de los datos y donde se realiza el procesamiento complejo dictado por las necesidades del sistema. Este nivel constituye el motor del enfoque de tres niveles.

3.El nivel de datos: Con frecuencia se denomina módulo de servicio y desempeña un papel importante en el almacenamiento de la información para satisfacer las solicitudes de los dos niveles anteriores. En numerosas ocasiones, aunque no en todas, es una base de datos de tipo relacional y optimizada específicamente para recibir datos del nivel de aplicaciones, a medida que los usuarios interactúan con los sistemas.

3.2.1.- Características del Oracle

Son muchas las características que han catapultado a la Empresa Oracle a lo alto de la creciente comunidad de empresas e instituciones dedicadas a la gestión y manejo de la información. A continuación se describirá algunas de las características del Oracle (Abbey, 2000).

a) *Soporte para la Toma de Decisiones.* En un sentido amplio, se define a los Sistemas de Toma de Decisiones (Decision Support System, DSS), como un conjunto de programas y herramientas que permiten obtener de manera oportuna la información que se requiere, empleando un proceso de toma de decisiones necesaria durante un ambiente de incertidumbre. Ayuda a la toma de decisiones a los Administradores de Bases de Datos, a la combinación de los datos con modelos analíticos sofisticados y un software amigable en un solo sistema poderoso puede dar soporte a la toma de decisiones semiestructuradas o no estructuradas.

b) *Gestión de Grandes Volúmenes de Datos.* Oracle siempre ha presentado especial atención a la gestión de grandes volúmenes de datos, especialmente desde la versión 7.3 que apareció alrededor del año 1997.

c) *Mecanismos de Seguridad*. Los sofisticados mecanismos de seguridad de Oracle controlan el acceso a los datos sensibles utilizando un conjunto de privilegios que son otorgados de manera selectiva a los usuarios, empleando para tal selección de usuarios el “*nombre de usuario*”. En función al nombre que se le otorga a un usuario este puede conectarse a la base de datos, con derechos o privilegios para hacer únicamente consultas o también hacer consultas y modificar datos. Los administradores usan estos mecanismos para asegurarse de que ciertos usuarios puedan consultar los datos de carácter sensible; mientras que, ha otros se les puede negar dicha posibilidad, lo que permite que los datos originales no sean distorsionados consciente o inconscientemente por los usuarios al momento de interactuar con los ellos.

d) *Realización de Copias de Seguridad y Recuperación*. Oracle proporciona sofisticados procedimientos de realización de copias de seguridad y recuperación de datos. Las copias de seguridad permiten crear una copia secundaria de los datos de Oracle, con la cual, mediante procedimientos de recuperación se restaura la información.

e) *Gestión de Espacio*. Oracle ofrece una gestión flexible del espacio de disco. Esto significa que, el administrador puede asignar un cierto espacio de disco para el almacenamiento de los datos y controlar las subsiguientes asignaciones instruyendo a Oracle sobre cuanto espacio reservar para los requerimientos futuros. Asimismo, se tiene una serie de características que fueron diseñadas teniendo en cuenta las necesidades de las bases de datos de muy gran tamaño.

f) *Conectividad de Carácter Abierto*. Oracle proporciona conectividad de carácter abierto hacia y desde paquetes software de otros fabricantes. Con Oracle Access Manager, las instalaciones pueden integrar fácilmente sus diversos sistemas corporativos ejecutados en una amplia gama de productos de diferentes fabricantes. Debido a esta característica, se puede acceder a los datos de Oracle de manera transparente desde aplicaciones escritas en lenguajes de tercera y cuarta generación, como el Cobol o el C. También se admite el lenguaje PL/SQL, lo que permite invocar procedimientos almacenados en ubicaciones remotas con respecto a Oracle.

3.3.- El Servidor Oracle8i

3.3.1.- Introducción

El objetivo cuando se desarrollo Oracle8i era gestionar los datos corporativos, independientemente del tipo de datos que fuera, esto incluye datos estructurados y no estructurados. Oracle 8i puede almacenar dichos datos según el modelo más apropiado para sus aplicaciones y situación.

La compañía Oracle reconoce que existen muchas formas de utilizar una tecnología en las empresas o instituciones. En algunas, se necesita procesar transacciones interactivas; en otras son solo herramientas para la toma de decisiones. Independientemente de la necesidad de cada tipo distinto de trámite u operación, Oracle8i se ha desarrollado para ocuparse de las aplicaciones empresariales e institucionales. La Tabla 3.1 ilustra las características que la compañía Oracle ha introducido en la base de datos Oracle8i, que le permite satisfacer de forma completa las necesidades de cualquier empresa o institución que maneje datos.

Tabla_3.1. Cualidades de Oracle8i.

Respecto a:	Permite
Tipos de Datos	Escalar, texto, video, espacial, imagen, definido por el usuario.
Modelo de datos	Relacional, Multidimensional, orientando a objetos.
Aplicación	Operaciones, toma de decisiones, colaboración, comercio.

3.3.1.1.- Áreas clave del enfoque de Oracle8i

Las áreas clave de enfoque del Oracle8i son las siguientes:

- Requerimientos de procesamiento de transacciones iterativas de gama alta (OLTP,

on line transaction processing) y de almacén de datos. Esta opción permite realizar operaciones con la base de datos vía Internet desde una maquina distinta a la maquina donde radica la base de datos.

- Extensiones Objeto – Relaciones. Esto quiere decir que las bases de datos sacarán provecho tanto de la tecnología Orientada a Objetos como de la Relacional.
- Numerosas mejoras funcionales, de rendimiento y de gestión.

3.3.1.2.- Capacidad de usuarios y tamaño

Oracle8i puede admitir unos 10000 usuarios y una base de datos con un tamaño máximo de 100 terabytes (siendo 1 terabyte = 1.099.511.627.776 bytes). Esta capacidad es clave para dar servicio a entornos OLTP de gama alta y proyectos de almacenar bases de datos muy grandes.

3.3.1.3.- Partición y Paralelismo

Una de las formas más fáciles que se ha encontrado para trabajar en un problema complicado es dividir el problema en problemas más sencillos y por ende más pequeños. En esto se basa el concepto de partición; es decir, en la capacidad de dividir la base de datos en partes más pequeñas y más manejables, para trabajar después de forma independiente con cada una de ellas. Oracle 8i pone ahora a su disposición esta posibilidad.

Al igual que la partición, se ha visto que cuando se quiere hacer un trabajo rápido inevitablemente se pide ayuda. De esto se trata la ejecución en paralelo; es decir, la capacidad de dividir un trabajo en muchos trabajos más pequeños, que pueden hacerse en paralelo. Los trabajos que pueden llevar horas pueden hacerse de esta manera en minutos.

3.3.1.4.- Configuraciones del servidor

Oracle8i Server es un entorno avanzado que permite la gestión de la información fácil y segura, ya que permite almacenar grandes cantidades de datos y proporciona a los usuarios acceso rápido a los mismos. Asimismo, el Servidor Oracle8i permite la compartición de datos entre aplicaciones; es decir, la información se almacena en cierto lugar y puede ser utilizada por muchos sistemas. El Servidor Oracle considera las siguientes configuraciones (Abbey, 2000):

- *Basada en host.* Los usuarios se conectan directamente a la misma computadora en la que reside la base de datos.
- *Cliente/Servidor.* Los usuarios acceden a la base de datos desde su computadora personal (cliente) a través de una red, ya que la base de datos se encuentra en una computadora diferente (servidor).
- *Procesamiento distribuido.* Los usuarios acceden a una base de datos que reside en más de una computadora. La base de datos está repartida entre varias máquinas y los usuarios no tienen por qué conocer la ubicación física de los datos con los que están trabajando.
- *Computación compatible con la Web.* La posibilidad de acceder a los datos desde una aplicación basada en Internet.

Oracle vende su tecnología de servidores con cierto número de extensiones opcionales que amplifican la funcionalidad del servidor. El producto básico proporciona toda la funcionalidad necesaria para satisfacer las necesidades de la mayor parte de los usuarios. Cuando éstos necesitan funcionalidad adicional, Oracle dispone de una serie de opciones que se pueden adquirir. En las siguientes secciones se va a detallar algunas de estas características mas no las opciones.

En general, una característica es algo que se incluye en el producto base de Oracle8i Server, mientras que una opción es algo que se puede añadir al producto base por un precio adicional (Abbey, 2000).

3.3.2.- Características de Oracle8i Server

Las características de su tecnología de servidor han permitido que Oracle8i logre su posición y prestigio. Aquí se tratan algunas de las características que han ayudado a catapultar a Oracle a lo más alto en cuanto a software de manejo de base de datos.

a) Accesibilidad de los Datos. Al adquirir Oracle Server, se adquiere una serie completa de funcionalidades básicas que ayudan a almacenar los datos y mantenerlos accesibles. Esto incluye utilidades para realizar copias de seguridad de los datos, las cuales se pueden llevar a cabo incluso mientras que los usuarios están trabajando con ellos. El término empleado de manera usual para explicar este concepto es el de "copia de seguridad en caliente", pero el término oficial que emplea Oracle para designar esta capacidad es "copias de seguridad en modo archivo" (archive mode backups). Esto permite que no se interrumpa el acceso a los datos por parte de las aplicaciones mientras se realiza la copia de seguridad.

Oracle Server también se encarga de la integridad de los datos. Si se produce cualquier tipo de fallo mientras que un usuario está cambiando los datos en una base de datos, esta tiene la capacidad de deshacer o cancelar cualquier transacción sospechosa. Oracle Server no permite dudas en lo referente al estado de una transacción determinada. El servidor incluye también un bloqueo completo por filas de todos los datos almacenados.

b) Capacidades Procedimentales. Oracle8i Server cuenta con muchas características procedimentales las cuales están basadas en el lenguaje de programación de Oracle, PL/SQL. Con el lenguaje PL/SQL y este Oracle Server se pueden implementar las siguientes funciones:

- *Procedimientos almacenados.* Se pueden almacenar programas (o segmentos de código) en la base de datos Oracle, para realizar funciones de importancia para el sistema.
- *Disparadores de base de datos (triggers).* Son segmentos de código almacenados en la base de datos y que se disparan como respuesta a sucesos que tienen lugar en las aplicaciones.
- *Paquetes (packages).* Los procedimientos se suelen agrupar, almacenándose el código como una única unidad de programación en la base de datos.

c) Procesamiento Distribuido. La *Capacidad Distribuida* del servidor Oracle permite que la información pueda estar ubicada en más de una computadora sin que esto signifique una dificultad. Si un usuario necesita acceder a los datos como si todos estuvieran en una sola computadora central, dicha operación para el Servidor Oracle no representa ningún problema. Esto significa que la ubicación física de los datos no es de conocimiento del usuario.

d) Consulta en Paralelo. La opción de consulta en paralelo (Parallel Query) permite sacar partido del procesamiento de consultas en computadoras dotadas de más de una unidad central de proceso (procesador). En las máquinas con un solo procesador, así como en aquellas con múltiples procesadores sin la opción de consulta en paralelo, sólo un proceso puede acceder a la base de datos y seguidamente mostrar dichos datos al usuario según el criterio de selección definido. El procesamiento tiene lugar como se muestra en la Figura 3.2. Cuando se utiliza la opción de consulta en paralelo en una máquina con múltiples procesadores, Oracle ejecuta varios procesos de consulta los cuales se reparten el procesamiento de la consulta y trabajan de manera simultánea. Una vez que los resultados están preparados, se integran y son presentados al usuario. La Figura 3.3 muestra los fundamentos de esta opción. Con esta capacidad de procesar consultas en paralelo, una consulta que podía tardar antes una hora puede ser procesada en cuestión de minutos, sacando partido de toda la potencia de procesador disponible.

e) **Enterprise Manager.** Oracle Enterprise Manager es un conjunto de herramientas de gestión diseñadas para ayudarle al Administrador de Base de Datos (DBA) a gestionar todo el entorno Oracle, lo que incluye los sistemas, aplicaciones, redes y bases de datos. Además de permitirle al DBA gestionar diferentes entornos Oracle con una única herramienta. También dispone de agentes inteligentes y de una interfaz abierta que le ayudará a sacar el máximo partido de productos de terceras fuentes (productos que no pertenecen al paquete del software).

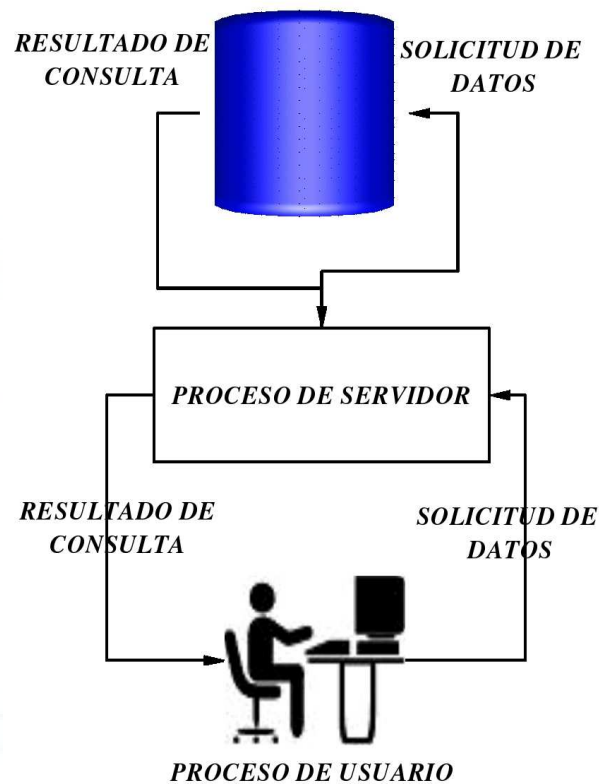


Figura 3.2. Esquema que muestra el procesamiento de consulta sin la opción de consulta en paralelo.

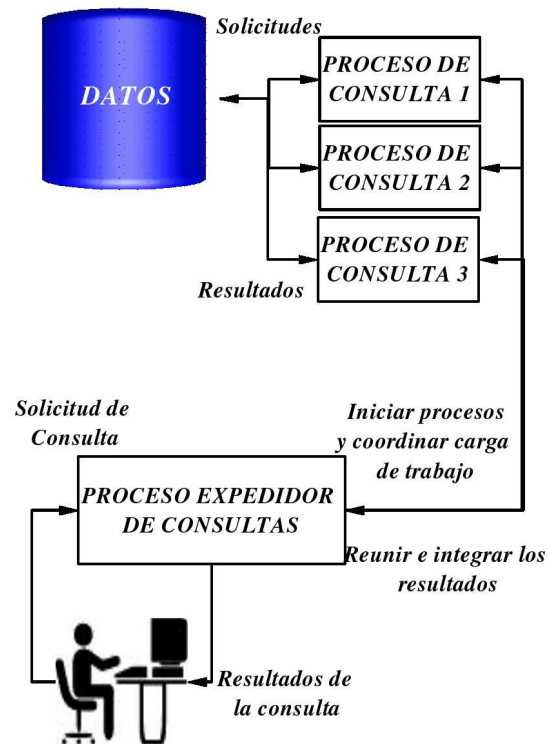


Figura 3.3. Esquema que muestra el procesamiento de consulta con función de consulta en paralelo.

Desde la consola del gestor, el administrador de la base de datos puede gestionar todo el entorno Oracle a través de una interfaz gráfica. Existen cuatro componentes principales en la consola de Oracle Enterprise Manager:

- ⑩ *El Navigator (Explorador)* que permite ver y manipular todos los nodos y servicios de la red en forma de árbol.
- ⑩ *El Map Window (Ventana de Asignación)* que permite al administrador de la base de datos ver y asignar subconjuntos de objetos de una manera gráfica. Esta herramienta permite tomar un entorno muy complejo y distribuirlo en subconjuntos más coherentes.
- ⑩ *El Job Scheduling System (Sistema de Planificación de Tareas)*, que permite a los

Administradores de Bases de Datos (DBA) automatizar las tareas repetitivas. Esto quiere decir que desde una ubicación se puede automatizar una tarea en otra ubicación remota. Por ejemplo, el administrador de la base de datos puede automatizar las tareas de realización de copias de seguridad, lo que le permite ahorrar tiempo para dedicarlo a otras tareas.

⑩ *El Event Management System (Sistema de Gestión de Sucesos)*, que permite al DBA supervisar de manera remota sucesos del sistema o de la base de datos y de acuerdo con ciertos umbrales predefinidos, lanzar tareas correctivas mediante el sistema de planificación de tareas. Por ejemplo, puede definirse un suceso para que supervise la cantidad de espacio libre en la base de datos, con el fin de asegurarse de que una base de datos nunca se queda sin espacio.

Cada base de datos Oracle gestionada, tiene un proceso denominado agente inteligente que se ejecuta de modo continuo. Este proceso supervisa cada una de las bases de datos gestionadas por el OEM, recopila estadísticas de rendimiento de la base de datos y guarda la información en el depósito de Oracle para que OEM la use y el DBA pueda ejecutar acciones correctivas. OEM utiliza a este agente para realizar sus tareas. El agente inteligente es de carácter abierto; es decir, que los productos de terceras fuentes también pueden servirse de él. Al adquirir Oracle8i Server se cuenta con el Oracle Enterprise Manager.

3.3.3.- Oracle8i y la tecnología Objeto - Relacional

En este punto, se tratará los conceptos correspondientes a las bases de datos Relacionales y a la base de datos Orientadas a Objetos, siendo estos el fundamento del Oracle8i, para comprender la tecnología que ofrece el software de base de datos Oracle8i.

3.3.3.1.- El enfoque Relacional

El modelo relacional de datos supuso un gran avance con respecto a los modelos anteriores. Este modelo está basado en el concepto de relación. Por concepto, una relación es un conjunto de n-tuplas. Una tupla, al contrario que un segmento, puede representar tanto entidades como interrelaciones (N:M). Los lenguajes matemáticos sobre los que se asienta el modelo Relacional, el álgebra relacional y el cálculo relacional, aportan un sistema de acceso y consulta orientado al conjunto. La repercusión del modelo en los DBMS (Database Management System, Sistemas Gestores de Bases de Datos) comerciales y actuales ha sido enorme, estando hoy en día la gran mayoría de los gestores de bases de datos basados en mayor o menor medida en el modelo Relacional.

3.3.3.1.1.- Conceptos básicos: tuplas, relaciones, atributos

Como ya se ha mencionado, el modelo relacional está basado en la teoría de conjuntos. En este modelo, los datos se organizan en un tipo especial de conjunto denominado *relación*, (relation) que se define de la siguiente manera: Sean los conjuntos D_1, \dots, D_n , denominados *dominios*, que no tienen por qué ser distintos entre sí; una relación definida sobre D_1, \dots, D_n es cualquier subconjunto R de D , donde n es el *grado* o *aridad* de R . Los dominios son en principio conjuntos finitos de datos. Por tanto, a menos que se indique lo contrario, se presume que las relaciones son también finitas. Los elementos de una relación se denominan *tuplas*. Formalmente, una tupla es: $\langle d_1, \dots, d_n \rangle$, donde $d_1 \in D_1, \dots, d_n \in D_n$.

El número de tuplas en una relación es la *cardinalidad* de la relación. Puesto que una relación es un conjunto, los elementos de este conjunto, las tuplas, han de ser por fuerza distintas. Esto también implica que el orden de las tuplas es irrelevante. El conjunto vacío es una relación particular: la relación *nula* o *vacía*. Cada tupla de una relación, junto con el nombre de la relación, representa una *aserción* o *afirmación* (en el sentido lógico).

Las relaciones también pueden ser vistas como *tablas*, en las que cada tupla es una fila de la tabla. Los nombres de las columnas de la tabla, por otra parte, son los *atributos*. El conjunto (ordenado) de todos los atributos de una relación *R* es el *esquema* de *R*. Se puede referir a los atributos de una relación mediante su *nombre* o por la *posición* (número de columna) que el atributo ocupa en el esquema de la relación. Las tuplas, por tanto, pueden ser consideradas como matrices de pares *atributo:valor*.

Los términos formales del modelo relacional a menudo son sustituidos por otros de uso más común debido a que estos términos son demasiado abstractos para ser usados en la práctica. Así se obtiene las equivalencias mostradas en la tabla 3.2.

Tabla 3.2. Términos relacionales y equivalentes informales.

<i>Termino Relacional Formal</i>	<i>Equivalente Informal</i>
Relación	Tabla
Tupla	Fila o registro
Cardinalidad	Número de filas o registro
Atributo	Columna o campo
Grado	Número de columnas o campos
Clave primaria	Identificador único
Dominio	Fondo de valores legales

Aun así, se debería usar estos términos con gran cautela. Codd (1990), indica el hecho de que las relaciones puedan ser percibidas como tablas y puesto que una tabla es parecida a un fichero plano, puede crear la falsa ilusión de que la misma libertad de acciones permitida para las tablas o ficheros planos está permitida para la manipulación de relaciones. Por ejemplo en una relación no pueden existir dos tuplas exactamente iguales. Sin embargo, existen DBMS comerciales, supuestamente relacionales, tales como la familia dBase® de Ashton-Tate (hasta su versión IV) y en general, los productos xBase que permiten este tipo de acciones no permitidas en enfoques

relacionales puros.

En términos de representación tabular se dice que una relación consta de dos partes: *cabecera (heading)* y *cuerpo (body)*. La cabecera es el conjunto de atributos (columnas) y el cuerpo es el conjunto de tuplas (filas). En la *Tabla 4.2*, la primera línea es la cabecera (en negrita) y el resto de las filas el cuerpo.

Tabla 3.3. *Tabla, Cabecera y cuerpo.*

<i>Id</i>	<i>Profesor</i>	<i>Curso</i>	<i>Año</i>	<i>Departamento</i>
1	Date	Base de Datos	2001	Informática
2	Miller	Sismología	2002	Geofísica
3	Booch	Tectonofísica	2003	Geofísica
4	Sag	Gravimetría	2004	Geofísica
5	Sinclair	Métodos Numéricos	2005	Matemática

Finalmente, se aclara que las *relaciones* no sólo son las "*tablas*" base de las bases de datos, sino que también son los distintos tipos de relaciones que se pueden generar mediante consultas a las relaciones base. En general, se puede distinguir los siguientes tipos de relaciones:

- 1.*Relaciones base o reales*: Es lo que corresponde al concepto de tabla. El conjunto de éstas son las que componen realmente la base de datos.
- 2.*Conjunto dinámico de datos*: No poseen datos almacenados propios y están representadas únicamente dentro del sistema mediante su definición en términos de otras relaciones; es decir, mediante consultas conocidas como *dynasets*.
- 3.*Instantáneas (snapshots)*: Iguales que las anteriores, pero los datos que contienen no son virtuales, sino que están realmente almacenados en la instantánea. Se utilizan

para manejar datos susceptibles de cambios.

4.*Resultados de consultas*: La relación resultante de consultar una o más relaciones base.

5.*Resultados intermedios*: El resultado de una operación anidada en una consulta, estos resultados son usados por la consulta externa para otra operación. La facilidad de anidar consultas dota de gran potencia al lenguaje de consulta relacional (SQL).

3.3.3.1.2.- Claves Primarias

Puesto que las tuplas son irrepetibles, una relación necesita un identificador único para cada una de las tuplas, esta es la clave (*primaria*) de la relación que se define como un subconjunto C de los atributos de R y cuyos valores no pueden ser repetidos. Una clave primaria debe ser *mínima*, en el sentido que en su composición no intervengan más que los atributos estrictamente requeridos para identificar las tuplas de forma única. Puesto que una relación es un conjunto de tuplas, se debe dar la condición de que toda relación deba tener una clave primaria, al menos el conjunto de los atributos de una relación conforma la clave de esa relación. Además, una clave primaria puede ser simple (formada por un solo atributo) o compuesta (formada por más de uno). Las dos características definitorias son por tanto, la *unicidad* y la *minimalidad* (Date, 1990).

3.3.3.1.3.- Interrelaciones. Claves ajenas

Un RDBMS ofrece la posibilidad de interrelacionar dos o más relaciones existentes en una base de datos. De hecho, es ésta la facultad que dota de mayor potencia al modelo. Hay varios aspectos que conviene resaltar sobre el establecimiento de interrelaciones.

En primer lugar, la manera en que el modelo relacional interrelaciona las

relaciones existentes en una base de datos no es la que cabría esperar. Normalmente, cuando en una aplicación informática se desea referenciar dos elementos cuales sean, se utiliza los denominados *punteros (pointers)*, haciendo que un elemento apunte a la dirección de memoria de otro y/o viceversa. Sin embargo, el modelo relacional funciona de forma diferente.

En realidad, no existe ningún tipo de puntero o enlace que el usuario pueda percibir. Todas las interrelaciones se realizan mediante la comparación de valores, ya sean éstos identificadores de objetos (claves primarias de las relaciones) o atributos de estos objetos. Para que esto sea posible, es condición indispensable que los valores a comparar pertenezcan al mismo dominio (y por tanto contengan el mismo tipo de datos (numéricos, texto, booleanos, etc.).

Por otra parte, algunos RDBMS comerciales, (por ejemplo Microsoft Access o ADABAS/D) permiten al usuario definir interrelaciones, incluso de forma gráfica. Pero esto no significa ningún cambio en cuanto a la estructura de la base de datos, sino que hace que el DBMS ayude más al usuario a la hora de hacer consultas, interrelacionando automáticamente las tablas incluidas en una consulta determinada y estableciendo las reglas de integridad referencial.

Por ello, una interrelación puede ser considerada a todos los efectos como otra relación (que normalmente se manifiesta como un conjunto dinámico de datos o una instantánea).

La mejor manera de comprender como funcionan las interrelaciones en una base de datos relacional es mediante un ejemplo, que también ayudará a comprender la diferencia entre una relación y una tabla plana.

Considerando la relación de profesores expuesta anteriormente en la Tabla 4.2. Sería conveniente que la base de datos a la que pertenece esta relación contuviese también información sobre los datos personales de los profesores, descripción de los

cursos ofrecidos y descripción de los distintos departamentos. Pues bien si se quisiera incluir toda esta información en una tabla plana, esta debería contener al menos, los siguientes atributos(columnas):

PROFESOR_COD	PROFESOR_NOMBRE
PROFESOR_DIRECCIÓN	PROFESOR_TELÉFONO
PROFESOR_DEPTO	DEPTO_COD
DEPTO_NOMBRE	DEPTO_DESC
CURSO_COD	CURSO_NOMBRE
CURSO_DESC	CURSO_NIVEL

El modelo relacional ofrece una buena solución a este problema, el cual permite reducir la redundancia de datos al mínimo y agilizar las operaciones de consulta y actualización. Lo que se haría es separar la información que se refiere a las tres entidades que se tiene (profesores, cursos y departamentos) en tres relaciones independientes, y después relacionarlas entre sí. De este modo se obtendría una disposición parecida a la de la Figura 3.4. Los recuadros indican relaciones base, mientras que las flechas indican las interrelaciones entre ellas. Estas interrelaciones, en realidad, no existen al nivel de la base de datos, se usan sólo al nivel de representación gráfica. Los nombres de algunos atributos (*Prof_ID*, *Depto_ID*, *Curso_ID*) sugieren el tipo de claves que se ha usado: *claves subrogadas o primarias*.

A partir de estas tres relaciones, y mediante el mecanismo de comparación de valores antes mencionados, se tiene acceso a toda la información que se desea sin redundancia alguna. Los "1" y "M" que etiquetan las flechas hacen referencia al tipo de interrelación "uno a muchos", así en la tabla PROFESOR no hay valores repetidos para el atributo Prof_ID (existe un solo conjunto de atributos para describir un determinado profesor), pero existen varios de ellos en la tabla CURSO (un profesor puede dar varios cursos). Igualmente, un departamento consta de varios profesores. Las interrelaciones

que se ha mostrado en la Figura 3.4 son todas del mismo tipo: de uno a muchos (one-to-many). Ésta es la interrelación más frecuente. Además se tiene otras como:

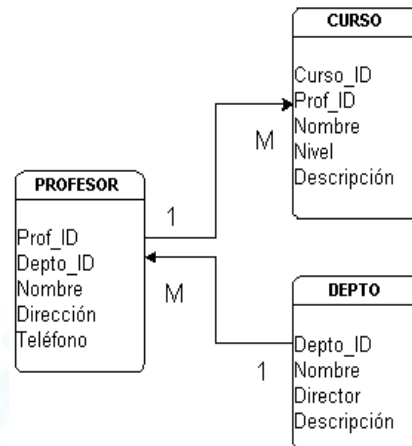


Figura 3.4. Ejemplo de disposición relacional.

- **Muchos a muchos(many-to-many)**: en una relación de este tipo, la tabla A puede tener más de un registro coincidente en la tabla B y un registro en la tabla B puede tener más de un registro coincidente en la tabla A. Para detectar las relaciones "muchos a muchos" entre las tablas, es importante observar la relación en los dos sentidos. Este tipo de relación requiere cambiar el diseño de la base de datos; es decir, a nivel físico, ya que en la realidad esto no es factible. La forma de implementar este tipo de interrelación es mediante una tercera tabla que sirva de puente entre las dos. La tercera tabla contendrá información procedente de las otras dos interrelacionando los registros pertinentes.
- **Uno a uno(one-to-one)**: en una relación de este tipo, un registro en la tabla A no puede tener más de un sólo registro coincidente en la tabla B y viceversa. Este tipo de interrelación es muy poco frecuente, ya que en la mayoría de los casos la información de las dos tablas puede ser combinada en una sola tabla. Tan sólo es apropiado cuando el número de campos de la tabla B es muy alto y concerniente a un determinado tipo de información. En estos casos es aconsejable tener esa

información en una tabla separada.

Las interrelaciones de uno a muchos se implementan mediante el uso de claves ajenas, también llamadas externas o foráneas (foreign keys). Una clave ajena es un atributo (posiblemente indexado y posiblemente compuesto) de una relación R2 cuyos valores han de concordar con los de alguna clave primaria en otra relación R1. R1 y R2 no han de ser necesariamente distintas.

Los atributos Prof_ID, en la tabla PROFESOR, Curso_ID en la tabla CURSO y Depto_ID en la tabla DEPTO, son claves primarias, mientras que los atributos Prof_ID en la tabla CURSO y Depto_ID en la tabla PROFESOR, son claves externas.

3.3.3.1.4.- Integridad Relacional

Ahora que ya se conoce el funcionamiento de las claves primarias y las claves ajenas se esta en posición de estudiar las reglas de integridad. Con este nombre se designa aquellas reglas que han de ser aplicadas a una base de datos para asegurar que los datos introducidos sean consistentes con la realidad que se pretenden modelar. Existen dos reglas generales que aporta el modelo relacional. Estas dos reglas son muy simples y son las siguientes:

- 1.Regla de integridad de las entidades: ningún componente de la clave primaria de una relación base puede aceptar valores nulos.
- 2.Regla de Integridad Referencial: la base de datos no debe contener valores de clave ajena sin concordancia.

La primera de estas reglas impide la existencia de una tupla sin identificador único. La segunda impide que, por ejemplo, en la “base de datos académica” exista un profesor adscrito a un departamento inexistente o un curso impartido por un profesor

inexistente. Se debe recordar que sólo los productos puramente relacionales implementan realmente estas dos reglas generales de integridad relacional. En otros, destinados al mercado doméstico, estas incongruencias son admitidas sin problemas.

Además, muchos RDBMSs añaden un buen número de características que ayudan al DBA a mantener más fácilmente la integridad de los datos. Mediante estos mecanismos es posible añadir reglas específicas para cada base de datos, siendo estas denominadas *restricciones de integridad definidas por el usuario* (Codd,1990). Por ejemplo, se podría determinar que un profesor no pueda ser menor de x años o que un curso sólo pueda pertenecer a los niveles 1, 2 ó 3. El resultado sería que al intentar introducir un valor fuera de este rango, el DBMS rechazaría la información introducida mostrando un mensaje de error.

3.3.3.1.5.- Formas normales

Además de las restricciones impuestas por las reglas generales del modelo relacional y de las reglas específicas impuestas por el DBA para una determinada base de datos, sería conveniente la observación de otras "reglas" que reforzarán el modelo, ayudaran a mantener la integridad de los datos y evitarán la redundancia. Esto es lo que se conoce como *normalización*.

Según Codd (1972) existen tres formas normales básicas en la primera versión del modelo Relacional conocidas como 1NF, 2NF y 3NF respectivamente, más otras tres que fueron añadidas con posterioridad (BCNF, 4NF y PJ/NF ó 5NF). En realidad, BCNF (la forma normal de Boyce/Codd) expuesta por Codd (1974) no es más que un intentar mejorar la 3NF, y durante un tiempo fue llamada simplemente 3NF. Las dos restantes, 4NF y PJ/NF, no fueron definidas por este autor, sino por Fagin (1977 y 1979). A continuación se expone únicamente las tres formas normales básicas (Codd, 1972).

1.1NF: una relación R está en primera forma normal (1NF) si y sólo si todos los

dominios simples subyacentes contienen únicamente valores atómicos (no divisibles).

2.2NF: una relación R está en segunda forma normal (2NF) si y sólo si R está en 1NF y además todos los atributos no clave (es decir, los que no forman parte de la clave primaria) dependen por completo de la clave primaria.

3.3NF: una relación R está en tercera forma normal (3NF) si los atributos no clave (si los hay) son: mutuamente independientes, y dependientes por completo de la clave primaria.

3.3.3.1.6.- Lenguajes relacionales. Consultas

Para crear las relaciones, modificarlas, eliminarlas, recuperar los datos almacenados en ellas y para manipularlas en general, se necesita un lenguaje formal que facilite el acceso, de lo contrario un DBA se vería obligado a trabajar a bajo nivel, o nivel de máquina. Este lenguaje debe ser lo suficientemente expresivo para permitir llevar a cabo todas estas operaciones, y debe estar basado en formalismos que cumplan con todas las premisas expuestas en los apartados anteriores respecto a reglas de integridad, formas normales, etc. Existen dos tipos básicos de formalismos para expresar las consultas sobre las relaciones de una base de datos relacional: el álgebra relacional y el cálculo relacional

El lenguaje de consulta de bases de datos relacionales por antonomasia es como ya mencionó anteriormente, el llamado SQL (Structured Query Language). Este lenguaje actúa de interfaz entre el usuario y la base de datos y facilita realizar todas las operaciones permitidas. El lenguaje fue diseñado para que mediante un número muy reducido de comandos y una sintaxis simple, fuese capaz de realizar un gran número de operaciones. El aprendizaje de SQL es realmente rápido. Además, SQL es bastante flexible en el sentido de que cláusulas SQL pueden ser anidadas indefinidamente dentro de otras cláusulas SQL facilitando así las consultas que utilizan varias relaciones, vistas

u otras consultas. En capítulos posteriores de este estudio se tratará con mas énfasis sus aplicaciones y sentencias.

3.3.3.2.- Base de Datos Orientada a Objetos

Oracle8i fue la primera versión de la base de datos Oracle que incorpora la tecnología orientada a objetos y la primera base de datos objeto – relacional de esta empresa. Una base de datos objeto – relacional no es una base de datos orientada a objetos únicamente, ni tampoco es una base de datos relacional, sino que representa un híbrido de las dos; es decir, que al evolucionar a llegado hasta un punto en la que contiene ambas tecnologías, relacional y de objetos. Con el objetivo de apreciar y comprender lo que Oracle 8i significa para las instituciones y porque las bases de datos objeto - relacional empiezan a convertirse en un estándar, se hablará de las tecnologías orientadas a objetos.

A diferencia de las bases de datos relacionales, que tratan los datos en el nivel mas bajo posible; es decir, como una serie de columnas y filas que conforman las tablas, el método orientado a objetos trata los datos en un nivel mucho mas alto: “trata los objetos que circundan a los datos”. En una base de datos orientada a objetos, cuando se trata con el cliente, se trabaja con un objeto denominado “cliente”. Cuando se trata con un pedido, se hace referencia a un objeto denominado “pedido”. Puesto que una base de datos de objetos entiende el objeto cliente y todas sus relaciones, puede fácilmente tratar con el objeto cliente y todo aquello necesario para trabajar con él.

En el modelo relacional, un pedido es realmente una combinación de diferentes *tablas de datos*, con *tablas de intersección* las cuales contienen todos los atributos necesarios para admitir y mantener dicho pedido. A diferencia del modelo de objetos, en el que la base de datos conoce las interrelaciones entre estos, en el modelo relacional no ocurre así. Si el modelo relacional se efectúa un cambio, usualmente se traduce en que hay que desarrollar una nueva serie completa de tablas si se quiere continuar trabajando con dicho modelo. Un diseñador de base de datos es quien debe rehacer estas relaciones.

Por ejemplo, cuando un cliente introduce un pedido a una base de datos relacional se necesitan varias tablas para permitir esta actividad, estas deben ser: una tabla de clientes, una tabla inventario, una tabla de precios, una tabla inventario – precios, una tabla elemento - línea, una tabla historia – cliente, y así sucesivamente. Para manipular estas tablas el programador debe elaborar el código necesario junto con los vínculos requeridos entre tablas.

Como se puede ver, el simple acto de introducir un pedido requiere una serie de tablas. Las filas y columnas de las tablas contienen la información necesaria para procesar el pedido. Un único cambio en el proceso puede tener un impacto importante en las tablas subyacentes, siendo necesario que un diseñador de base de datos elabore las nuevas relaciones y las tablas asociadas que representen dichas relaciones. Esto no es así en el modelo orientado a objetos. De hecho, se asume que las relaciones cambiarán, y dichos cambios son solo una progresión y un suceso natural.

Seguidamente, se definirá y tratará algunos de los conceptos y características que se manejan en el modelo de base de datos Orientado a Objetos, a fin de ver la potencia de este modelo de base de datos.

a) Objeto. Los objetos son representaciones software de las entidades del mundo real. Para reflejar las funciones y las capacidades del mundo real, los objetos contienen información operacional y de sus atributos. En una base de datos orientada a objetos, el *Diccionario de Datos de Oracle* no sólo almacena y permite conocer las relaciones entre los objetos, sino que también entiende los comportamientos de cada uno de ellos.

b) Clase. Cuando hay objetos similares entre sí, en cuanto a sus comportamientos y otros atributos, pueden ser colocados juntos dentro de una clase. Este concepto de clases, clases padre, subclases y superclases proporcionan un cierto nivel de abstracción para el agrupamiento de objetos. Al pensar en una clase como en una plantilla para objetos, ayudará a gestionar los objetos muy complejos. Esta capacidad de agrupación mediante las clases permite a los objetos aprovecharse de la similitud de los

comportamientos y en otras características que compartan.

c) Encapsulación. Uno de los fundamentos del modelo orientado a objetos es que permite la encapsulación. La encapsulación es la característica por la cual los datos están vinculados al objeto de modo que el acceso a ellos solo pueda ocurrir por medio de los comportamientos aprobados o aceptados por dicho objeto. El beneficio directo de esto es que se protege a los datos de accesos ilegales.

Desde el punto de vista del DBA, los objetos son una encapsulación de datos y comportamientos. Se pueden tratar como cajas negras de programación. Considere a los objetos como colecciones de código y datos que tienen la capacidad, muy potente, de funcionar independientemente. La encapsulación incluye la capacidad de acceder a los objetos y a sus datos solo a través de los comportamientos aprobados y aceptados por el objeto.

Además, en el modelo orientado a objetos, los comportamientos de los datos se almacenan dentro de la base de datos y no son externos a la misma. Dado que los comportamientos de los datos se encuentran dentro de la base de datos, no se compromete la independencia de las aplicaciones con respecto a los datos. Esta es la base de la ley de independencia de los datos: las aplicaciones, SQL o de otro tipo, son independientes de los propios datos.

c)Extensibilidad. La extensibilidad es la capacidad de una base de datos orientada a objetos para añadir nuevos objetos y sus comportamientos asociados, sin afectar a los demás objetos y aplicaciones. Dado que los datos pueden encapsularse con objetos, esta propiedad de extensibilidad proporciona al modelo de objetos la capacidad de manejar datos no estándar.

d) Herencia. Se piensa en la herencia como en una forma de compartición de código. Cuando se define una nueva clase de objetos, esta se puede definir basándose en una clase existente, a la que se denomina clase base. A medida que se crean objetos de nivel

inferior, estos heredan o acceden a los datos y comportamientos asociados con todas las clases por encima de ellos.

La clase de objetos derivada de la clase base, normalmente, aumenta o redefine la estructura y el comportamiento existente de la clase base. Otra forma de interpretar la herencia es que se crean clases nuevas como especialización de las clases existentes. generalmente estas clases se obtienen por las siguientes razones:

- 1.Si se desea implementar el mismo código, pero con diferentes comportamientos asociados.
- 2.Si se quiere ampliar de forma incremental el comportamiento de la clase base. Cuando ya se ha creado una clase especializada, se pueden confeccionar los segmentos de código que proporcione a la clase especializada más capacidades especiales. Estas es una forma de añadir funcionalidad al sistema.
- 3.Se necesita realizar diferentes implementaciones del sistema.

Existen dos tipos de herencia:

- *Herencia de datos*: tiene lugar cuando cada elemento de datos puede heredar atributos adicionales de otros elementos de datos de la clase.
- *Herencia función/objeto*: tiene lugar cuando un objeto puede heredar datos y atributos de otro objeto de su clase.

d) Polimorfismo. El polimorfismo es la capacidad de dos objetos diferentes para comportarse de forma distinta cuando reciben el mismo mensaje. El concepto de polimorfismo refleja el mundo real donde se reciben mensajes idénticos todos los días y se reacciona de manera diferente.

Como se puede apreciar, estas características proporcionan una gran potencia a una base de datos. Esta base de datos puede aprender de sí misma y reutilizar elementos cuando sea preciso. Asimismo, puede tratar objetos muy complejos de una forma muy simple. Esta introducción a las tecnologías orientadas a objetos servirá para entender mejor a Oracle 8i y hacia donde se dirige el futuro de las tecnologías de base de datos. Oracle 8i es una implementación de una base de datos objeto – relacional cuyos fundamentos se encuentran en la tecnología relacional. El futuro de esta base de datos esta en una evolución hacia las tecnologías orientadas a objetos. Es preciso apreciar y comprender que las capacidades de Oracle 8i para dar servicio a aplicaciones corporativas de misión crítica requiere mucho mas que la capacidad de crear objetos.

3.4.- Arquitectura de Oracle

Una vez hecha la introducción al conocimiento del Oracle Server y de la tecnología Objeto - Relacional, a continuación se tratará sobre la arquitectura de la base de datos Oracle8i con la finalidad de comprender algunos conceptos que ayuden a entender el funcionamiento de la base de datos.

Según Scott Martín (Abbey, 2000), uno de los principales desarrolladores de Oracle, definió una base de datos de la siguiente manera: “Es un montón de programas que manipulan archivos de datos”. Esta afirmación se acerca mucho a la realidad. Una base de datos es una colección de archivos de datos y el software que los manipula.

3.4.1.- Diccionario de datos

Oracle mantiene un *Diccionario de Datos*, que contienen información importante sobre la base de datos. Oracle lo utiliza para saber quien esta autorizado a registrarse en la base de datos, que archivos de datos están asociados con la misma y otras informaciones necesarias para que el sistema pueda funcionar. Oracle también almacena información sobre el aspecto de los datos de la base de datos.

3.4.2.- Archivos de datos.

La base de datos Oracle esta formada por uno o más archivos de datos, los cuales se agrupan para formar un espacio de tablas. Es especialmente importante destacar que los archivos de datos contienen todos los datos almacenados en la base de datos, semejante a las unidades de disco de una PC. Los archivos contenidos en dichas unidades de disco representan toda la información actualmente disponible en la PC.

3.4.2.1.- Datos de usuario y Datos del sistema

En los archivos de datos asociados con la base de datos se almacena dos tipos de datos: los datos de usuario y los datos de sistema.

- Los *datos de usuario* son los datos de la aplicación, junto con toda la información relativa a las aplicaciones. Esta es la información que una empresa almacena en una base de datos. La Tabla 3.4 muestra algunos tipos comunes de datos de usuario.
- Los *datos de sistema* constituyen la información que la base de datos necesita para gestionar los datos de usuario y a sí misma. Por ejemplo, con los datos de sistema, Oracle sabe que el campo de una tabla esta formado solo por números y ninguna letra, y que es un campo obligatorio. Los datos de sistema también dicen a Oracle cuales son los usuarios validos de la base de datos, cuales son sus contraseñas, cuantos archivos de datos forman parte de la base de datos y donde se localizan. La Tabla 3.5 muestra una serie de datos del sistema típicos.

3.4.3.- Espacio de tablas. Archivadores de Oracle

Puesto que una base de datos es una colección de archivos de datos, es muy importante entender como Oracle agrupa archivos en una base de datos. Lo hace mediante un objeto de base de datos denominado *espacio de tablas*. Antes de poder

introducir datos en una base de datos Oracle, hay que crear un espacio de tablas y luego un objeto dentro de dicho espacio para guardar los datos. Cuando se crea el objeto, se tiene que incluir toda la información sobre el tipo de datos que se desea almacenar.

Tabla 3.4. *Tipos comunes de datos de usuario.*

Tipos de datos	Contiene información sobre
Información de cliente	Apellidos, nombres, número de teléfono
Información de producto	Nombre de producto, disponibilidad, precio
Información Medica	Resultados de laboratorio, nombre de médico, nombre de enfermera
Información de almacén	Cantidad en almacén, cantidad bajo pedido.
Información de un sismo	Fecha, hora, latitud, longitud, profundidad, etc.

Tabla 3.5. *Tipos comunes de datos del sistema.*

Tipo de dato	Contiene información sobre
Tabla	Los campos de las tablas y el tipo de información que contienen.
Espacio	Cantidad de espacio físico que ocupan los objetos de la base de datos.
Usuario	Nombres, contraseñas, privilegios.
Archivos de datos	Nombre, localización, último tiempo de utilización.

La mejor analogía para explicar los conceptos de base de datos, espacio de tablas, archivos de datos, tabla y datos es la imagen de un archivo físico. Los cajones del archivo son los espacios de tablas; las carpetas de cada cajón son los archivos de datos; las hojas de papel de cada carpeta son las tablas y otros objetos de la base de datos y la información escrita en las hojas de cada carpeta son los datos. Los espacios de tablas son una forma de agrupar los archivos de datos.

3.4.3.1.- Nombres y contenido de los espacios de tablas

Oracle nombra con total libertad a los espacios de tablas, los cuales son muy descriptivos para cada espacio de modo que permite dar a conocer el tipo de dato que contienen. Los nombres que aquí se proporcionan son meramente un convenio aceptado y el cual el administrador de la base de datos no está obligado a utilizar, dado que el objetivo de los espacios de tablas es ayudar a organizar su base de datos.

a) Espacio de tablas del sistema. El espacio de tablas del sistema “*system*” es una pieza que necesitan todas las bases de datos Oracle. Aquí, Oracle almacena toda la información que necesita para gestionarse por sí sola, tal como los nombres de los espacios de tablas y los archivos de datos que cada espacio de tabla contiene.

b) Espacio de tablas temporales. El espacio de tablas temporales “*temp*” es donde Oracle almacena todas las tablas temporales. Es como el papel borrador de la base de datos. Del mismo modo, algunas veces es necesario un sitio donde apuntar algunos números para luego sumarlos, Oracle también necesita espacio de disco de forma periódica. En el caso de tener una base de datos muy activa, debe disponerse de más de un espacio de tablas temporales; por ejemplo, temp01, temp02 y temp03.

c) Espacio de tablas de herramientas. El espacio de tablas de herramientas “*tools*” es el lugar donde se almacena los objetos de la base de datos que necesitan las herramientas que se usan con la base de datos, tales como Oracle Reports que contiene su propio conjunto de tablas. La mayoría de los administradores de base de datos coloca las tablas que necesitan las herramientas en este espacio de tablas.

d) Espacio de tablas de usuario. El espacio de tablas de usuarios “*users*” contiene la información personal de los usuarios. Por ejemplo, mientras se hacen pruebas en el software los objetos se pueden ir ubicando en este espacio de tablas.

e) Espacio de tablas de datos e Índices. El nombre de los espacios de tablas

designados para la información de datos o de índices corresponde a la denominación que el administrador quiera darle.

f) Espacio de tablas de anulación. Todas las bases de datos Oracle necesitan un lugar para almacenar la información para deshacer. Este espacio de tablas, que contiene los segmentos de anulación, se denomina normalmente rollback o rbs. Una de las principales razones de utilizar un sistema de gestión de base de datos tal como Oracle es su capacidad para recuperar transacciones incompletas o interrumpidas, como parte de su principal funcionalidad.

3.4.4.- Registro de Rehacer: El Registro de Transacciones

Además de los archivos de datos asociados con un espacio de tablas, Oracle tiene otros archivos del sistema operativo asociado con dicho espacio, llamados *registros de rehacer en línea*. Otro término común para designar los registros de rehacer es registros de transacciones. Se trata de archivos del sistema operativo en los que Oracle registra todas las modificaciones o transacciones que se producen en la base de datos. Cuando se producen cambios en la base de datos, dichos cambios se realizan en memoria. Oracle mantiene en memoria estos cambios por razones de rendimiento. Una operación de E/S (entrada / salida) a disco es 1000 veces más lenta que una acción en memoria. Dado que siempre se registra una copia de todas las transacciones en los registros de rehacer en línea. Oracle puede tomarse su tiempo para registrar de nuevo en el archivo de datos original los cambios de datos que se produzcan en memoria. Al final, la copia definitiva del cambio de datos se registra de nuevo en el archivo de datos físicos. Puesto que todas las transacciones se registran en los registros de rehacer en línea, la base de datos siempre puede recuperarse desde dichos registros de transacción. Es un requisito que toda base de datos Oracle tenga al menos dos registros de rehacer en línea.

3.4.4.1.- Funcionamiento de los registros de rehacer

Los registros de rehacer trabajan en forma circular. Si se tienen una base de datos con dos registros de rehacer en línea, logA y logB, entonces, cuando las transacciones crean, borran y modifican los datos de la base de datos, primero quedan registradas en logA. Cuando logA esta lleno, se produce otra conmutación. Entonces, las transacciones se registran en logB. Cuando logB esta lleno, se produce otra conmutación, y de nuevo todas las transacciones se registran en logA. Dado que los registros de rehacer se usan en forma cíclica, cuando Oracle usa de nuevo logA la información sobre transacciones existentes en logA se sobrescribe.

Este punto lleva a la explicación de cómo funciona la base de datos Oracle en modo ARCHIVELOG y en modo NOARCHIVELOG, los cuales tienen una correlación directa con los registros de rehacer en línea.

a) Modo Archivelog. Cuando una base de datos se ejecuta en modo archivelog, se guardan todos los registros de rehacer de las transacciones. Esto quiere decir que se dispone de una copia de cada transacción realizada en la base de datos, y que incluso cuando los registros de rehacer trabajan de forma circular se hace una copia de cada registro de rehacer antes de sobrescribirlo. En el caso de que sea necesario conmutar antes de haber hecho la copia, Oracle se prepara hasta que se haya completado esta acción. Oracle no permitirá que se sobrescriba el registro de transacciones hasta tener una copia de él. Al disponer de una copia de todas las transacciones, la base de datos esta protegida contra todo tipo de fallos, incluyendo los errores de usuario o los fallos de disco. Este es el modo mas seguro de trabajar con la base de datos. El servidor de base de datos con el cual se ha desarrollado la estructura de la Base de Datos Sísmicos esta configurado en este modo.

b) Modo Noarchivelog. Cuando una base de datos se ejecuta en modo noarchivelog (modo predeterminado), no se guardan los registros de rehacer antiguos. Dado que no se guardan todos los registros de rehacer, solo esta protegido contra casos tales como fallos

de alimentación. Si se tiene en cuenta que cuando se llena un registro se produce una conmutación, y cuando se vuelve al registro original, la información previa se pierde.

Una base de datos que funcione en modo noarchive log no guardara una copia de los registros de rehacer antes de sobrescribirlos. Este modo establece el nivel mínimo de recuperación. Esta diseñado fundamentalmente para protección frente a fallos de instancia.

3.4.5.- Archivos de Control

Toda base de datos debe tener al menos un archivo de control, aunque es altamente recomendable disponer de dos o más por si alguno de ellos llegara a fallar mientras la base de datos esta funcionando. Si tiene un único archivo de control, se vera en apuros para garantizar el acceso de sus usuarios a la base de datos.

Un archivo de control es un pequeño archivo que contiene información clave sobre todos los archivos asociados con la base de datos Oracle. Los archivos de control preservan la integridad de la base de datos y ayudan a identificar que registros de rehacer son necesarios en el proceso de recuperación.

La mejor analogía en la que se puede pensar para ilustrar este punto es la revisión anual de un coche. Todos los años, cuando los coches tienen mas de 10 años, sus conductores deben llevarlos a una revisión para asegurarse de que cumplan las normatividades de seguridad y polución.

3.4.5.1.- Funcionamiento de los archivos de control

Antes de que la base de datos comience a trabajar, acude al archivo de control para determinar si la base de datos se encuentra en un estado aceptable. Por ejemplo, si falta un archivo de datos o se ha alterado un archivo determinado mientras que la base

de datos no se estaba usando, el archivo de control informa a la base de datos de que no ha pasado la inspección. Si esto ocurre, al igual que en la inspección del coche, no se podrá continuar hasta que se haya corregido el problema.

Si Oracle lee el archivo de control y, en función a la información que contiene, determina que el estado de la base de datos no es correcto, no permitirá que la base de datos entre en funcionamiento.

Si se realiza un punto de comprobación de la base de datos o si hay un cambio en la estructura de la misma, el archivo de control se actualiza. Si no se dispone de un archivo de control válido, la base de datos no arrancará.

3.4.6.- Programas

Se ha definido anteriormente una base de datos como “un montón de programas que manipulan archivos de datos”. Ahora se va a tratar los programas, que preferentemente se van a denominar procesos, dado que cada vez que un programa se inicializa en la base de datos se comunica con Oracle a través de un proceso. Existen dos tipos de procesos Oracle: usuario y servidor.

a) Procesos de usuario (Cliente). Los procesos de usuario trabajan con el cliente de la base de datos solicitando información a los procesos de servidor. Ejemplos de procesos de usuario son: Oracle Developer y SQL*Plus. Estas herramientas son comunes y cualquier usuario de la base de datos puede utilizar para comunicarse con ella.

b) Procesos de servidor. Los procesos de servidor reciben las solicitudes de los procesos de usuario y se comunican con la base de datos. A través de esta comunicación, los procesos de usuario trabajan con los datos de la base de datos. Una buena forma de pensar en el proceso cliente/servidor es imaginarse en un restaurante. El cliente comunica al camarero lo que desea tomar. Este comunica a la cocina su solicitud. El trabajo del personal de cocina es preparar la comida, decir al camarero

cuando esta lista y hacer inventario. El camarero entonces la lleva la comida. En este símil, el camarero representa el proceso de cliente y el personal de cocina representa el proceso de servidor.

3.4.7.- Procesos Auxiliares de la Base de Datos

A continuación se va a describir un conjunto especial de procesos de servidor que ayudan a operar a la base de datos.

a) Escritor de base de datos (DBWR). El escritor de base de datos es un proceso obligatorio que escribe los bloques de datos modificados en los archivos de la base de datos. Es uno de los dos únicos procesos que permiten escribir en los archivos de datos y constituyen la base de datos Oracle. Con ciertos sistemas operativos, por razones de rendimiento, Oracle permite tener múltiples escritores de bases de datos.

b) Punto de comprobación (Checkpoint, CKPT). Cuando los usuarios están trabajando con una base de datos Oracle realizan solicitudes para acceder a los datos. Dichos datos se leen de los archivos de la base de datos y se colocan en un área de memoria donde los usuarios puedan consultarlos. Finalmente, algún usuario realiza un cambio en los datos que debe registrarse de nuevo en los archivos de datos originales. Cuando se realiza una conmutación de los registros de rehacer, se produce un checkpoint o punto de comprobación. En ese momento, Oracle consulta la memoria y escribe la información de los bloques de datos modificados en el disco. También notifica al archivo de control que se ha producido una conmutación de los registros de rehacer. Estas son las tareas que realiza el proceso de punto de comprobación.

c) Escritor de registro (LGWR). El escritor de registro es un proceso obligatorio que escribe los datos de rehacer en los registros de rehacer. Gracias a ello, Oracle puede recuperarse de distintos tipos de fallo. Además, puesto que una copia de cada transacción se escribe en el registro de rehacer, Oracle no gasta constantemente sus recursos escribiendo las modificaciones de los datos en los archivos de datos de forma

inmediata. Esto mejora el rendimiento. El escritor de registro es el único proceso que escribe en los registros de rehacer, y también, es el único proceso en una base de datos Oracle que lee los registros de rehacer.

d) Monitor de sistema (SMON). El monitor de sistema es un proceso obligatorio, que lleva a cabo cualquier recuperación de información necesaria durante la inicialización. En el modo servidor paralelo base de datos Oracle con diferentes computadoras que comparten el mismo conjunto de discos, véase Figura 3.4 (Abbey et al., 2000). También puede realizar la recuperación de una base de datos que haya fallado en otra computadora. Recuerde que las dos bases de datos comparten los mismos archivos de datos.

e) Monitor de proceso (PMON). El monitor de proceso es un proceso obligatorio que lleva a cabo una tarea de recuperación cuando un usuario de la base de datos falla. Asume la identidad de usuario, libera todos los recursos de la base de datos que tenía el usuario y deshace la transacción interrumpida.

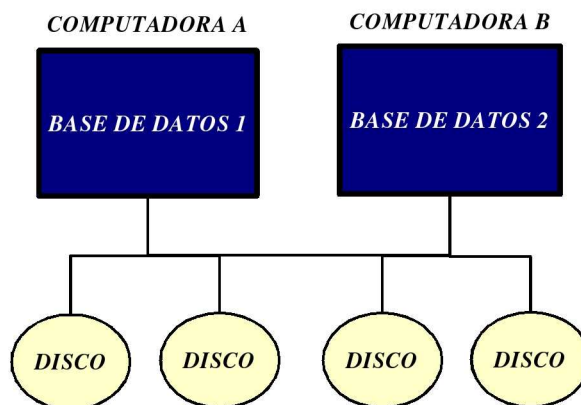


Figura 3.4. Esquema de Oracle en modo servidor paralelo.

f) Archivador (ARCH). El archivador es un proceso opcional. Tal como se ha

comentado anteriormente, los registros de rehacer (transacciones) se escriben en forma secuencial. Cuando un registro se llena, se conmuta al siguiente registro de rehacer disponible. Cuando una base de datos esta funcionando en modo archivelog, la base de datos hace una copia del archivo de rehacer para que, cuando la base de datos conmute de nuevo a ese registro de rehacer, se disponga de una copia del contenido de ese archivo para propósitos de recuperación. Este es el trabajo del proceso de archivador. Hace una copia del archivo de forma similar a una fotocopidora.

g) Bloqueo (LCKn). El bloqueo es un proceso opcional. Cuando la base de datos Oracle esta funcionando en modo servidor paralelo se crean múltiples procesos lck. En este modo (véase Figura 3.4), los bloqueos ayudan a que las bases de datos se comuniquen.

h) Recuperador (RECO). Este proceso opcional sólo se emplea cuando la base de datos se ejecuta con la opción distribuida de Oracle. Una transacción distribuida es aquella en la que dos o más ubicaciones de los datos deben mantenerse sincronizadas.

i) Expedidor (Dispatcher, Dnnn). Los expedidores son procesos opcionales de segundo plano y están presentes sólo cuando se usa una configuración de servidor multihebra. Para cada protocolo de comunicación en uso (por ejemplo, TCP/IP, SNA) se crea al menos un proceso expedidor (D000, ..., Dnnn). Cada proceso expedidor es responsable del encaminamiento de las solicitudes desde los procesos de usuarios conectados hasta los procesos de servidor compartidos disponibles y del retorno de las respuestas al proceso de usuario apropiado.

j) Procesos de cola de trabajo (SNPn). Estos procesos son los que proporcionan la tecnología de instantáneas de Oracle. Se activan regularmente y refrescan aquellas instantáneas que estén configuradas para actualizarse periódicamente. El DBA puede configurar hasta 36 procesos de este tipo y si se genera más de uno se puede compartir el refresco de instantáneas entre los procesos disponibles.

k) Procesos de monitor de cola (QMn). Este proceso sirve para implementar la opción de cola avanzada de Oracle8i (Advanced Queuing, AQ), e integra un sistema de mensajes en cola. Con esta opción, los mensajes se pueden almacenar en la base de datos para su envío inmediato o rezagado. Los sistemas acceden a la funcionalidad que proporciona AQ usando lenguajes como java, PL/SQL o Visual Basic. La opción AQ proporciona muchas otras funciones, entre las que se podría destacar la prioridad y ordenación de mensajes en la cola y la capacidad de enviar mensajes a colas remotas o locales.

3.4.8.- Estructura de Memoria

Hasta aquí se ha tratado de archivos de datos y programas. También se ha tratado los procesos del servidor y de los procesos de cliente. En este punto se va a tratar la forma cómo se comunican los procesos de servidor y de cliente entre sí, mediante las estructuras de memoria. Como su nombre lo indica, se trata de áreas de memoria especiales donde un proceso puede comunicarse consigo mismo y con otros procesos.

Oracle utiliza dos tipos de estructuras de memoria: El área global del sistema o SGA y el área global de programa o PGA.

3.4.8.1.- Área Global del Sistema (SGA)

El SGA es el lugar de la memoria donde la base de datos Oracle almacena la información sobre sí misma. Lo hace así dado que la memoria es la forma más rápida y más eficiente de permitir que los procesos se comuniquen. Esta estructura de memoria es accesible para todos los procesos de usuario y de servidor. En la Figura 3.5 se muestra como el SGA se encuentra en el centro de todas las comunicaciones (Oracle8i Concepts, 2000).

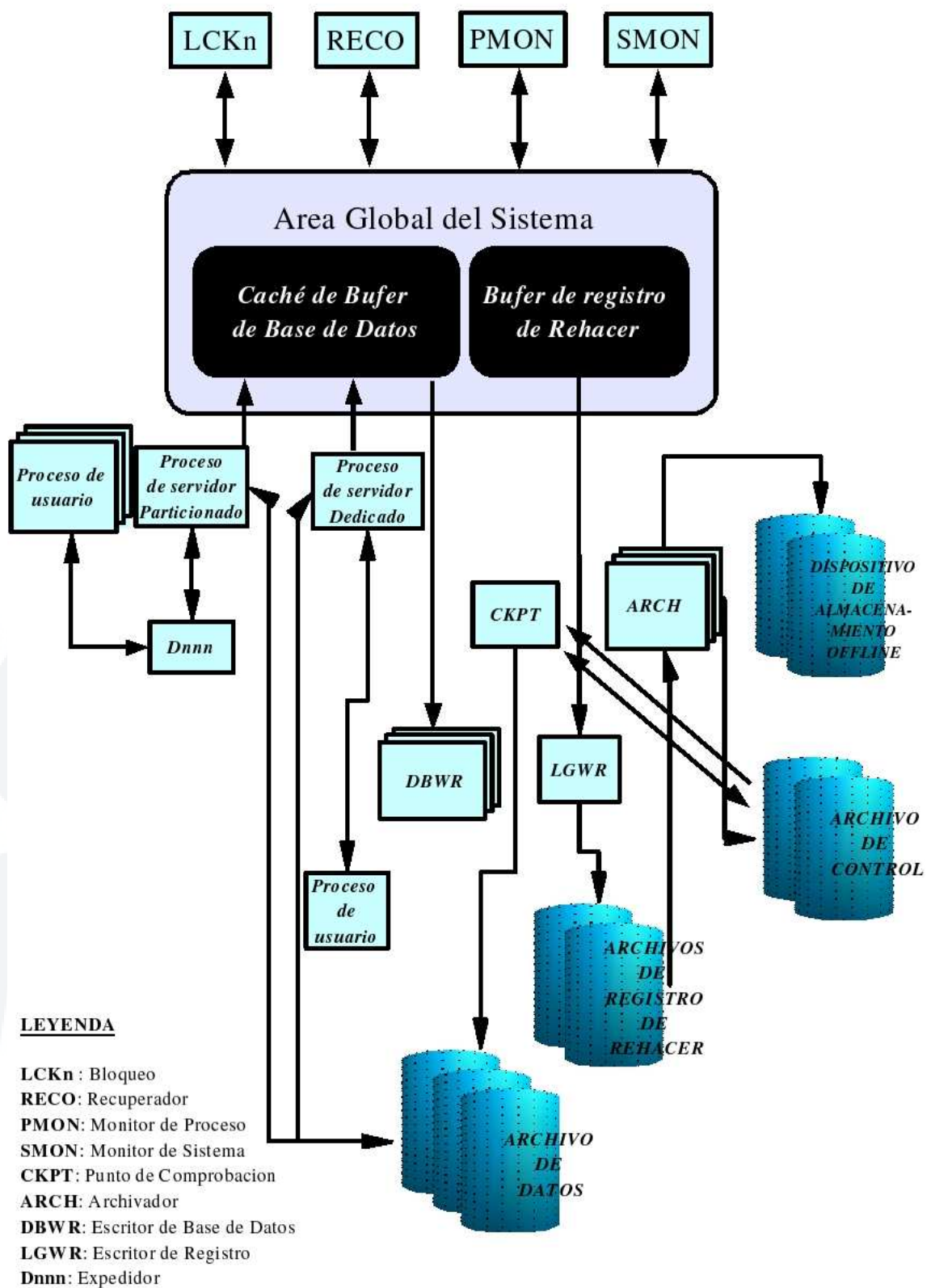


Figura 3.5. Esquema de la estructura de memoria y de los procesos de Oracle.

Dado que el SGA es el mecanismo por el que distintos procesos de cliente y de servidor se comunican, es importante que se entienda sus diferentes componentes. El SGA de Oracle server se divide en las siguientes componentes fundamentales.

a) Caché del búfer de datos. La caché de búfer de datos es donde Oracle almacena los bloques de datos mas recientemente utilizados. En otras palabras, es la memoria caché de datos. Cuando se introduce información en la base de datos, estos se almacenan en bloques de datos. La caché de búfer de datos es un área de memoria en la que Oracle coloca dichos bloques de datos para que un proceso de usuario pueda acceder a ellos. Antes de que ningún proceso de usuario pueda acceder a parte de los datos, estos deben residir en la caché del búfer de datos. El tamaño de la caché del búfer de datos tiene un limite físico y cuando Oracle la haya llenado deja los bloques de mayor uso en la memoria caché y elimina los bloques de menor uso. Este se hace mediante el algoritmo LRU (menos uso recientemente).

Un punto importante que se debe clarificar es que, si un proceso de cliente necesita cierta información que no esta en la memoria caché, la base de datos accede a la unidad de disco físico, lee los bloques de datos necesarios y los coloca en el caché de búfer de datos. Esto hace que todos los procesos de servidor y de cliente puedan aprovecharse de la lectura de disco físico.

b) Caché de diccionario (caché de fila). Un caché de diccionario esta formado por filas extraídas del diccionario de datos. El diccionario de datos contiene toda la información que Oracle necesita para gestionarse; es decir, información sobre los usuarios que tienen acceso a la base de datos Oracle, sobre los objetos que la base de datos posee y donde se ubican dichos objetos.

c) Búfer de registro de rehacer. Recuerde que otro nombre común para designar a los registros de rehacer es de registros de transacciones. Cualquier transacción que pueda registrarse en el registro de rehacer (los registros de rehacer en línea se necesitan para propósitos de recuperación), debe residir primero en el búfer de registro de rehacer. Se

trata de un área de memoria reservada para este propósito. Periódicamente, la base de datos, vacía este búfer en los registros de rehacer en línea.

d)Área SQL compartida. Se debe pensar en un área de SQL compartida como en la memoria caché del programa. Es el área donde se almacenan todos los programas. Los programas de una base de datos Oracle se basan en un lenguaje estándar denominado SQL. Esta memoria caché contiene todas las instrucciones SQL ya analizadas y que están listas para ejecutarse.

Resumiendo, el SGA es el gran comunicador. Es el lugar de la memoria donde se coloca la información a la que pueden acceder los procesos de servidor y de cliente. Esta dividido en varias áreas: la memoria caché de datos, la caché de registro de rehacer, la caché de diccionario y la caché SQL compartida. En la Figura 3.6 se muestra las memorias caché que Oracle mantiene en el SGA. Se denomina *sqlarea* a la memoria caché SQL pudiendo usar ambos términos indistintamente.

3.4.8.2.- Área Global de Programa (PGA)

Un PGA es un área de memoria utilizada por un único proceso Oracle. El *área global de programa* no se comparte ya que contiene datos e información de control para un proceso único. Esta área contiene información tal como matrices internas y variables de secciones de proceso. Al igual que un sistema de intercomunicadores domestico, las distintas partes del proceso pueden comunicarse entre sí, pero no con el mundo exterior.

3.4.9.- Instancias

Para acceder a los datos de la base de datos, Oracle utiliza los procesos en segundo plano o procesos auxiliares que comparten todos los usuarios. Además, se comparte las estructuras de memoria que almacenan los datos consultados recientemente (Memoria Caché de datos). Conociendo esto, se puede definir a una instancia de base de datos (también conocida como *servidor*) como un conjunto de

estructuras de memoria y procesos en segundo plano que acceden a un conjunto de archivos de base de datos (Loney, 2000). Dicho sencillamente, una instancia Oracle es un conjunto de procesos de servidor Oracle que tienen su propia área global del sistema y un conjunto de archivos de base de datos asociados con ello. Por ejemplo, si se tiene una computadora que contiene dos bases de datos denominados DB1 y DB2, cada base de datos tiene su propio SGA y su propio conjunto de procesos de servidor Oracle, entonces tiene dos instancias de la base de datos, tal como se muestra en la Figura 3.7.

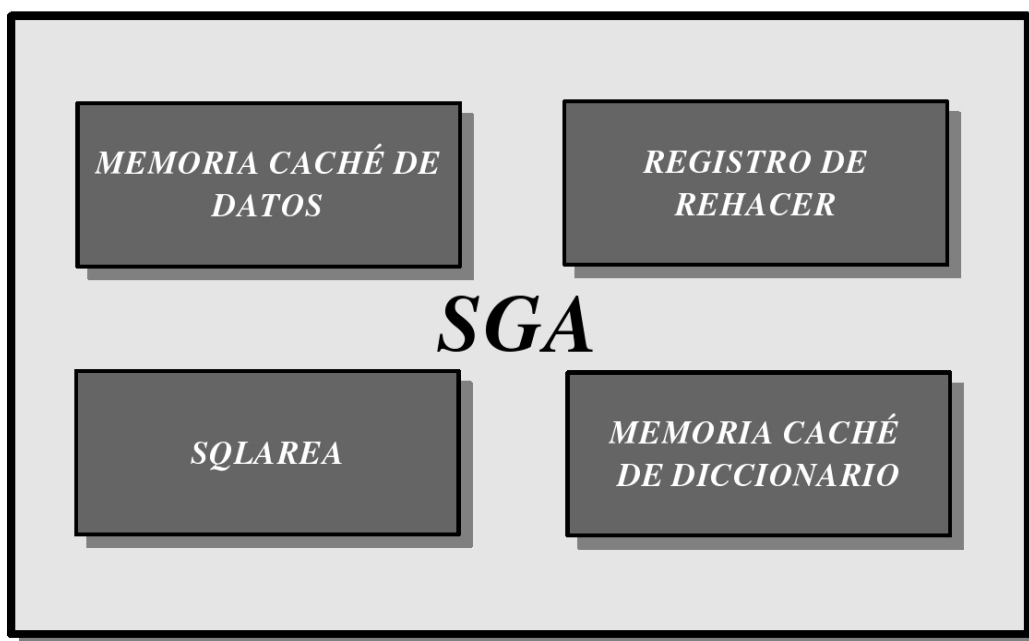


Figura 3.6. El Area Global del Sistema: Una vista más detallada.

Para que la base de datos no se confunda, cada instancia se identifica por medio de lo que se conoce como SID (Identificador de Sistema). En la mayor parte de las computadoras Unix se define mediante la variable “ORACLE_SID”. Luego, se nombra a cada uno de los procesos de servidor de acuerdo con el SID correspondiente. Por ejemplo, en la base de datos DB1, los procesos deberían nombrarse de la siguiente forma: ora_BD1_dbw0, ora_BD1_pmon, ora_BD1_smon, ora_BD1_lgwr, etc.

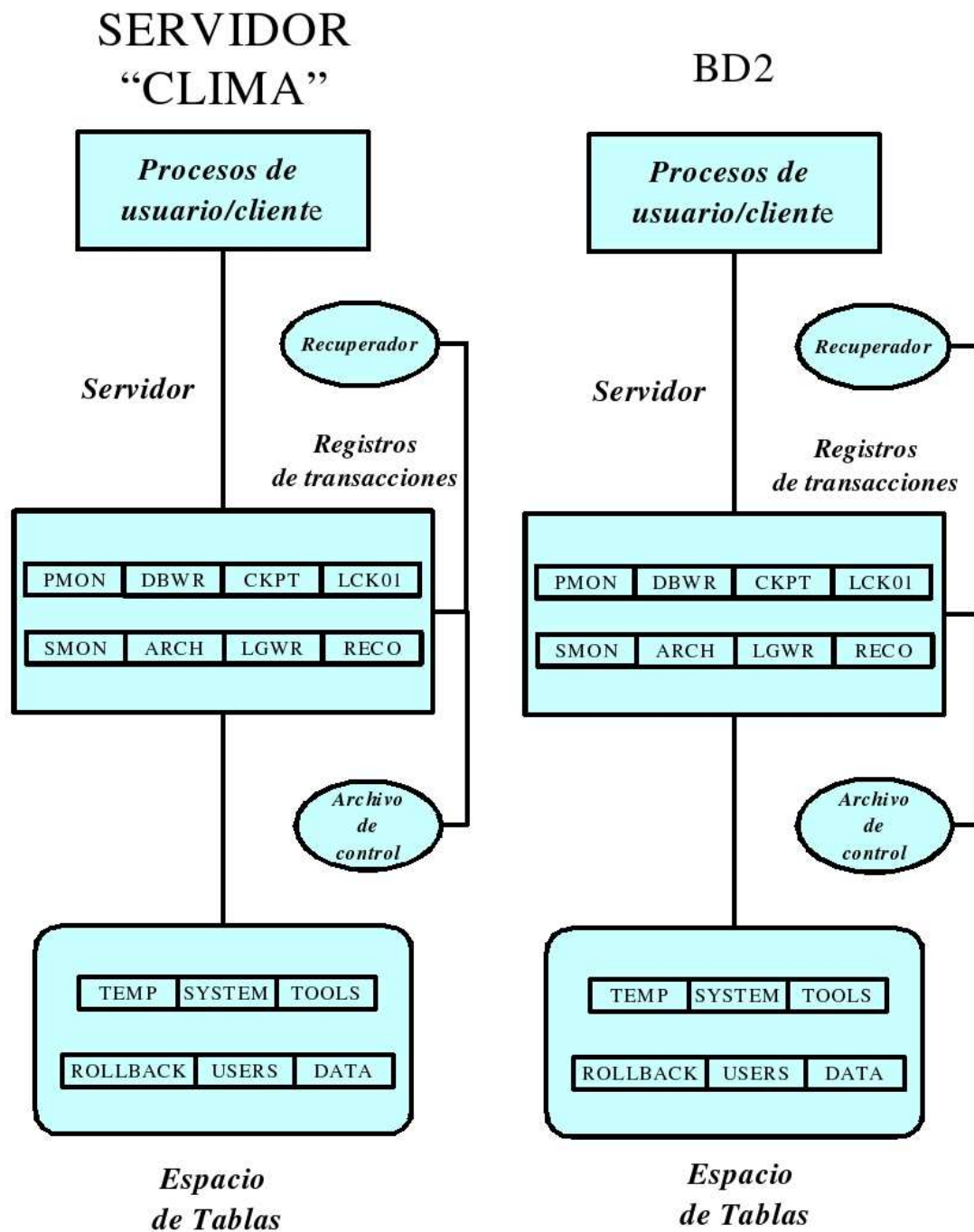


Figura 3.7. Esquema que muestra 2 instancias de una base de datos Oracle.

3.5.- El servidor de base de datos “Clima”

La Base de Datos Sísmicos ha sido desarrollada dentro de una instancia de Oracle denominada “Clima”; es decir, el Administrador de la Base de Datos Sísmicos, DBA_CNDG, representa un usuario del servidor “Clima”, y comparte el área de memoria, los procesos auxiliares y los espacios de tablas del sistema con los demás usuarios (DBA_CLIMA, etc.).

3.5.1.- Características del Hardware del Servidor de Base de Datos

El Hardware del servidor de base de datos tiene las siguientes características:

- Servidor Base de Datos Centralizado denominado “CLIMA”
- Workstation Compaq AlphaServer DS20
- Sistema Operativo OSF1 V4.0F Tru64 UNIX
- Oracle 8i Standard Edition, Versión 8.1.6 para Compaq Tru64 UNIX
- Procesador 21264, Velocidad CPU 500MHz
- 1GB RAM, Memoria Cache 4MB
- Velocidad de Bus 5.2GB/s
- 3 disco duro de 9.1 GB
- Unidad de Cinta 4mm 14GB, CD-ROM 32X
- Interface de Red Ethernet 10/100 RJ-45
- Interface SCSI Ultra Fast Wide
- Adaptador Gráfico 8MB

- Para las PCs: Oracle 8i Client, Versión 8.1.6 para Microsoft Windows 98/NT/2000
- Número IP: 200.4.215.164

3.5.2.- Seguridad del Servidor de Base de Datos “Clima”

El Servidor Clima emplea los mecanismos de seguridad que Oracle le ofrece; es decir, en función al nombre de usuario con el que se conecta a la base de datos. Este mecanismo de seguridad consiste en otorgar a determinados usuarios atributos o de asignarles algunas restricciones las cuales son dadas por el Administrador de Base de Datos. Por ejemplo, EL DBA_CNDG es un usuario encargado de implementar la Base de Datos Sísmicos, entonces a este se le concederá atributos que le permitan realizar dicha labor. El usuario denominado GUEST_CNDG es tan solo un consultor de datos, por lo que ha este solo se le dará únicamente atributos de selección de datos. Es probable que exista otro tipo de usuarios a los que se les dé únicamente acceso a la información de algunos de los objetos de datos y no a otros, esta restricción también es posible. Toda la información referente a las restricciones y concesiones que se les da a los usuarios son almacenadas en el Diccionario de datos de Oracle y por esta razón, es muy difícil que un usuario pueda realizar tareas dentro de la base de datos que Oracle no lo permita.

CAPITULO 4

ESTRUCTURA Y DISEÑO DE LA BASE DE DATOS SISMICOS

4.1.- Introducción

El presente capítulo tratará la creación de los objetos de la Base de Datos Sísmicos que permitan almacenar la información sísmica producida en el estudio de los sismos que ocurren en Perú, desarrollado por el CNDG - Sismología. Para tal objetivo, se empezará tratando sobre el Lenguaje de Consulta Estructurado (Structured Query language SQL) o lenguaje estándar para todos los software de bases de datos y como a través de un conjunto de instrucciones ejecutadas desde una interfaz amigable proporcionada por Oracle (denominada SQL*Plus), le permite al usuario (DBA_CNDG) desarrollar objetos dentro de la base de datos e interactuar con la información que almacena en ellos. Asimismo, se tratarán las relaciones existentes entre los objetos *tablas* que almacenan los datos, con el fin de garantizar la Relacionalidad y la Integridad Relacional de la información (http://ar.geocities.com/r_niella/Document/t_cap5.htm). Igualmente se explicara la creación de algunos objetos de la base de datos adicionales a las tablas (vistas, índices y otros) y que conforman la estructura de almacenamiento de la base de datos.

4.2 SQL*Plus, La Interfaz de Usuario Amigable

SQL*Plus es una herramienta de Oracle que permite trabajar directamente con la base de datos. El SQL*Plus es una interfaz, tanto para el lenguaje PL/SQL como para las sentencias o comandos del Lenguaje de Consulta Estructurado, SQL (Structured Query Language). Además, es una herramienta que realiza la tarea de elaborar los reportes que se envían a los clientes de la base de datos y al servidor Oracle. Con SQL*Plus, el usuario

puede comunicarse interactivamente con una base de datos Oracle empleando sentencias SQL. Este lenguaje es un estándar adoptado por todos los fabricantes de Softwares de Base de Datos. Además de esto, el SQL*Plus de Oracle es un superconjunto del SQL estándar; es decir, cumple con el estándar de los lenguajes compatibles con SQL y tiene además una serie de extensiones específicas de Oracle, de ahí el nombre: SQL + Plus.

Usando el SQL*Plus, se puede ingresar tres tipos de sentencias desde su línea de comandos:

- *Comandos SQL*Plus*; estos comandos son usados para establecer opciones para SQL*Plus, formato de reportes, editar archivos, editar los comandos del buffer y otras operaciones más. Estos comandos no son interactivos con la base de datos. Estos comandos no terminan en punto y coma (;) como los comandos SQL.
- *Comandos SQL*; estos comandos interactúan directamente con los datos de las bases de datos.
- *Bloques PL/SQL*; son programas o segmentos de programas con los cuales se manipula la información de las base de datos Oracle.

4.2.1.- El lenguaje de Consulta Estructurado, SQL

El Structured Query Language o SQL, es un lenguaje que provee una interfaz para los *Sistemas de Base de Datos Relacionales*. Cuando Oracle Server trabaja únicamente entiende las instrucciones expresadas mediante sentencias SQL; además, cuando alguna herramienta de Oracle interactúa con la base de datos, lo único que transmite para su procesamiento son instrucciones SQL. La implementación Oracle de SQL (SQL*Plus) cumple con los estándares ANSI (American National Standards Institute) e ISO

(International Standards Organization).

Las instrucciones SQL estándar que se admiten en las bases de datos Oracle, se realizan con los siguientes fines:

a) *Lenguaje de Definición de datos o DLL (Data Definition Language)*. Permite realizar las siguientes tareas:

- La creación de un objeto de base de datos.
- La eliminación de un objeto de base de datos.
- La modificación de un objeto de base de datos.
- La concesión de privilegios sobre un objeto de base de datos.
- La revocación de privilegios sobre un objeto de base de datos.

Las instrucciones que pertenecen a la categoría DDL se confirman automáticamente, esto significa que cuando Oracle8i envía un mensaje “Revoke succeeded” (revocación realizada con éxito), la orden se ha completado y no se puede deshacer. Por lo tanto, el lenguaje de definición de datos es un conjunto de ordenes SQL que crea y define objetos de base de datos y almacena sus definiciones en el Diccionario de Datos. En la Tablas 4.1 se presentan algunas sentencias DDL.

b) *Lenguaje de manipulación de datos o DML (Data Manipulation Language)*. Se emplean para crear nuevos datos, trabajar con los datos ya existentes, eliminar información o simplemente para seleccionar algunos de ellos o todos. En la Tabla 4.2 se muestra algunas

de las sentencias utilizadas generalmente.

Tabla 4.1. Lista parcial de ordenes DDL.

Orden SQL.	Proposito
<i>CREATE.</i>	Crea objetos en la Base de datos
<i>ALTER.</i>	Cambia la estructura de la Base de Datos
<i>DROP.</i>	Borrar objetos de la base de datos
<i>TRUNCATE.</i>	Remueve todos los registros de una tabla, inclusive, todos los espacios asignados para los registros son removidos.
<i>COMMENT.</i>	Añade comentarios al Diccionario de datos
<i>GRANT.</i>	Concede privilegios de acceso de usuarios a la base de datos
<i>REVOKE.</i>	Retira los privilegios de acceso concedidos con el comando Grant.

Tabla 4.2. Lista parcial de ordenes DML.

Orden SQL	Propósito
<i>SELECT</i>	Permite ver o recuperar datos de una base de datos.
<i>INSERT</i>	Inserta datos dentro de una tabla de una base de datos.
<i>UPDATE</i>	Actualiza los datos existentes en una tabla.
<i>DELETE</i>	Borra todos los registros de una tabla. Los espacios de los registros permanecen.
<i>CALL</i>	Llama o ejecuta un subprograma en PL/SQL o Java.
<i>EXPLAIN PLAN</i>	Explica la ruta de acceso a los datos.
<i>LOCK TABLE</i>	Control de concurrencia.

c) *Lenguaje de Control de Datos o DCL (Data Control Language)*. Se emplean para el control de los datos. Estos son vistos de la Tabla 4.3.

Tabla 4.3. Lista de órdenes DCL.

Orden SQL	Propósito
<i>COMMIT</i>	Guarda los cambios y las labores realizadas en la base de datos.
<i>SAVEPOINT</i>	Identifica un punto en una transacción que puede ser recuperada o restablecido.
<i>ROLLBACK</i>	Restaura la base de datos considerando la información y la configuración desde el último COMMIT.
<i>SET TRANSACTION</i>	Cambia opciones de transacción que el segmento de Rollback usa.

Una vez conocidas algunas de las sentencias SQL, a continuación se definirá algunos de los objetos de la base de datos y como se pueden generar empleando sentencias SQL desde el SQL*Plus de Oracle.

4.3.- Objetos de la Base de Datos

Los objetos de la base de datos Oracle se definen como elementos dotados de significado en el que se almacena información. Normalmente se habla de tipos de objetos siendo los más comunes las *Tablas* y las *Vistas*, pero son importantes también los *Sinónimos*, los *Indices* y los *Papeles*.

4.3.1.- Tablas

Una Tabla es un objeto de la base de datos que almacena información. En el momento en que se crea una tabla en una base de datos Oracle, se almacena información de

esta en el “*Diccionario de Datos de Oracle*”, el cual contiene toda la información referente a todas las tablas creadas dentro de la base de datos. Dicha información se refiere básicamente al nombre de las columnas que conforman dicha tabla y principalmente el tipo de dato asociado a cada una de estas. Una lista y descripción de los tipos de datos en los que pueden estar definidas las columnas de una tabla en Oracle se ve en la Tabla 4.4. Un ejemplo de como se genera una tabla haciendo uso del SQL*Plus de Oracle se ve a continuación:

```
SQL> create table CATALOGO_1983_ACT
(
  tiempo_id      varchar2(25),
  fecha          date,
  latitud         number,
  error_lat       number,
  longitud        number,
  error_lon       number,
  profundidad     number,
  error_pro       number,
  mb1             number,
  mb2             number,
  gap             number,
  rmc             number,
  estaciones      number,
  fasesP         number,
  fasesS         number,
  sismoid         varchar2(9),
  primary key (sismoid)
);
commit;
```

El nombre de la tabla es *catalogo_1983_act*, y tiene 16 campos o columnas (*tiempo_id*, *fecha*, *latitud*, *error_lat*, *longitud*, *error_lon*, etc.), definidas cada una de ellas con un tipo de dato (*varchar2*, *number*, *date*, etc.).

Cuando se define una tabla en una base de datos, los campos o columnas pueden tener *restricciones*. Dichas *restricciones* le van a permitir a dicha tabla relacionarse con otras tablas, determinar si se puede incluir valores nulos en dicho campo o simplemente que cumplan un cierto criterio impuesto por el Administrador de la Base de Datos. Las *restricciones* de la base de datos ayudan a garantizar la *Integridad Referencial* de los datos. De esta forma, uno puede estar seguro de que todas las referencias de la base de datos son validas y de que se cumplan todas las restricciones. Estas restricciones son:

- *Clave Primaria o Primary Key*; hacen que la columna o columnas sea única y obligatoria; es decir, no acepta valores nulos.
- *Clave Foránea o Foreign Key*; esta hace referencia a una Clave Primaria definida previamente en alguna otra tabla de la base de datos.
- *Obligatorio o Not Null*; significa que esta columna no puede estar vacía.
- *Default*; permite generar un valor en una columna cuando se inserta una fila en la tabla, sin especificar un valor.
- *Check*; se utiliza para asegurarse que los valores de una determinada columna cumplan un cierto criterio.
- *Unique*; se utiliza para especificar la unicidad de columnas que deben ser univocas, pero que no formen parte de la clave primaria.

Tabla 4.4. Tipos de datos en los que puede estar definido un campo o columnas de una tabla Oracle.

Tipos de Datos	Descripción
CHAR	Campo de longitud fija, con un máximo de 2000 caracteres.
VARCHAR	En la actualidad idéntico a VARCHAR2, aunque su funcionalidad puede cambiar en versiones futuras. Por tanto, para almacenar cadena de caracteres de longitud variable debe utilizarse el tipo de dato VARCHAR2
VARCHAR2	Campo de longitud variable, con una longitud máxima de 4000 caracteres.
DATE	Campo de una longitud fija de 7 bytes, que se utiliza para almacenar todas las fechas. La hora se almacena como parte de la fecha. En las consultas, la fecha aparece en el formato DD-MES-AA. Por ejemplo, 13-APR-99.
NUMBER	Columna numérica de longitud variable. Los valores permitidos son el cero y los números positivos y negativos comprendidos entre 1.0E-130 y 9.99E125.
LONG	Campo de longitud variable, con una longitud máxima de 2 Gb.
RAW	Campo de longitud variable para datos binarios, de una longitud máxima de 4000 caracteres.
LONG RAW	Campo de longitud variable para datos binarios, de una longitud máxima de 2 Gb.
BLOB	Objeto binario de gran tamaño, de hasta 4 Gb de longitud.
CLOB	Objeto de caracteres de gran tamaño, de hasta 4Gb de longitud.
NCLOB	Tipo de datos CLOB para conjuntos de caracteres multibyte, de una longitud máxima de 4Gb.
BFILE	Archivo binario externo. El sistema operativo dicta el tamaño máximo.

4.3.2.- Vistas

Una vista es un objeto de la base de datos que permite crear una selección personalizada de una tabla o de una colección de tablas. A diferencia de una Tabla, una vista no tiene ningún dato: únicamente contiene una consulta SQL. Los datos que se recuperan de esa consulta se presentan en forma de tabla. Al igual que una tabla, se puede insertar, actualizar, borrar y seleccionar los datos de una vista. Las vistas son muy importantes y pueden ser utilizadas para las siguientes razones:

- Las vistas pueden proporcionar un nivel adicional de seguridad de los datos.
- Las vistas permiten ocultar la complejidad de los datos. Una base de datos puede estar conformada por varias tablas y se puede recuperar de una o mas de estas la información que se desee sin que el usuario que realiza la consulta sepa de que tablas las obtuvo.
- Las vistas ayudan a mantener la claridad de la nomenclatura. A menudo cuando se asigna los nombres de columnas en una tabla, los DBA asignan nombres que pueden resultar complicados a la hora de redactar la consulta en SQL.
- Las vistas permiten cambiar la estructura de las tablas subyacentes sin necesidad de cambiar el código de la aplicación. Si una vista asociada con dos tablas, la cual presenta tres columnas de una tabla y cuatro de la otra. Si se altera el contenido de la primera con una columna mas la definición de la vista no se ve afectada y cualquiera de las aplicaciones que den referencia a dicha vista no precisara revisión.

En Oracle una vista se crea con la orden *create view*. Un ejemplo de como se crea una vista a partir de una tabla (*catalogo_1983_act*), se ve a continuación:

```
SQL> create view catalogo_1983_act_vis as
      select
      timpo_id, latitud, longitud, profundidad, mb1, mb2
      from catalogo_1983_act;
```

Esta vista denominada *catalogo_1983_act_vis*, esta conformada por las columnas *fecha*, *latitud*, *longitud*, *profundidad* y *mb1* y *mb2* de la tabla *catalogo_1983_act*. Esto favorece el trámite de la consulta de estos parámetros, en el sentido de que este se realizará con mayor rapidez.

4.3.3 Índices

Del mismo modo como el índice de un libro ayuda a buscar rápidamente la información, un índice asociado a una tabla ayuda a recuperar los datos rápidamente. Un Índice es una minicopia de una tabla. Los elementos de índice de una tabla permiten a Oracle acceder rápidamente a los datos contenidos en una tablas.

En si, los índices permiten consultar una parte de la información de una tabla especificada por cada columna sin tener que recorrer por la información restante. Por ejemplo, si en la tabla *catalogo_1983_act*, se desea consultar la información de la columna *time_id* únicamente, Oracle necesariamente recorrerá toda la tabla. Sin embargo, utilizando los Índices esta consulta solo se hará en dicha columna sin necesidad de recorrer por toda la información restante, lo que permitirá hacerla de manera rápida.

Oracle permite crear índices concatenados, que se forman con mas de una columna. Muchas veces en las tablas, jamas se consulta una columna sin consultar otra en particular, es para estos casos en que es necesario concatenar las columnas. La instrucción SQL que permite realizar un índice concatenado empleando la tabla creada anteriormente es:

```
SQL> create index catalogo_1983_act_ind on catalogo_1983_act  
      (time_id, latitud, longitud, profundidad, mb1, mb2);
```

Este índice denominado *catalogo_1983_act_ind*, esta hecho en base a las columnas *time_id*, *latitud*, *longitud*, *profundidad*, *mb1*, *mb2* de la tabla *catalogo_1983_act*.

4.3.4.- Sinónimos

Un sinónimo es un objeto de la base de datos que permite crear nombres alternativos para las vistas y tablas de Oracle. El objetivo de crear Sinónimos de una tabla

Oracle es básicamente por las siguientes razones:

- Para ocultar el nombre del creador de la tabla o nombre de una tabla.
- Porque se quiere o se necesita ocultar la verdadera ubicación de la tabla.
- Para proporcionar a los usuarios nombres de tabla menos complicados que los nombres reales.

La instrucción SQL que permite crear un sinónimo es *create synonym*. Un ejemplo de como se crea un Sinónimo a partir de la tabla *catalogo_1983_act* es la siguiente:

```
SQL> create synonym cat_83_act for dba_cndg.parametros;
```

El sinónimo denominado *cat_83_act* reemplaza en las sentencias SQL al nombre de la tabla *dba_cndg.catalogo_1983_act*, donde *dba_cndg* representa el nombre del creador y propietario de la tabla que opcionalmente no se desea hacer saber a los usuarios.

Una vez definidos los objetos que pueden integrar una base de datos Oracle, se empieza a describir las tablas que conforman la estructura de almacenamiento de los datos sísmicos.

4.4.- Estructura de Tablas para la Base de Datos Sísmicos

Como ya se mencionó anteriormente, la información y datos que se ha considerado para el desarrollo de la estructura de tablas es la existente en la dirección del CNDG - Sismología del Instituto Geofísico del Perú, los mismos que fueron descritos en el Capítulo 2. La estructura de tablas va a permitir almacenar la información de los sismos de magnitud elevada (Intensidad Máxima, *Imax*, mayor al grado IV en la escala de Mercalli Modificada), los cuales son fuente de estudios especiales con relación a los de magnitud menor; es decir, evalúan los efectos que produce el sismo además de conocer el origen y las

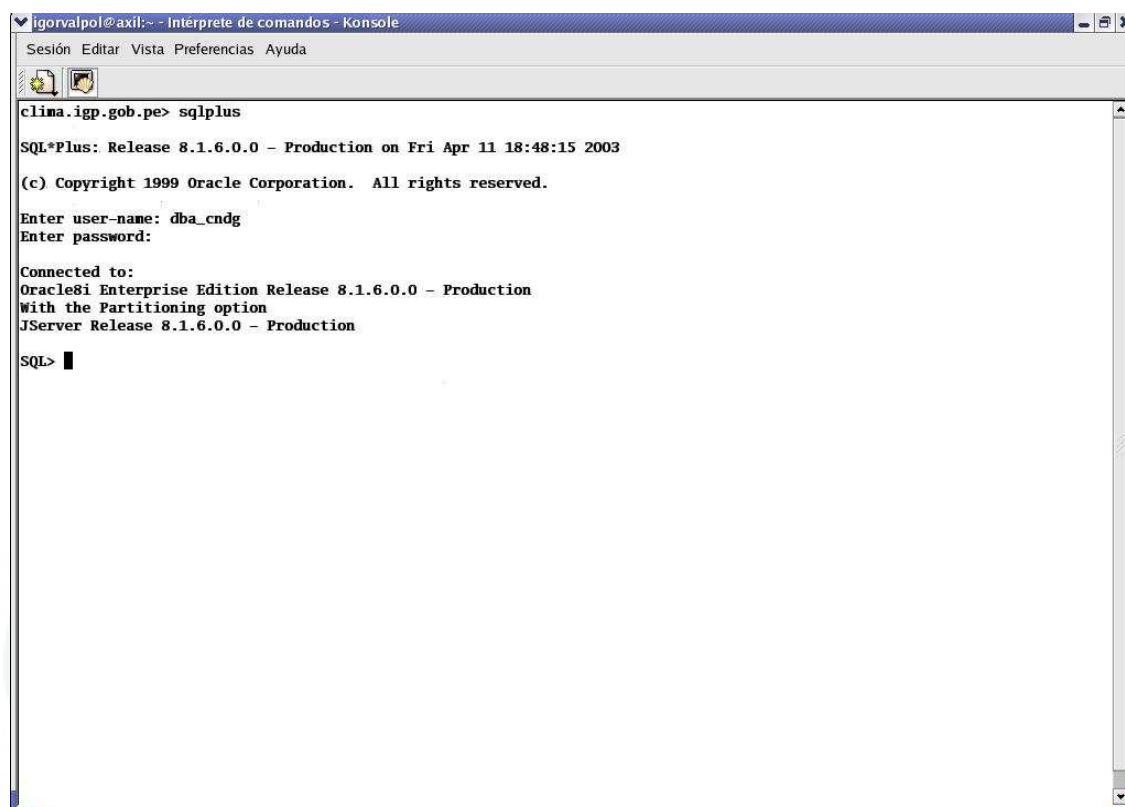
características del proceso de ruptura, obteniéndose archivos de extensión gráfica y documentos de texto.

Para empezar a desarrollar la estructura de tablas, inicialmente se debe de conectar al servidor Oracle haciendo uso del *nombre de usuario* previamente definida por el Administrador de la Base de Datos; es decir, “*dba_cndg*”. Una vez que el *dba_cndg* se conecta a la base de datos de Oracle por medio del SQL*Plus, este puede crear todos los objetos anteriormente definidos. La Figura 4.1 muestra una pantalla con el SQL*Plus debidamente conectado a Oracle empleando la sesión del *dba_cndg*.

El desarrollo de las tablas que conforman la estructura de almacenamiento de la información se hicieron considerando los siguientes aspectos:

- La estructura de tablas de la base de datos sísmicos conserva un orden y una coherencia entre ellos. Cada dato de una tabla estará relacionado con uno o más datos de las demás tablas (Relacionalidad).
- Las tablas en conjunto almacenarán toda la información que generan los sismos de gran magnitud ($I_{max} > IV$ en MM), y por consiguiente se podrá almacenar toda la información proveniente de los sismos más pequeños ($I_{max} \leq IV$ en MM).
- Cada tabla estará encargada de almacenar la información proveniente de cada una de las diferentes fuentes de adquisición de información sísmica; es decir, los archivos generados por el monitoreo con estaciones de banda ancha irán en una tabla especial, la información generada por el monitoreo con las estaciones de periodo corto en otra, los archivos de datos de entrada y salida de los algoritmos de procesamiento sísmico en otra y la información del Catálogo Sísmico del Perú actualizado, en otra tabla.

Considerando lo anterior, la estructura de tablas ideada para la información sísmica esta dada como se describe en los siguientes puntos.



```
igorvalpol@axil:~ - intérprete de comandos - Konsole
Sesión Editar Vista Preferencias Ayuda

clima.igp.gob.pe> sqlplus

SQL*Plus: Release 8.1.6.0.0 - Production on Fri Apr 11 18:48:15 2003

(c) Copyright 1999 Oracle Corporation. All rights reserved.

Enter user-name: dba_cndg
Enter password:

Connected to:
Oracle8i Enterprise Edition Release 8.1.6.0.0 - Production
With the Partitioning option
JServer Release 8.1.6.0.0 - Production

SQL>
```

*Figura 4.1. Pantalla que muestra el inicio de una sesión con la cuenta del DBA_CNDG en el SQL*Plus. Esta conexión se hizo desde una consola de linux hacia el servidor de Base de Datos denominado “Clima”.*

4.4.1.- Tabla Catalogo_1471_1982

Esta Tabla esta conformada por un grupo de campos o columnas las cuales van ha contener la información proveniente del Catalogo Sísmico de Perú en su versión Corregida y Actualizada para el periodo 1471 – 1982. Una tabla con el resumen de los campos descritos se ve en la Tabla 4.5.

Tabla 4.5. Descripción de los campos de la Tabla catalogo_1471_1982.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Sismoid	Varchar2	PK	Código del sismo.
Fuente	Varchar2		Fuente de información para la obtención de datos.
Fecha	date		Fecha del sismo.
Hora	Varchar2		Hora del sismo.
TQF	Varchar2		Calidad del parámetro tiempo (Hora).
Latitud	Number		Latitud del epicentro del sismo.
Longitud	Number		Longitud del epicentro del sismo.
M	Varchar2		Calidad de los parámetros de ubicación epicentral.
Profundidad	Number		Profundidad del hipocentro del sismo.
EQF	Varchar2		Calidad del parámetro profundidad.
Fue	Varchar2		Fuente de información para la obtención de los datos de ubicación y profundidad.
Mb	Varchar2		Magnitud de ondas de cuerpo del sismo.
Ms	Varchar2		Magnitud de ondas superficiales del sismo.
Mw	Varchar2		Magnitud energía del sismo.
M_sismico	Varchar2		Momento sísmico calculado para el sismo.
N_est	Number		Número de estaciones que entraron en el análisis del sismo.
Reg_flin	Number		Código que representa la Región sísmica de Flinn.
Idata	Varchar2		Símbolo que identifica la existencia de datos de intensidad del sismo.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Int_max	Varchar2		Valor de Intensidad máxima del sismo.
Aut_int	Varchar2		Código que representa a la fuente de evaluación de la intensidad del sismo.
Efectos	Varchar2		Efectos provocados por el sismo.

Los campos que conforman esta tabla son:

Sismo_id: Este campo contiene un nombre con una nomenclatura especial que se le da a los sismos con la intención de identificarlos de forma única. Este campo está definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4) y representa la *clave primaria* de esta tabla. Su nombre está formado de la siguiente forma: dos letras que representan el país de donde es el sismo: por ejemplo, PE en Perú o BR en Brasil; y un número que se incrementa normalmente de sismo en sismo de uno en uno, pero en algunos casos los hace de 5 en 5 a fin de integrar en el futuro sismos que no fueron considerados en versiones iniciales del catálogo. Ejemplo: PE12625, PE00005.

Fuente: Este campo contiene un dato que representa la fuente de donde se obtuvo la información que caracteriza al sismo, siendo esta de 3 letras. Este campo está definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4). Ejemplo: IGO, CGS, IGF, etc.

Fecha: Este campo contiene la fecha en que ocurrió el sismo y en general se encuentra en el formato YYYY/MM/DD (YYYY, año; MM, mes; DD, día); sin embargo, para sismos históricos únicamente se considera el año o el año y el mes. Este campo está definido con el tipo de dato Oracle *date* (ver Tabla 4.4). Ejemplo: 1471, 1576/07, 1952/07/12.

Hora: Este campo contiene la hora en que ocurrió el sismo, en el formato hh:mm:ss.ss.

Para los sismos históricos los valores de hora de minuto y segundo son iguales a cero. Lo cual significa que este no posee ese valor. Este campo esta definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4). Ejemplo: 14:35:12.45, 00:00:00.00 (no cuenta con hora).

TQF: Este campo contiene un valor que representa la calidad de los datos del sismo en cuanto al la fecha y hora. Donde X representa a los sismos no instrumentales y cuyo cálculo de tiempo es aproximado. Este campo esta definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4).

Latitud: Este campo contiene la latitud del sismo expresada en grados, representa una de las coordenadas geográficas del sismo. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4). Ejemplo: -13,5.

Longitud: Este campo contiene el valor de la longitud del sismo expresado en grados, representa una de las coordenadas geográficas de sismos. El tipo de dato Oracle en que esta definido es el *number* (ver Tabla 4.4). Ejemplo: -69,7.

M: Este campo contiene y define el valor que representa la calidad de los datos que indican la ubicación epicentral del sismo, indicado por valores F, G, H, I, M y X (Huaco, 1986). Este campo esta definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4).

Profundidad: Este campo contiene el valor que indica la profundidad del sismo expresada en kilómetros. El tipo de dato Oracle en que esta definido este campo es el tipo *number* (ver Tabla 4.4). Ejemplo: 52.

EQF: Este campo contiene un valor que representa la calidad de los datos que indican la profundidad del sismo y están indicados por valores similares al campo M de esta tabla. Este campo esta definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4).

Fue: Este campo contiene el código de la fuente de información de donde se obtuvo los parámetros de ubicación del sismo. Este campo esta definido con el tipo de dato Oracle *varchar2* (ver Tabla 4.4). Ejemplo: LO.

Mb: Este campo contiene el valor de la magnitud del sismo obtenido a partir de las ondas corporeas y esta acompañado de tres letras que representan la fuente de donde se obtuvo dicho valor. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: 4.1 CGS, 4.3 ISC, 3.8IGP.

Ms: Este campo contiene el valor de la magnitud del sismo obtenido a partir de las ondas superficiales y esta acompañado de tres letras que representan la fuente de donde se obtuvo dicho valor. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: 6.2 PAS, 5.6 CGS.

Mw: Este campo contiene el valor de la magnitud momento del sismo y al igual que los casos anteriores, esta acompañado de tres letras que representan la fuente de donde se obtuvo dicho valor. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: 7.5 DOS, 8.2 BEC.

M_sismico: Este campo contiene el valor del momento sísmico del sismo, esta expresado en *dinas.cm* y también esta acompañado de tres letras que representan la fuente de donde se obtuvo dicho valor. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: 27.0E+27KAN, 1.2E+26DOS.

N_est: Este campo contiene el número de estaciones sísmicas que registraron el sismo y de las cuales se obtuvo los parámetros sísmicos. El tipo de dato Oracle en que esta definido

este campo es el tipo *number* (ver Tabla 4.4). Ejemplo: 12, 28.

Reg_flin: Este campo contiene el código que representa la región sísmica en la cual ocurrió el sismo según la reorganización propuesta por Flinn et. al. (1984). El tipo de dato Oracle en que esta definido este campo es el tipo *number* (ver Tabla 4.4). Ejemplo: 117, 116.

Idata: Este campo esta contiene el signo asterisco “*” que indica si ese sismo posee datos de Intensidad. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: *.

Int_max: Este campo contiene el número que representa el valor de intensidad máxima que alcanzo una región afectada por el sismo en la escala en la que fue evaluada. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: 8M, 10M.

Aut_int: Este campo contiene un código que representa el nombre del autor o la fuente de información de la que se obtuvo el valor máximo de intensidad del sismo. El tipo de dato Oracle en que esta definido este campo es el tipo *varchar2* (ver Tabla 4.4). Ejemplo: DH. LO, H*O.

Efectos: Este campo contiene un código el cual indica que tipo de efectos que produjo el sismo.

4.4.2.- Tabla Catalogo_1983_act

Esta Tabla esta conformada por un grupo de campos o columnas las cuales van ha contener la información proveniente del Catalogo Sísmico de Perú en su versión Corregida y Actualizada para el periodo 1983 a la actualidad. Un resumen de la descripción de la tabla

catalogo_1983_act se aprecia en la Tabla 4.6.

Tabla 4.6. Descripción de los campos de la Tabla *catalogo_1983_act*.

Campo	Tipo de dato	Restricción	Descripción
Tiempo_id	Varchar2	PK	Fecha y hora del Sismo.
Fecha	Date		Fecha del sismo.
Latitud	Number		Latitud del epicentro del sismo.
error_lat	Number		Error en el valor de la latitud dado por el programa de procesamiento sísmico (PPS).
Longitud	Number		Longitud del epicentro del sismo.
error_lon	Number		Error en el valor de la longitud dado por el programa de procesamiento sísmico (PPS).
Profundidad	Number		Profundidad del hipocentro del sismo.
Magnitud	Number		Magnitud del evento dado por el PPS.
	Number		Magnitud del evento dado por el PPS.
Intensidad	Varchar2		Valores de Intensidad en localidades cercanas al epicentro del sismo.
Gap	Number		Espacio angular máximo entre las estaciones que registraron al sismo.
Rmc	Number		Raíz Media Cuadrática que resulta del cálculo de parámetros sísmicos con el PPS.
N_est	Number		Número de estaciones que registraron el sismo.
N_fasesp	Number		Número de fase P leídas en todas las estaciones.
N_fasess	Number		Número de fase S leídas en todas las estaciones.
sismoid	Varchar2	PK	Clave principal que se relaciona con otras tablas.

Los campos que conforman esta tabla son:

Tiempo_id: Este campo contiene información sobre la fecha y hora en que ocurrió el sismo, el cual esta definida como un tipo de dato Oracle denominado *varchar2* (ver Tabla 4.4). Este campo pudiera estar definido como un tipo de dato Oracle Date, pero este no abarca dentro de sus diferentes formatos los decimales de los segundos. Ejemplo: 2000/01/25 15:23:49,35

Fecha: Este campo contiene información que representa la fecha del sismo indicando día, mes y año. El tipo de dato Oracle en el que esta definida esta columna es *date* (ver Tabla 4.4). Ejemplo: 2000/01/25.

Latitud: Este campo contiene el valor de la latitud del sismo expresada en grados representa una de las coordenadas geográficas del sismo. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4). Ejemplo: -13,5.

Error_lat: Este campo contiene el valor que representa el rango de error que puede tener la latitud y se encuentra expresado en kilómetros, y que resulta del procesamiento de cálculo hecho con programas especializados. El tipo de dato Oracle en que esta definida esta columna es el *number* (ver Tabla 4.4). Ejemplo: 0,5

Longitud: Este campo contiene el valor de la Longitud del sismo expresado en grados, la cual representa una de las coordenadas geográficas del sismo. El tipo de dato Oracle en que esta definido es el *number* (ver Tabla 4.4). Ejemplo: -69,7.

Error_lon: Al igual que el campo Error_lat, este campo contiene el valor que representa el rango de error que puede tener la longitud y se encuentra expresado en kilómetros, y que resulta del cálculo hecho con programas especializados. El tipo de dato en que esta definida

esta columna es el number (ver Tabla 4.4). Ejemplo: 1,3.

Profundidad: Este campo contiene el valor de profundidad del sismo expresado en kilómetros. El tipo de dato Oracle en que esta definido este campo es el tipo *number* (ver Tabla 4.4). Ejemplo: 52.

Error_pro: Al igual que el campo Error_lat y el campo Error_lon, este campo contiene el valor que representa el error obtenido en el cálculo de la profundidad expresado en kilómetros. El tipo de dato Oracle en que esta definida esta columna es el *number* (ver Tabla 4.4). Ejemplo: 5.

Mb1: Este campo de la tabla contiene el valor de magnitud del sismo. Este valor se obtiene a partir de la duración del registro del sismo utilizando una relación o ecuación. El tipo de dato Oracle en que esta definida esta columna es el *number* (ver Tabla 4.4). Ejemplo: 2,3.

Mb2: Este campo de la tabla contiene el valor de magnitud del sismo obtenido a partir de la amplitud y periodo de las fases del sismo empleando para esto una relación especial. El tipo de dato Oracle en que esta definida esta columna es el *number* (ver Tabla 4.4). Ejemplo: 2,3.

Intensidad: Este campo contiene el nombre de las ciudades que fueron afectadas por el sismo y sus valores de Intensidad máximas. Este valor es obtenido a través de encuesta en las ciudades cercanas al epicentro del sismo. Esta información es tanto numérica como alfanumérica y debido ha esto, se le ha asignado ha esta columna el tipo de dato Oracle *varchar2* (ver Tabla 4.4). Ejemplo: “III – IV en Camaná, V en Arequipa”.

Gap: Este campo contiene la información referente al espaciamiento angular máximo que existe entre las estaciones que registraron el evento sísmico y se encuentra expresado en

grados sexagesimales. Su valor es numérico y por lo tanto se le asignó el tipo de dato Oracle *number* (ver Tabla 4.4). Ejemplo: 235.

Rmc: Este campo contiene la información referente a la Raíz Media Cuadrática que resulta del cálculo de los parámetros sísmicos de un determinado sismo empleando un programa especializado. Este valor numérico es importante debido a que según el valor de este se puede tomar la precisión del cálculo de los parámetros epicentrales. El tipo de dato Oracle que define este campo es el *number* (ver Tabla 4.4). Ejemplo: 0,1.

N_estaciones: Este campo representa el número de estaciones que registraron y que participaron en el cálculo de los parámetros sísmicos del sismo. El tipo de dato Oracle que define este campo es el *number* (ver Tabla 4.4). Ejemplo: 5.

N_fasesp: Este campo define el número de fases de la onda P leídos en todas las estaciones que registraron el sismo. El tipo de dato Oracle que define este campo es el *number* (ver Tabla 4.4). Ejemplo: 5.

N_fasess: Este campo define el número de fases de la onda S leídos en todas las estaciones que registraron el sismo. El tipo de dato Oracle que define este campo es el *number* (ver Tabla 4.4). Ejemplo: 4.

Sismoid: Este campo contiene el valor que caracteriza un sismo. Este valor está comprendido por la fecha del evento y un número que le acompaña y que representa al número de sismo ocurrido en ese día. Esta numeración comienza en el 1 y puede llegar hasta el número 99 y si el número de sismos en el día excede este valor, a los siguientes eventos, se les condicionará las letras A1, A2, y así sucesivamente hasta llegar a Z99. Este campo representa otra Clave o llave Primarias (PK) de la tabla y a través de este campo es posible relacionarla con las demás tablas de la base de datos. El tipo de dato Oracle que

define este campo es *varchar2* (ver Tabla 4.4). Un ejemplo es: 0012504.

4.4.3.- Tabla **Program_files**

Esta tabla esta constituida por campos o columnas destinados a almacenar los archivos que resultan del procesamiento de la información, ya sea este empleando un software de lectura de fases y duración del sismo (Sismalp, Winquake, etc.) o por los programas de procesamiento de datos sísmicos (Epi, Epigraf, hipoinverse y hipoelipse).

Los campos que conforman esta tabla son:

Sismo_id. Este campo representa la Clave Primaria (PK) y además la Clave Foránea (FK) que se relaciona con la columna que lleva el mismo nombre en la tabla catalogo_1983_act, por lo que, su formato es el mismo. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Datos_ent. Este campo se relaciona con los Archivos de Entrada a los programas de procesamiento. La manera de insertar información en este campo es especial y se tratará mas adelante. El tipo de dato Oracle en la cual esta definida esta columna es *clob* (ver Tabla 4.4).

Datos_sal. Este campo se relaciona con los archivos de Salida de los programas de procesamiento. Al igual que el campo Datos_ent, este campo el mecanismos de inserción de la información es diferente a los datos comunes (números, nombres, etc.). El tipo de dato Oracle en la cual esta definida esta columna es *clob* (ver Tabla 4.4).

Un resumen de la descripción de la tabla program_files se aprecia en la Tabla 4.7.

Tabla 4.7. Descripción de los campos de la Tabla *program_files*.

Campo	Tipo de dato	Restricción	Descripción
Sismoid	Varchar2	PK y FK	Clave Primaria y Foránea de la tabla. Se relaciona con la tabla catalogo_1983_act.
Datos_ent	Clob		Campo relacionado con los Archivos de entrada a los PPS.
Datos_sal	Clob		Campo relacionado con los Archivos de salida de los PPS.

4.4.4.- Tabla Informes_1471_1982

En esta tabla se almacenarán los informes o publicaciones elaborados para los sismos correspondientes al Catálogo Sísmico del Perú para el periodo 1471 – 1982. Además del mapa de intensidades que pudiera tener cada uno de estos sismos. Esta tabla esta conformada por los siguientes campos:

Sismo_id. Este campo representa la Clave Primaria (PK) y además la Clave Foránea (FK) que se relaciona con la columna que lleva el mismo nombre en la tabla catalogo_1471_1982, por lo que su formato es el mismo. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Final_inf: Este campo contiene el archivo con el informe del sismo. El tipo de dato Oracle en la cual esta definida esta columna es *blob* (ver Tabla 4.4).

Map_int: Este campo contiene el archivo gráfico del mapa de Intensidades del sismo. El

tipo de dato Oracle en la cual esta definida esta columna es *blob* (ver Tabla 4.4). Un resumen de esta información se muestra en la Tabla 4.8.

Tabla 4.8. Descripción de los campos de la Tabla *informes_1471_1982*.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Sismoid	Varchar2	PK y FK	Clave Primaria y Foránea de la tabla. Se relaciona con la tabla catalogo_1471_1982.
Final_inf	Blob		Archivo con el informe del sismo.
Map_int	Blob		Archivo con el mapa de intensidades del sismo.

4.4.5.- Tabla Informes_1983_act

En esta tabla se almacenarán los informes o publicaciones realizados sobre los sismos correspondientes al Catálogo Sísmico del Perú periodo 1983 a la actualidad. Esta tabla esta conformada por los siguientes campos:

Sismo_id. Este campo representa la Clave Primaria (PK) y además la Clave Foránea (FK) que se relaciona con la columna que lleva el mismo nombre de la tabla catalogo_1983_act, por lo que su formato es el mismo. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Preliminar_inf: Este campo contiene el archivo con informes preliminares de los sismo. El tipo de dato Oracle en la cual esta definida esta columna es *blob* (ver Tabla 4.4).

Final_inf: Este campo contiene los archivos con publicaciones realizadas para los sismos.

El tipo de dato Oracle en la cual esta definida esta columna es *blob* (ver Tabla 4.4). Un resumen de la descripción de la tabla *informes_1983_act* se aprecia en la Tabla 4.9.

Tabla 4.9. Descripción de los campos de la Tabla *informes_1983_act*.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Sismoid	Varchar2	PK y FK	Clave Primaria y Foránea de la tabla. Se relaciona con la tabla catalogo_1471_1982.
Preliminar_inf	Blob		Archivo con el informe preliminar del sismo.
Final_inf	Blob		Archivo con el informe final del sismo.

4.4.6.- Tabla Graphics

En esta tabla se almacenarán los archivos gráficos de los sismos correspondientes al Catálogo Sísmico del Perú para el periodo 1983 a la actualidad. Esta tabla esta conformada por los siguientes campos:

Sisgraf_id. Este campo representa una Clave Primaria (PK). Su nombre esta formado por el sismoid del sismo acompañado de un número. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4). Ejemplo: 0112000305.

Sismoid. Este campo representa la Clave Foránea (FK) que se relaciona con la columna que lleva el mismo nombre en la tabla catalogo_1983_act, por lo que su formato es el mismo. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Nombre: Este campo contiene el nombre del archivo gráfico del sismo. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Grafico: Este campo contiene el archivo con un gráfico correspondiente al sismo, el cual puede ser un mapa de intensidades, una fotografía, o cualquier otro gráfico. El tipo de dato Oracle en la cual esta definida esta columna es *blob* (ver Tabla 4.4). Un resumen de la descripción de la tabla *Graphics* se aprecia en la Tabla 4.10.

Tabla 4.10. Descripción de los campos de la Tabla *Graphics*.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Sisgraf_id	Varchar2	PK	Clave Primaria de la tabla.
Sismoid	Varchar2	PK	Clave Foránea de la tabla. Se relaciona con la tabla catalogo_1471_1982.
Nombre	Varchar2		Nombre del archivo Gráfico.
Grafico	Blob		Archivo con el gráfico correspondiente al sismo.

4.4.7.- Tabla Estaciones_param

Esta tabla almacena la información referente a las características instrumentales y los parámetros que indican la ubicación de los equipos que conforman la RSN. Los campos que conforman dicha tabla son los siguientes: Un resumen de esta información se muestra en la Tabla 4.11.

Tabla 4.11. Descripción de los campos de la Tabla Estaciones_param.

Campo	Tipo de dato	Restricción	Descripción
Codigo_nac	Varchar2	PK	Código nacional de la estación.
Codigo_internac			Código internacional de la estación.
Nombre	Varchar2		Nombre de la estación.
Latitud_est	Number		Latitud de la ubicación geográfica de la estación.
Longitud_est	Number		Longitud de la ubicación geográfica de la estación.
Elevación_est	Number		Elevación de la ubicación geográfica de la estación.
Sensor	Varchar2		Código del sensor de la estación.
Equipo	Varchar2		Tipo de equipo.
Tipo_est	Varchar2		Tipo de Estación.
Ancho_banda	Varchar2		Ancho de banda de la estación.
Rango_dinam	Number		Rango dinámico de la estación.
Muestreo	Number		Muestreo de la estación.
Pre_event	Number		Pre – evento de la estación.
Post_event	Number		Post – evento de la estación.
Gain_Amp	Varchar2		Ganancia de la amplitud de la estación.
Long_regist	Varchar2		Longitud de registro de la estación.
Estado	Varchar2		Estado en el que se encuentra la estación.
Estacion_grup	Varchar2		Indica el nombre de las estaciones que la integran. Solo se da cuando el Codigo_nac

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
			indica a una Sub – Red de estaciones de periodo corto.

Codigo_nac. Este campo contiene los datos que corresponden al Código Nacional de cada estación sísmica. Este código representa un “alias” que sirve para identificar a cada una de las estaciones de la RSN en los diferentes informes, archivos de datos y otros documentos. Así mismo, en el software de procesamiento de datos se trabajan con esta nomenclatura para identificar de que estación proviene la información que se desea procesar. Esta columna o campo representa la Clave Primaria (PK) de dicha tabla. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4). Ejemplo: CON (Conima), NNA (Ñaña), CAM (Camacho), etc.

Codigo_internac: Este campo contiene un código internacional con el que se le reconoce a la estación a nivel mundial. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4). Ejemplo: PP21 (Conima), NNA (Ñaña), PT10 (Camacho), etc.

Nombre. Este campo contiene los nombres de las estaciones de la RSN. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4). Ejemplo: Conima, Ñaña, Camacho, etc.

Latitud_est. Este campo contiene el valor de la latitud en grados y que corresponde a una de las coordenada geográfica, para cada estación de la RSN. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Longitud_est. Este campo contiene el valor de la longitud en grados y que corresponde a

una de las coordenada geográfica para cada estación de la RSN. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Elevación_est. Este campo contiene el valor de la altitud de la estación en metros con respecto al nivel del mar para cada estación de la RSN. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Sensor. Este campo contiene el valor referente al tipo de sensor que utiliza cada una de las estaciones sísmicas de la RSN, en una nomenclatura estándar. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4). Conociendo este valor, se pueden inferir muchas otras características físicas del sensor.

Equipo. Este campo contiene la información referida tipo de equipo de cada estación de la RSN, el cual se encuentra expresado en una nomenclatura estándar o conocida. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Tipo_est: Este campo contiene la información referida al tipo de estación de la RSN; es decir si es de Banda ancha, si es de periodo corto o si esta es una Sub-red. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Ancho_banda: Este campo contiene el valor correspondiente al ancho de banda de frecuencias en la cual se registra un sismo en una estación de Banda ancha. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Rango_dinam: Este campo contiene el valor correspondiente al Rango Dinámico de la estación sísmica expresado en decibeles. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Muestreo: Este campo contiene el valor correspondiente al Muestreo de los datos registrados por las estaciones sísmicas, expresado en *mps* (numero de muestras por segundos). El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Pre_event: Este campo contiene el valor correspondiente al *Pre - evento* del registrador sísmico en cada estación, expresado en segundos. Se entiende este valor como el tiempo que retornara el registrador sísmico a partir de la ocurrencia de un sismo para empezar a grabarlo. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Post_event: Este campo contiene el valor correspondiente al *Post - evento* del registrador sísmico en cada estación, expresado en segundos. Se entiende este valor como el tiempo adicional que se le da al registrador sísmico para terminar de grabar el sismo, una vez identificado el final de este. El tipo de dato Oracle en el que esta definido este campo es *number* (ver Tabla 4.4).

Gain_amp: Este campo contiene el valor correspondiente a la Ganancia o sensibilidad en amplitud de la estación. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Long_regist: Este campo contiene el valor correspondiente a la longitud de registro del sismo expresado en segundos en cada estación. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Estado: Este campo almacena un nombre que indica el estado de la estación, ya sea este operativo o desactivado. El tipo de dato Oracle en el que esta definido este campo es *varchar2* (ver Tabla 4.4).

Estacion_grup. En este campo se indica el nombre de todas las estaciones de periodo corto que conforman esta Sub – Red. El tipo de dato Oracle en el que se encuentra definido este campo es *Varchar2* (ver Tabla 4.4).

4.4.8.- Tabla Estaciones_inform

Esta tabla contiene la información correspondiente a las estaciones sísmicas de la RSN, en las que se incluye el informe técnico de la estación y la curva de magnificación. Sus campos son:

Codigo_nac: Este campo contiene el código nacional de la estación. Representa la clave Primaria de la tabla y la clave foránea que esta relacionada con la tabla estaciones_param. El tipo de dato Oracle en la cual esta definida esta columna es *varchar2* (ver Tabla 4.4).

Curva_magnif: Este campo contiene el archivo que contiene la curva de magnificación de la estación sísmica. El tipo de dato Oracle en el que se encuentra definido es *blob* (ver Tabla 4.4).

Informe_est: Este campo contiene el archivo que contiene el informe técnico elaborado para cada estación sísmica. El tipo de dato Oracle en el que se encuentra definido es *blob* (ver Tabla 4.4).

Un resumen de la descripción de la tabla *Estaciones_inform* se aprecia en la Tabla 4.12.

Tabla 4.12. Descripción de los campos de la Tabla Estaciones_inform.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Codigo_nac	Varchar2	PK	Clave primaria y foránea de esta tabla. Se relaciona con la tabla estaciones_param
Curva_magnif	Blob		Archivo con la curva de magnificación.
Informe_est	Blob		Archivo con el informe técnico de la estación.

4.4.9.- Tabla Ondas_sis

Esta tabla contiene la información que proviene del registro de los sismos empleando las redes telemetrías de periodo corto de la RSN. Los campos que conforman esta tabla son:

Ondassis_id. Este campo representa la clave primaria (PK) de esta tabla. Su formato esta dado por el valor del campo *Sismoid* unido al valor de campo *Codigo_nac* de esta tabla. El tipo de dato Oracle en el que se encuentra definido es *Varchar2* (ver Tabla 4.4).

Sismoid. Este campo representa la Clave Foránea (FK) de la tabla. Este campo relaciona la tabla con la tabla *catalogo_1983_act*. Su formato y tipo de dato Oracle esta dado por la columna a la cual esta relacionada (*Sismoid* de la tabla *catalogo_1983_act*).

Codigo_nac. Este campo representa otra clave Foránea (FK) de esta tabla. Este campo relaciona la tabla con la tabla *Estaciones_param*. Su formato y tipo de dato Oracle esta dado por la columna con la cual esta relacionada (*Codigo_nac* de la tabla *Estaciones_param*).

Eventosis. Este campo esta relacionado a los archivos binarios de las formas de onda de los sismos registrados por las estaciones de periodo corto de la RSN. Al igual que los campos *Datos_ent*, *Datos_sal*, Informe en la tabla *Texto_Grafico*, o cualquier otra información representada por un archivo, este se almacena de manera diferente en la base de datos. En el Capítulo 2 se señaló la nomenclatura que se utiliza para identificar cada uno de los archivos de este tipo. Esta misma nomenclatura se utiliza para identificar cada archivo almacenado en este campo. El tipo de dato Oracle que define esta columna es *bfile* (ver Tabla 4.4).

Ndxfile. Este campo esta relacionado con los archivos NDX (*.ndx) que ya fueron tratados en el Capítulo 2 de este estudio. Al igual que el campo *Eventos_sis*, el nombre que lleva dicho archivo sirve para identificarlo. El tipo de dato Oracle que define esta columna es *clob* (ver Tabla 4.4).

Depfile. Este campo esta relacionado a los archivos DEP (*.dep) de los que ya se trato anteriormente. Al igual que los campos *Eventos_sis* y *Ndxfile*, el nombre que llevan estos archivos sirve para identificarlos dentro de su campo. El tipo de dato Oracle que define esta columna es *clob* (ver Tabla 4.4).

Un resumen de esta información se muestra en la Tabla 4.13.

Tabla 4.13. Descripción de los campos de la Tabla *Ondas_sis*.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Ondassis_id	Varchar2	PK	Clave primaria de esta tabla.
Sismoid	Varchar2	FK	Clave foranea que se relaciona con la Tabla Parametros.
Codigo_nac	Varchar2	FK	Clave foranea que se relaciona con la Tabla estaciones.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Eventosis	Bfile		Campo que se relaciona con los archivos binarios de las formas de onda de las estaciones de periodo corto.
Ndxfile	Bfile		Campo que se relaciona con los archivos NDX.
Depfile	Bfile		Campo que se relaciona con los archivos DEP.

4.4.10.- Tabla Ondas_sac

Esta tabla contiene la información que proviene del registro de los sismos por las estaciones de banda ancha de la RSN. Los campos que conforman esta tabla son:

Ondassac_id. Este campo representa la clave primaria (PK) de la tabla. Su formato esta dado por el valor del campo Sismoid unido al valor del campo Codigo_nac de la tabla. El tipo de dato Oracle en el que se encuentra definido es *Varchar2* (ver Tabla 4.4).

Sismoid. Este campo representa la Clave Foránea (FK) de esta tabla. Este campo relaciona esta tabla con la tabla *catalogo_1983_act* de la misma forma que en la tabla *Ondas_sis*. Su formato y tipo de dato Oracle esta dado por la columna a la cual esta relacionada (*Sismoid* de la tabla *catalogo_1983_act*).

Codigo_nac. Este campo representa otra clave Foránea (FK) de la tabla. Este campo relaciona esta tabla con la tabla *Estaciones_param*. Su formato y tipo de dato Oracle esta dado por la columna a la cual esta relacionada (*Codigo_nac* de la tabla *Estaciones_param*).

Eventosacz. Este campo esta relacionado con el archivo binario de la forma de onda de un

sismo registrado por una estación de banda ancha, pero en su componente vertical (Z). Al igual que el campo Eventosis, el nombre del archivo servirá para identificar a cada uno de estos archivos. El tipo de dato Oracle que define esta columna es *bfile* (ver Tabla 4.4).

Eventosacn. Este campo esta relacionado con el archivo binario de la forma de onda de un sismo registrado por una estación de banda ancha, pero en una de sus componentes horizontales, la componente Norte - Sur (N). El tipo de dato Oracle que define esta columna es *bfile* (ver Tabla 4.4).

Eventosace. Al igual que en los dos campos anteriores, este campo esta relacionado con el archivo binario, pero en su otra componente horizontal, la componente Este - Oeste (E). El tipo de dato Oracle que define esta columna es *bfile* (ver Tabla 4.4).

Un resumen de esta información se muestra en la Tabla 4.14.

Tabla 4.14. Descripción de los campos de la Tabla Ondas_sac.

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
Ondassac_id	Varchar2	PK	Clave primaria de esta tabla.
Sismoid	Varchar2	FK	Clave foranea que se relaciona con la Tabla Parametros.
Codigo_nac	Varchar2	FK	Clave foranea que se relaciona con la Tabla Estaciones.
Eventosacz	Bfile		Campo que se relaciona con los archivos binarios de las formas de onda de los sismos en las estaciones de banda ancha. Componente vertical.
Eventosacn	Bfile		Campo que se relaciona con los archivos binarios de las formas de onda de los sismos en las

<i>Campo</i>	<i>Tipo de dato</i>	<i>Restricción</i>	<i>Descripción</i>
			estaciones de banda ancha. Componente horizontal N - S.
Eventosace	Bfile		Campo que se relaciona con los archivos binarios de las formas de onda de los sismos en las estaciones de banda ancha. Componente Horizontal E - W.

4.5.- Relaciones entre Tablas de la Base de Datos Sísmicos

En el Capítulo 3, se habló de una base de datos Oracle objeto – relacional. Esta característica permite sacar provecho tanto de la tecnología orientada a objetos como de la de tipo relacional en la base de dato. La *Relacionalidad* es una característica importante dentro de la estructura de una base de datos ya que permite organizar mejor la información en un sistema de tablas relacionadas entre si, tanto para facilitar el almacenamiento de la información, como la gestión de los datos y las consultas respectivas.

Las tablas se relacionan entre si mediante las columnas que tienen en común. La base de datos puede utilizarse para forzar estas relaciones por medio de la Integridad Referencial. Si se utiliza las funciones orientadas a objetos de Oracle, las filas podrían estar relacionadas entre si por medio de referencias internas llamadas identificadores de objeto (OID, Object Identifier). La integridad Referencial se impone en el nivel de la base de datos por medio de restricciones. Es así que, dos tablas están relacionadas mediante las restricciones de Clave Primaria (Primary Key, PK) y Clave Foránea (Foreign Key, FK) las cuales han sido definidas anteriormente y que han sido definidas al momento de la creación de dichas tablas. Un ejemplo de como se define una relación entre dos tablas al momento de crearlas se presenta a continuación:

```
SQL> create table ONDAS_SIS
      (sismoid      varchar2(9),
      codigo_nac varchar2(9),
      evento_sis   bfile,
      ndxfile      bfile,
      depfile      bfile,
      foreign key (sismoid) references CATALOGO_1983_ACT)
      commit;
```

4.5.1.- Diagrama Entidad - Relación

El diagrama en el que se muestran las relaciones entre tablas de una base de datos se denomina “Diagrama Entidad – Relación”. En este diagrama cada tabla o entidad esta representada mediante rectángulos y cada uno de los campos de la tabla son representadas por elipses. Las relaciones se representan a través de rombos y en sus extremos llevan valores que representan la relación entre los datos de ambas tablas. El diagrama Entidad – Relación para la base de datos sísmicos previamente definida se muestra en la Figura 4.2.

4.6.- Vistas de la Base de Datos Sísmicos

Las vistas que conforman la base de datos sísmicos van a permitir optimizar el tramite de la información, así como también, permitirá al administrador de la base de datos tener una mejor organización de la consulta a los datos y la seguridad de los mismos.

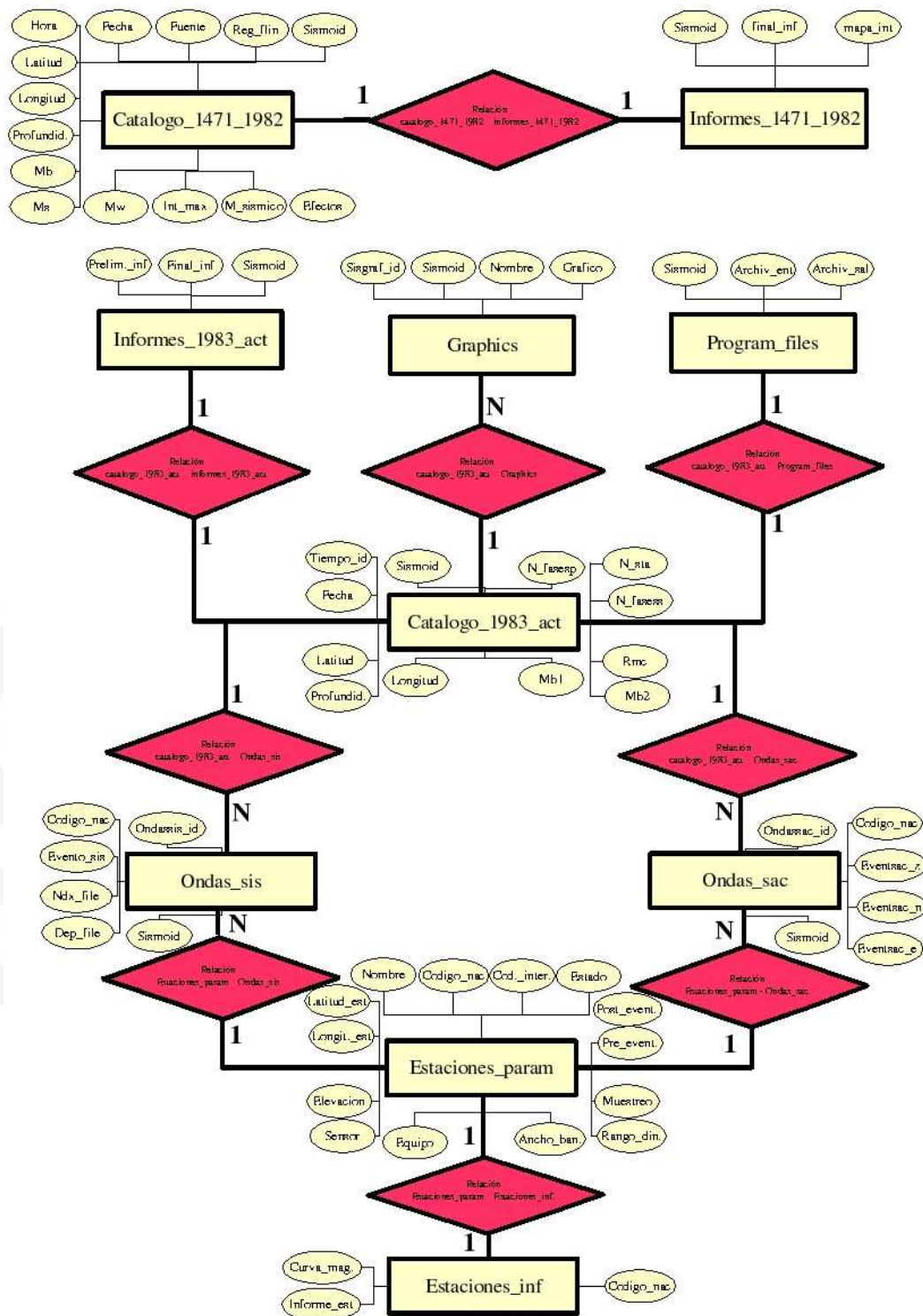


Figura 4.2. Diagrama Entidad – Relación de la Base de Datos Sísmicos.

Las vistas pueden ser aplicadas a los campos de las tablas cuyo contenido sea únicamente de tipo dato y no un archivo, es decir, números, caracteres, fechas o la combinación de estos tipos de datos. En la estructura de tablas de la base de datos sísmicos, la tabla *catalogo_1471_1982*, *catalogo_1983_act* y la tabla *estaciones_param* son las únicas que posee solamente estos tipos de datos, por lo que es en ellas en donde se van a aplicar el desarrollo de vistas. Las vistas que conforman la base de datos sísmicos son:

- La vista *catalogo_1471_1982_vis*, es una vista que contiene todos los campos de la tabla *catalogo_1471_1982*.
- La vista *catalogo_1983_act_vis*, es una vista que contiene todos los campos de la tabla *catalogo_1983_act*.
- la vista *estaciones_param_vis*, es una vista que contiene todos los campos de la tabla *estaciones_param*.

El desarrollo de más vistas dependerá de la necesidad que tengan los usuarios, dada la capacidad de Oracle de incluir nuevas vistas aun después de definir la base de datos.

CAPITULO 5

MECANISMOS DE INSERCIÓN Y ALMACENAMIENTO DE LOS DATOS SÍSMICOS

5.1.- Introducción

En este Capítulo se describirá y tratará todo lo relacionado a la inserción de la información sísmica generada por el área del CNDG – Sismología dentro de la Base de Datos Sísmicos empleando las herramientas y funciones que Oracle proporciona para dicho fin. Para esto se empleará la herramienta de Oracle denominada SQL*Loader para la inserción de datos, ya sean estos datos estructurados como datos no - estructurados. Además, se tratará el uso de la función *BFILENAME* que permite realizar la inserción de la información definida en la base de datos como tipo de dato Oracle *bfile* (archivos binarios ubicados fuera de la base de datos) empleando la sentencia *Insert* desde el SQL*Plus y también, la inserción de este tipo de información empleando la función *BFILE*, cuando la inserción de estos archivos se realiza empleando la herramienta SQL*Loader. Asimismo, el capítulo considera la descripción secuencial de los pasos a seguir para almacenar archivos de diferentes formatos dentro de la base de datos y de algunos programas elaborados por el DBA_CNDG (Administrador de la Base de Datos Sísmicos) para crear *Archivos de Datos* que faciliten la inserción de los archivos empleando el SQL*Loader.

5.2.- El SQL*Loader

El SQL*Loader es una herramienta de Oracle que se encarga de leer e insertar los datos en la Base de Datos Oracle de acuerdo con las instrucciones que recibe de un “*archivo de control*”, el cual le indica donde colocar los datos y que tipo de datos se va a insertar.

En general, las características más importantes de la herramienta SQL*Loader son:

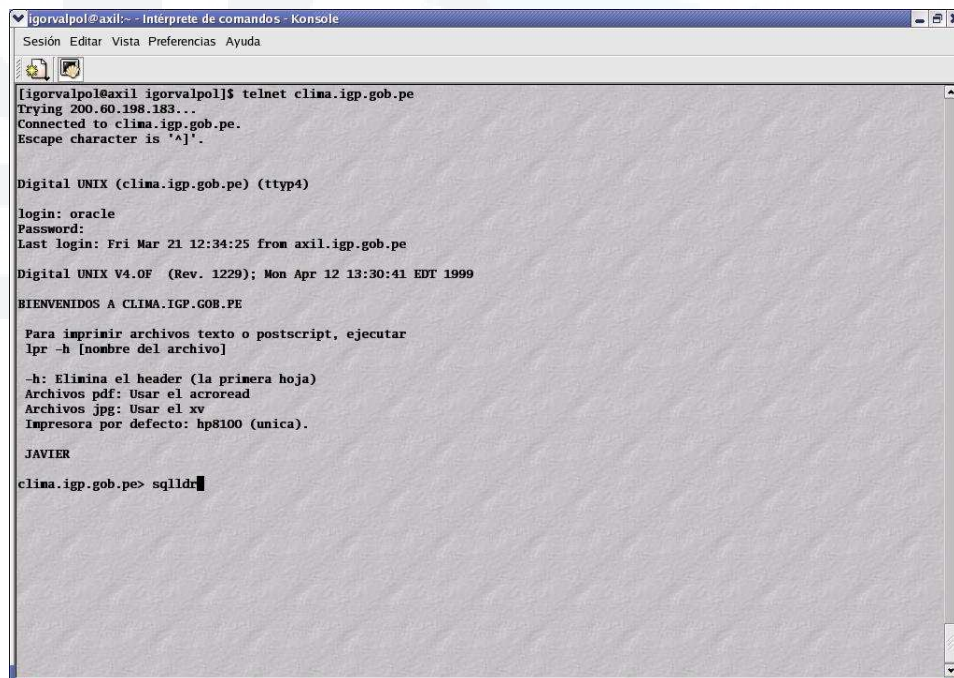
- Tiene un poderoso motor de análisis de datos que considera poca limitación en cuanto al formato de los datos del archivo de datos.
- Permite insertar datos leídos de múltiples archivos de datos en la misma sesión de inserción.
- Permite insertar datos dentro de múltiples tablas durante la misma sesión de carga.
- Permite especificar el grupo de caracteres de los datos.
- Permite seleccionar los datos cargados.
- Permite la manipulación de los datos leídos de una archivo antes de ser cargados, usando sentencias SQL.
- Permite generar valores claves secuenciales únicos en columnas especificadas.
- Permite la utilización del sistema de archivos del Sistema Operativo para acceder los archivos de datos.
- Permite insertar datos desde unidades de discos, cintas y otros.
- Proporciona sofisticados y detallados reportes de error.
- Soporta dos formas de inserción, una Convencional y otra Directa.
- Puede cargar arbitrariamente datos objeto – relacional complejos; es decir, la fuente

de datos de donde los lee el SQL*Loader pueden tener a la vez información en modo dato como en modo archivo.

- Soporta *Archivos de Datos Secundarios* para insertar datos Oracle de tipo LOB (Large Object) como archivos binarios, archivos de texto o archivos gráficos.

5.2.1.- Palabras Claves en Línea de Comandos SQL*Loader

Al igual que con el SQL*Plus, el Administrador de Base de Datos (DBA) se conecta al servidor Oracle empleando el *Telnet*, ya sea desde el DOS en Windows ó una desde una consola en Linux /Unix. Una vez conectado, el SQL*Loader puede ser invocado y modificado desde la Línea de comandos del sistema operativo usando *Palabras Claves (Keywords)*. En la Figura 5.1, se presenta una pantalla desde un terminal de Linux conectado al servidor de Base de Datos denominado “Clima” empleando una conexión de tipo Telnet.



```
igorvalpol@axil:~ - intérprete de comandos - Konsole
Sesión Editar Vista Preferencias Ayuda

[igorvalpol@axil igorvalpol]$ telnet clima.igp.gob.pe
Trying 200.60.198.183...
Connected to clima.igp.gob.pe.
Escape character is '^['.

Digital UNIX (clima.igp.gob.pe) (ttty4)

login: oracle
Password:
Last login: Fri Mar 21 12:34:25 from axil.igp.gob.pe

Digital UNIX V4.0F (Rev. 1229); Mon Apr 12 13:30:41 EDT 1999

BIENVENIDOS A CLIMA.IGP.GOB.PE

Para imprimir archivos texto o postscript, ejecutar
lpr -h [nombre del archivo]

-h: Elimina el header (la primera hoja)
Archivos pdf: Usar el acroread
Archivos jpg: Usar el xv
Impresora por defecto: hp8100 (unica).

JAVIER

clima.igp.gob.pe> sqlldr
```

Figura 5.1. Pantalla que muestra la línea de comandos del servidor “Clima” en un terminal de Linux conectados mediante una comunicación vía TELNET.

La palabra utilizada para invocar SQL*Loader depende del sistema operativo. El nombre que se emplea en Unix/Linux es el “*sqlldr*” y para Windows es “*sqlload*”. En vista que la base de datos se desarrolla en una plataforma Unix la palabra que invocará a SQL*Loader desde ahora será *sqlldr*. Si se introduce *sqlldr* sin especificar nada mas, la pantalla muestra una ayuda con todas las *Palabras Claves* disponibles y sus valores por defecto considerados por SQL*Loader, tal como se muestra a continuación:

```
clima.igp.gob.pe> sqlldr
SQL*Loader: Release 8.1.6.0.0 - Production on Tue Mar 18 16:41:20 2003
(c) Copyright 1999 Oracle Corporation. All rights reserved.
Usage: SQLLOAD keyword=value [,keyword=value,...]
Valid Keywords:
  userid -- ORACLE username/password
  control -- Control file name
  log -- Log file name
  bad -- Bad file name
  data -- Data file name
  discard -- Discard file name
discardmax -- Number of discards to allow (Default all)
  skip -- Number of logical records to skip (Default 0)
  load -- Number of logical records to load (Default all)
  errors -- Number of errors to allow (Default 50)
  rows -- Number of rows in conventional path bind array or
         between direct path data saves
         (Default: Conventional path 64, Direct path all)
  bindsize -- Size of conventional path bind array in bytes
             (Default 65536)
  silent -- Suppress messages during run
           (header,feedback,errors,discards,partitions)
  direct -- use direct path (Default FALSE)
  parfile -- parameter file: name of file that contains parameter
specifications
  parallel -- do parallel load (Default FALSE)
  file -- File to allocate extents from
skip_unusable_indexes -- disallow/allow unusable indexes or index
partitions (Default FALSE)
skip_index_maintenance -- do not maintain indexes, mark affected
indexes as unusable (Default FALSE)
```

```
commit_discontinued -- commit loaded rows when load is discontinued
(Default FALSE)
readsize -- Size of Read buffer (Default 1048576)
```

PLEASE NOTE: Command-line parameters may be specified either by position or by keywords. An example of the former case is 'sqlload scott/tiger foo'; an example of the latter is 'sqlload control=foo userid=scott/tiger'. One may specify parameters by position before but not after parameters specified by keywords. For example, 'sqlload scott/tiger control=foo logfile=log' is allowed, but 'sqlload scott/tiger control=foo log' is not, even though the position of the parameter 'log' is correct.

Cuando se da un valor personalizado a las *Palabras Clave*, estas son separadas por comas y pueden ser ingresadas en cualquier orden. Las palabras claves son seguidas de argumentos válidos después de un signo “=”, por ejemplo;

```
clima.igp.gob.pe>sqlldr CONTROL=foo.ctl, LOG=bar.log, BAD=baz.bad,
DATA=etc.dat USERID=scott/tiger, ERRORS=999, LOAD=2000,
DISCARD=toss.dis, DISCARDMAX=5;
```

A continuación se describirá algunas de las *Palabras Claves* más comunes a fin de conocer su función.

USERID (username/password). Es usado para proveer el nombre de usuario y contraseña de la cuenta propietaria de la tabla en la que se va a realizar la inserción de los datos. Si durante una operación de inserción esta es omitida esta información, SQL*Loader le solicitará que se ingrese dicha información.

CONTROL (Control file). Especifica el nombre del *Archivo de Control* que indica como insertar los datos. Si la extensión de un archivo o el tipo de archivo no está especificado, este será por defecto CTL (*.ctl). Si el nombre del archivo es omitido durante una operación de inserción empleando el SQL*Loader, este le solicitará que lo ingrese.

DATA (Datafile). Especifica el nombre del *Archivo de Datos* que contiene los datos a ser insertados. Si el nombre del *archivo de datos* no está especificado durante una operación de inserción, el nombre del *archivo de control* es usado por defecto, con la extensión DAT (*.dat).

LOG (File log). Especifica el nombre del *Archivo LOG* que SQL*Loader tomará para almacenar información estadística de la inserción. En este archivo se detallan los resultados de la operación de inserción. Si el nombre de un archivo no es especificado durante la operación de inserción, el nombre del *archivo de control* es usado por defecto añadiéndole la extensión LOG (*.log).

BAD (Bad file). Especifica el nombre del archivo empleado por SQL*Loader para almacenar registros que producen errores durante la inserción y no pueden ser insertados ya que están formateados inadecuadamente. Si el nombre de archivo no está especificado, SQL*Loader toma el nombre del *archivo de datos* por defecto con la extensión BAD (*.bad). Este archivo tiene el mismo formato que el *archivo de datos* de ingreso, a fin de que pueda ser cargado por el mismo *archivo de control* una vez realizadas las correcciones necesarias.

ERRORS (Errores permitidos). Especifica el número máximo de errores de inserción permitidos. Si el número de errores excede el valor del parámetro ERRORES, SQL*Loader termina la operación de inserción. Por defecto el número es 50. Para no permitir ningún error, se le debe asignar el valor cero (ERRORES = 0). En todos los casos SQL*Loader escribe los registros erróneos en el *archivo Bad* (bad file).

DISCARDFILE (File name). Especifica el nombre de un *archivo de descarte* (opcional) donde SQL*Loader almacena los registros que fueron negados a ser insertados dentro de una tabla o son rechazados dada una condición de ingreso. Si el nombre del *archivo de descarte* no es especificado, este por defecto es DSC. Este

archivo tiene el mismo formato que el *archivo de datos* de ingreso, así, la información de este podrá ser insertada por el mismo archivo de control después de que se hagan las actualizaciones o las correcciones apropiadas.

DISCARDMAX (Integer). Especifica el número de registros descartados que SQL*Loader admite antes de que la operación de inserción de los datos este terminada. El valor por defecto es todos (all). Por ejemplo, para que una sección de inserción se detenga en el primer registro descartado, especifique el valor uno (1).

LOAD (registro a cargar). Especifica el número máximo de registros lógicos a cargar, después de obviar el número especificado de registros. Por defecto todos los registros son cargados.

PARALLEL (Parallel load). Especifica si la carga directa puede operar en secciones múltiples simultáneas para cargar los datos dentro de la misma tabla.

PARFILE (Parameter File). Especifica el nombre de un archivo que contiene las *Palabras Claves* con valores personalizados que se desea configurar durante una operación de inserción. Por ejemplo;

```
SQLldr PARFILE=example.par
```

Donde, el archivo de parámetros *example.par* tiene el siguiente contenido:

```
userid=scott/tiger
control=example.ctl
errors=9999
log=example.log
```

DIRECT (Data path). Especifica el modo de inserción de los datos; es decir, el método de inserción de datos a utilizar. Estos pueden ser de modo Convencional o

modo Directo. El valor *TRUE* especifica el modo Directa y el valor *FALSE* especifica el modo Convensional. El valor por defecto es *FALSE*.

El método Convencional crea una instrucción SQL *insert* para cada fila del *archivo de datos*. El método Directo crea bloques de datos en memoria y los coloca en la base de datos sin necesidad de generar una instrucción *insert* para cada fila. El valor predeterminado de este parámetro es *FALSE* (falso). Para ejecutar una carga directa hay que incluir la orden **direct=true**. La inserción en modo Diercto (*TRUE*) se ejecuta más rápidamente que la inserción en modo convencional.

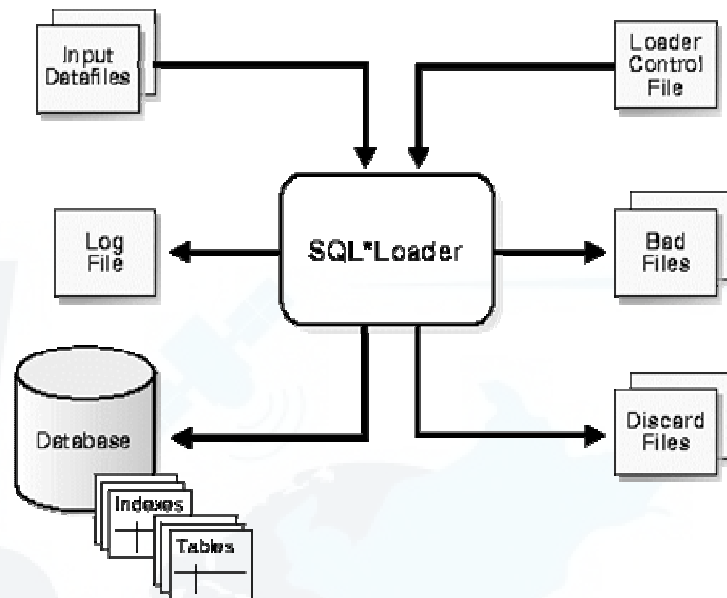
El administrador de la base de datos puede realizar todas las modificaciones que le permita hacer SQL*Loader empleando las *Palabras claves* descritas anteriormente. Sin embargo, para realizar una operación de inserción con los valores por defecto que toma SQL/Loader, la sintaxis que se ingresa en línea de comandos es;

```
sqlldr userid=<yourName>/<yourPasswd> control=<ctlFile>;
```

donde: *<yourName>* representa el nombre de usuario del Administrador de Base de Datos Sísmicos propietario de la tabla, *<yourPasswd>* es la contraseña de ingreso y *<ctlFile>* es al nombre del archivo de control con extensión **.ctl*. Sin embargo, se puede indicar únicamente el nombre de usuario y dejar que SQL*Loader pida el ingreso del password para mayor seguridad en la operación.

Como ya se mencionó, SQL*Loader emplea para el ingreso de información dentro de la base de datos Oracle un *Archivo de Control* y uno o más *Archivos de Datos*. Este *Archivo de control*, valga la redundancia, controla el comportamiento de SQL*Loader y uno o más Archivos de Datos. Los resultados o la salida que arroja SQL*Loader es una base de datos Oracle (donde los datos están cargados), un *archivo Log*, un *archivo Bad* y potencialmente un *archivo de descarte*, de los cuales se tratará posteriormente. Un esquema que muestra este funcionamiento global se aprecia en la Figura 5.2. A continuación, se va ha tratar cada uno de estos archivos por separado, a

fin de comprender más sobre las posibilidades que brinda el uso del SQL*Loader y entender su funcionamiento.



*Figura 5.2. Esquema del trabajo realizado por el SQL*Loader.*

5.3.- Archivo de Control

El *archivo de control* es un archivo en formato asccii, escrito empleando sentencias que SQL*Loader entiende. El *archivo de control* describe la labor que SQL*Loader realiza durante un proceso de inserción. El *archivo de control* le dice a SQL*Loader donde encontrar los datos, como hacer un análisis sintáctico para interpretar los datos y donde insertar los datos. Aunque no esta definido precisamente, se puede decir que un *archivo de control* tiene cuatro partes o secciones principales, tal como se muestra en la Figura 5.3.

a) *Primera Sección o Load Data*. La palabra clave *Load Data* (Carga de datos) inicia la mayor parte de los archivos de control de SQL*Loader, independientemente del contenido del resto del archivo. Asimismo, sirve como punto de partida para el resto del archivo.

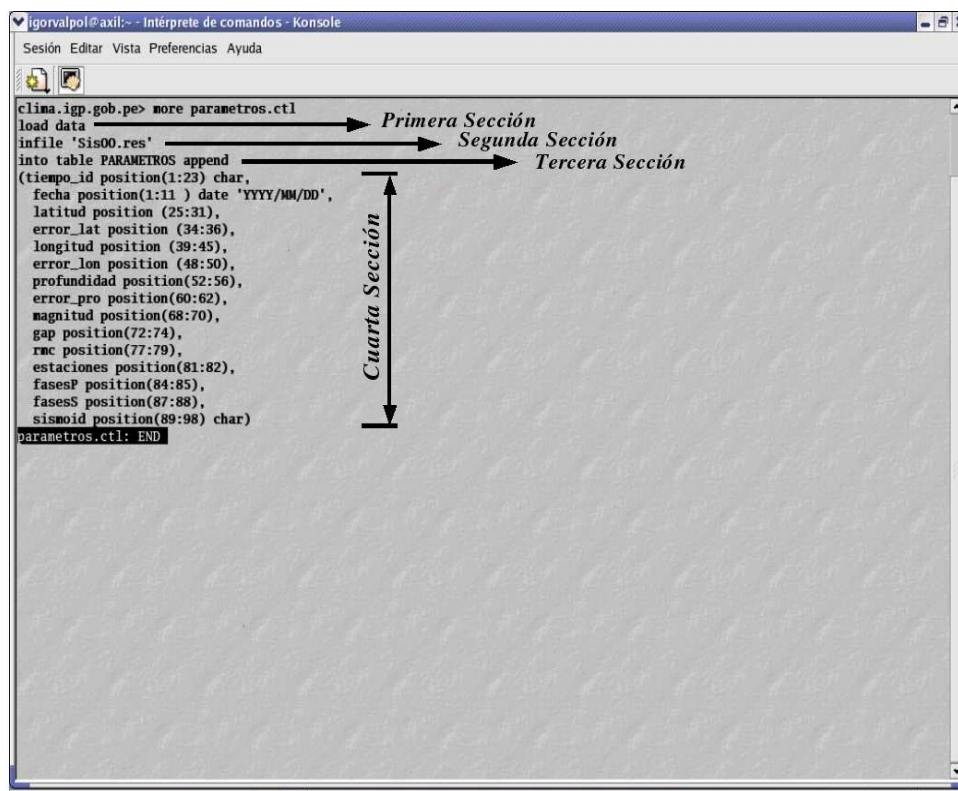


Figura 5.3. Secciones que conforman un Archivo de Control.

b) *Segunda Sección o Infile.* Esta línea nombra al *archivo de datos*, el cual se encierra entre comillas sencillas.

c) *Tercera Sección o Into Table.* Esta línea instruye a SQL*Loader sobre donde colocar los datos cuando se insertan dentro de Oracle (nombre de la tabla). En el *archivo de control* existen cuatro modificadores para la orden **into table**:

1. **Insert**, es el modificador predeterminado e indica que la tabla donde se cargarán los datos estará vacía cuando comience la carga.
2. **Append**, inserta nuevas filas a una tabla que ya contienen información.
3. **Replace**, borra las filas previamente insertadas de una tabla e ingresa las nuevas filas.

4. **Truncate**, se comporta igual que **Replace**.

Normalmente, no se incluye el calificador **Insert** ya que es el valor predeterminado. El error más común con que se puede encontrar un DBA que intenta insertar datos en una tabla en la que anteriormente se ha insertado filas, es no incluir en esta sección algún modificador **Append**, **Replace**, o **Truncate**. Si esto ocurre, SQL*Loader devuelve error.

d) *Cuarta Sección, Especificaciones de columna y campo*. Esta sección del *archivo de control* establece la correspondencia entre los caracteres del *archivo de datos* de entrada y las columnas de la tabla de destino (tabla *catalogo_1983_act*). En esta especificación, cada línea tiene cuatro partes: el nombre de la columna de la tabla de destino, la palabra clave **position**, las posiciones de los caracteres inicial y final, y el tipo de dato comprendido entre dichos caracteres del *archivo de datos*. La palabra clave **position** seguida por las especificaciones de número de caracteres cobra importancia cuando se desea insertar solo una parte de cada línea del *archivo de datos*, en lugar de la línea completa. En la Figura 5.4 se aprecia el contenido de un *archivo de datos*, el mismo que corresponde a la *tabla catalogo_1983_act* cuya estructura de campos fue previamente definida en el capítulo anterior. El contenido del *archivo de control*, que permitirá la carga de datos del *archivo de datos* de la Figura 5.4 a la *Tabla catalogo_1983_act* se presenta a continuación. Intencionalmente se colocó en la cabecera del archivo una fila que señala la posición de los datos partiendo de izquierda a derecha.

```
load data
infile 'Sis00.res'
into table CATALOGO_1983_ACT append
(tiempo_id position (1:23) char,
fecha      position (1:11 ) date 'YYYY/MM/DD',
latitud    position (25:31),
error_lat  position (34:36),
longitud   position (39:45),
error_lon  position (48:50),
profundidad position (52:56),
error_pro  position (60:62),
mbl        position (64:66),
```

```

mb2          position (68:70),
gap          position (72:74),
rmc          position (77:79),
estaciones   position (81:82),
fasesP       position (84:85),
fasesS       position (87:88),
sismoid      position (89:98) char)

```

igorvalpol@axil:~ - intérprete de comandos - Konsole

Sesión Editar Vista Preferencias Ayuda

*****1*****2*****3*****4*****5*****6*****7*****8*****9*****0

2000/01/01 7:44:23.08

-11.303 2.8

-76.791 4.0

133.7 2.5

2.6 327

0.2 3 3 3

00001001

2000/01/01 10:23: 9.27

-13.309 1.3

-75.916 2.7

83.0 3.4

3.2 193

0.3 6 6 4

00001002

2000/01/01 13:26:24.02

-13.934 1.9

-76.189 2.0

42.1 1.6

3.1 164

0.5 11 11 4

00001003

2000/01/01 19:49: 0.55

-6.259 1.0

-79.598 1.6

78.5 1.3

2.3 199

0.1 4 4 4

00001004

2000/01/02 3:26:55.39

-10.950 2.1

-73.998 1.7

27.0 0.0

333

0.3 6 6 2

00002001

2000/01/02 7:50:33.00

-4.596 3.6

-79.830 5.6

79.5 7.6

3.6 348

0.2 4 4 2

00002002

2000/01/02 8:40:21.78

-10.973 2.1

-74.671 1.7

156.7 2.9

3.4 295

0.4 10 10 8

00002003

2000/01/02 14:34:18.40

-11.033 2.3

-74.345 3.1

47.8 1.7

329

0.4 7 7 5

00002004

2000/01/02 21:16:40.18

-14.929 0.5

-75.836 1.3

61.8 1.1

2.9 297

0.1 3 3 2

00002005

2000/01/02 22:53:22.54

-15.203 5.0

-75.576 7.5

43.8 2.4

3.6 336

0.3 6 6 1

00002006

2000/01/03 6:12:52.33

-12.437 1.8

-75.177 1.6

100.4 3.1

3.0 107

0.4 11 11 4

00003001

2000/01/03 9:20:20.07

-15.021 4.6

-75.518 6.6

22.3 6.7

2.5 341

0.5 3 3 3

00003002

2000/01/03 11:53:54.10

-10.569 2.9

-74.187 3.2

57.5 0.0

338

0.3 4 4 2

00003003

2000/01/03 12:56:27.01

-11.163 5.1

-72.978 3.4

55.7 2.1

340

0.3 5 5 4

00003004

2000/01/03 15:56:59.33

-3.378 3.8

-80.479 9.9

4.0 1.0

3.7 355

0.3 4 4 4

00003005

2000/01/03 17: 6:52:45

-14.473 0.3

-75.745 1.1

30.0 1.2

2.1 172

0.1 3 3 2

00003006

2000/01/03 21:43:31.41

-6.269 1.7

-79.916 2.6

82.1 2.6

2.6 275

0.2 4 4 4

00003007

2000/01/04 1: 4:53.43

-13.962 0.5

-76.088 0.6

53.0 0.5

2.1 152

0.1 5 5 4

00004001

2000/01/04 4:34:18.15

-12.053 5.5

-77.572 3.1

52.4 3.4

2.4 316

0.3 3 3 3

00004002

2000/01/04 4:46:18.98

-14.730 4.1

-74.770 3.9

33.0 4.3

2.8 316

0.5 4 4 4

00004003

2000/01/04 13:56: 5.94

-14.806 0.7

-74.061 0.8

91.5 1.8

3.4 315

0.1 4 4 3

00004004

2000/01/04 18: 3:58.31

-13.417 0.4

-76.336 1.1

33.8 2.0

1.9 168

0.1 3 3 3

00004005

2000/01/05 7:33:51.66

-10.693 3.6

-73.778 4.6

28.3 0.0

341

0.5 4 4 4

00005001

2000/01/05 12:12:49.34

-8.931 2.0

-79.533 6.6

16.6 4.5

4.0 213

0.5 8 8 5

00005002

2000/01/06 2:13:48.83

-6.297 1.4

-80.566 1.4

15.2 2.4

2.8 316

0.2 4 4 3

00006001

2000/01/06 8:32: 5.83

-7.367 2.4

-77.556 1.9

94.2 3.9

3.3 340

0.1 4 4 2

00006002

2000/01/06 12:23:34.20

-7.208 1.5

-80.183 1.5

10.9 1.9

2.3 314

0.3 4 4 4

00006003

2000/01/06 18:12:29.61

-8.678 1.4

-79.729 4.7

9.0 3.7

3.8 217

0.5 6 6 5

00006004

2000/01/06 18:41:10.47

-14.560 3.4

-75.567 3.6

71.5 2.3

3.7 112

0.6 10 10 5

00006005

2000/01/06 21:37:47.87

-17.858 7.7

-69.871 5.4

201.8 6.3

3.9 347

0.4 7 7 6

00006006

2000/01/07 7:27:19.41

-11.696 0.7

-73.315 1.3

36.5 1.0

345

0.4 3 3 2

00007001

2000/01/07 9:34:10.05

-5.617 0.7

-80.023 0.6

59.0 0.7

2.6 336

0.0 3 3 2

00007002

2000/01/07 15:46: 2.85

-14.376 0.5

-76.119 0.6

16.0 3.2

1.8 218

0.1 3 3 3

00007003

2000/01/07 23: 8:17.26

-11.916 0.2

-74.343 0.3

13.5 1.2

314

0.0 4 4 3

00007004

2000/01/08 1: 3:33.64

-4.748 3.0

-77.908 4.9

232.7 5.6

3.1 335

0.2 4 4 4

00008001

2000/01/08 8:26: 7.23

-12.309 2.6

-74.437 2.2

18.7 4.1

302

0.2 5 5 4

00008002

2000/01/08 9:51:24.16

-12.016 7.4

-74.849 5.1

98.6 4.3

248

0.3 5 5 4

00008003

SQLLoader Concepts - Konsole

Gimp

igorvalpol@axil:~ - intérprete de comandos - Konsole

VCLSalFrame

12:45

Figura 5.4. Contenido de un Archivo de Datos visto desde un terminal Linux.

Cabe señalar que el formato predeterminado para el tipo de dato *date* en Oracle es DD-MON-YY, siendo DD el día; MON, los tres primeros caracteres del mes en inglés y YY los dos últimos dígitos del año. Si el formato de la fecha del *archivo de datos* no están en ese formato, se debe indicar a Oracle en que formato se encuentra la fecha en dicho archivo. En el caso del valor asignado a la fecha del *archivo de datos* mostrado en la Figura 5.4, este se encuentra en un formato YYYY/MM/DD, indicado dentro de la estructura *archivo de control* visto anteriormente, Así será identificado e

ingresado a la base de datos de manera adecuada. Dado que hay reglas para las fechas (por ejemplo, el mes 13 no existe, así como el 31 del mes de junio), Oracle rechazará los datos del *archivo de datos* que violen dichas reglas.

5.4.- Datos de Ingreso y archivo de Datos.

El otro ingreso a SQL*Loader, al igual que el *archivo de control*, son los datos contenidos en los *Archivos de Datos* (ver Figura 5.4). SQL*Loader lee los datos de uno o más archivos especificados en el *archivo de control* y los carga o almacena en la base de datos. Para tal caso, desde la perspectiva del SQL*Loader, los datos en el archivo de datos están organizados en registros.

5.4.1.- Registros Físicos.

Se defiende a los *Registros Físicos* como filas de información formadas por todos los elementos de datos almacenados en una tabla de Oracle. En un archivo de datos cada línea de datos separados por un “enter” representa un *registro físico* (ver Figura 5.4).

5.4.2.- Registros Lógicos.

SQL*Loader organiza los datos de entrada en registros físicos de acuerdo a un formato de registro especificado. Por defecto, un registro físico es un registro lógico, pero para adicionar mayor flexibilidad, SQL*Loader puede ser instruido para combinar un número de registros físicos en un sólo registro lógico si es que el DBA lo desea.

5.4.3.- Campos de datos.

Una vez formado un *registro lógico*, el ajuste de los campos en el *registro lógico* es realizado. El *ajuste del campo* es el proceso donde el SQL*Loader, basado en las

especificaciones de campos del *archivo de control*, determina qué parte de los datos en dicho registro lógico corresponden a qué campo de la tabla definidos en el *archivo de control*. Es posible que dos o más especificaciones de campos exijan los mismos datos; además un registro lógico puede contener datos que no son demandados por alguna especificación de campo del archivo de control; es decir, que cuando se ingrese información de un registro lógico dentro de una tabla de la base de datos no necesariamente será toda la información que contenga dicho registro, es de carácter selectivo.

La mayoría de las especificaciones hechas en los campos del *archivo de control* exigen una parte particular del registro lógico. Este señalamiento selectivo se puede hacer de las formas siguientes:

- La *posición del byte de inicio y final* de los datos de un campo o ambos puede ser especificada. Esta forma de especificación no es la más flexible pero proporciona un funcionamiento para el ajuste de campo muy alto.
- Los *delimitadores de cadenas* (encerrando y/o terminando) de un dato de campo particular pueden ser especificados. Un dato de campo delimitado se asume que comienza donde el último dato de campo termina, a menos que la posición del byte donde comienza el dato del campo se especifique.
- La *posición del byte inicial y/o la longitud del campo de datos* pueden ser especificados. Así, cada campo se inicia con un número que indica la *posición del byte inicial* especificado o donde el último campo terminó y contiene toda la información que abarca la *longitud de campo* especificado.
- Los tipos de datos que señalan la *Longitud máxima* y *Valor fijo* que debe tener el campo de datos de la tabla pueden ser utilizados. En este caso, el primer número de x bytes de un campo de datos determina cuan grande es el resto del campo de datos.

La Figura 5.5 muestra las etapas en las cuales los datos de un *archivo de datos* pasan a las columnas de las tablas de la base de datos Oracle, durante una carga de forma convencional.

La Figura 5.5 también muestra la división del trabajo entre el SQL*Loader y el servidor de la base de datos Oracle. Las especificaciones de campo dicen a SQL*Loader cómo interpretar el formato del *archivo de datos*. Una vez realizada la tarea del SQL*Loader, el servidor de la base de datos Oracle convierte esos datos a los datos definidos por Oracle, usando para esto los tipos de datos de la columna en los que esta definida la tabla y por ultimo los inserta en las columnas de la base de datos. Debe tenerse presente la distinción entre el campo en un *archivo de datos* y una columna en la base de datos. Asimismo, los tipos de datos del campo definidos en un *archivo de control* del SQL*Loader no son iguales que los tipos de datos de la columna en una tabla Oracle.

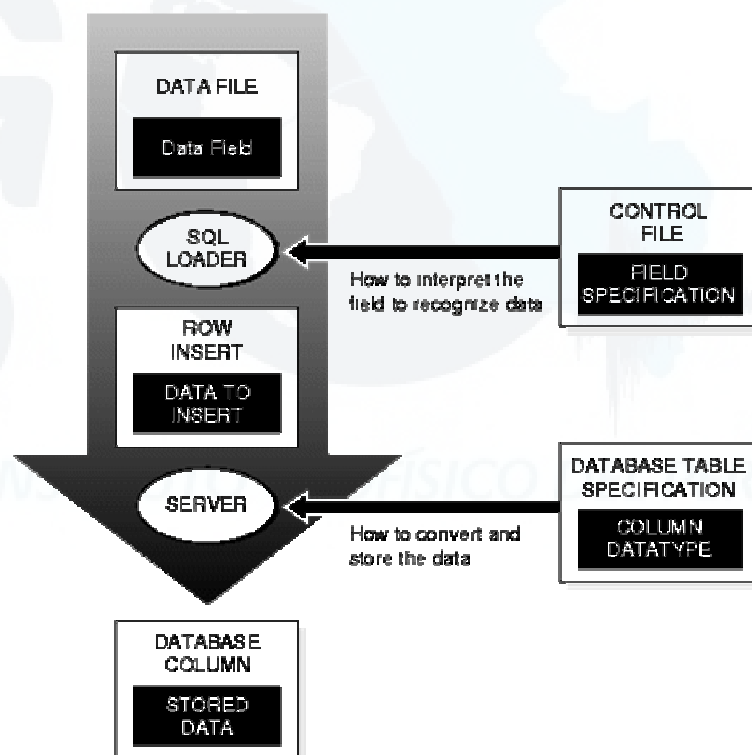


Figura 5.5 Etapas por la que pasan los datos de una Archivo de Datos desde que son leídos por el Archivo de Control empleando el SQL*Loader, hasta que son ingresados a las tablas dentro de la base de datos en el servidor

El SQL*Loader utiliza las especificaciones de campo en el archivo de control para analizar los datos de entrada y para conformar el arreglo vinculado que corresponden a una sentencia *Insert* de SQL usando esos datos. La sentencia *Insert* entonces es ejecutada por el servidor de base de datos Oracle para ser almacenada en la tabla. El servidor de base de datos Oracle utiliza el tipo de dato de la columna para convertir los datos en su formato de almacenamiento final. Hay dos pasos a seguir para la conversión:

- 1.El SQL*Loader identifica un campo en el archivo de datos, interpreta los datos, y los pasa al servidor de base de datos Oracle usando un almacenador intermedio ó buffer.
- 2.El servidor de base de datos Oracle acepta los datos y los almacena en la base de datos.

En la Figura 5.6, se muestra un ejemplo con dos campos CHAR definidos para un registro de datos. Las especificaciones de campo están definidas en el *archivo de control*. En dicha figura se observa que la especificación CHAR del archivo de control no es igual que la especificación CHAR de la base de datos. Un campo de datos definido como CHAR en el archivo de control simplemente le dice a SQL*Loader cómo crear la inserción de la fila. Para este ejemplo, los datos podrían ser insertados en una columna definida con los tipos de datos CHAR, VARCHAR2, NCHAR, NVARCHAR, o aún en una columna definida con el tipo de dato NUMBER en la base de datos con el servidor de Oracle8i.

Por defecto, el SQL*Loader quita espacios que se arrastran de los datos CHAR del *archivo de datos* antes de pasarlo a la base de datos. Así, en la Figura 5.6 el campo 1 esta definido con el tipo de dato CHAR y el campo 2 con el tipo de dato VARCHAR, ambos al ser insertados a la base de datos en las columnas de la tabla señalan una diferencia.

La columna 1 se define en la base de datos como una columna de longitud fija (*Char* de longitud 5). Los datos (aaa) se justifican a la izquierda de esa columna, la cual sigue siendo de cinco caracteres de par en par. El espacio adicional a la derecha se rellena con los espacios en blanco. La columna 2, sin embargo, se define como un campo cuya longitud varía hasta un máximo de cinco caracteres (*Varchar* de longitud máxima 5). Los datos para esa columna (bbb) también se justifican a la izquierda, pero la longitud del dato permanece con tres caracteres sin espacios en blanco.

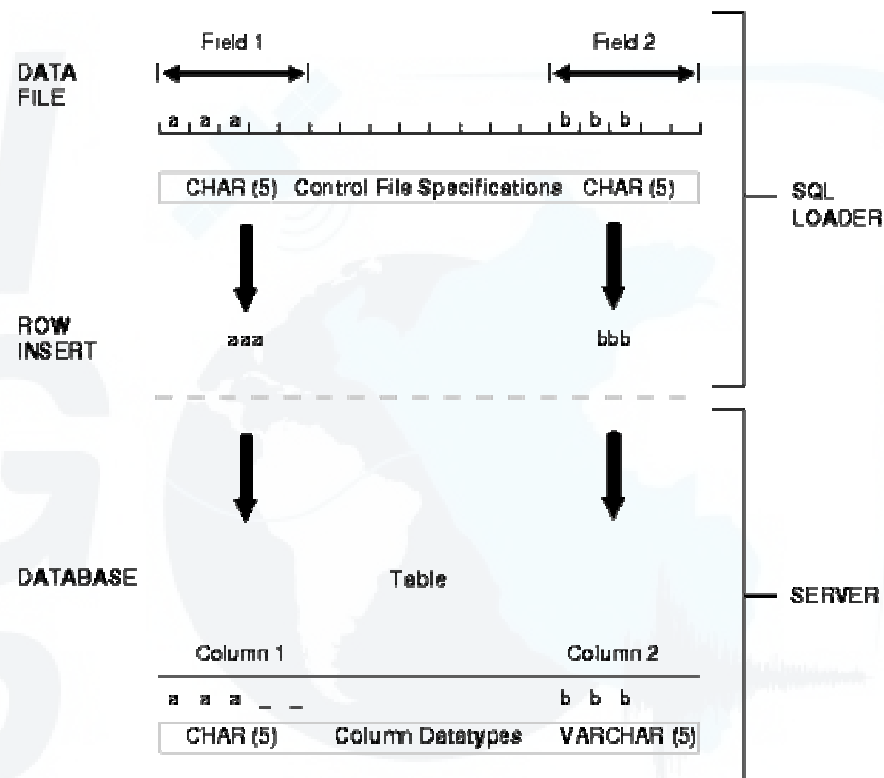


Figura 5.6. Diferencia en el ingreso de un mismo dato entre dos campos definidos con los tipos de datos Char y Varchar.

En resumen, se tiene lo siguiente:

- El nombre del campo de datos corresponde al nombre de la columna de la tabla en la cual los datos deben ser cargados.
- El tipo de dato del campo le dice a SQL*Loader cómo tratar los datos en el archivo de datos. Este no es igual que el tipo de dato de la columna de la tabla. Los tipos de

datos del *archivo de datos* de entrada al SQL*Loader son independientes del tipo de dato de la columna.

- Los datos se convierten del tipo de dato especificado en el *archivo de control* al tipo de dato de la columna en la base de datos.
- Hay una distinción entre los registros lógicos y los físicos.

5.5.- Salidas de SQL*Loader

Cuando se realiza una operación de inserción de datos con SQL*Loader, este escribe una serie de archivos que reflejan como ha transcurrido la inserción de información a la base de datos. De manera predeterminada, SQL*Loader genera un *archivo Log* en función al éxito o fracaso de la operación de inserción y cuyo contenido representa los resultados estadísticos y los parámetros utilizados durante el proceso de inserción. Además, genera un archivo de registros descartados o *archivo de descarte* (discard) y un *archivo de fallos* (bad). Como ya se mencionó anteriormente, a no ser que se especifique los nombres de los *archivos log, de descarte y de fallos*, estos tres archivos llevarán el mismo nombre que el *archivo de control* empleado en el proceso de inserción y añadiéndole una extensión respectiva; es decir, *.log, *.bad y *.dsc, respectivamente.

5.5.1.- Archivo Log: Carga completa

Cuando SQL*Loader empieza su ejecución crea un *archivo de log* y si no puede crearlo simplemente termina su ejecución. El *archivo log* contiene un resumen detallado del proceso de inserción, incluyendo la descripción de algún error que se haya producido durante el proceso de inserción. Un ejemplo donde se muestra el contenido de un *archivo log* se ve a continuación (La numeración que aparece al costado izquierdo se a colocado únicamente como referencia para la descripción del contenido del archivo):

```

1 SQL*Loader: Release 8.1.6.0.0 - Production on Mon Nov 11 12:01:08 2002
2 (c) Copyright 1999 Oracle Corporation. All rights reserved.
3 Control File: parametros.ctl
4 Data File: Sis00.res
5 Bad File: Sis00.bad
6 Discard File: none specified
7 (Allow all discards)
8 Number to load: ALL
9 Number to skip: 0
10 Errors allowed: 50
11 Bind array: 64 rows, maximum of 65536 bytes
12 Continuation: none specified
13 Path used: Conventional
14 Table PARAMETROS, loaded from every logical record.
15 Insert option in effect for this table: APPEND
16 Column Name Position Len Term Encl Datatype
17 -----
18 TIEMPO_ID 1:23 23 CHARACTER
19 FECHA 1:11 11 DATE YYYY/MM/DD
20 LATITUD 25:31 7 CHARACTER
21 ERROR_LAT 34:36 3 CHARACTER
22 LONGITUD 39:45 7 CHARACTER
23 ERROR_LON 48:50 3 CHARACTER
24 PROFUNDIDAD 52:56 5 CHARACTER
25 ERROR_PRO 60:62 3 CHARACTER
26 MAGNITUD 68:70 3 CHARACTER
27 GAP 72:74 3 CHARACTER
28 RMC 77:79 3 CHARACTER
29 ESTACIONES 81:82 2 CHARACTER
30 FASESP 84:85 2 CHARACTER
31 FASESS 87:88 2 CHARACTER
32 SISMOID 89:98 10 CHARACTER
33 Record 1567: Discarded - all columns null.
34 Table PARAMETROS:
35 1566 Rows successfully loaded.
36 0 Rows not loaded due to data errors.
37 0 Rows not loaded because all WHEN clauses were failed.
38 1 Row not loaded because all fields were null.
39 Space allocated for bind array: 8192 bytes(64 rows)
40 Space allocated for memory besides bind array: 0 bytes
41 Total logical records skipped: 0
42 Total logical records read: 1567
43 Total logical records rejected: 0
44 Total logical records discarded: 1
45 Run began on Mon Nov 11 12:01:08 2002
46 Run ended on Mon Nov 11 12:01:10 2002
47 Elapsed time was: 00:00:02.49
48 CPU time was: 00:00:00.12

```

SQL*Loader presenta las líneas 1 y 2 al principio de todos los archivos de registro. Las líneas 3 a 15 informan el valor de los parámetros cuando se ejecuta la sesión. Las líneas 16 a 32 describen la información de la tabla y columnas que se especifican en el archivo de control. Las líneas 33 a 38 muestran el número de filas

insertadas con éxito y el número de filas rechazadas. Las líneas 39 a 48 muestran el tamaño de algunas estructuras de memoria de SQL*Loader, la hora a la que comenzó y finalizó la sesión, y el tiempo que duro el proceso.

5.5.2.- Archivo Log: Carga incompleta

Cuando se rechazan una o más filas de datos no validos, SQL*Loader escribe las filas erróneas en un *archivo de fallos* (bad). El *archivo log* de una carga incompleta es parecido al que se ha visto para una carga completa, pero con alguna información adicional. Las información adicional es:

- **Fila Rechazada.** Cuando SQL*Loader escribe las filas rechazadas en su *archivo de fallos*, incluye un bloque de información en el archivo de registro de la sesión, similar al siguiente:

```
Record 785: Rejected - Error on table catalogo_1983_act.
```

```
ORA-0001: unique constraint (PARAMETROS.SISMOID) violated
```

La tabla de destino establece que la columna de valores denominada *sismoid* debe ser única; por la tanto los datos del registro 785 se rechazan ya que contienen información duplicada para la columna *sismoid* de la tabla *catalogo_1983_act*. Por lo general, los registros son rechazados al ingreso cuando los datos no cumplen las restricciones que tienen las columnas de las tablas (not null, unique, etc.).

- **Estadística de Carga.** Cuando se rechaza una o más filas, los valores estadísticos cambian para reflejar estos datos. Por ejemplo, si se produce el rechazo de la fila anteriormente mencionada la estadística comprendida en las filas 34 a 38 cambiarían al siguiente texto:

```
34 Table PARAMETROS:
```

```
35 1565 Rows successfully loaded.
```

```

36 1 Rows not loaded due to data errors.
37 0 Rows not loaded because all WHEN clauses were failed.
38 1 Row not loaded because all fields were null.

```

En este ejemplo se ve un registro rechazado (38), debido a que el *archivo de datos* tenía una fila sin datos, la cual también fue considerada en el ingreso por el *archivo de control*.

5.5.3.- Archivo de Fallos

El *archivo de fallos* (bad file) solo se escribe cuando uno o más registros del archivo de entrada son rechazados. Para el ejemplo anterior, si se rechaza algún registro durante la carga, ésta se incluye en el *archivo de fallos* denominado *Sis00.bad*. El formato de los registros en el *archivo de fallos* es el mismo que presenta el *archivo de datos* (ver Figura 5.4). En el *archivo de fallos* no se incluye ningún texto descriptivo al error, por lo que se debe recurrir a los mensajes de error del *archivo de log*. Cuando se emplea el mismo *archivo de control* para realizar varias operaciones de inserción de datos de diversos *archivos de datos* en diferentes sesiones, es importante tener en cuenta los registros del *archivo de fallos* de cada sesión antes de iniciar la siguiente, ya que el archivo se sobre escribe cada vez que se utiliza.

5.5.4.- Archivo de descarte (discard)

El contenido del *archivo de control* presentado anteriormente indica a SQL*Loader que insertara todos los registros del archivo de entrada. SQL*Loader generará un *archivo de descarte* únicamente si se le indica; es decir, haciendo uso del parámetro **discard=<nombre _del_archivo.dsc>** cuando se invoque a SQL*Loader.

5.6.- Ingreso de los archivos Binarios a la Base de Datos Sísmicos.

Hasta ahora se ha venido tratando el ingreso de los datos de tipo numérico, caracteres y alfanumérico en los diferentes ejemplos y sentencias mostradas con la

intención de explicar el funcionamiento del SQL*Loader. A partir de este punto, se va a tratar el ingreso a la base de datos de la información no - estructurada.

Oracle emplea una forma especial de almacenamiento para este tipo de información y consiste en relacionar o vincular el campo o columna de una tabla con un archivo ubicado ya sea dentro de la base de datos o fuera de ella. Como se vio en el Capítulo anterior, estos tipos de datos se almacenan en columnas definidas con los tipos de datos Oracle *LOB* (Large Object).

Las secciones siguientes discuten el uso de SQL*Loader para insertar los tipos de datos LOB internos (BLOB's, CLOB's, y NCLOB's) y los LOBs externos (BFILEs).

5.6.1.- LOBs Internos.

Debido a que los LOBs pueden ser bastante grandes, SQL*Loader es capaz de insertar datos LOB desde cualquiera de los *archivos de datos principales* (leyendo la información del archivo LOB y almacenándola en los campos de la tabla) o desde LOBFILES. Para insertar información LOB de los archivos de datos principales, se puede usar los formatos estándar de SQL*Loader. Sin embargo, la forma más concreta de almacenar estos tipos de datos es empleando el LOBFILE.

5.6.1.1.- Cargando datos LOB internos usando LOBFILE.

Los datos LOB internos pueden ser muy grandes, lo que hace sensible insertarlos desde un LOBFILE. Empleando LOBFILE, los datos LOB están considerados a estar estáticos dentro de los campos (es decir, tamaño predeterminado y delimitados), los mismos que no están organizados dentro de registros (el concepto de registros no existe con LOBFILE). Por lo tanto, el procesamiento global de la transacción con registros es evitado. Este tipo de organización de datos es ideal para cargar datos LOB.

A continuación se muestra un *archivo de control* que permitirá la carga de datos tipo LOB internos empleando *LOBFIL*E, y en donde cada *LOBFIL*E es el origen de un LOB interno simple. Para insertar datos LOB internos organizados de la manera como se muestra en un *archivo de datos* mostrado seguidamente, en el cual se sigue el nombre de la columna o de campo con las especificaciones del tipo de dato de *LOBFIL*E.

```
LOAD DATA
INFILE 'sample.dat'
  INTO TABLE person_table
  FIELDS TERMINATED BY ','
  (name          CHAR(20),
(1)--> ext_fname    FILLER CHAR(40),
(2)--> "RESUME"     LOBFIL(ext_fname) TERMINATED BY EOF)
```

donde el *archivo de datos* tiene la siguiente distribución de campos:

```
Johny Quest,jqresume.txt,
Speed Racer,'/private/sracer/srresume.txt',
```

- (1)El *campo de relleno* (FILLER) indica un *campo de datos* con longitud de 40 bytes a ser leído usando el tipo de dato CHAR de SQL*Loader. Almacena el nombre del LOB interno.
- (2) SQL*Loader obtiene el nombre del LOB para el *LOBFIL*E del campo de relleno (FILLER) *ext_fname*. Este entonces carga los datos del *LOBFIL*E (usando el tipo de dato CHAR) desde el principio hasta el final del archivo(EOF; End of File). Si ningún *LOBFIL*E existente es especificado, el campo “RESUME” es asignado como vacío.

Para realizar el ingreso de datos LOB desde *LOBFIL*Es, se debe tener presente lo siguiente:

- Un fallo al cargar un LOB particular no da lugar al rechazo del registro que contiene

- ese LOB, por el contrario se tendrá un registro que contiene un LOB nulo.
- No es necesario especificar la longitud máxima del campo que corresponde a una columna del tipo LOB; sin embargo, si una longitud máxima es especificada, SQL*Loader usa esta información como una insinuación para optimizar el uso de la memoria. Por lo tanto, es importante que la especificación de la máxima longitud no minimice la longitud máxima verdadera.

5.6.2.- LOB Externos.

El ingreso de la información de tipo BFILE dentro de la Base de Datos se realiza de una forma especial y distinta a la empleada para los tipos de datos numéricos, carácter o alfanuméricos descritos anteriormente. Oracle permite insertar información de tipo BFILE haciendo uso de la función “BFILENAME” con la siguiente sintaxis:

BFILENAME(DIRECTORY, <Nombre_del_Archivo>)

Donde DIRECTORY es un nombre de alias lógico para un directorio físico existente el sistema de archivos del servidor; es decir, es la ruta de un directorio dentro del servidor Oracle en la cual se encuentran todos los archivos BFILE que se desee ingresar a la base de datos. La sintaxis usada por Oracle para crear un “DIRECTORY” es:

`CREATE DIRECTORY <nombre> AS <path>`

Donde <nombre> especifica como se le va a conocer a DIRECTORY y <path> es la ubicación del directorio en el sistema de archivos del servidor. Esta orden se ejecuta desde el SQL*Plus, y un ejemplo de ello es como sigue:

```
SQL> create DIRECTORY eventos0101 AS 'users/cndg/0101';
```

Una vez definido este directorio o lugar físico de almacenamiento, se ubica allí toda la información de tipo BFILE. Luego, desde SQL*Plus se ejecuta el comando SQL de inserción de datos (*INSERT*) que incluye la función BFILENAME de la siguiente forma:

```
SQL> INSERT INTO tabla (campo1, campo2)
      values (valor1, BFILENAME(DIRECTORY, archivo_binario));
```

ejemplo:

```
SQL> INSERT INTO DATOS(sismoid, eventosis, ndxfile)
      values('01026001', BFILENAME(eventos0101,
      '01012601.sis'), BFILENAME(eventos0101,
      '01012601.ndx'));
```

Una vez realizada esta operación con cada uno de los datos contenidos en la carpeta “DIRECTORY, la información recién será almacena dentro de la Base de Datos.

5.6.2.1.- Ingreso de LOB Externos (BFILE) empleando SQL*Loader.

Los tipos de datos BFILE almacenan datos binarios no estructurados en archivos de sistemas operativos fuera de la base de datos. Un atributo o columna BFILE almacena un localizador de archivo que apunta al archivo externo que contiene los datos. El archivo a ser localizado como un BFILE no tiene que existir necesariamente al momento de ser cargado, este puede ser creado posteriormente. SQL*Loader asume que los objetos del directorio, “*DIRECTORY*”, necesarios ya han sido creados.

Un campo del archivo de control correspondiente a una columna BFILE consiste en el nombre de la columna seguido por la cláusula BFILE. La cláusula BFILE toma como argumentos el nombre de un DIERCTORY seguido por el nombre de un archivo tipo BFILE, ambos pueden ser proveídos como cadenas constantes o pueden ser dinámicamente cargados a través de algún otro campo.

A continuación se muestran dos ejemplos de *archivos de control* que permiten la carga de BFILES, el primero solo tiene el nombre de archivo especificado dinámicamente; mientras que el segundo muestra la especificación del BFILE y el DIRECTORY dinámicamente.

Ejemplo 1.

```
LOAD DATA
INFILE sample.dat
INTO TABLE planets
FIELDS TERMINATED BY ','
  (pl_id      CHAR(3),
   pl_name    CHAR(20),
   fname      FILLER CHAR(30),
(1)--> pl_pict  BFILE(CONSTANT "scoTT_dir1", fname))
```

donde el archivo de datos tiene el siguiente formato:

```
1,Mercury,mercury.jpeg,
2,Venus,venus.jpeg,
3,Earth,earth.jpeg,
```

(1)El nombre del directorio esta citado. Por lo tanto, la cadena es usada como esta y no es renombrada.

Ejemplo 2.

```
LOAD DATA
INFILE sample.dat
INTO TABLE planets
FIELDS TERMINATED BY ','
  (pl_id      NUMBER(4),
   pl_name    CHAR(20),
   fname      FILLER CHAR(30),
```

```
(1)--> dname      FILLER CHAR(20));  
      pl_pict    BFILE(dname, fname),
```

donde el archivo de datos tiene el siguiente formato:

```
1, Mercury, mercury.jpeg, scott_dir1,  
2, Venus, venus.jpeg, scott_dir1,  
3, Earth, earth.jpeg, scott_dir2,
```

(1)*dname* indica el campo del archivo de datos que contiene el nombre del directorio correspondiente al archivo que esta siendo cargado.

5.7.- Aplicaciones diseñadas para la generación de archivos de datos.

Como se ha venido diciendo en este Capítulo, la herramienta de Oracle SQL*Loader es muy importante y práctica a la hora de cargar cualquier tipo de dato; sin embargo, es importante tener como punto de partida un *archivo de datos* en un formato ideal para que este pueda ser leído adecuadamente por el *archivo de control* que el administrador de la base de datos diseñe.

Para diseñar estos *Archivos de Datos* se han elaborado aplicaciones o pequeños programas, los mismos que van a realizar la tarea de leer los nombres de los archivos BFILE que se desea insertar de un determinado directorio con la intención de darles un formato adecuado y fácil de ser leído por un *archivo de control*. Este formato será adecuado para el ingreso de la información en las tablas anteriormente definidas en el Capítulo 4 de este trabajo.

Los programas han sido diseñados principalmente en función a la ubicación que tienen los archivos BFILE; es decir, a la forma como van a estar conformado los directorios que almacenan a los BFILE y LOB's dentro y fuera del servidor.

5.7.1.- Organización de directorios dentro del servidor.

La organización de los directorios dentro del servidor esta realizada con el fin de optimizar, tanto el almacenamiento de la información de tipo archivos LOB, como también para optimizar la consulta de esta información, la cual será tratada posteriormente. Un diagrama que muestra como se encuentran organizados los diferentes directorios que contienen a los archivos LOB dentro del servidor se muestra en la Figura 5.7. En esta Figura, todos los directorios se encuentran dentro de un directorio principal denominado “data”. El nombre que llevan estos directorios representa la información que almacenan y en algunos casos es el mismo nombre de las tablas a las cuales corresponde esta información y que fueron previamente definidas en el capítulo anterior.

El directorio FORMAS_DE_ONDA contiene los directorios ONDAS_SIS y ONDAS_SAC. En estos dos directorios se almacenan los archivos provenientes de la Red Sismica Nacional; es decir, *Ondas_sis* para los registros de las estaciones de Periodo Corto y *Ondas_sac* para las de banda ancha. Asimismo, el directorio *Ondas_sis*, esta dividido en los directorios Norte, Lima, Sur y Huan, los cuales corresponden a los nombres de la Sub-Redes de estaciones de Periodo Corto de la RSN. Dentro de estos directorios existen más directorios los cuales se nombran con el año y el mes correspondiente a la información que contienen y estos a vez están divididos por el directorio nombrado por un número el cual indica el día. Mientras que, el directorio *Ondas_sac* esta dividido en directorios con los nombre correspondientes a los códigos nacionales de todas estas estaciones de Banda Ancha de la RSN y también, cada uno de estos se encuentran divididos por diferentes directorios con los nombre que indica el año y el mes correspondiente a la información que contienen (ver Figura 5.7).

Asimismo, el directorio GRAFICOS contiene dos directorios con los nombres FOTOS y MAP_INT y estos a su vez están conformados por un conjunto de directorios con los nombres que indican la fecha a la cual corresponde la información y dentro de cada uno de estos un conjunto de directorios con nombres que corresponden al sismo

(sismoid), tal como se ve en la Figura 5.7.

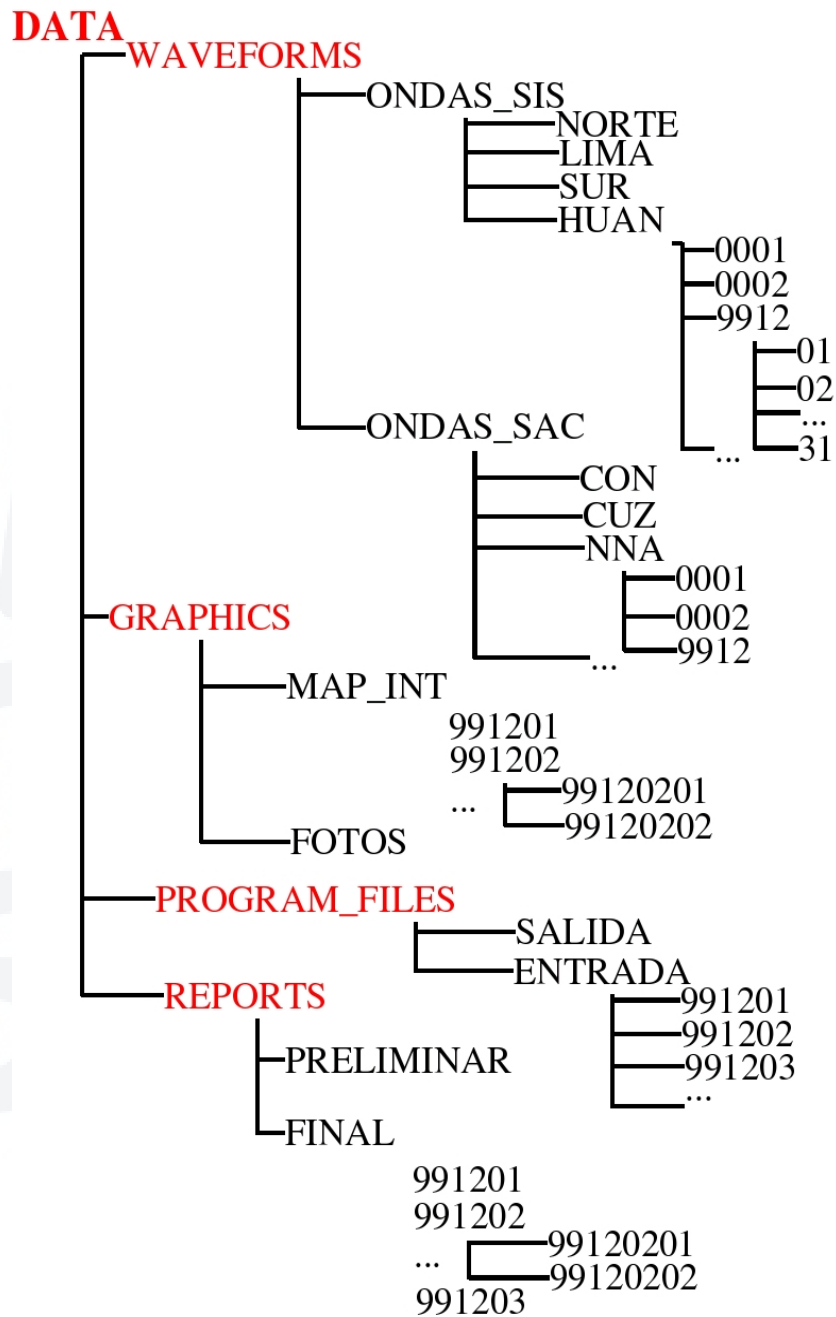


Figura 5.7. Diagrama que muestra el esquema con los nombres de los directorios que contiene la información LOB. Donde se presentan los puntos suspensivos, se indica que existe más información similar a la que corresponde a su nivel correspondiente.

El directorio ARCHIVO_DE_ALGORITMOS contiene los directorios ENTRADA y SALIDA que almacenan los archivos de entrada a los programas de procesamiento sísmico y de salida respectivamente. Cada uno de estos directorios están divididos en un conjunto de directorios cuyos nombres representan la fecha correspondiente a esta información (ver Figura 5.7).

Por ultimo, el directorio INFORMES contiene estos los archivos LOB con los Informes Sísmicos, este directorio esta formado por dos directorios llamados PRELIMINAR Y FINAL y divididos en directorios nombrados por la fecha y a su vez, estos son separados en directorios con nombres que representan al sismo al cual corresponde la información (ver Figura 5.7).

Conociendo el esquema de directorios se puede generar pequeños programas que permitan obtener un *Archivo de Datos*, que con la ayuda del SQL*Loader y un *archivo de control* pueden ser ingresados a las diferentes tablas de la Base de Datos Sísmicos.

5.7.2.- Archivos de Datos generados por programas

El programa denominado *datafile.exe* se encargan de leer específicamente los nombres de los archivos referidos a las formas de onda los cuales representan un volumen de datos muy grande y el cual seria muy complicado ingresar a la base de datos uno por uno. Este programa crear la estructura del *archivo de datos* siguiendo los ejemplos vistos en este capítulo anteriormente (*Ejemplo1* y *Ejemplo2*).

La presentación en pantalla del programa *datafile.exe* que desarrolla *Archivos de Datos* para el ingreso de los archivos LOB como las Formas de Onda tanto de estaciones de periodo corto como de banda ancha a sus tablas respectivas (ver Figura 5.8). En esta Figura se puede distinguir primeramente que el programa pide al DBA ingresar el nombre de la *tabla* donde desea que la información se almacene, así como

también el *Codigo Nacional* de la Sub – Red o de la estación de banda ancha, dependiendo de que archivos quiera almacenar y por último el *año* y el *mes*. Una vez terminado este ingreso, el programa realiza sus operaciones internamente y muestra como resultados el *DIRECTORY* y el nombre del *Archivo de Datos*. Con el *DIRECTORY* y el *Archivo de Datos* se puede generar el *archivo de control* que permitirá realizar el ingreso en bloque de esta información con el SQL*Loader. La Figura 5.9 muestra el contenido de los *archivos de datos* correspondiente a la información del mes de enero del año 2000 que se ingresará en las tablas *Ondas_sis* y *Ondas_sac* de la Base de datos Sísmicos.

5.7.3.- Los Archivos de Control.

Para los dos casos de *archivos de datos* anteriores, los *archivos de control* tienen el siguiente contenido:

- *Archivo de Control para los archivos de las Estaciones de Banda Ancha*

```
LOAD DATA
INFILE con0001.dat
INTO TABLE ondas_sac
FIELDS TERMINATED BY ','
  (ondassacid CHAR(15),
   sismoid CHAR(15),
   codigo_nac CHAR(20),
   fname_1 FILLER CHAR(20),
   fname_2 FILLER CHAR(20),
   fname_3 FILLER CHAR(20),
   eventosacz BFILE(CONSTANT "../data/formas_de_onda/ondas_sac/con/0001", fname_1),
   eventosacz BFILE(CONSTANT "../data/formas_de_onda/ondas_sac/con/0001", fname_2),
   eventosacz BFILE(CONSTANT "../data/formas_de_onda/ondas_sac/con/0001", fname_3))
```

- *Archivo de Control para los archivos de las Estaciones de Periodo Corto*

```
LOAD DATA
INFILE huan0001.dat
INTO TABLE ondas_sis
FIELDS TERMINATED BY ','
  (ondassisid CHAR(15),
   sismoid CHAR(15),
   sub_red CHAR(20),
   fname_1 FILLER CHAR(20),
```

```

fname_2    FILLER CHAR(20),
fname_3    FILLER CHAR(20),
dname_1    FILLER CHAR(100),
eventosis  BFILE(dname, fname_1),
eventosis  BFILE(dname, fname_2),
eventosis  BFILE(dname, fname_3))

```

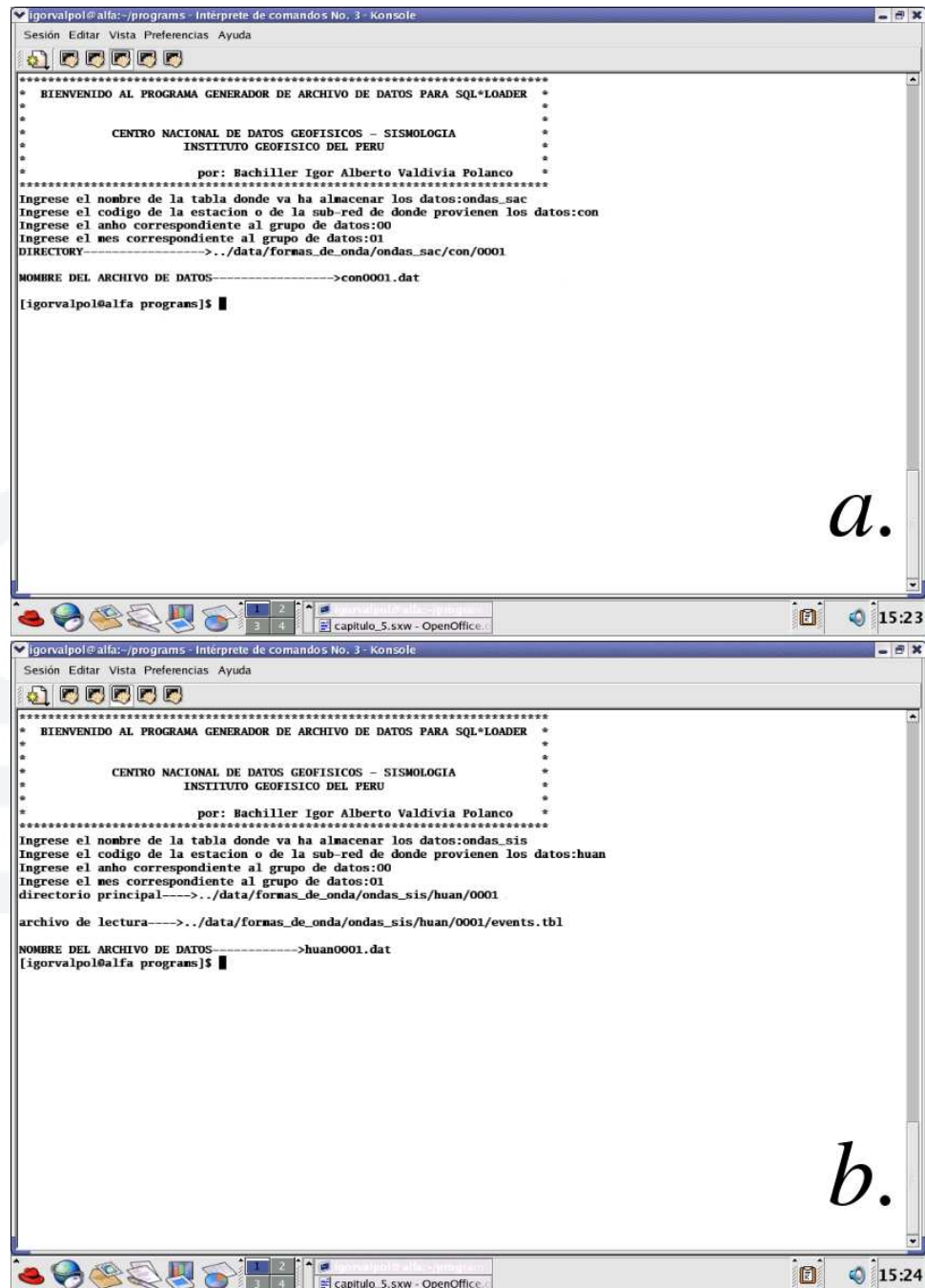


Figura 5.8. Pantalla que muestra la presentación del programa y la ejecución para los archivos de las estaciones de periodo corto(a) y de banda ancha(b).

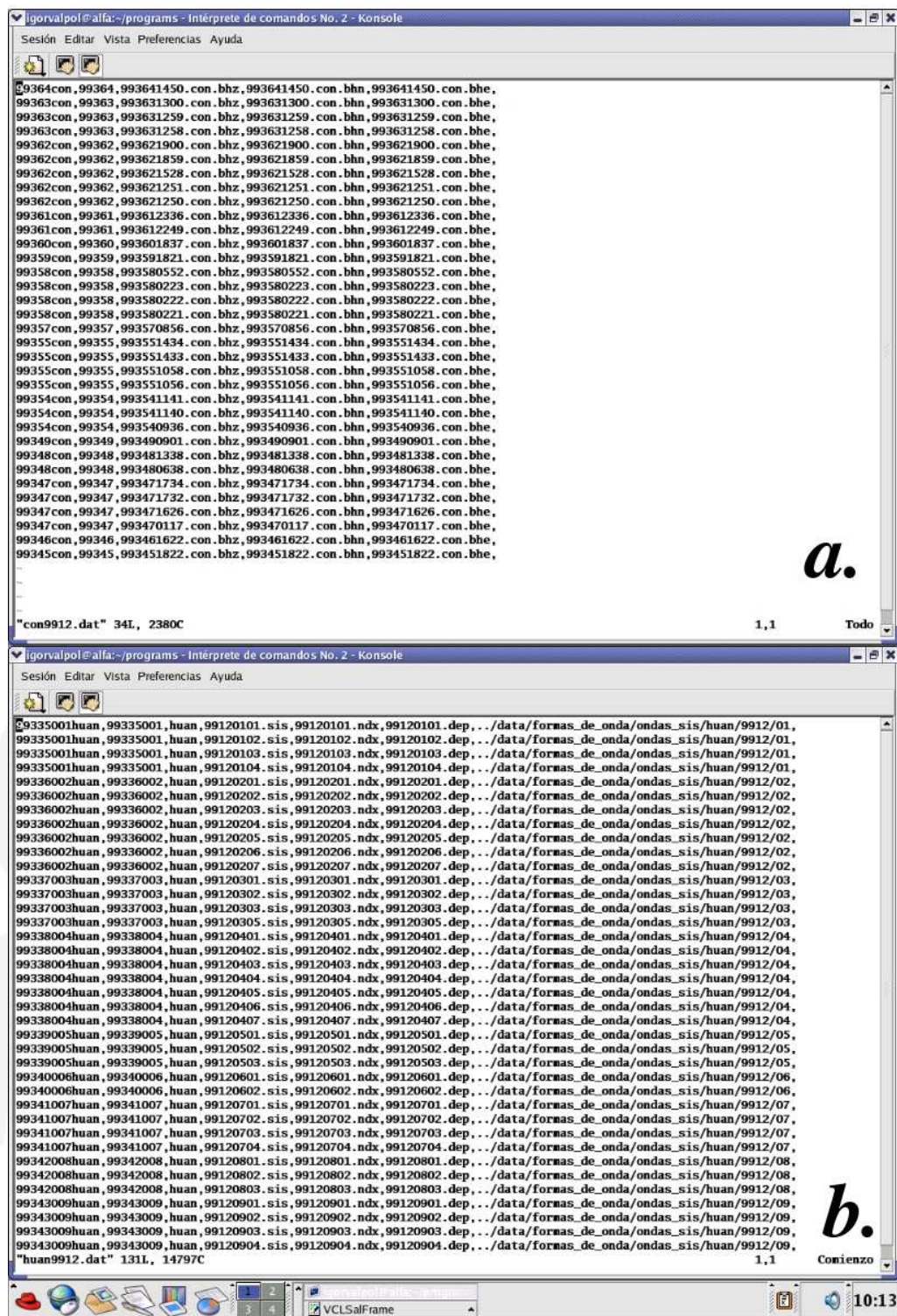


Figura 5.9. Contenido de los Archivos de Datos generados por el programa datafile.exe para el ingreso de archivos de las estaciones de a) banda ancha y b) periodo corto con el SQL*Loader.

CAPITULO 6

CONSULTA A LA INFORMACION ALMACENADA EN LA BASE DE DATOS SISMICOS

6.1.- Introducción

En este capítulo se va a tratar todo lo que respecta a la consulta de la información sísmica que ha sido almacenada dentro de la Base de Datos Sísmicos, tanto para la información de tipo estructurado (datos numéricos, fechas, caracteres o datos alfanuméricos) como para la de tipo no — estructurado (archivos binarios), la cual va a ser realizada desde una computadora exterior empleando la Internet como medio de conexión entre dicha computadora y la computadora que contiene la Base de Datos Sísmicos. Para realizar dicha labor, se va a emplear una serie de programas ubicados dentro del servidor y los cuales son conocidos como Interfaz Común de Compuertas (Common Gateway- Interface, CGI).

6.2.- Interfaz Común de Compuertas. CGI

Los programas CGI constituyen una característica extremadamente poderosa de interacción entre visualizadores y servidores de Web que pueden cambiar por completo la idea sobre las presentaciones en Web. Los programas CGI le permiten al cliente consultor interactuar con las paginas web; es decir, buscar información en cierta base de datos, opinar sobre lo que se ha escrito o seleccionar varios componentes de formularios y recibir a cambio una respuesta personalizada (Lemay, 1995). Cuando se llena un formulario o se realiza una búsqueda a través de una ventana de diálogo en Web, ya se está utilizando los programas CGI sin que esto se de a conocer al cliente o consultor, ya que estos programas trabajan dentro del servidor y solo muestran los resultados de su trabajo. El Administrador de la información que se desea mostrar es el encargado de crear todas las partes del programa: la parte que ve el usuario, el manejo de los datos

que se ingresan y el formato de los datos que se muestran.

De manera sencilla, un programa CGI es aquel programa que se ejecuta en un servidor de Web que se pone en funcionamiento por una orden de entrada proveniente del visualizador. El programa CGI suele ser un vínculo entre el servidor y algún otro programa que se ejecuta dentro del sistema; por ejemplo, una base de datos.

Los programas CGI pueden ser de diferente naturaleza, dependiendo de los que pueda soportar el servidor web donde se ejecuta; pueden ser programas compilados o archivos por lotes o cualquier otra entidad ejecutable. Los programas CGI son utilizados por los servidores Web CERN y NCSA en Unix para permitir la interacción entre servidores y programas; Unix fue el primero en aplicarlas, por lo que es tal sistema operativo el que establece el estándar.

6.2.1.- Lenguaje de Programación del CGI

Dentro de los lenguajes de programación, el lenguaje C es el de mayor difusión y uso en las plataformas de servidor. El "C" como mejor se le conoce, resulta ser uno de los más eficientes lenguajes de programación para desarrollar programas CGI. No obstante, cualquier lenguaje de programación es útil para realizar este tipo de programas. Por tal razón, todos los programas CGI que se han realizado para efectuar la consulta de la información sísmica almacenada en la Base de Datos Sísmicos, van a estar hechas en el Lenguaje de programación C el cual va a estar compilado empleando el compilador que proporciona el servidor Unix donde radica la Base de Datos Sísmicos.

6.2.2.- Mecanismo de Trabajo del CGI

La Interfaz Común de Compuertas (Common Gateway Interface) ó CGI, surgió como una opción importante para la elaboración de documentos de Internet completamente dinámicos empleando programas externos instalados dentro del sistema en el servidor de Internet. En tal sentido la función del programa CGI es tomar

la Información de consulta, que es ingresada por un usuario empleando un documento HTML elaboran la sintaxis de consulta dentro de la base de datos y de obtener los resultados de la consulta para mostrarlos al usuario el resultado en otro documento HTML que el mismo programa genera.

A continuación se señala paso a paso el mecanismo de los CGI's:

- 1.Un URL establece una ruta hacia un programa CGI de la misma manera como lo hace con cualquier documento localizado dentro del servidor o fuera de él. A su vez, el visualizador solicita ese URL de un servidor de la misma manera como lo haría con cualquier documento.
- 2.El servidor recibe la respuesta, advierte que el URL establece una ruta hacia un CGI (basándose en la localización del archivo o en su extensión eso depende del servidor) y ejecuta el CGI referido.
- 3.El CGI realiza cierta acción con base en los datos de entrada, si hay alguno, proveniente del visualizador. Esta acción puede incluir una búsqueda en una base de datos, calcular un valor o tan solo llamar a otro programa residente en el sistema.
- 4.Los resultados de las acciones realizadas por el CGI aparecen de tal manera que son interpretados por el servidor Web.
- 5.El servidor Web recibe el resultado del CGI y lo pasa al visualizador , el cual lo forma y lo despliega para el lector.

En la figura 6.1 se muestra gráficamente un esquema de cómo trabajan los programas CGI dentro del Servidor (Boutell, 1996).

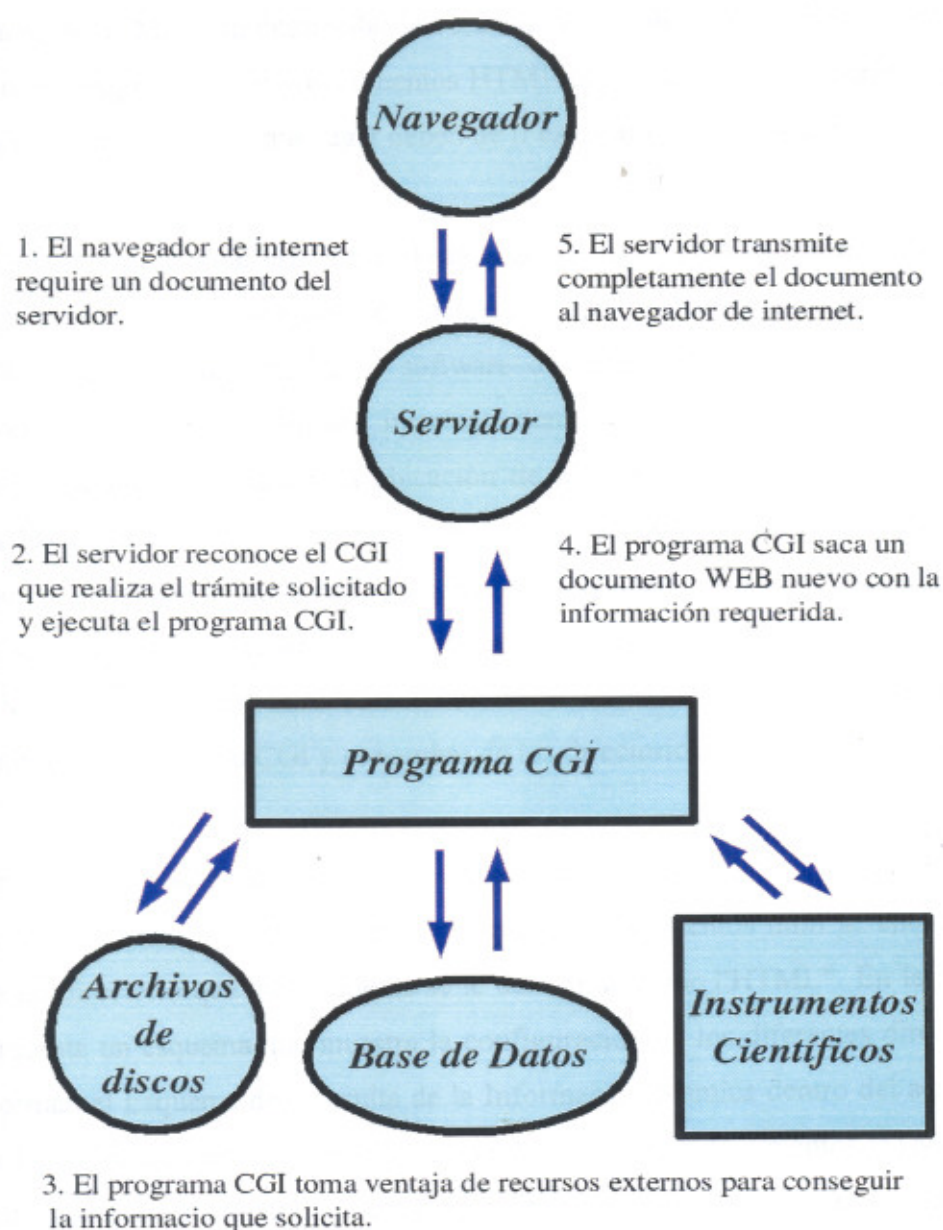


Figura 6.1. Esquema que muestra el mecanismo de los CGI.

6.3.- Estructura del Esquema de Consulta de la Información Sísmica

Tal como se mencionó anteriormente, los programas CGI requieren integrarse con documentos HTML para desarrollar la consulta de la información. Por lo tanto, es necesario tener programas CGI y documentos HTML dentro del servidor. Sin embargo, se debe de conocer como y en que lugar deben de ir estos archivos y programas.

El servidor en el cual esta instalada la Base de datos Oracle, además de servir como lugar donde reside dicha Base de Datos, funciona en forma paralela como un servidor Web para lo cual emplea el software de configuración Web denominado Apache versión 2.18. Este software instalado dentro del servidor Principal, es el encargado de definir el nombre y la ubicación de los directorios donde van ha estar instalados los programas CGI y los archivos HTML. El Apache tiene un archivo de configuración (*conf*) denominado *httpd.conf* escrito en lenguaje SHELL y es allí donde se hacen las modificaciones necesarias y se definen los parámetros que van ha permitir al servidor hacer labores web; tales como: nombre del servidor, nombre el directorio HTML, nombre el directorio CGI y el nombre de los directorios con acceso restringido.

En general, los programas CGI son instalados dentro del Servidor en un directorio denominado "CGI—BIN", mientras que los documentos html se encuentran dentro de un directorio que normalmente se le denomina como "HTML". En la Figura 7.2, se presenta un esquema que muestra la configuración de los diferentes directorios que conforman el Esquema de Consulta de la Información Sísmica dentro del servidor. En dicha figura se muestra que tanto los programas CGI, los documentos HTML y la base de datos se manejan dentro del servidor "Clima" el cual es a la vez un servidor de Base de Datos y un servidor Web. Por lo tanto, los programas CGI van ha poder interactuar con la base de datos y toda la información sísmica almacenada dentro de ella prácticamente de forma directa, respetando las condiciones de seguridad que implanta la base de datos Oracle.

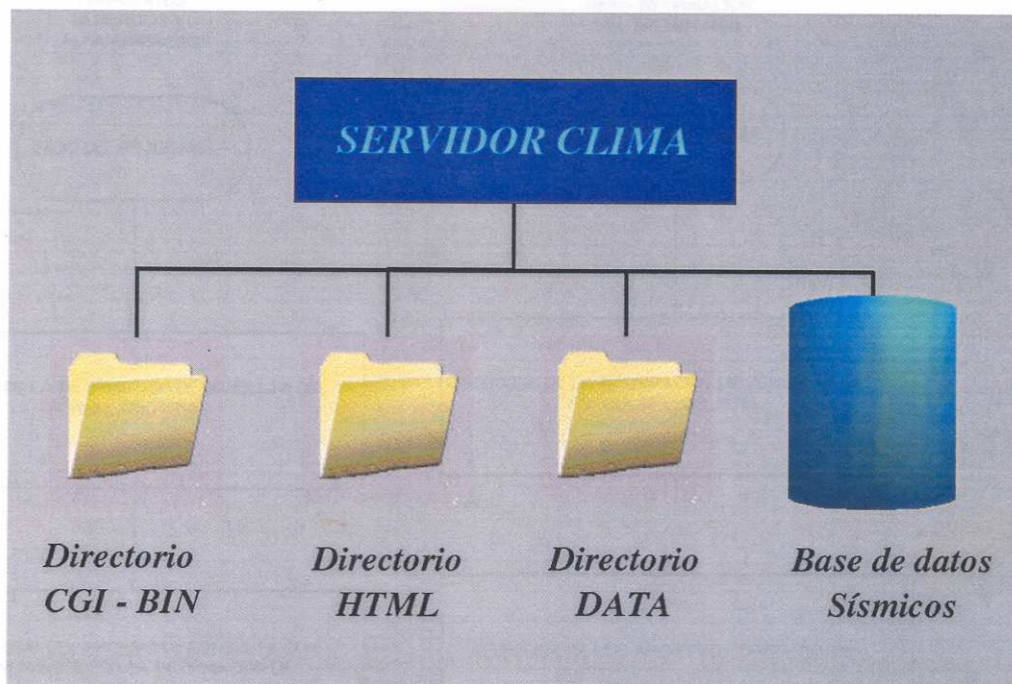


Figura 6.2. Esquema del servidor "Clima". Este esquema muestra la configuración de los directorios de la Estructura de Consulta de la Información Sísmica y la Base de Datos Sísmicos.

6.3.1.- Estructura de los programas CGI dentro del Esquema de Consulta

Los programas CGI empleados para la consulta de información pueden ser básicamente de dos tipos: los que realizan la consulta de la información sísmica de tipo estructurado y otros para información de tipo no – estructurado y por tal motivo su estructura es diferente, tal como se muestra en el esquema de al Figura 6.3.

DIAGRAM DEL CGI
PARA DATOS
NUMERICOS Y

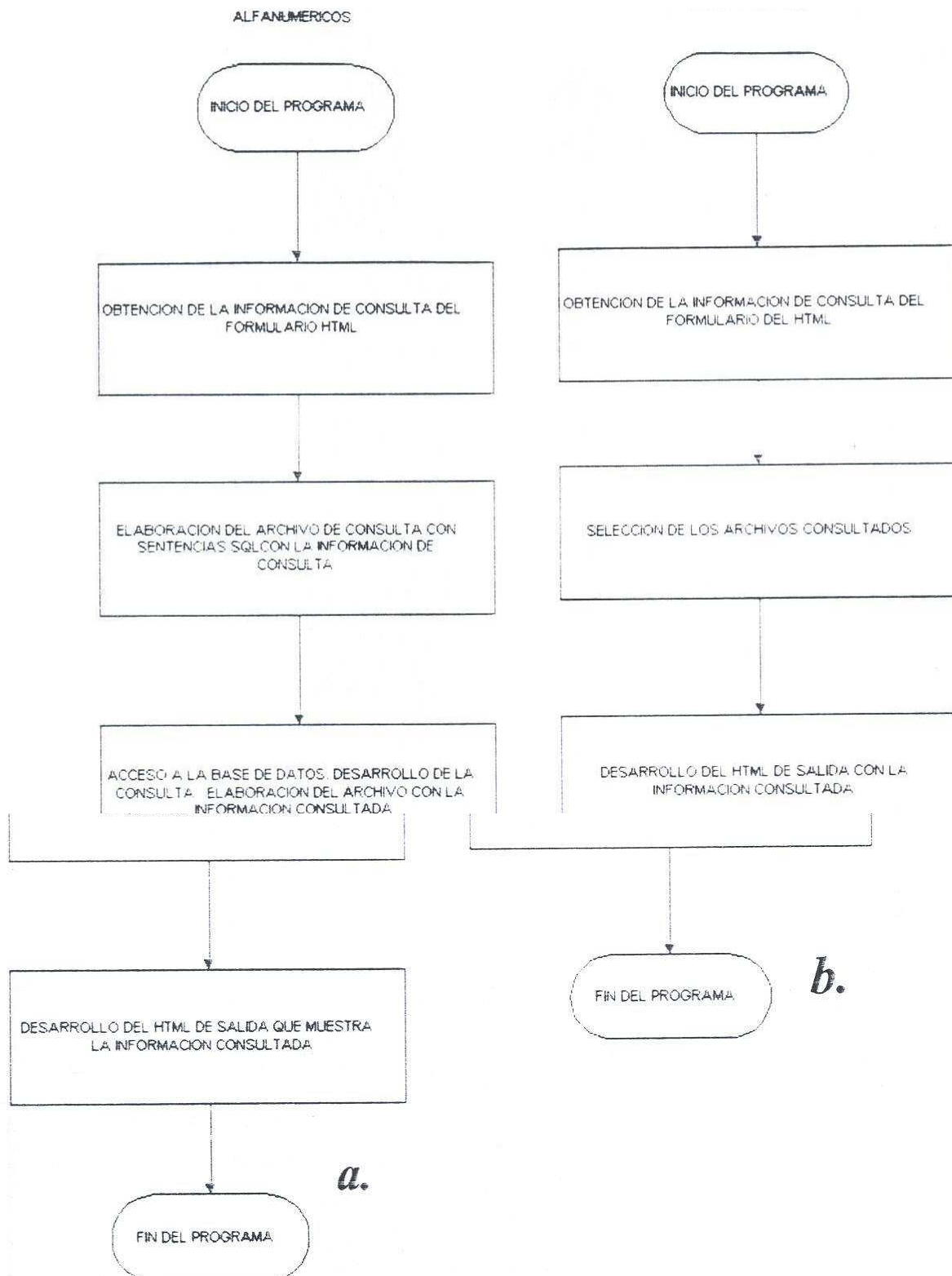


Figura 6.3. Diagrama de flujo de los programas CGI empleados en este estudio.
a) Información estructurada; b) Información no – estructurada.

a) Programas CGI para los datos Estructurados

Estos programas realizan la consulta en forma directa a los objetos que almacenan la información estructurada de la Base de Datos 'Sísmicos'; es decir, a las tablas definida previamente y que contienen únicamente este tipo de información (números, fechas, caracteres, etc.). Para realizar la consulta se emplea las herramientas de Oracle "SQLPLUS", la misma que utiliza los comandos del lenguaje de bases de datos denominado SQL. El código de estos programas puede verse en el Anexo A.

1.La primera parte. Esta parte del código del programa es la que se encarga de recepcionar los datos ingresados por el usuario desde un formulario html y de identificar cada uno de ellos con un nombre y valor. Para esto se emplea la función PARSEFORM definida por el programador y que se encarga básicamente de crear dos arreglos (arrays) de caracteres denominados *names* y *values* que contienen los nombres y valores de las variables.

2.La segunda parte. Esta parte del código del programa es la que se encarga de elaborar un archivo cuya extensión es SQL (*.sql) y cuyo contenido va a ser la consulta a la base de datos en lenguaje SQL empleando los datos ingresados por el usuario. Para esto, el programa hace uso de la función del lenguaje C denominada FOPEN con la cual crea un fichero con comandos que almacenan la consulta escrita empleando sentencias SQL y la información proveniente del formulario del HTML definido en variables, tal como se ve a continuación:


```

out = fopen ("queryl471.sql", "w+");

fprintf (out, "TIITLE LEFT ***** - \n" ) ;

fprintf (out, " ***** CATALOGO SISMICO DEL PERU 1471 -
1982 --- CENTRO NACIONAL DE DATOS GEOFISICOS *****'-\n");

fprintf (out, " ` ***** '\n");

fprintf (out, "set linesize 500\n");

fprintf (out, "set pagesize 10000; \n");

fprintf (out, "select      sismoid, fuente, to_char (fecha,
'YYYY/MM/DD' ), hora,      tqf,  longitud, m,
profundidad, egf, fue, mb, ms, mw, m_sismico, n_est,
reg_flin,      idata,      int_max,      aut_int,      efectos      from
catalogo_1471_1982 \n");

fprintf (out, "where longitud between %s and %s and\n",
values [lon_izqIndex]-, values [lon_derIndex]);

fprintf(out,"longitud      between      %s      and      %s;      \n",
values[lon_izqIndex], values{lon_derIndex});

fprintf (out, "fecha >=  '%s' and fecha <=  '%s';  \n",
values [fec_iniIndex], values [fec_finIndex]);

fprintf(out,"profundidad      between      %s      and      s';      \n",
values[pro_menIndex], values[pro_mayIndex]);

fprintf(out, "exit\n")

fclose (out);

```

Donde *set linesize* y *set pagesize* son parámetros que determinan el ancho y el largo del archivo de salida con la información consultada. Además, se aprecia la sentencia SELECT DE SQL que realiza la selección de los datos.

3.La tercera parte. Esta parte del código del programa es la que se encarga de invocar el SQLPLUS y de ejecutar la consulta de la información, generar el archivo con la información que resulta de la consulta hecha y posteriormente mostrarla. Para realizar esta tarea el programa emplea la función del lenguaje C *SYSTEM*, la que tiene por función ejecutar comandos del sistema Unix y ejecutar programas

instalados en el servidor Unix “Clima”. La forma como se invoca el SQL* Plus empleando esta función es de la siguiente forma.

```
sqlplus -s dba_cndg/password @query1471.sql" );
```

b) Programas CGI para los datos no - Estructurados

Estos programas no hacen la consulta a la base de datos empleando sentencias SQL como en el caso anterior, si no que esta se hace directamente a la ubicación donde radican estos archivos dentro del, servidor y- el directorio asignado a almacenar todos estos archivos. Estos programas constan de tres partes principalmente (Figura 6.3b):

1. *La primera parte.* Al igual que para los programas anteriormente descritos, se encarga de recepcionar los datos ingresados por el usuario desde un formulario html y de identificar cada uno de ellos con un nombre y valor, utilizando la función PARSEFORM.

2. *La segunda parte.* Esta parte del código del programa es la que se encarga de realizar la ubicación del archivo o los archivos que se desea consultar empleando como parámetros de búsqueda la información que ingresa el usuario en la primera parte. Esta parte del programa se apoya en la configuración de los directorios que contienen esta información.

3. *La tercera parte.* Esta parte del código del programa es la que se encarga de construir el archivo html que se va a mostrar al usuario y que contiene la información que resultó de la consulta. La página de salida de la consulta debe ser creada empleando comandos HTML, los cuales empiezan siempre con el siguiente texto:

```
printf ("Content - type: text/html\n\n");
```

y seguidamente elaboran la salida en formato HTML:

```
printf ("html>\n");
printf ("head>\n");
...
Printf ("</body></html>\n");
```

6.3.1.-Estructura de los documentos HTML

La estructura de los documentos HTML esta dada para que la consulta de la información sea para el usuario una labor sencilla y fácil. La página principal de consulta lleva el nombre de *Index.htm* y en ella se puede encontrar los vínculos hacia las páginas *Catalogo.htm*, *estaciones.htm*, *informes.htm* y *wf_formato.htm*. La presentación de esta página en un navegador de Internet se muestra en la figura 6.4. La dirección electrónica de la página es: <http://clima.igp.gob.pe/cndg/index.htm>

En la página *catalogo.htm* se puede encontrar los vínculos hacia los siguientes documentos html: *area_rec.htm*, *area_circ.htm*, *area_rec_hist.htm*, *area_circ_his1.htm*, los cuales contienen los formularios de consulta a la información contenida en las tablas *catalogo_1471_1982* y *catalogo_1983_act* (ver Figura 6.5). Dicha consulta puede ser realizada haciendo una selección de datos para una área rectangular y para un área circular, haciendo uso de programas CGI. El vínculo se realiza desde el documento HTML dentro de las características del formulario tal como se muestra a continuación (en negrita):

```
<form name="form12" method="post" action="catalogo_83._act.cgi">
```

En la página *estaciones.htm* se aprecia una tabla con los parámetros de las estaciones que conforman la RSN y la misma que esta vinculada a una página que contiene el mapa del Perú con las ubicaciones de cada una de las estaciones sísmicas (ver Figura 6.6).

En la página *informes.htm* se aprecia un formulario que muestra una selección de información referente a los informes sísmicos almacenados dentro de la base de datos (Figura 6.7).



Figura 6.4. *Página principal de la Estructura de documentos HTML para la consulta de información de la Base de Datos Sísmicos.*

En la Pagina ef_fomato.htm se encuentra los vínculos a la páginas wf_sac.htm y wf_sis.htm, las cuales realizan la consulta a los archivos que contienen las forma de onda de los sísmos registrados en las estaciones de banda ancha y de periodo corto respectivamente (ver Figura 6.8).

6.3.2.- Documento HTML de salida

Como ya se mencionó anteriormente, estos documentos son generados automáticamente por los programas CGO, cuando se ingresa la información de consulta a través de los formularios contenidos en las páginas web antes mencionadas.

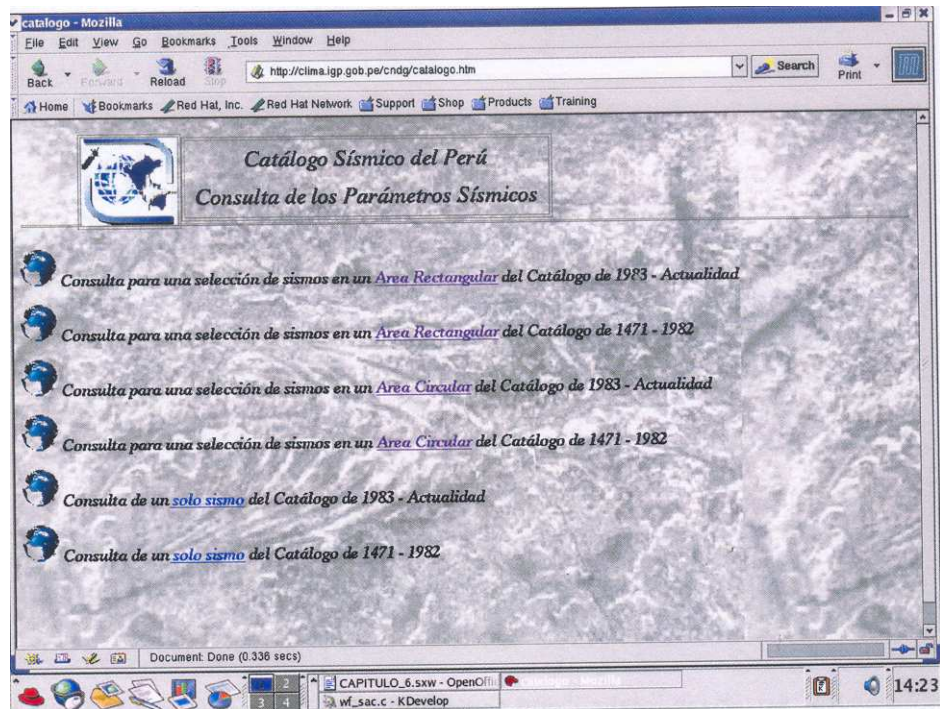


Figura 6.5. Documento HTML “catalogo.htm” visto desde un navegador de Internet. Este vincula a las páginas web que permiten el ingreso de la consulta al Catálogo Sísmico.

ESTACION	CODIGO Nacional	CODIGO Internac.	LAT. °S	LON. °W	ELEV. Msnm	SENSOR	EQUIPO
Atocpunta	ATP	PT14	12.416	74.816	4592	KMI-SS1	Telemetría
Cajamarca	CAJ	PP16	7.13	78.516	2750	GURALP	REFTEK
Camacho	CAM	PT10	12.075	76.969	274	L4C	VR2
Cayma	CAY	PP13	16.376	71.545	2350	L4C	RV301
Conima	CON	PP21	15.467	69.431	3900	KMI-SS1	REFTEK
Cusco	CUS	PP19	13.478	71.958	3858	STRECKEISEN STS-2	REFTEK
Guadalupe	GUA	PT03	13.998	75.79	690	KMI-SS1	Telemetría
Huarmaca	HCA	PT16	5.585	79.486	3300	KMI-SS1	Telemetría
Huancayo	HUA	HUA	12.038	75.322	3330	GURALP	REFTEK
Huaylas	HLS		8.847	77.889	3230		REFTEK
La Yarada	LYAR		18.135	70.568	363		REFTEK
Maychil	MCH	PT20	6.519	79.186	1180	KMI-SS1	Telemetría
Mal Paso	MPA	PT22	6.663	79.443	500	KMI-SS1	Telemetría
Misti	MIS		16.301		71.43	4486	
Montañita	MTA	PT21	6.849	79.136	1200	KMI-SS1	Telemetría
Ñaña	NNA	NNA	11.987	76.839	575	BENNTOS	VR3
Paracas	PAR	PT06	13.829	76.332	150	KMI-SS1	Telemetría

Figura 6.6. Documento HTML “estaciones.htm” visto desde un navegador de Internet. Esta página web muestra la tabla con las estaciones sísmicas de la RSN.

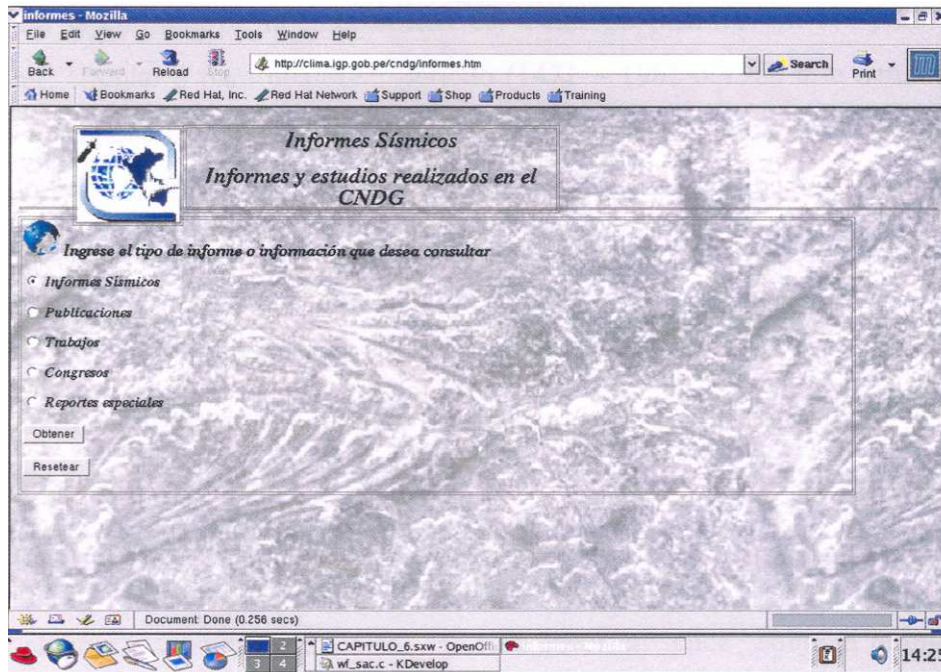


Figura 6.7. Documento HTML “informes .htm” visto desde un navegador de Internet. Esta página web muestra un formulario de consulta para diversos tipos de informes.

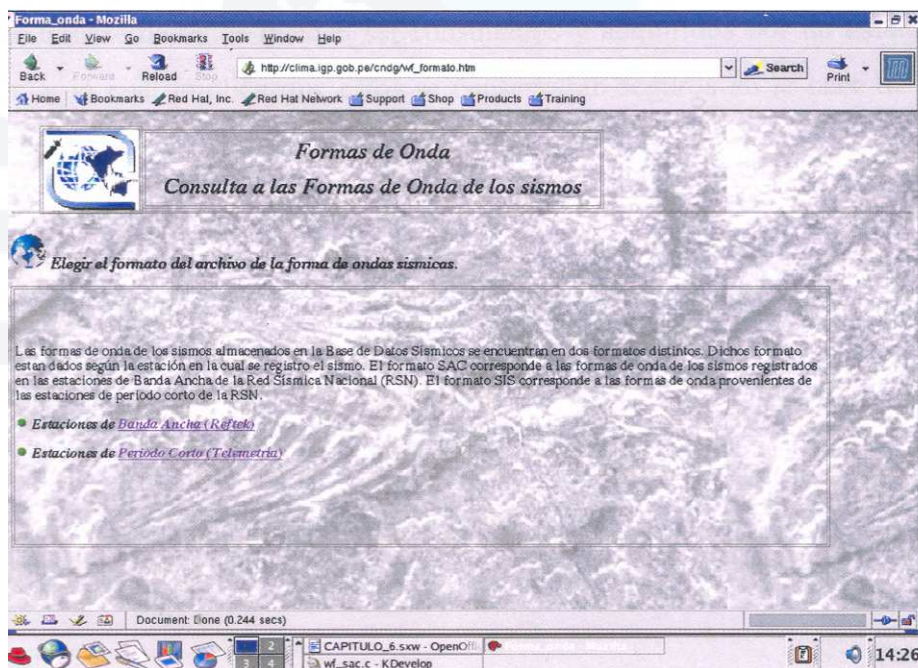


Figura 6.8. Documento HTML “wf_formato.htm” visto desde un navegador de Internet. Esta página web muestra 2 vínculos hacia las paginas que permitirán la consulta de las formas de onda de los sismos almacenados en al Base de Datos Sísmicos.

Para la consulta de la información del Catálogo Sísmico, ya sea para un área rectangular, circular ó para cualquier periodo (1471 — 1982 o 1983 a la actualidad) se empleará el mismo formato. Básicamente, este formato estará dado por un título donde irá el periodo de los datos del catálogo del cual se extrajo la información. Seguidamente se mostrará el nombre de los campos seleccionados de acuerdo a la consulta y los valores obtenidos en ella (ver Figura 6.9).

Para la consulta de los archivos que contienen las formas de onda de los sismos se presenta como título la fecha en la cual ocurrió el sismo y seguidamente, aparecerá la lista con los nombres de todos los archivos que corresponden a la fecha de consulta. Dichos nombres estarán vinculados con la ubicación física de estos archivos y de donde se podrá obtener cada uno de ellos con tan solo hacer un “click” sobre el nombre. Este archivo podrá ser almacenado en cualquier unidad de almacenamiento disponible de la computadora desde donde se ha realiza la consulta (ver Figura 6.10).

Observando como se adquiere dicha información se puede suponer que todos los archivos de la base de datos pueden ser consultados y adquiridos por un usuario de la misma forma, lo cual implica que estos se conservaran seguros en la base de datos y podrán ser administrados con facilidad.

Area_rec - Mozilla

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop http://clima.igpp.gov.pe/cndg/area_rec.htm Search Print

Home Bookmarks Red Hat, Inc. Red Hat Network Support Shop Products Training

Catálogo Sísmico del Perú 1983 - Actualidad

Consulta de los Parámetros Sísmicos

Ingresar los contornos del Rectángulo

-3.0 Latitud Superior -16.0 Latitud Inferior

-68.0 Longitud Derecha -70.0 Longitud Izquierda

Ingresar los parámetros Sísmicos

01-jan-00 Fecha Inicial 31-dec-01 Fecha Final

0 Profundidad Menor 100 Profundidad Mayor

0 Magnitud Menor 7 Magnitud Mayor

Aceptar

Mozilla

File Edit View Go Bookmarks Tools Window Help

Back Forward Reload Stop http://clima.igpp.gov.pe/cgi-bin/cndg/catalogo_83_act.cgi Search Print

Home Bookmarks Red Hat, Inc. Red Hat Network Support Shop Products Training

SISNOID	TIEMPO	ID	CATÁLOGO SÍSMICO DEL PERÚ 1983 - ACTUALIDAD	CENTRO NACIONAL DE DATOS GEOFÍSICOS	MB1	MB2			
			LATITUD E_LAT LONGITUD E_LON PROFUNDIDAD E_PRO						
00001003	2000/01/01	10:23:9.27	-13.309	1.3	-75.916	2.7	83	3.4	3.2
00001003	2000/01/01	13:25:24.02	-13.934	1.9	-76.189	2	42.1	1.6	3.1
00002005	2000/01/02	15:40:18	-14.929	5	-75.836	1.2	61.8	1.1	3.9
00002006	2000/01/02	22:53:22.54	-15.203	5	-75.576	7.5	43.8	2.4	3.5
00003002	2000/01/03	9:20:20.07	-15.021	4.6	-75.518	6.6	22.3	6.7	3.5
00003006	2000/01/03	17:6:52.45	-14.473	3	-75.745	1.1	30	1.2	2.1
00004001	2000/01/04	1:4:53.43	-13.962	5	-76.089	6	53	5	2.1
00004002	2000/01/04	4:34:18.15	-12.053	5.5	-77.572	3.1	52.4	3.4	2.4
00004003	2000/01/04	4:46:18.98	-14.73	4.1	-74.77	3.9	33	4.3	2.8
00004004	2000/01/04	13:55:5.94	-14.806	7	-74.061	8	91.5	1.8	3.4
00004005	2000/01/04	18:3:58.31	-13.417	4	-76.336	1.1	33.8	2	1.9
00006002	2000/01/06	8:32:5.83	-7.367	2.4	-77.556	1.9	94.2	2.9	3.3
00006005	2000/01/06	18:41:10.47	-14.56	3.4	-75.567	3.6	71.5	3.7	3.7
00007003	2000/01/07	15:46:2.85	-14.376	5	-76.119	6	16	3.2	1.8
00009001	2000/01/09	1:30:3.84	-13.613	3.1	-76.154	5.4	58.5	5.9	2.2
00009007	2000/01/09	13:45:13.07	-12.023	4.7	-77.088	5.9	71.3	5.6	2.9
00010003	2000/01/10	4:41:38.93	-5.702	1.1	-77.821	5	22.4	4	3.1
00010004	2000/01/10	7:32:13.07	-5.668	1.1	-77.818	1.1	2.2	1.1	3.1
00010006	2000/01/10	13:43:46.86	-14.863	1	-75.672	1.3	25.9	1.1	2.1
00010007	2000/01/10	14:12:57.05	-13.16	1.4	-75.806	3.3	77.5	4.5	2.6
00011001	2000/01/11	1:43:35.70	-15.42	8.5	-73.772	6.9	50.2	0	3.9
00011003	2000/01/11	6:21:1.91	-13.105	1.1	-76.14	3.3	58.1	3.1	2.6
00011004	2000/01/11	12:46:8.89	-14.491	3	-76.179	4	18.9	1.9	1.8
00011005	2000/01/11	13:41:38.72	-15.731	6.6	-74.946	4.7	31.6	6.1	3.7
00011006	2000/01/11	14:40:39.30	-14.077	2.4	-76.708	3.9	17.3	3.2	2.5
00012004	2000/01/12	13:30:8.76	-14.544	1.4	-76.216	1.8	16.9	1.9	2.5
00013003	2000/01/13	8:28:26.52	-15.883	2.5	-71.476	2	94.3	1.2	3.6
00013004	2000/01/13	9:56:24.40	-14.356	1.3	-76.18	1.7	5.1	0	2.4
00013006	2000/01/13	22:42:9.31	-14.902	3.3	-74.722	2.4	55.4	0	3.3
00014002	2000/01/14	11:37:4.22	-8.232	3.8	-76.536	4.3	18.6	8.7	4.5
00014003	2000/01/14	18:53:49.78	-12.857	2.5	-76.777	4.3	47.6	2.3	2.5
00015001	2000/01/15	2:44:28.33	-11.928	1.2	-76.393	1.1	21.5	1.6	2.5
00015002	2000/01/15	10:17:37.63	-13.588	4.3	-75.856	7	61.4	4.6	3.3
00015006	2000/01/15	18:4:8.59	-13.024	2.6	-73.927	4	41.4	3.2	3.3
00016001	2000/01/16	0:18:52.48	-12.61	3.5	-73.322	2.8	7.9	2.6	3.8
00016005	2000/01/16	9:39:52.17	-12.567	2.9	-76.786	5.1	45.7	3.5	4
00017001	2000/01/17	3:1:19.63	-12.478	1.1	-76.713	2.1	38.8	4.4	1.9
00017002	2000/01/17	4:7:54.85	-14.527	6.1	-75.09	9.8	39.5	3.7	3.1
00017005	2000/01/17	20:32:11.76	-14.669	4.5	-75.081	5.8	49.4	2.8	2.6
00017006	2000/01/17	19:22:37.87	-15.512	8.6	-72.365	6.4	75.6	0	3.6
00017007	2000/01/17	23:37:2.67	-14.743	1.6	-75.877	4.2	45.1	4.7	2.2

Document Done (1.261 sec)

CAPITULO_6.sxw - OpenOffice

wf_sac.c - KDevelop

14:29

Figura 6.9. Consulta a los datos del Catálogo Sísmico del Perú desde un navegador de Internet. En la parte superior se aprecia el ingreso de los datos de consulta en un formulario; mientras que en la parte inferior los resultados de la consulta arrojados por el programa CGI.

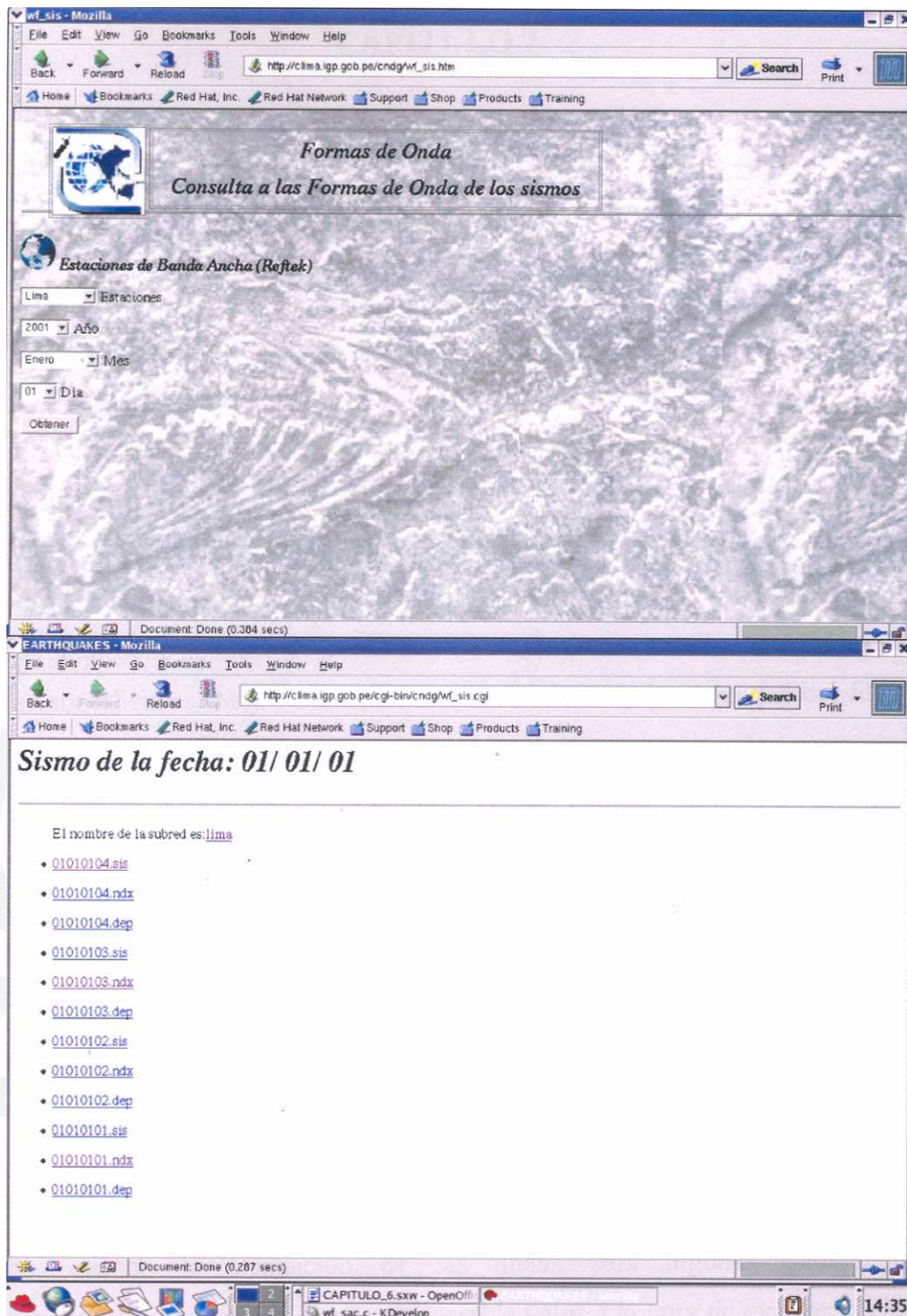


Figura 6.10. Consulta a los archivos con las formas de onda de los sismos contenidos en la base de datos desde un navegador de Internet. En la parte superior el formulario muestra el ingreso de los datos de consulta; mientras que en la parte inferior se muestran los resultados de la consulta arrojados por el programa CGI.

CAPÍTULO 7

CÁLCULO DE LA RELACIÓN DE ATENUACIÓN – INTENSIDAD PARA SISMOS EN PERÚ.

7.1.- Introducción

En el presente capítulo se obtendrá una Relación *Atenuación – Intensidad* para sismos ocurridos en Perú, utilizando la metodología propuesta por Ambraseys (1996), la misma que ha sido aplicada con éxito en diferentes regiones del mundo. Para la aplicación de dicha metodología es necesario disponer de un conjunto de sismos para los cuales se dispone de mapa de intensidad y valores de magnitud obtenidos a partir de las ondas superficiales (M_s). Asimismo, se debe conocer la profundidad del foco para cada uno de los sismos considerados. Esta información, junto con los parámetros de ubicación del sismo (latitud y longitud), va a permitir proponer una relación *Atenuación – Intensidad* que permita estimar el valor teórico de intensidad para un sismo de magnitud “ M_s ” ubicado a una determinada distancia epicentral.

Sin duda, es importante conocer el valor de atenuación de la intensidad para una determinada región, pero este valor se vuelve irrelevante cuando depende de la dirección de propagación de las ondas sísmicas. La atenuación de la energía liberada por un sismo en un medio con características elásticas heterogéneas, es diferente en todas las direcciones; es decir, esta dependerá de la dirección de propagación de las ondas. Sin embargo, cuando la atenuación de las ondas para algunas direcciones es significativamente mayor, esta toma considerable valor a la hora de evaluar la cantidad de energía que puede afectar a una determinada región. Los mapas de Intensidad reflejan la atenuación de la energía, tanto con la distancia como con la dirección mostrando normalmente tendencias circulares cuando la heterogeneidad del medio no difiere mucho, como es el caso de los sismos continentales con origen en la deformación de la corteza, y en otros casos elipsoidales cuando existe una mayor

heterogeneidad, como es el caso de los mapas de intensidad de los sismos con origen en el proceso de subducción. Considerando la forma como se distribuyen en los mapas de intensidad las curvas de isosistas de sismos ocurridos en Perú, se puede definir dos modelos de atenuación de la energía liberada por los sismos: un *modelo elipsoidal*, para aquellos sismos cuyo epicentro se encuentra en la zona de subducción, y un *modelo circular* para los sismos cuyo epicentro se encuentra localizado en el interior del continente y son debidos a deformaciones corticales. Estimar las relaciones Atenuación – Intensidad para estos sismos con diferente origen es el objetivo del presente estudio.

En este capítulo se desarrolla la secuencia seguida para la obtención de las relaciones Atenuación – Intensidad partiendo de los conceptos de la tectónica los cuales permitirán tener idea de los diferentes materiales que constituyen las distintas unidades tectónicas en Perú. Seguidamente, se da una definición de los conceptos necesarios para comprender la intensidad sísmica, las escalas de intensidad y los mapas de intensidad. Posteriormente, se hace una descripción general de los mapas de intensidad de los sismos empleados en este estudio, para luego realizar una descripción y discusión acerca de los modelos de atenuación de la energía para los sismos con origen en el fenómeno de subducción o en el interior del continente.

7.2.- Tectónica y Sismicidad

Existe un conjunto de fenómenos geofísicos que revelan el proceso de interacción entre la placa Sudamericana y la de Nazca, las mismas que según la "Tectónica de Placas" obedecen a un proceso de subducción de una placa oceánica (placa de Nazca) debajo de una continental (placa Sudamericana). Este proceso ha provocado la ocurrencia de sismos de diversos grados de magnitud frente a la línea de costa, tal como se detalló anteriormente en el Capítulo 2 de este estudio. Uno de los resultados de este proceso, a través del tiempo, es la presencia en el borde occidental de Sudamérica de una estructura importante conocida como la Cordillera de los Andes (Tavera y Buforn, 1998).

La evolución de la Cordillera de los Andes generó la presencia de importantes unidades estructurales de las que destacan principalmente la Franja de la Costa, la Cordillera Occidental, la Cordillera Oriental, el Altiplano y la Zona Subandina, división hecha por Audebaud et al.(1973) y Dalmayrac et al (1987). La Figura 7.1 muestra un mapa con las diferentes unidades estructurales en Perú. Las características más relevantes de estas unidades estructurales se describen a continuación:

La Franja Costera, es una zona estrecha de aproximadamente 40 km. de ancho que se extiende de norte a sur y esta constituida en su mayoría por suaves plegamientos volcánicos y rocas sedimentarias del mesozoico. En la zona Sur, esta formada por basamentos de rocas cristalinas fuertemente plegadas y sujetas a deformaciones desde el Precámbrico.

La Cordillera Occidental, se constituye del batolito plutónico andino de mayor volumen y continuo desde Venezuela hasta Tierra del Fuego en Chile. En el Perú se distribuye de norte a sur paralelo a la línea de costa. La parte mas elevada de esta cordillera (4200-4500m) está formada por series del Mesozoico, mas o menos plegados y recubiertos de manera heterogénea por una capa volcánica del Cenozoico. Esta cordillera aumenta notablemente su anchura en la región sur del Perú.

El Altiplano, se encuentra situado entre las Cordilleras Occidental y Oriental. En la región Sur tiene un ancho de 200 Km extendiéndose hacia el norte hasta 9°S aproximadamente, en donde alcanza un ancho de 50 Km y después desaparece. Esta región está formada por una serie de cuencas intramontañosas del Cenozoico que se prolongan hacia el altiplano boliviano. La zona Sur de esta unidad esta invadida por estructuras volcánicas activas de Terciario Superior.

La Cordillera Oriental, en promedio menos elevado que la Cordillera Occidental (3700-4000m) y corresponde principalmente a un extenso anticlinal formado esencialmente por depósitos intrusivos del Precámbrico. En la región Sur, se contornea en dirección EW para luego continuar paralela a las unidades mencionadas anteriormente.

La Zona Subandina, es una zona de anchura variable en donde se amortiguan las estructuras andinas. La zona Subandina se encuentra entre la Cordillera Andina y la Llanura Amazónica; está formada por una cobertura de sedimentos del Mesozoico y del Cenozoico, fuertemente afectada por pliegues de gran longitud de onda.

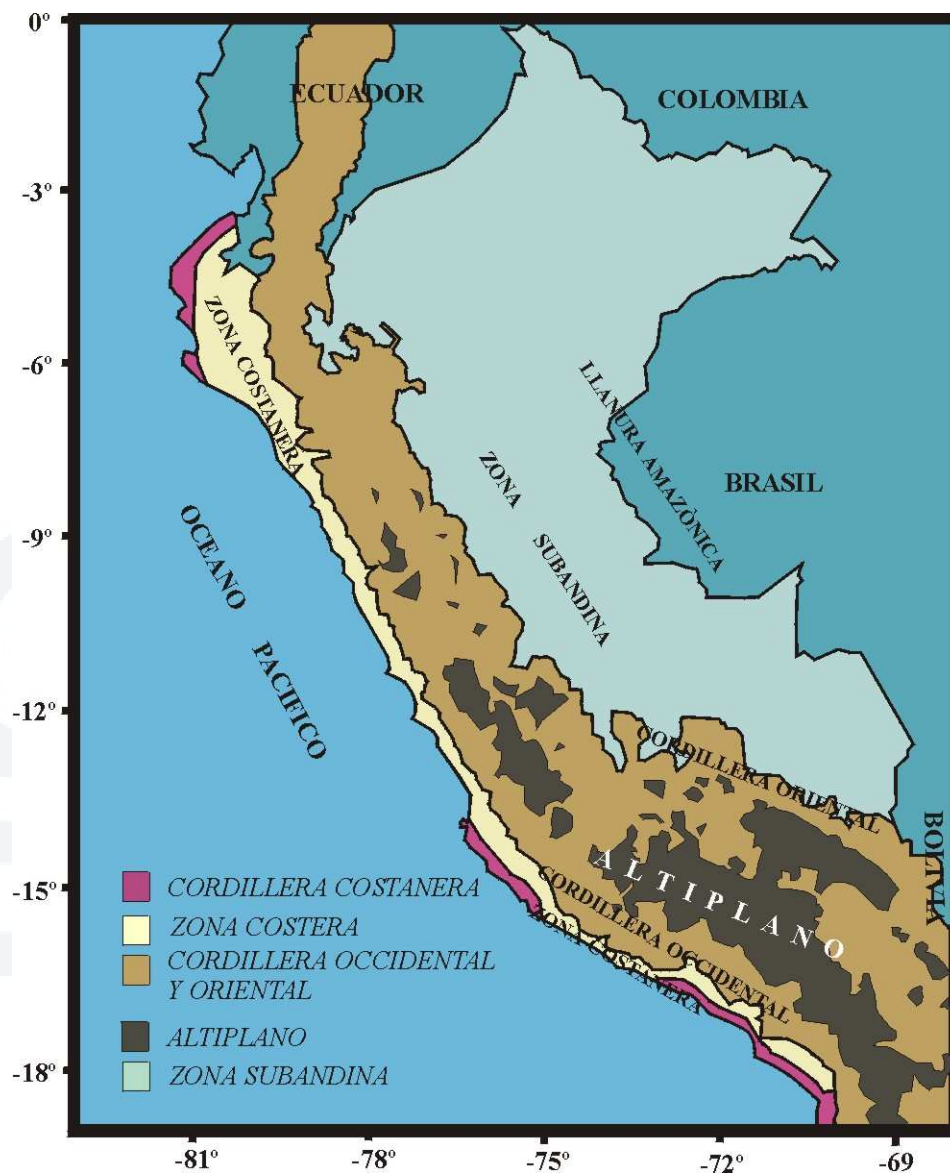


Figura 7.1. Unidades estructurales presentes en el Perú según Audebaud et al, (1973).

Los diferentes materiales que constituyen cada una de las unidades estructurales que conforman la Cordillera de los Andes son el resultado de su evolución continua, la cual también se ve reflejada por la ocurrencia de sismos en su interior. Además, estos diferentes materiales dan una idea de la heterogeneidad del medio por donde se propaga la energía liberada por los diferentes sismos en Perú y cuyas características pueden ser evaluadas y conocidas a partir de las curvas de intensidad.

7.3.- La Intensidad

7.3.1.- Definición

Una forma de describir el tamaño de un sismo es por los efectos que este produce en las personas y en el medio que les rodea; es decir, por los daños ocasionados en edificios y estructuras construidas por el hombre o por las consecuencias sobre el terreno. La intensidad de un sismo en un punto determinado de la superficie de la Tierra, es la fuerza con que se siente en dicho punto. Este concepto no difiere, por tanto, del de intensidad de un campo cualquiera de fuerzas, aunque la forma de medirse sea indirecta (Udias y Mezcuca, 1986). Así mismo, debe entenderse que la intensidad del sismo en un punto cualquiera dependerá de la magnitud del mismo, de la distancia al epicentro y de la constitución litológica del medio afectado por el sismo.

7.3.2.- Escalas de Intensidad Sísmica

Para medir el grado de intensidad de un sismo existen diversas escalas de intensidad establecidas de manera empírica y que son de uso en la actualidad. En general, estas escalas de intensidad sísmica se dividen en grados referidos a los efectos producidos y sentidos por las personas, los mismos que son descritos en términos universales con la finalidad de estandarizar el uso de dichas escalas sísmicas. Los términos que definen los grados de intensidad son:

- Efectos o descripciones de como son sentidos y percibidos los sismos por las personas en su medio ambiente.

- Daños producidos en las construcciones y edificaciones hechas por el hombre, según sus diversos tipos.
- Cambios advertidos en la naturaleza.

El desarrollo de las escalas de intensidad para medir el tamaño de un sismo se realizó en forma progresiva a partir del siglo XIX. Así, las primeras escalas de intensidad se deben a los trabajos de S. de Rossi y F. A. Forel en Italia y Suiza respectivamente y quienes proponen la escala Rossi – Forel dividida en diez grados (I al X) en el año de 1883. Una modificación de esta escala es la propuesta por G. Mercalli en 1902, primero con diez grados y a propuesta de Cancani, de 12 grados (I al XII). Esta escala fue la base para sustentar a las usadas en la actualidad. En América se utiliza la llamada escala de Mercalli Modificada (MM) propuesta por H. Wood y F. Newmann en 1931, la cual posteriormente fue modificada por C. F. Richter en 1956. Para Europa la escala usada esta basada en los trabajos de S. V. Medvedev, W. Sponheuer y V. Karnik en la URSS, y recibe el nombre de MSK, la misma que también tiene doce grados y es equivalente a la de Mercalli Modificada (Udias y Mezcua, 1986). Para Perú, Ocola (1979 y 1988) realizó algunas modificaciones a la escala MSK, con intención de que sea empleada para evaluar las intensidades de los sismos de Perú.

a) Escala de Mercalli Modificada (MM)

La escala de intensidad de Mercalli Modificada fue propuesta por Harry O. Wood y Frank Newman en el año de 1931 y luego por C. F. Richter en 1956, la cual resulta de una modificación hecha a la escala propuesta por G. Mercalli en 1902. Esta escala, al igual que las demás, no se basa en los registros instrumentales del terremoto, sino en la forma como perciben las personas el movimiento y en la evaluación de los daños y efectos que produce este en las estructuras y en el medio ambiente. La nomenclatura de esta escala es expresada en números romanos y los valores de sus grados son aproximadamente proporcionales; es decir, una intensidad de grado IV equivale aproximadamente al doble de una intensidad de grado II.

La escala de Mercalli Modificada es la más difundida en los países americanos y por lo tanto, es la escala con la que se han elaborado la mayoría de mapas de intensidad de los sismos de Perú, tanto los históricos como los recientes. Dado a que esta escala permite evaluar la intensidad de un evento sísmico a partir de la simple descripción de daños, efectos en las personas, en estructuras y principalmente en la naturaleza, es que investigadores han podido deducir mapas de intensidades a partir de los relatos y crónicas hechas por testigos presenciales para los diferentes sismos históricos. Actualmente, los sismos evaluados con esta escala se hacen mediante una evaluación de daños en construcciones, efectos en la naturaleza, además de entrevistas a diferentes pobladores de la región afectada y según la descripción se buscará que grado de intensidad de esta escala coincide. Una descripción de las características de cada grado de la escala de Mercalli Modificada se encuentra en el Anexo B.

b) Escala MSK

En la mayoría de los países de Europa, la escala de intensidad utilizada es la M.S.K propuesta en 1964 por S. V. Medvedev, W. Sponheuer y V. Karnik en colaboración con un grupo de trabajo constituido por la XIII Asamblea General de la U.G.G.I. Esta escala desarrollada en Europa realiza la descripción de sus grados usando los tipos de construcciones típicas de allí, así como los daños propios de estas estructuras. A fin de que esta escala pueda ser aplicada en Perú, Ocola (1979) modificó las descripciones de cada grado con la inclusión de las características de las construcciones propias de Perú. Así mismo, añadió además del valor de grado n los valores de grado $n+$ y $n-$, que simbolizan un valor de intensidad de $n+1/4$ para $n+$ y un valor $n-1/4$ para $n-$.

Al evaluar las intensidades mediante la escala MSK a diferencia de la escala de intensidad MM, se requiere adicionalmente información del tipo de suelo y condiciones geológicas en donde se hace la evaluación, el nivel freático, el material de la construcción afectada, la antigüedad de esta y la calidad de la construcción referente al diseño de la estructura. Toda esta información influye notablemente para dar un valor

final de intensidad. Todos estos términos obligan a que se haga una evaluación *in situ* de la intensidad de forma calificada y adecuadamente. Una descripción general de las características de la escala MSK se presenta en el Anexo B.

c) Comparación entre MM y MSK

La intensidad es un valor cualitativo que corresponde a una descripción hecha por un observador de los efectos producidos por un sismo en las personas y en las construcciones. Considerando lo anterior, es posible que dos observadores que evalúan la intensidad de un determinado lugar, discrepen en sus apreciaciones, y mas aún cuando esta es interpretada con diferentes escalas. Reiter (1990), elaboró un cuadro de equivalencias entre las escalas de intensidades sísmicas más conocidas, tomando como patrón la de Mercalli Modificada (Figura 7.2). En esta comparación, Reiter represento el valor de cada grado por medio de celdas equivalentes al tamaño de los efectos en cada grado de intensidad y para cada escala de intensidad. Según Reiter, entre las escalas MM y MSK existe una equivalencia total para los grados de intensidad que van del grado IV al grado XII al igual que para el grado I. Para los grados II y III, según las celdas, la diferencia sería significativamente variable; es decir, el tamaño de la celda de grado III en la escala MSK es mayor comparado con la celda que representa el grado III en la escala MM en un tercio de grado y el tamaño de la celda del grado II en la escala MSK es menor que la que comprende a la escala MM en la mitad (Figura 3). Estas diferencias muestran que el grado III en MSK, considera características que se describen en la escala II de MM.

MERCALLI MODIFICADA	ROSSI FOREL	JMA	MERCALLI CANCAM SIEBERG	MSK
I	I	I	II	I
II	II		III	II
III	III		IV	III
IV	IV		V	IV
V	V	II	VI	V
VI	VI	III	VII	VI
VII	VII	IV	VIII	VII
VIII	VIII	V	IX	VIII
IX	IX	VI	X	IX
X	X		XI	X
XI		XII	XI	
XII		VII	XII	

Figura 7.2. Escala de intensidades sísmicas y su equivalencia (Reiter, 1990).

7.4.- Análisis de los Mapas de Intensidad.

El análisis hecho a los mapas de intensidad permitirá hacer un reconocimiento visual de la forma que predomine en cada una de las curvas de las isosistas de los sismos. Esto permitirá deducir modelos de atenuación y a partir de estos modelos estimar las relaciones de atenuación necesarias que mejor expresen el comportamiento de la energía liberada por los sismos en el medio que se propaga. Los parámetros hipocentrales de los sismos empleados para este estudio corresponden a un periodo comprendido entre 1940 y 2001, para los cuales se cuenta con 19 sismos con magnitudes “Ms” entre 6.0 y 8.1 y cuyos parámetros hipocentrales se muestran en las tablas 7.1 y 7.2 para los sismos de subducción y los sismos continentales

respectivamente. Así también, en la Figura 7.3 se presenta la ubicación epicentral de los sismos utilizados en este estudio. De este análisis se a excluido a los sismos ocurridos en Ancash (1946), Satipo (1947) y Arequipa (1960). El sismo de 1946 a pesar de ser continental, sus isosistas muestran tendencias elipsoidales que no obedecen al patrón definido por los sismos continentales y que es debido a que su epicentro estuvo ubicado en medio de un valle y con la presencia de la Cordillera Andina por ambos extremos que atenúan la energía liberada por el sismo, por lo que se prefirió dejar de lado en el cálculo. El sismo de 1947 no corresponde a un sismo continentales ni es un sismo de subducción sino a un sismo con origen en la deformación interna de la placa de Nazca por debajo del continente a una profundidad de 100 kms., por lo que se determino no considerarlo en los cálculos. El sismo de 1960 también es excluido del análisis y del cálculo debido a que la distribución de sus isosistas indica que no hubo información suficiente para su elaboración.

Tabla 7.1. *Parámetros hipocentrales de los sismos de subducción de Perú.*

EVENUTO	FECHA DD/MM/AA	LATITUD (°)	LONGITUD (°)	PROF. (KM.)	Ms	REFERENCIA
1	24/05/40	11.22 S	77.79 W	50	8.0	LIMA
2	24/08/42	15.55 S	75.74 W	60	8.1	NAZCA
3	12/12/53	03.88 S	80.43 W	30	7.8	TUMBES
4	17/10/66	10.83 S	78.65 W	37	8.0	LIMA
5	31/05/70	09.27 S	78.84 W	71	7.8	CHIMBOTE
6	09/12/70	04.06 S	80.66 W	28	7.1	TUMBES
7	03/10/74	12.28 S	77.54 W	21	7.8	LIMA
8	18/04/93	11.75 S	76.62 W	94	5.6	LIMA
9	12/11/96	15.99 S	75.67 W	14	6.6	ICA
10	03/04/99	16.61 S	72.82 W	92	6.0	CAMANA
11	23/06/02	16.20 S	73.75 S	38	7.9	AREQUIPA

Tabla 7.2. *Parámetros hipocentrales de los sismos continentales de Perú.*

EVENTO	FECHA DD/MM/AA	LATITUD (°)	LONGITUD (°)	PROF. (KM.)	Ms	REFERENCIA
1	15/01/58	16.50 S	72.00 W	60	7.0	AREQUIPA
2	19/06/68	5.57 S	77.13 W	16.3	6.9	MOYOBAMBA
3	01/10/69	11.75 S	75.15 W	14.5	6.2	HUANCAYO
4	05/04/86	13.41 S	71.79 W	50	5.4	CUSCO
5	29/05/90	06.16 S	77.23 W	24	6.5	MOYOBAMBA
6	04/04/91	05.99 S	77.09 W	19	6.8	MOYOBAMBA
7	16/02/79	16.58 S	72.57 W	53	6.9	AREQUIPA



Figura 7.3. *Ubicación epicentral de los sismos utilizados en este estudio. Los triángulos rojos representan los sismos de subducción, mientras que los triángulos azules representan los sismos continentales.*

7.4.1.- Mapas de Intensidad de sismos de Subducción

Al analizar las formas de las isosistas en los mapas de intensidad de los sismos con origen en el proceso de subducción en Perú, se aprecia que estas se asemejan a elipses circunscritas con el eje mayor orientado en dirección paralela a la línea de costa y eje menor con dirección perpendicular a la misma, tal como se muestra en la Figura 7.4. Este análisis permite deducir un mayor grado de atenuación de las ondas que se propagan en dirección perpendicular a la línea de costa. Esta diferencia es atribuida principalmente a la presencia de la Cordillera de los Andes que se orienta a la dirección de propagación de las ondas y que debido a su masa de gran volumen y con espesores de hasta 70 Km. y anchos del orden de 50 a 250 Km., representa una estructura importante en la atenuación de la energía liberada por los sismos ocurridos en el Perú. De acuerdo a esta característica de las curvas de isosistas se cree conveniente calcular dos relaciones de Atenuación – Intensidad; es decir, una relación para la dirección paralela a la orientación de la Cordillera y otra para la dirección perpendicular a la misma.

7.4.2.- Mapas de Intensidad de sismos Continentales

Para los sismos con origen en deformaciones continentales (Figura 7.5) no ocurre lo mismo que para el caso anterior, ya que las curvas isosistas presentan formas circulares, que sugieren un mismo grado de atenuación para cualquier dirección de propagación de la energía liberada por estos sismos. Esta forma circular puede ser atribuida principalmente a la poca influencia de la Cordillera de los Andes, al valor de magnitud en estos sismos, comparado con los que tienen origen en el proceso de subducción, que es relativamente bajo y en su mayoría son menores a 25 Km (Tabla 7.2).

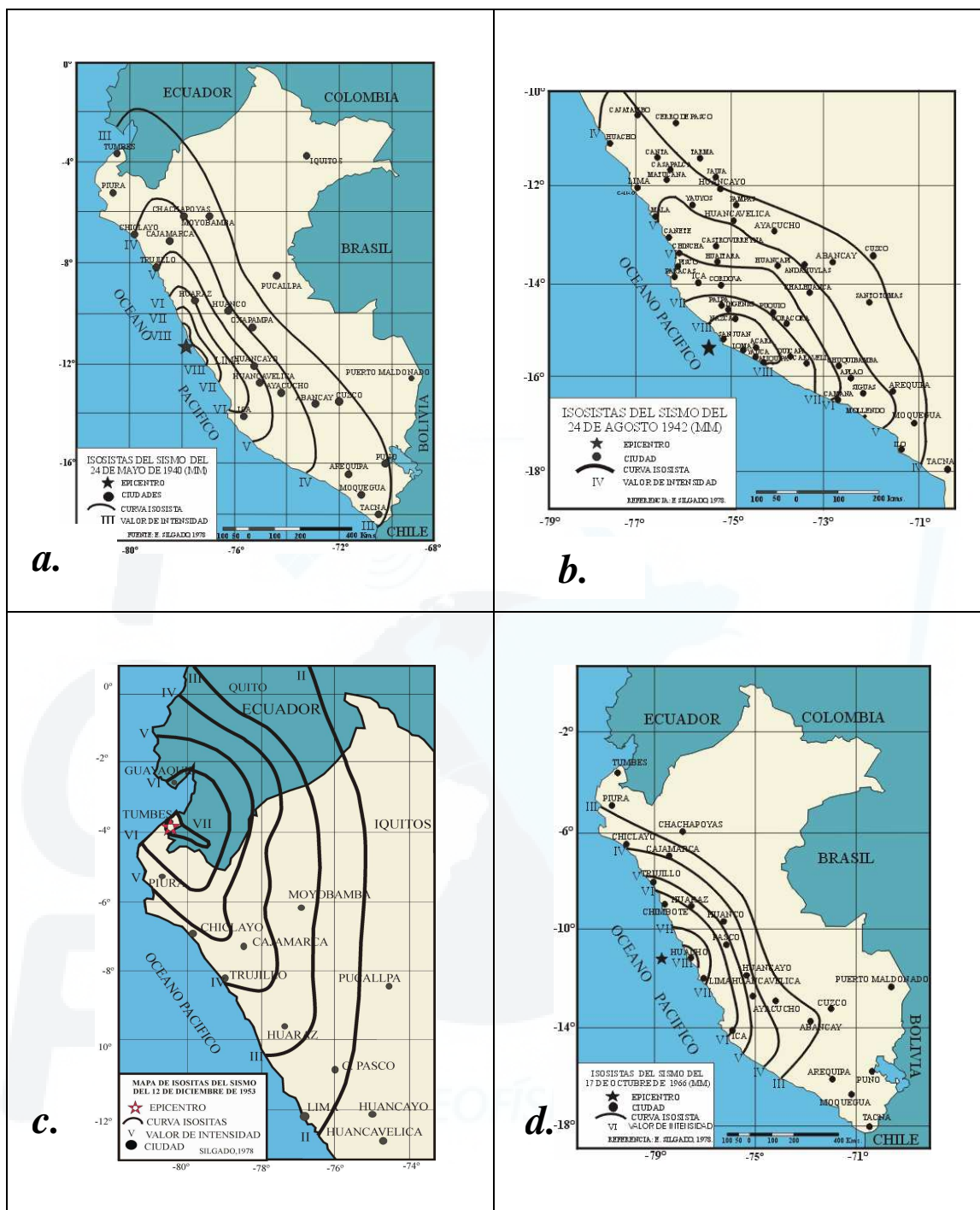


Figura 7.4. Mapa de intensidad de los sismos ocurridos en la zona de subducción de Perú. a) sismo del 24 de mayo de 1940; b) 24 de agosto de 1942; c) 12 de diciembre de 1953; d) 17 de octubre de 1966; e) 31 de mayo de 1970; f) 9 de diciembre de 1970; g) 3 de octubre de 1974; h) 18 de abril de 1993; i) 12 de noviembre de 1996; j) 3 de abril de 1999; y k) 23 de junio del 2001.

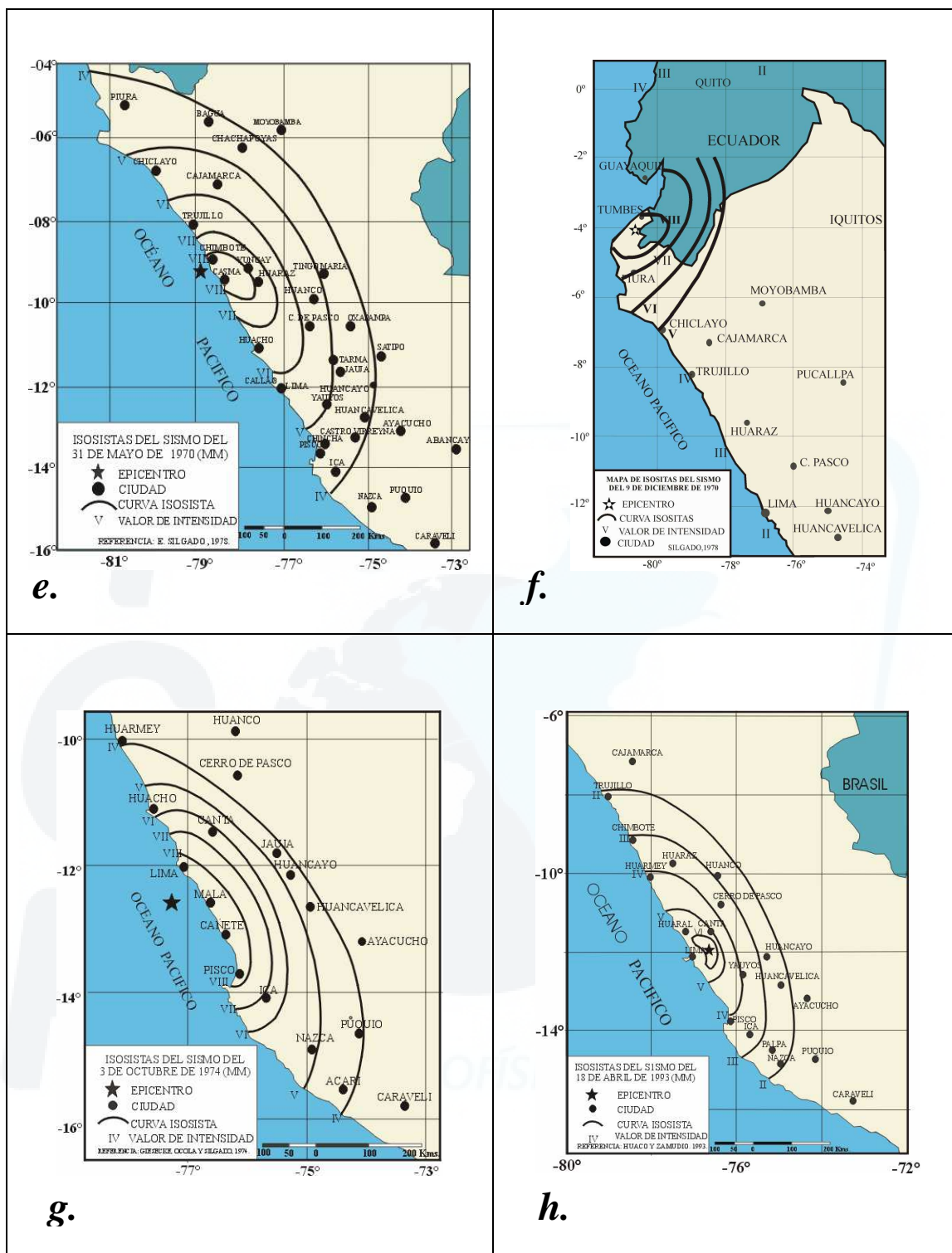


Figura 7.4. Continuación.

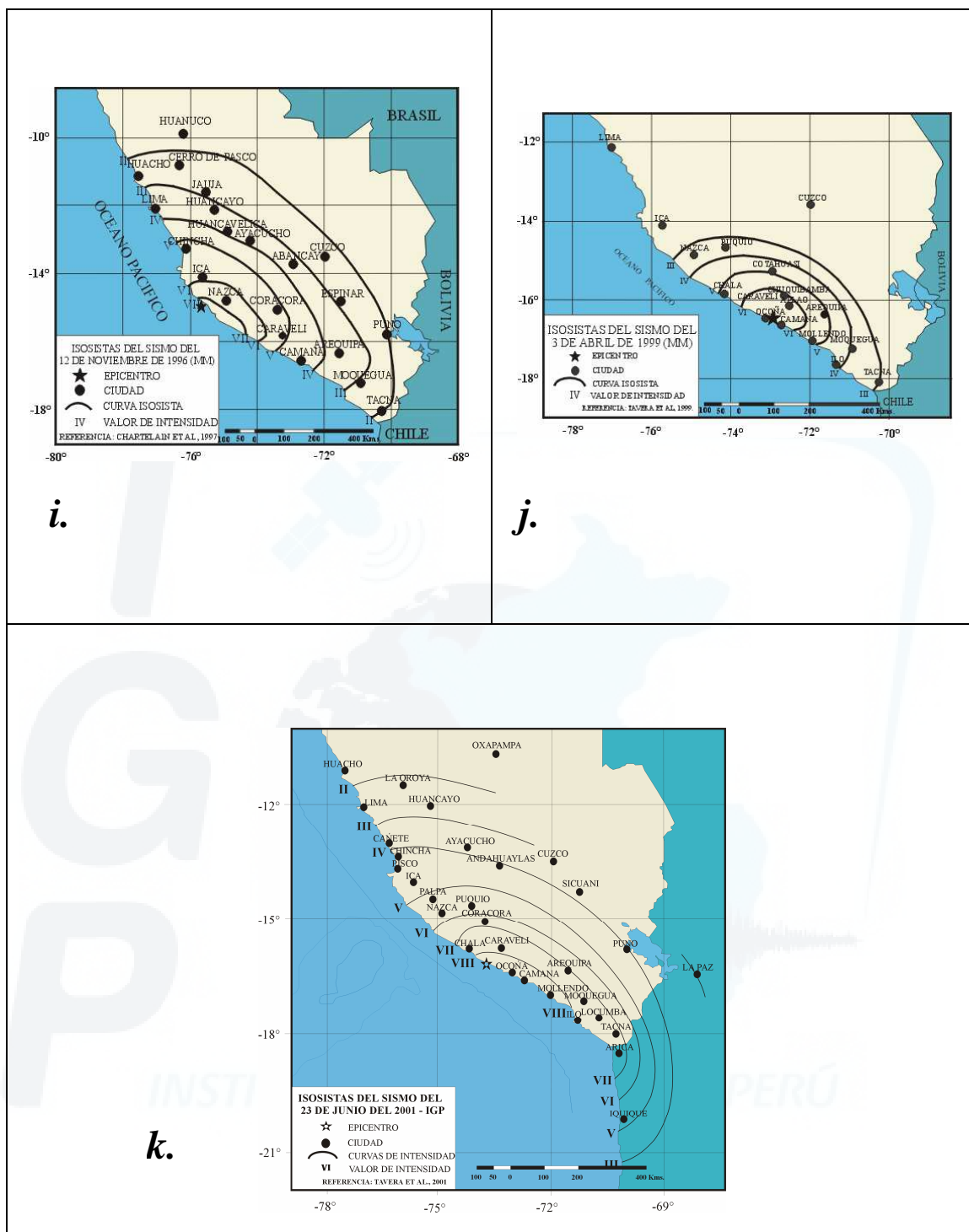


Figura 7.4. Continuación.

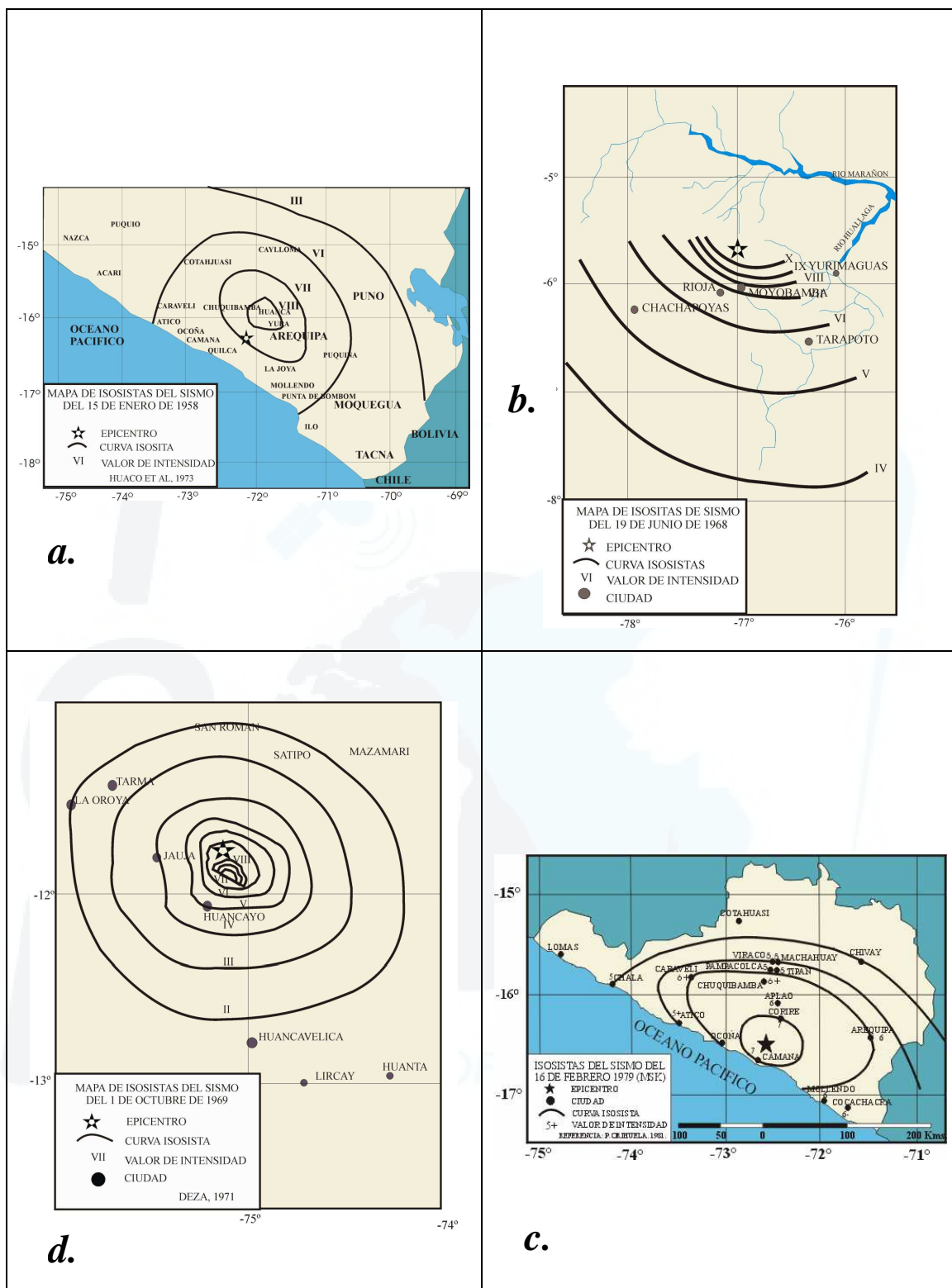


Figura 7.5. Mapas de intensidad de los sismos ocurridos en la zona Subandina de Perú. a) sismo del 15 de enero de 1958; b) 19 de junio de 1968; c) 1 de octubre de 1969; d) 16 de febrero de 1979; e) 5 de abril de 1986; f) 29 de mayo de 1990; y g) 4 de abril de 1991.

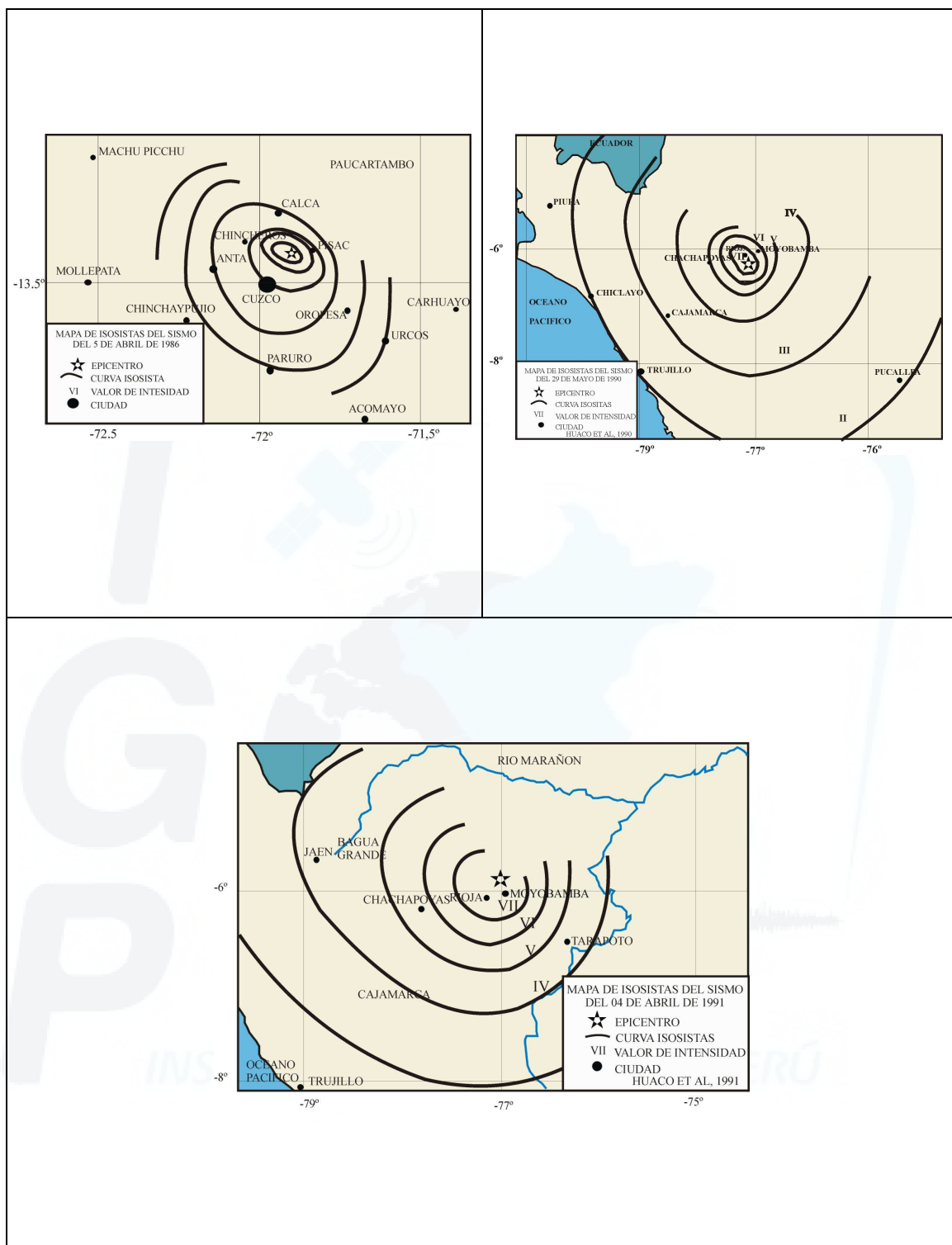


Figura 7.5. Continuación.

7.5.- Modelos de Atenuación y cálculo de las Relaciones de Atenuación - Intensidad

Durante muchos años se ha tenido como objetivo principal, determinar una relación de atenuación de la energía liberada por un sismo con respecto a la distancia epicentral o hipocentral y que esta este asociada a otros parámetros del sismo.

Según Ambraseys (1985), existe una relación de atenuación de la energía liberada por un sismo, en la cual involucra la intensidad que provocada un evento sísmico en superficie y la respectiva distancia hipocentral; en la que también se relaciona a la magnitud obtenida a partir de las ondas superficiales “Ms” calculada para el evento. La relación de atenuación estará definida por;

$$I = B_1 + B_2 (Ms) + B_3(R) + B_4 \text{Log} (R) \quad (7.1)$$

donde B_1 , B_2 , B_3 , B_4 son los coeficientes a ser determinados, Ms es la magnitud del sismo obtenido a partir de las ondas superficiales, R es la distancia focal o hipocentral que corresponde al radio epicentral $D_I = (R_I^2 - h_0^2)^{1/2}$ de la Isosista I expresado en Km y h_0 representa la profundidad focal promedio, obtenida a partir de las profundidades de los sismos empleados en este estudio, la cual también se expresa en Km.

Considerando la teoría propuesta por Ambraseys (1985), y las características de las curvas de las isosistas en los mapas de intensidad de los sismos ocurridos en Perú se ve por conveniente considerar dos modelos de atenuación; el *modelo elipsoidal* y el *modelo circular*. Para los cuales se determinará las relaciones de Atenuación – Intensidad con la intención de que dichos modelos se adapten de la mejor manera a la geometría presentada por las curvas de las isosistas representados en cada sismo.

7.5.1.- Modelo Elipsoidal de Atenuación

Como ya se mencionó en este capítulo, para los sismos con origen en el proceso de subducción de Perú, una sola relación de Atenuación – Intensidad no va a ser la que mejor represente la atenuación de la energía en todas sus direcciones ya que las curvas de isosistas se asemejan más a una elipse que a un círculo. En general, en la dirección paralela a la línea de costa la elongación de las curvas de isosistas es notoriamente mayor comparada con la dirección perpendicular a esta, poniendo en claro la diferencia de atenuación de la energía para esas dos direcciones.

Considerando esta característica, si se toma cada curva de isosista como una elipse (Figura 7.6), el centro de la misma sería el epicentro del sismo (E). El eje mayor de la elipse, AA', estaría dado en dirección paralela a la línea de costa y el eje menor, BB', en dirección perpendicular. Considerando que el límite costero no presenta una dirección única, se tomara como eje paralelo a la costa una línea que pase por el epicentro del sismo. Como es evidente, para la mayoría de los sismos únicamente se tendrá la mitad de la elipse con referencia al eje mayor, debido a que los epicentros de los sismos están en el mar o muy cerca de él. Esto trae como consecuencia la existencia de dos semiejes en la dirección paralela a la costa (EA y EA') y un solo semieje en la dirección perpendicular a la línea de costa (EB'). Considerando lo indicado anteriormente, la metodología seguida en este estudio es como sigue:

- *En dirección paralela a la línea de costa.* Una vez identificado el eje mayor de la elipse, se procede a medir la distancia comprendida entre el epicentro o punto central de la elipse y cada uno de los puntos donde se intercepte cada curva isosista con el eje mayor AA' (distancia *epicentro – Isosista*). En esta dirección se va a tener dos valores, debido a los dos sentidos de propagación, estos se promediarán para obtener al final el valor " D_i " para la curva de la isosista i . Para los casos en que el eje mayor (AA') pase por el mar y no se intercepte con las curvas de las Isosistas, se prolonga el punto máximo de elongación de la misma

de forma perpendicular al eje mayor (AA') y se toma la medida del punto de intersección hasta el epicentro (ver Figura 7.6).

- *Para la dirección perpendicular a la línea de costa.* Se procederá de la misma forma que para el caso anterior, con la diferencia de que solo se obtendrá una sola medida definida por " d_i ".

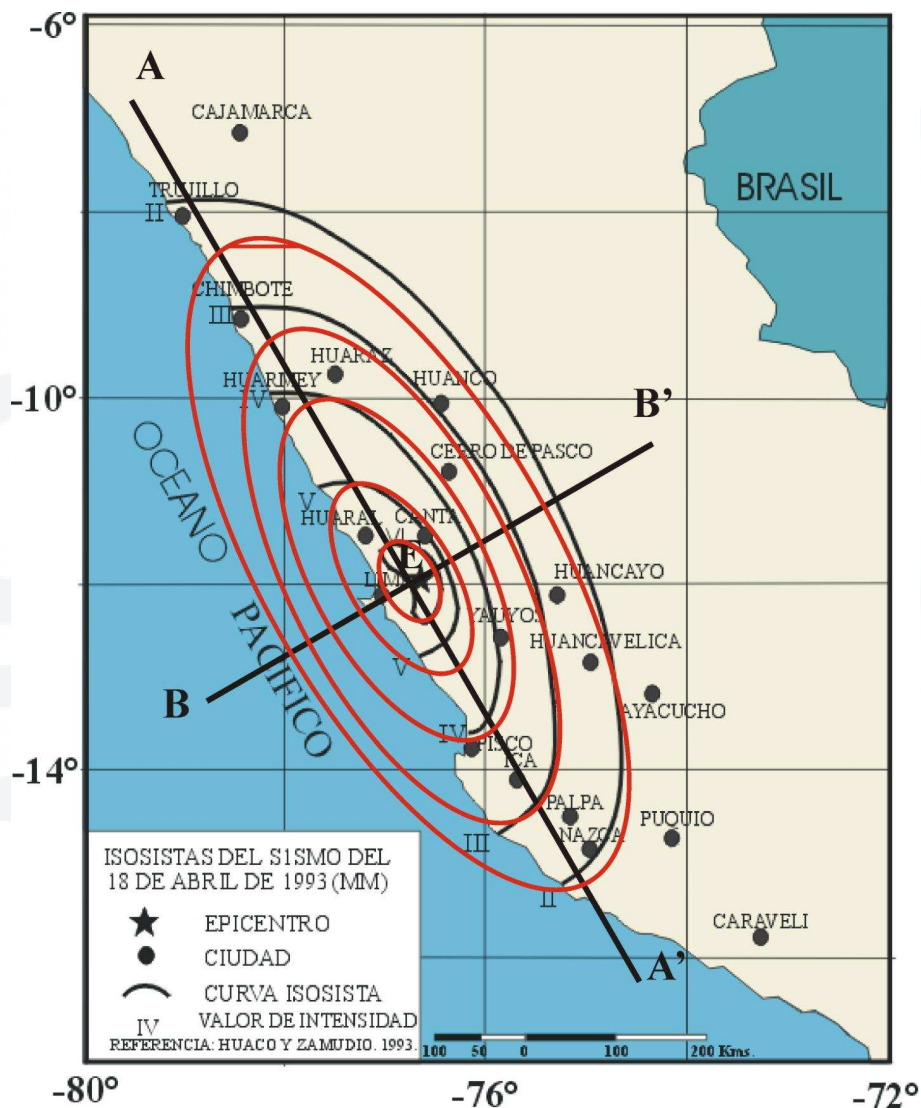


Figura 7.6. Curvas de un mapa de intensidad para un sismo con origen en el proceso de subducción (1993). Se muestra la tendencia elipsoidal de las isosistas y se señala los ejes de la elipse (AA' y BB') y el centro de la misma (E)

7.5.1.1.- Relación Atenuación – Intensidad

Partiendo de la relación dada por Ambraseys (1985), se puede generar tantas ecuaciones como curvas de isosistas se han evaluado para todos los sismos empleados en este estudio, tanto para la dirección paralela a la línea de costa (Tabla 7.3a) como para la dirección perpendicular (Tablas 7.3b). A partir de estas ecuaciones, es posible calcular los coeficientes de la relación definida en la ecuación 7.1 utilizando un modelo de regresión lineal para relaciones que tenga 2 o más variables independientes llamado “Modelo de Regresión Lineal Múltiple” (Hines y Montgomery, 1993).

Los resultados obtenidos para cada dirección, son:

- Para la dirección AA', paralela a la línea de costa:

$$I = 7.8723 + 1.0821(Ms) - 0.0023(R) - 3.9977(\log R) \pm 0.54 \quad (7.2)$$

$$\text{para } R = (D_I^2 - 48.5^2)^{1/2}$$

- Para la dirección BB', hacia el interior del continente (a la Cordillera Andina):

$$I = 6.7085 + 1.2803(Ms) - 0.0050(R) - 4.3579(\log R) \pm 0.55 \quad (7.3)$$

$$\text{para } R = (d_I^2 - 48.5^2)^{1/2}$$

Siendo I la intensidad calculada, Ms magnitud obtenida a partir de las ondas superficiales del sismo, R la distancia hipocentral correspondiente al radio epicentral d_I y a la profundidad focal promedio de $h_0 = 48.5$ Km, obtenida promediando las profundidades de todos los sismos empleados en este modelo.

Con estas dos relaciones se puede obtener valores de intensidad teóricos tanto para la dirección paralela a la línea de costa como para la perpendicular, considerando una distancia epicentral D_i o d_i y la magnitud de ondas de superficie Ms del sismo. La distancia hipocentral se calculará utilizando la profundidad promedio calculada en este estudio para este modelo (48.5 Km) y la distancia epicentral mediante el Teorema de Pitágoras.

Tabla 7.3. Radio epicentro – isosista promedio para cada una de las isosistas en todos los sismos analizados con el modelo elipsoidal. a) dirección paralela a la línea de costa (D_i) y b) dirección perpendicular a la línea de costa (d_i). “i” representa el grado de intensidad de la curva de la cual se ha tomado el radio.

EVENTO	FECHA	PROF. (KM)	Ms	RADIO EPICENTRAL MEDIO " D_i " (KM)						
				D_8	D_7	D_6	D_5	D_4	D_3	D_2
1	24/05/40	50	8.0	103.9	181.2	299.5	449.3	652.2	1033.8	-
2	24/08/42	60	8.1	87.2	180.0	253.2	376.9	573.8	-	-
3	12/12/53	30	7.8	-	68.6	179.7	326.8	457.5	647.1	839.9
4	17/10/66	37	8.0	77.8	127.8	366.7	461.1	583.3	755.5	-
5	31/05/70	71	7.8	80.8	151.5	282.8	441.1	656.6	-	-
6	09/12/70	28	7.1	75.9	174.1	259.2	359.2	-	-	-
7	03/10/74	21	7.8	133.7	174.9	216.0	312.8	390.9	-	-
8	18/04/93	94	5.6	-	-	52.9	113.8	230.2	338.6	447.1
9	12/11/96	14	6.6	-	97.0	153.5	246.5	359.6	476.8	593.9
10	03/04/99	92	6.0	-	-	90.9	168.3	249.2	340.1	-
11	23/06/02	38	7.9	134.1	297.1	347.8	445.6	554.3	507.2	630.4

EVENTO	FECHA	PROF. (KM)	Ms	RADIO EPICENTRAL MEDIO " d_i " (KM)						
				d_8	d_7	d_6	d_5	d_4	d_3	d_2
1	24/05/40	50	8.0	58.0	91.8	135.3	212.6	304.3	483.1	-
2	24/08/42	60	8.1	90.0	123.8	196.9	258.8	382.6	-	-
3	12/12/53	30	7.8	-	32.7	137.2	254.9	346.4	411.8	542.5
4	17/10/66	37	8.0	133.3	188.9	244.4	300.0	377.8	455.6	-
5	31/05/70	71	7.8	80.8	148.1	188.6	276.1	363.6	-	-
6	09/12/70	28	7.1	85.2	118.5	185.2	233.3	-	-	-
7	03/10/74	21	7.8	98.8	135.8	160.5	193.4	234.6	-	-
8	18/04/93	94	5.6	-	-	21.2	31.7	68.8	127.0	190.5
9	12/11/96	14	6.6	-	56.6	121.2	210.1	282.8	347.5	468.7
10	03/04/99	92	6.0	-	-	80.8	114.5	154.9	195.3	-
11	23/06/02	38	7.9	59.2	125.9	177.8	237.0	318.5	370.4	474.1

7.5.2.- Modelo Circular de Atenuación

Los sismos con origen en la deformación continental en Perú, presentan curvas Isosistas de forma casi circular; por lo tanto, con sólo una relación de Atenuación – Intensidad que se obtenga a partir de sus Isosistas, es posible describir la distribución radial de la energía descrita por las curvas de las isosistas de los mapas de intensidad.

Por lo tanto, con sólo una relación se podrá calcular el valor de la intensidad teórica para cualquier dirección de propagación.

En tal sentido, se considerará cada isosista como una circunferencia (Figura 7.7), con centro en el epicentro del sismo (E). Como la curva isosista no representa una circunferencia exacta, se medirán las distancias epicentro – isosista en varias direcciones las cuales estarán separadas en 20° una de la otra con el fin de cubrir lo mejor posible toda la curva. Seguidamente se promediarán estos valores y solo se tendrá una valor final “ D_i ”, para cada isosista i .

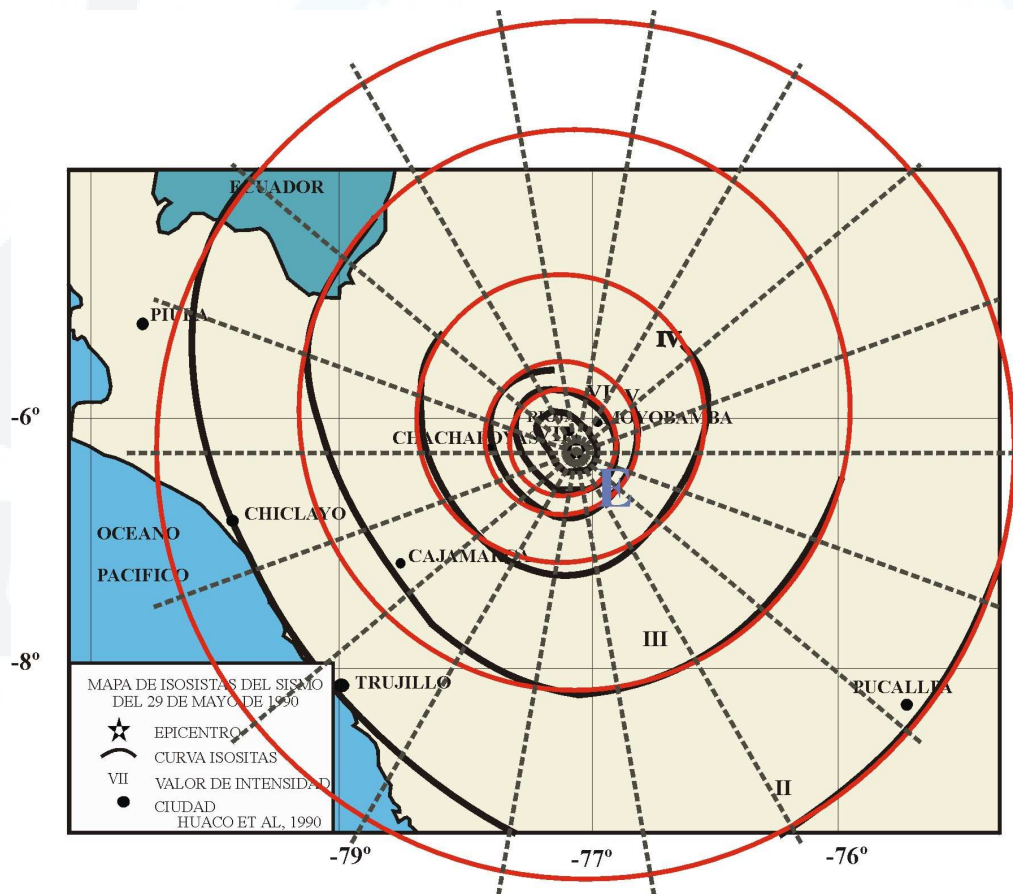


Figura 7.7. Mapa de intensidad de un sismo de la zona Subandina (1990) que muestra la tendencia circular de las curvas Isosistas. Cada línea discontinua representa una dirección que se consideró para obtener la distancia Epicentro – Isosista en este modelo.

7.5.2.1.- Relación Atenuación – Intensidad

Al igual que en el modelo elipsoidal con la relación de Ambraseys (1985), se generan ecuaciones empleando los datos de estos sismos y los valores de “ D_i ”, para cada isosista (Tabla 7.4). De la misma forma, se obtienen los coeficientes de la regresión empleando el modelo de Regresión Lineal Múltiple. El resultado obtenido es el siguiente:

$$I = 8.4209 + 2.5656(M_s) + 0.0183(R) - 11.3102(\log R) \pm 0.98 \quad (7.4)$$

Siendo I la intensidad calculada, M_s magnitud del sismo obtenida a partir de las ondas superficiales, R la distancia hipocentral correspondiente al radio epicentral d_i y a la profundidad focal promedio de $h_0 = 34$ Km, obtenida promediando las profundidades de todos los sismos empleados en este modelo.

Al igual que para el modelo anterior, con la relación 7.4 se puede obtener valores de intensidad teóricos para diferencias distancia epicentrales D_i con solo conocer la magnitud de ondas de superficie M_s del sismo. La distancia hipocentral R utilizada corresponde a la profundidad promedio calculada en este estudio para este modelo (34 Km).

Tabla 7.4. Radio epicentro – isosista promedio para cada una de las isosistas para los sismos analizados en el modelo circular. “ i ” representa el grado de intensidad de la curva de la cual se ha tomado el radio.

EVENTO	FECHA	PROF. (Km)	M_s	RADIO EPICENTRAL MEDIO " D_i " (KM)								
				d_{10}	d_9	d_8	d_7	d_6	d_5	d_4	d_3	d_2
1	15/01/58	60	7.0	-	-	-	-	167.6	-	-	243.5	-
2	19/06/68	16.3	6.9	16.5	27.6	36.7	49.2	87.3	142.6	232.9	-	-
3	01/10/69	14.5	6.2	-	-	-	12.9	19.7	27.6	42.9	67.0	93.1
4	16/02/79	53	6.9	-	-	-	49.5	103.4	149.7	-	-	-
5	05/04/86	50	5.4	-	-	3.5	7.0	10.9	20.2	36.6	43.8	-
6	29/05/90	24	6.5	-	-	-	23.6	42.7	63.2	118.1	222.0	348.1
7	04/04/91	19	6.8	-	-	-	45.0	72.2	108.9	169.7	258.9	-

CAPITULO 8

CONCLUSIONES Y RECOMENDACIONES

Una vez culminado la estructuración de la Base de Datos Sísmicos y el cálculo de la relación Atenuación – Intensidad para los sismos de Perú, se puede llegar a las siguientes conclusiones:

- Con este estudio se ha discutido la gran importancia de desarrollar una base de datos apoyándose en un software de base de datos como es el Oracle para almacenar, gestionar y administrar la información generada en un centro de investigación como lo es el Instituto Geofísico del Perú.
- Este estudio permitió proponer una metodología para estructurar y administrar una base de datos con la ayuda de un software de base de datos de alto nivel como lo es Oracle.
- Con el desarrollo de este trabajo se ha adquirido los conocimientos suficientes para entender y manejar el Software de Base de Datos Oracle en su versión 8i y su tecnología Objeto - Relacional, en modo administrador y operador; permitiendo crear objetos, gestionar la información y desarrollar mecanismos de consulta dinámicos.
- Asimismo, se ha podido organizar los datos del Centro Nacional de Datos Geofísicos sección Sismología de manera adecuada ya que toda la información se encuentra almacenada en una estructura que muestra un esquema Relacional en forma ordenada.
- Con los conocimientos adquiridos durante el desarrollo de este estudio se ha podido conocer la forma de adquisición de la información sísmica y los diferentes formatos

en los cuales se encuentra, los mismos que se deben al desarrollo constante de esta labor de investigación.

- Conociendo la información sísmica se ha elaborado tablas que almacenen de forma relacional toda la información sísmica y además de insertar fácilmente los datos y archivos en las tablas creadas, empleando las herramientas que proporciona el software de base de datos Oracle. Además, se desarrollaron aplicaciones adicionales principalmente para insertar los archivos binarios en grupo dentro de la Base de Datos Sísmicos con ayuda de SQL*loader.
- Empleando la tecnología de los programas CGI, se han elaborado las consultas de la información desde la Internet, la cual se ha realizado dada la arquitectura que Oracle ofrece, y el servidor que es el motor de su rendimiento.
- Con el desarrollo de programas CGI para la consulta se ha logrado obtener conocimientos en programación estructurada como es la programación en lenguaje C y la programación orientada a objetos.
- Con el desarrollo de la Base de Datos Sísmicos y la administración de la información sísmica se ha manipulado datos y archivos a la vez. Este tipo de información es actualmente la más manejada por todas las áreas de investigación. Esto hace suponer que este mismo trabajo puede ser aplicado a la información de cualquier área de investigación que este en desarrollo en el Perú, de tal forma que se ponga a disposición de toda la comunidad científica mundial.
- Este estudio va a servir como base para el posterior desarrollo de bases de datos y también servirá como orientación para futuras incursiones dentro del manejo del software Oracle y su tecnología Objeto - Relacional.
- En este estudio se ha obtenido la relación Atenuación - Intensidad para los sismos ocurridos en Perú, a partir de la metodología propuesta por Ambraseys (1985). Este

estudio permitió además definir dos modelos de atenuación para los sismos ocurridos en Perú: un modelo elipsoidal para los sismos con origen en el proceso de subducción y un modelo circular para los sismos con origen en el proceso de deformación cortical o de corteza.

- En este estudio se discutió la influencia que tiene la Cordillera Andina como estructura atenuante de la energía liberada por los sismos ocurridos en Perú.

Asimismo, el desarrollo de este estudio ha permitido proponer algunas recomendaciones:

- Para la administración, manejo y desarrollo de bases de datos empleando el software de base de datos Oracle es necesario partir de un conocimiento sobre bases de datos Relacionales y toda la teoría que esta comprende, pero principalmente tener conocimiento de la teoría de la programación orientada a objetos, la cual es la tendencia a futuro de las bases de datos y en sí de toda la programación actual.
- El desarrollo de la base de datos tiene que partir necesariamente del conocimiento de los datos a ser administrado y de las fuentes que tienen estos datos; sin embargo, es preferible partir del conocimiento del software y de las ventajas que este ofrece para administrar dicha información para poder determinar los objetivos finales.

BIBLIOGRAFIA

Abbey M., Corey M. Y Abramson I. (2000). Oracle 8i. Guía de Aprendizaje. Editorial McGraw Hill. España. 714p.

Abe, K. (1972). Mechanism and tectonic implications of the 1966 and 1970 Peru earthquake. Phys. Earth Planet. Inter., 5, 367 - 379.

Ambraseys, N. (1985). Intensity – Attenuation and Magnitude – Intensity Relationships for North West European Earthquake. Earthquake Eng. Struct. Dyn, 13, 733 – 778.

Audebaud, E., Capdevila, R., Dalmayrac, B., Laubaucher, G., Marocco, R., Mattauer, M., Megard, F., Paredes, J. (1973). Les traits geologiques essential des Andes Centrales (Perou - Bolivie). Revue Geographie Physique, Geologique et Dynamique., 15 (1 - 2), 73 - 114.

Barazangi, M. y Isacks, B. (1976). Spatial Distribution of Earthquakes Subduction of the Nazca Plate beneath America. Geology. 4, 686 - 692.

Barriga, V. (1939). Libro de los Cabildos de la Ciudad de Arequipa. Sesión del 10 de Agosto de 1555, reproducida en documentos para la historia de Arequipa.

Bernal, I. (2002). Aproximación a un Modelo Detallado de la Sismicidad en el Perú: Características y Evaluación de la Energía Sísmica Liberada. Tesis de Ingeniero. U.N.S.A. 170p.

Bernal, I. (2002). Propuesta de Optimización de la Red Sismica Nacional del Instituto Geofísica de Perú. Compendio de Tabajos de Investigación Realizados por los Estudiantes durante el Año 2001 (Practicass Pre - Profesionales). Volumen 3. 137 - 147.

Bernal I., Tavera H., y Antayhua Y. (2002). Zonas sismogénicas en el Perú: Volúmenes de deformación, gráficos polares y zonificación preliminar. Boletín de la Sociedad Geológica del Perú, V. 93, 31-44.

Boutell T. (1996). CGI Programming in C & Perl. Editorial Addison – Wesley Publishing Company. Australia. 93p.

Codd, E. F. (1972). "Further Normalization of the Data Base Relational Model", en Data Base Systems, Courant Computer Science Symposia Series, Vol. 6. Englewood Cliffs: Prentice-Hall.

Codd, E. F. (1974). "Recent Investigations into Relational Data Base Systems", en Proceedings IFIP Congress. Estocolmo, Suecia.

Chartelain, J.L., Guillier, B., Gueguen, P. y Bondoux, F. (1997). The Mw = 7.7 Nazca (Perú) Earthquake, November 12 1996. A repetition of the 1942 event?. Seism. Res. Lett., 68, 917 – 922.

Dalmayrac, B., Laubacher, G. y Morocco, R. (1987). Caracteres generaux de l'evolution geologique des Andes peruvienes. Trav. Doc. ORSTOM, 122, 501 págs.

Dorbath, L., Cisternas, A. y Dorbath, C. (1990B). Assesement of the size of large and great historical earthquake in Peru. Bull. Seism. Soc. Am., 80, 551 – 576.

Date, C. J. (1990). An Introduction to Database Systems, Volume 1, Fifth Edition Reading, Mass: Addison-Wesley Publishing.

Date, C. J. (1991). "Don' t Encode Information into Primary Keys!", en Relational Database Writings 1989-1991. Reading, Mass: Addison-Wesley Publishing.

Dorbath, L., Cisternas, A. y Dorbath, C. (1990). Assesement of the size of large and great historical earthquake in Peru. Bull. Seism. Soc. Am., 80, Nº 3, 551 - 576.

Fagin, R. (1977). "Multivalued Dependencies and a New Normal Form for Relational Databases", ACM TODS 2, No 4, December 1977.

Fagin, R. (1979). "Normal Forms and Relational Database Operators", Proceedings ACM SIGMOD. International Conference on Management of Data. Boston, Mass.

Flinn, E., Engdahl, E., and Hill, A. (1984). Seismic and Geographical Regionalization. Bulletin of the Seismological Society of America. Vol. 64. 771 – 993.

Hatori, T. (1968). Colombia - Peru tsunamis that observed along the coast of Japan, 1920 - 1967. Bull. Earthquake Res. Inst. Univ. Tokio, 6, 535 - 546.

Hines W. y Montgomery D., (1993). Probabilidades y Estadística para Ingeniería y Administración. Compañía Editorial Continental S.A. México. Pgs. 563 - 573.

Huaco, D. (1986). Catálogo Sísmico del Perú 1500 - 1982. Instituto Geofísico del Perú.

Huaco, P. y Zamudio, Y. (1993). Intensidades Sísmicas y otros efectos causados por el sismo del 18 de abril de 1993. Observatorio Sismológico de Camacho – 10 pp.

Lemay, L. (1995). Aprendiendo HTML para Web en una semana. 272pp.

Loney, K. (2000). Oracle8 Manual del Administrador. Editorial McGraw Hill. España. 6p.

Montessus De Ballore, F. (1911). Historia sísmica de los Andes Meridionales. Imprenta Universitaria, Santiago de Chile, Chile. 345 págs.

Ocola, L. (1979). Distribución de los daños en Arequipa Metropolitana. Simposio sobre el sismo del 16.02.79 y las construcciones de Arequipa. Colegio de Ingenieros del Perú Filial Arequipa. Arequipa.

Oracle Corporation (2000). On-Line Generic Documentation Installation and Usage Notes. "Oracle8i Concepts". Release 2 (8.1.6). A76965-01. CD-ROM.

Plafker, G., Ericksen, G. y Fernandez - Concha, J. (1971). Geological Aspect of the may 31, 1970 Peru earthquake. Bull. Seism. Soc. Am., 61, 543 - 578.

Reiter, L., (1990). Earthquake Hazard Analysis. Columbia University Press. New York.

Silgado, E., (1978). Historia de los sismos más notables ocurridos en el Perú, (1513 - 1974). Instituto Geológico Minero del Perú, Lima. Pgs. 56 - 107.

Stauder, W. (1975). Subduction of the Nazca Plate under Perú as evidenced by Focal Mechanism and by Seismicity, J. Geophys. Res. 80, 53 - 1064.

Tavera H. y Agüero C. (2000). Catalogo Sísmico del Perú, 1471 - 1982. Versión Revisada y Actualizada. Centro Nacional de Datos Geofísicos - IGP, 420p

Tavera H., Bufo E. (1998). Sismicidad y sismotectónica de Perú. Revista Física de la Tierra, N°10, 187-219.

Tavera, H., Fernandez, E., Perez - Pacheco, I., Salas, H., Rodríguez, S., Vilcapoma, L., Sánchez, B., Inza, A. y Agüero, C. (1999). Terremoto de Arequipa del 3 de abril de 1999 (Mw = 6.6). Centro Nacional de Datos - Sismología I.G.P.

Tavera, H. (2001). La Red Sísmica del Perú. Revista Geonoticias, Año VII, N°2, 32 - 33.

Tavera H., Bufo E. (2001). Source Mechanism of Earthquake in Perú. Journal of Seismology, V.5 N.4, 519-540.

Tavera H. Bufo E., Bernal, I., Antayhua, Y. y Vilcapoma, L. (2001). The

Arequipa (Peru) Earthquake of June 23, 2001. Journal of Seismology.

Udias A. y Mezcua J. (1986). Fundamentos de Geofísica. Editorial Alhambra. Madrid.
Pgs. 179 - 183.

Vargas, G. (1922). Monografía de Arica. Boletín de la Sociedad Geográfica de Lima,
Perú 63 - 77.

Web del Centro Nacional de Datos Geofísicos - Sismología del Instituto Geofísico del
Perú: <http://cns.igp.gob.pe>.

<http://www.geocities.com/SiliconValley/Pines/7894/introduccion/dss.html>

<http://www.orafaq.org/faqplus.htm#COMM>

http://ar.geocities.com/r_niella/Document/t_cap1.htm

http://ar.geocities.com/r_niella/Document/t_cap1.htm