

Diseño e implementación de una base
de datos de un

SISTEMA DE CONTROL ENERGÉTICO

Rosana Marín Berraondo

TFC del área de Base de datos relacionales

ÍNDICE

1.- INTRODUCCIÓN	5
1.1.- Alcance del proyecto	5
1.2.- Objetivos del proyecto.....	5
1.2.1.- Objetivos generales	5
1.2.2.- Objetivos específicos.....	5
1.3.- Planificación	7
1.3.1.- Calendario del proyecto	7
1.3.2.- Tareas en cada PEC.....	8
1.3.3.- Diagrama de GANNT	8
2.- DISEÑO DE LA BD	10
2.1.- Análisis de los requerimientos del sistema	10
2.2.- Diseño conceptual.....	10
2.2.1.- Modelo Entidad/Relación	10
2.2.2.- Descripción de las relaciones	12
2.3.- Diseño lógico.....	16
2.3.1.- Modelo relacional	16
2.3.2.- Explicación del modelo relacional	18
2.3.2.1.- Tablas principales	18
2.3.2.2.- Tablas auxiliares	21
2.3.2.3.- Tablas interrelacionales	24
2.3.2.4.- Tablas estadísticas	25
3.- IMPLEMENTACIÓN DE LA BBDD	29
3.1.- Aspectos generales de los scripts.....	30
3.1.1.- Nomenclatura.....	30
3.1.2.- Cabecera	30
3.1.3.- Tratamiento excepciones	31
3.1.4.- Tratamiento de datos de entrada	32
3.1.5.- Almacenamiento en el log	33
3.1.6.- Optimización de consultas.....	33
3.2.- Aspectos específicos de los scripts	33
3.2.1.- Descripción de los scripts específicos.....	33
3.2.2.- Script de creación de tablas	34
3.2.3.- Script de procedimientos ABM	35
3.2.4.- Script de creación de procedimientos estadísticos	41
3.2.6.- Script de creación de vistas materializadas	42
3.2.7.- Script de creación de los procedimientos de las consultas	43
3.2.8.- Script de creación de los procedimientos de estadísticas	43
3.2.9.- Script de listado de las estadísticas	44
3.2.10.- Script de carga de datos	44
3.2.11.- Script de borrado de información	44
3.2.12.- Script de borrado de objetos	45
3.2.13.- Script prueba listados consultas.....	45
3.2.14.- Script prueba estadísticas	45
4.- PLAN DE CONTINGENCIAS	45
5.- PLAN DE PRUEBAS	46
5.1.- Metodología a seguir	46
5.2.- Orden de los scripts	46
5.3.- Test de generación de objetos.....	46
5.4.- Plan de pruebas con ERROR	53
6.- VALORACIÓN ECONÓMICA Y RECURSOS NECESARIOS	53
6.1.- Recursos Humanos	53
6.2.- Recursos Técnicos y Materiales.....	54
6.3.- Valoración económica.....	54
7.- MEJORAS PROPUESTAS	56
8.- CONCLUSIONES	56
9.- BIBLIOGRAFIA	57
10.- ANEXOS	57

ÍNDICE IMÁGENES

Fig.1.- Tabla de calendario.....	7
Fig.2.- Diagrama de GANNT	10
Fig.3.- Modelo Entidad/Relación.....	12
Fig.4.- Relación Parte 1	12
Fig.5.- Relación Parte 2	12
Fig.6.- Relación Parte 3	13
Fig.7.- Relación Parte 4	13
Fig.8.- Relación Parte 5	14
Fig.9.- Relación Parte 6	14
Fig.10.- Relación Parte 7	15
Fig.11.- Relación Parte 8	15
Fig.12.- Relación Parte 9	15
Fig.12.- Relación entre modelos	16
Fig.13.- Modelo relacional.....	17
Fig.14.- Tabla contador	18
Fig.15.- Ejemplo tabla contador.....	18
Fig.16.- Tabla Central_produccion.....	19
Fig.17.- Ejemplo tabla central_produccion	19
Fig.18.- Tabla linea_comunica	19
Fig.19.- Ejemplo tabla linea_comunica	19
Fig.20.- Tabla cliente	20
Fig.21.- Ejemplo tabla cliente.....	20
Fig.22.- Tabla central_distribucion	21
Fig.23.- Ejemplo tabla central_distribucion.....	21
Fig.24.- Tabla modelo_contador	21
Fig.25.- Ejemplo tabla modelo_contador.....	21
Fig.26.- Tabla hist_lectura	22
Fig.27.- Ejemplo tabla hist_lectura.....	22
Fig.28.- Tabla hist_energia_emitida	22
Fig.29.- Ejemplo tabla hist_energia_emitida.....	22
Fig.30.- Tabla hist_energia_producida.....	23
Fig.31.- Ejemplo tabla hist_energia_producida	23
Fig.32.- Tabla tipo_central.....	23
Fig.33.- Ejemplo tabla tipo_central	23
Fig.34.- Tabla log	23
Fig.35.- Ejemplo tabla log.....	24
Fig.36.- Tabla contrato	24
Fig.37.- Ejemplo tabla contrato	24
Fig.38.- Tabla contador_central	24
Fig.39.- Ejemplo tabla contador_central	25
Fig.40.- Tabla E1	25
Fig.41.- Ejemplo tabla E1	25
Fig.42.- Tabla E2	26
Fig.43.- Ejemplo tabla E2	26
Fig.44.- Tabla E3	26
Fig.45.- Ejemplo tabla E3	26
Fig.46.- Tabla E4	27
Fig.47.- Ejemplo tabla E4	27
Fig.48.- Tabla E5	27
Fig.49.- Ejemplo tabla E5	27
Fig.50.- Tabla E6	28
Fig.51.- Ejemplo tabla E6	28
Fig.52.- Tabla E7	28
Fig.53.- Ejemplo tabla E7	29
Fig.54.- Pasos fase de implementación	29
Fig.55.- Nomenclaturas para los objetos.....	30
Fig.56.- Cabecera de un procedimiento	31
Fig.57.- Tratamiento de excepciones de forma manual	32
Fig.58.- Control de las excepciones.....	32
Fig.59.- Script de creación de tabla CONTADOR.....	35
Fig.60.- Script de creación de secuencia SEQ_COD_CONTADOR	35
Fig.61.- Objetos de 1_creacion_tablas_secuencias.sql.....	35
Fig.62.- Procedimiento de alta de un contador.....	36
Fig.63.- Procedimiento de baja de un modelo de contador	37
Fig.64.- Procedimiento de modificación de un contador	38
Fig.65.- Objetos de 2_creacion_procedimientos_ABM.sql.....	41

Fig.66.- Procedimiento P_ESTADISTICAS	42
Fig.67.- Objetos de 6_creacion_triggers.sql	42
Fig.68.- Objetos de 3_creacion_vistas_materializadas.sql	43
Fig.69.- Objetos de 7_creacion_procedimientos_consultas.sql	43
Fig.70.- Objetos de 4_creacion_procedimientos_estadisticas.sql	44
Fig.71.- Objetos de 9_creacion_procedimientos_estadisticas.sql	44
Fig.72.- Log de ejecución de creación_tablas_secuencias.....	47
Fig.72.- Tablas generadas.....	47
Fig.73.- Secuencias generadas	47
Fig.74.- Log de creación de los procedimientos ABM.....	48
Fig.75.- Procedimientos generados	48
Fig.76.- Log de creación de vistas materializadas.....	48
Fig.77.- Vistas materializadas generadas	49
Fig.78.- Log de creación de los procedimientos de actualización de estadísticas.....	49
Fig.79.- Procedimientos generados	49
Fig.80.- Log de creación del procedimiento de estadísticas.....	49
Fig.81.- Procedimiento P_ESTADISTICAS generado.....	50
Fig.82.- Log de creación de disparadores	50
Fig.83.- Disparadores generados con el script	50
Fig.84.- Log de creación de procedimientos de consulta.....	50
Fig.85.- Log de creación de procedimientos de listado de estadísticas	51
Fig.86.- Log de relleno de datos de las tablas.....	51
Fig.87.- Log del listado de consultas	51
Fig.88.- Log del listado de estadísticas	52
Fig.89.- Log del borrado de datos	52
Fig.90.- Ausencia de objetos en la base de datos.....	53

1.- INTRODUCCIÓN

1.1.- Alcance del proyecto

En este trabajo de fin de carrera del área de Base de Datos Relacionales (BDR) consiste en consolidar los conocimientos adquiridos en el estudio de las materias de Base de Datos I y II, Ingeniería de la programación y Sistemas gestores de Bases de Datos.

El material técnico informático que se utilizará es:

- Como base de datos: Oracle v10.
- Aplicación de desarrollo: SqlDeveloper.
- Desarrollo gráficos GANNT: GanntProyect.

El proyecto propuesto por la Comunidad Europea (CE) consiste en desarrollar e implementar una base de datos de un sistema de control energético que le permita almacenar información para una futura aplicación.

La información que el cliente requiere se puede distribuir en dos grupos:

- **Datos básicos:** clientes, modelos de contador, centrales de distribución, de producción, tipos de centrales de producción, líneas de comunicación, y la base de toda la información, los contadores.
- **Datos históricos:** Contratos, un histórico de lecturas de contadores, de energía producida por las centrales de producción e histórico de energía emitida por las centrales de distribución.

Además, la UOC propone a la CE la posibilidad de almacenar:

- Histórico de inspecciones de las centrales de producción.

1.2.- Objetivos del proyecto

El objetivo del proyecto es poder llevar un control sobre el consumo de energía que en un futuro sacará información estadística en una aplicación que trabajará sobre la información contenida en esta BD.

1.2.1.- Objetivos generales

Se ha de crear una BD con todos los requerimientos demandados que tenga total independencia con la futura aplicación que los gestionará.

La BD ha de documentar todos los procedimientos que hará servir para que puedan ser utilizados en la fase de programación.

1.2.2.- Objetivos específicos

Sobre los **datos básicos:**

Rosana Marín Berraondo

- Clientes: la CE necesita toda aquella información sobre los clientes que consumen la energía de sus contadores.
- Modelos de contador: Cada contador puede ser de un modelo diferente del que se ha de almacenar datos como el año de fabricación o la empresa que los fabrica.
- Centrales de distribución: datos sobre las centrales que emiten la energía hasta los contadores de cada cliente. Hay que tener en cuenta que una central de distribución emite energía a muchos contadores y un contador recibe energía de varias centrales de distribución, así se aseguran una alta disponibilidad de la energía.
- Centrales de producción: datos sobre las centrales que producen la energía que las centrales de distribución emiten. Para seguir consiguiendo esta "alta disponibilidad", una central de producción puede producir energía para varias centrales de distribución y una central de distribución puede recibir energía de varias centrales de producción. Esto se almacenará a través de lo que se denominará líneas de comunicación.
- Tipos de centrales de producción: información sobre el tipo de central de producción, así dependiendo de este tipo, sabremos que tipo de energía fabrica la central y por lo tanto que cantidad de energía produce o los residuos que produce o los paneles solares que contiene. Así, si es térmica tendremos los Kg. de CO₂, si es eólica el número de molinos.
- Contadores: información sobre su situación física, su consumo y por ejemplo modelo.
- Contratos: nos permiten relacionar los contadores con los clientes. Así, un cliente puede tener muchos contadores y un contador puede pertenecer a varios clientes a lo largo de su vida.

Además de la información, se ha de permitir poder operar con ella, realizando altas, bajas o modificaciones.

Sobre los **datos históricos**:

- Histórico de lecturas de contadores: almacena cada vez que se ha leído un contador, con su fecha y su consumo. Así, se pueden realizar las estadísticas de consumos por contador.
- Histórico de energía emitida: por una central de distribución en un tiempo determinado (en un mes). Así obtendremos listados y estadísticas sobre las centrales que más energía emiten, y a través de estas centrales, las zonas que más energía consumen.
- Histórico de energía producida: por una central de producción en un tiempo determinado (en un mes). Al igual que el resto de históricos será la información en la que nos basaremos en las estadísticas.

Además de la información, se ha de permitir poder operar con ella, realizando altas, bajas o modificaciones.

Sobre los **listados de consulta**:

- Listado que contenga la información sobre el contrato, la potencia máxima contratada y el tanto por ciento de consumo efectuado en relación al consumo medio de los contadores de una ciudad en una fecha determinada cuyo consumo mensual ha superado el 80% del consumo medio de todos los contadores de la misma ciudad.
- Listado que contenga los datos de dirección, energía emitida y energía máxima de las diez centrales de distribución que más energía emiten.
- Listado del identificativo, carga de la línea y energía a la que se puede ampliar (teniendo en cuenta las centrales de producción a la que está

conectada) de todas las 10 líneas de comunicación mas cargadas con respecto a su capacidad máxima.

- Listado con el DNI, código de contrato y tipo de contador de los clientes con servicio de alta disponibilidad.
- Listado con el consumo producido por una central en un periodo de tiempo para sus contadores.
- Listado con el porcentaje de lecturas presenciales y telemáticas en un intervalo de tiempo.
- Listado de contadores con una antigüedad específica.

El cliente está abierto a otros listados que la UOC le proponga y él considere útiles.

Sobre las **estadísticas**, se desea obtener:

- Consumo de los contadores de una central de producción específica en un año.
- Valor medio de la energía consumida de una línea de comunicación en un año.
- Línea con más energía consumida en el año.
- Porcentaje de líneas que superen el 50% de la energía consumida en un año.
- Número de centrales de producción que generen menos del 30% de la producción del año.
- Los diez contadores que han tenido más consumo en el año.
- Consumo medio de todos los clientes en el año.

Además, la BD ha de almacenar un histórico interno con la fecha, los procedimientos llamados y sus parámetros. (Tabla de Log) para uso interno.

1.3.- Planificación

Dividiremos el proyecto en PECs y éstas en tareas y una buena distribución de éstas a lo largo del tiempo garantizará un buen resultado.

El calendario de tareas se adaptará a las fechas impuestas por el cliente.

1.3.1.- Calendario del proyecto

En nuestro caso se trabajará de lunes a viernes.

El calendario indicado por el cliente es el siguiente:

CALENDARIO			
ACTIVIDAD	Fecha Inicio	Fecha Fin	SEM.
Inicio del proyecto (PEC1)	01/03/2012	18/03/2012	2
Control de planificación (PEC2)	19/03/2012	15/04/2012	4
Control de planificación (PEC3)	16/04/2012	20/05/2012	5
Entrega Final	21/05/2012	10/06/2012	3

Fig.1.- Tabla de calendario

En total, se trabajarán 14 semanas y un día (71 jornadas) , teniendo en cuenta mi situación personal, unas 15 horas a la semana, lo que nos resulta **213 horas** totales en el desarrollo del proyecto.

Si se trabaja 15 horas a la semana, y ésta tiene 5 días, el resultado es 3 horas al día de trabajo.

1.3.2.- Tareas en cada PEC

PEC1

- Lectura y comprensión de los requerimientos del cliente
- Instalación de programas necesarios
- Reunión cliente resolución de dudas del enunciado
- Redacción de parte de la memoria
 - Descripción del sistema y alcance total
 - Planificación estimada del proyecto
 - Distribución entregas PECs
 - Estudio de recursos necesarios
 - Análisis de riesgos y su control
 - Valoración inicial del coste total del proyecto
- Entrega de la PAC y revisión

PEC2

- Análisis de los requisitos funcionales
- Diseño conceptual de la BBDD
- Crear diagramas de Entidad/ Relación
- Diseño lógico
- Crear diagrama modelo lógico
- Scripts creación tablas
- Población de las tablas
- Índices de las tablas
- Procedimientos ABM de las tablas
- Redacción de parte de la memoria
- Entrega de la PAC y revisión

PEC3

- Procedimientos almacenados necesarios para los informes
- Disparadores
- Desarrollo módulo estadístico
- Creación juego de pruebas
- Testeo juego de pruebas
- Correcciones tras juego de pruebas
- Redacción de parte de la memoria
- Entrega de la PEC y revisión

DOCUMENTACIÓN FINAL

- Presentación en PowerPoint
- Redactar la memoria final
- Preparación del código y anexos
- Posible implementación de mejoras del sistema
- Entrega del proyecto

1.3.3.- Diagrama de GANNT

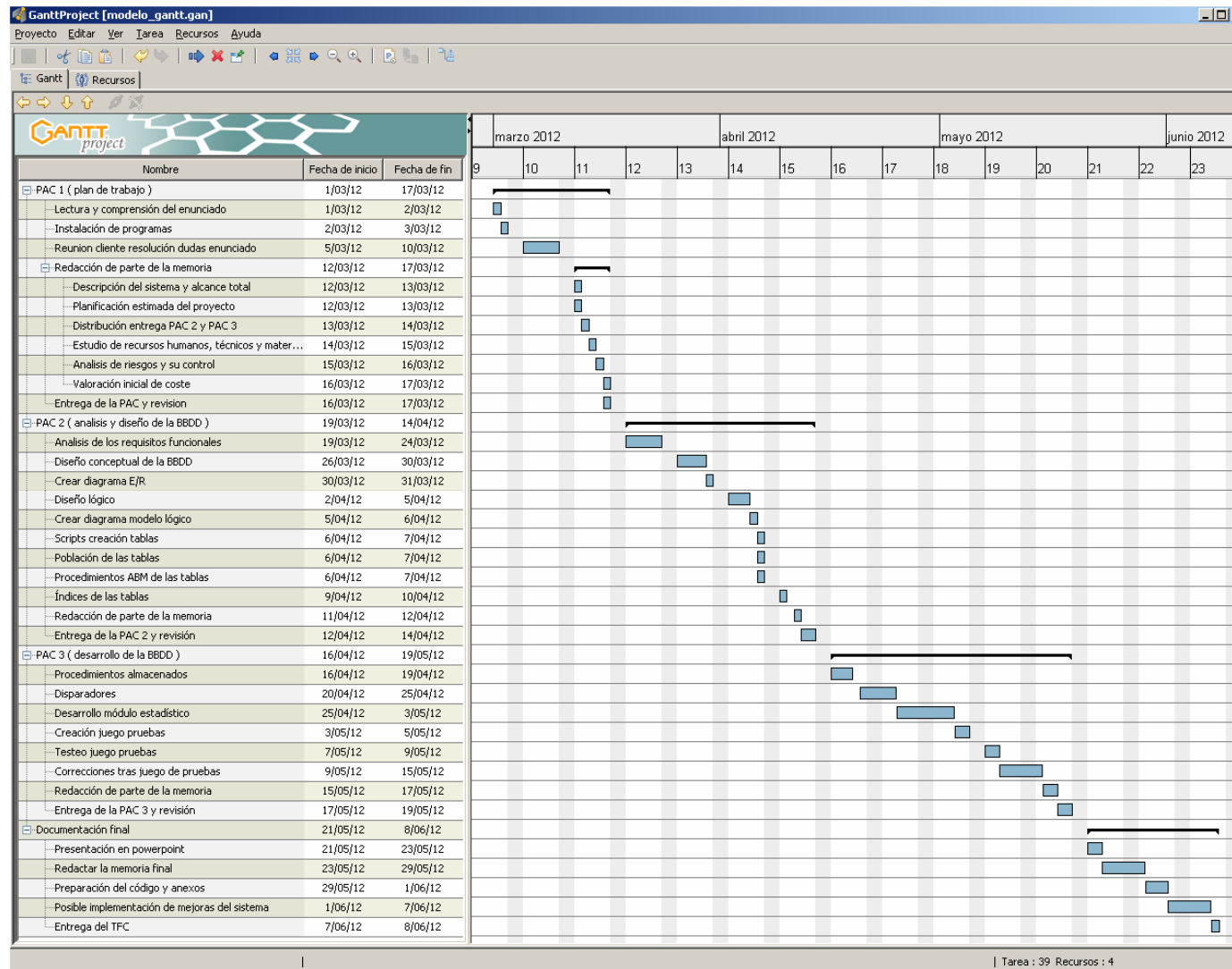


Fig.2.- Diagrama de GANNT

En la página anterior, se puede observar que en la parte izquierda del gráfico se presenta la planificación del proyecto mediante un calendario que marcará el reparto de tareas asociado a las PEC del semestre.

En la parte de la derecha del gráfico se puede observar el diagrama de Gantt asociado a la planificación. Cada rectángulo del diagrama corresponde a una tarea independiente, y se realizarán d manera secuencial.

Hay tareas que no están recogidas en la propuesta del plan de trabajo, porque no se sabe con exactitud cuando se realizarán, como por ejemplo tareas propias de gerencia, recursos humanos, nóminas...

2.- DISEÑO DE LA BD

2.1.- Análisis de los requerimientos del sistema

Los requisitos funcionales del sistema vienen descritos en el documento Enunciat_TFC_BDR_201201.doc.

2.2.- Diseño conceptual

El diseño conceptual consiste principalmente en la elaboración de una estructura de información independiente de la tecnología que se empleará en el desarrollo de la base de datos. Es decir, nos abstraemos de la base de datos que se va a utilizar, ni el sistema gestor de base de datos.

Esta estructura de información se plasmará en un modelo, que en nuestro caso es el modelo entidad-relación o E/R.

El modelo entidad-relación está formado por las entidades, los atributos y las interrelaciones de las entidades.

Este proyecto utilizará la notación UML (Unified Modeling Language) para la elaboración del modelo E/R.

2.2.1.- Modelo Entidad/Relación

A continuación se muestra el esquema conceptual asociado al sistema de control energético y se justificará el diseño expuesto mediante la explicación de cada una de las relaciones de este modelo.

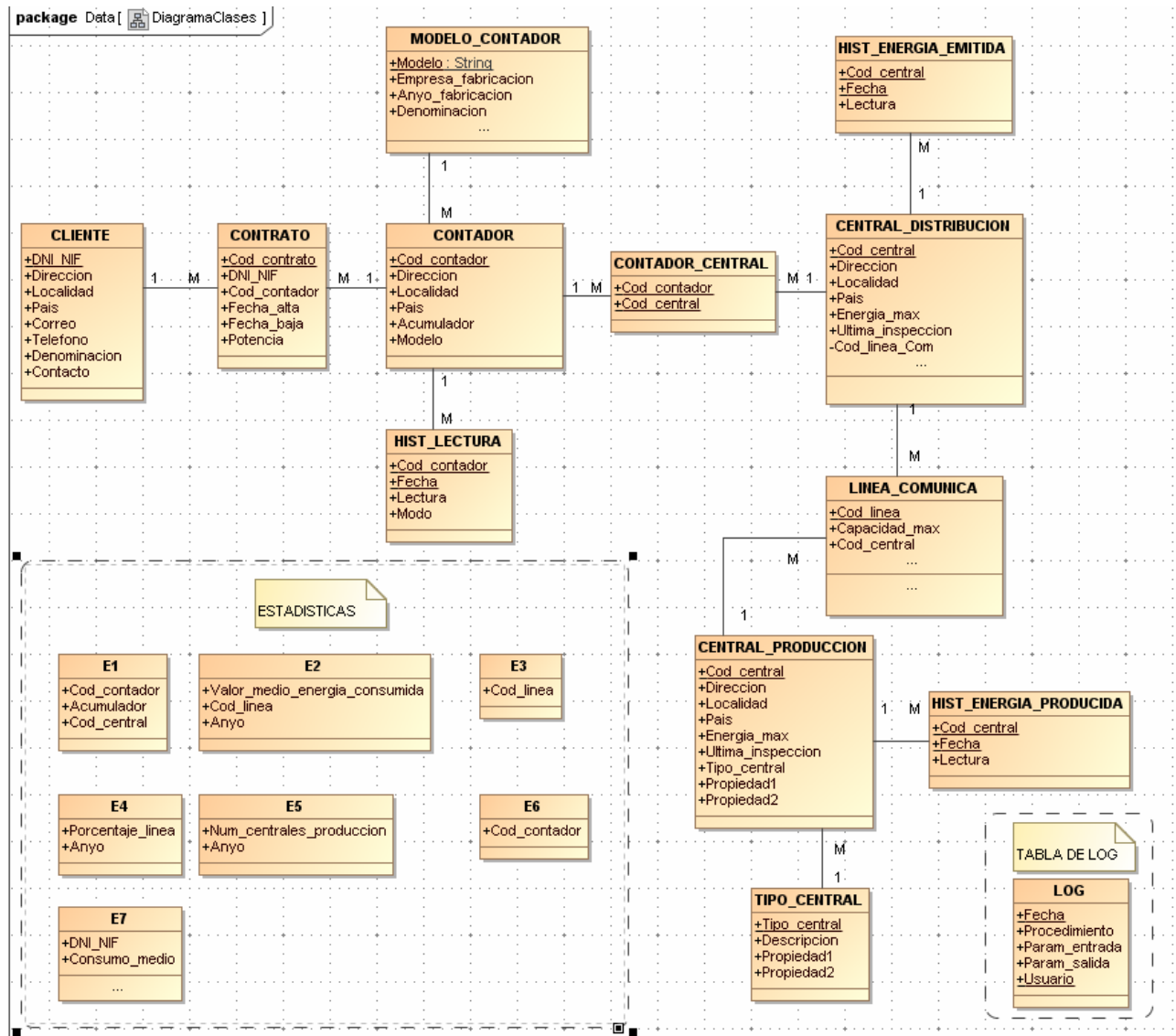


Fig.3.- Modelo Entidad/Relación

2.2.2.- Descripción de las relaciones

Se procede a la descripción de cada una de las relaciones:

PARTE 1:

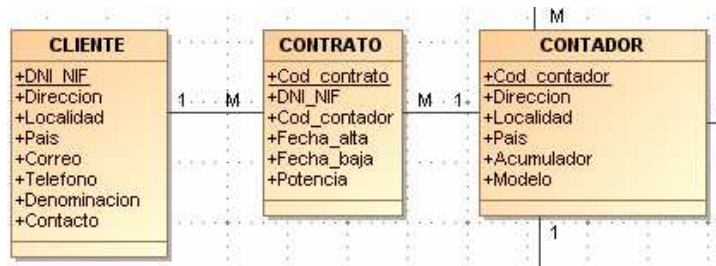


Fig.4.- Relación Parte 1

Un contador puede estar relacionado con un cliente en un momento dado, pero a lo largo de toda su vida ese contador puede pasar por varios clientes.

Un cliente puede tener asociado un contador, pero ese mismo cliente a lo largo de su vida puede tener varios contadores.

Al ser una relación M:M se crea una tabla de contratos que las relacionan. Así, un contador puede tener varios contratos a lo largo de su historia y un cliente puede tener varios contadores. Además se ve la necesidad de incluir una fecha de alta y de baja para saber los contratos que están activos en un momento del tiempo determinado.

PARTE 2:



Fig.5.- Relación Parte 2

Un contador sólo es de un modelo determinado, esto implica informar de los datos de fabricación del mismo.

Un modelo puede tener muchos contadores, ya que cuando se fabrican, no se hace un solo equipo por modelo, sino muchos.

PARTE 3:

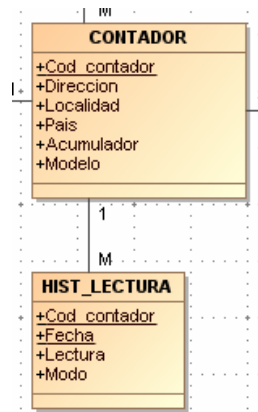


Fig.6.- Relación Parte 3

Un contador se lee mensualmente para apuntar su consumo, de manera Presencial o Telemática. Así, tenemos la entidad HIST_LECTURA con N lecturas de cada contador, según la duración de la vida del mismo.

PARTE 4:

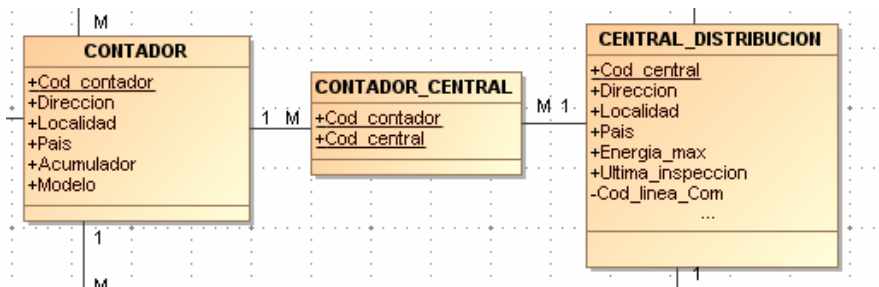


Fig.7.- Relación Parte 4

Un contador puede estar conectado a muchas centrales de distribución, de manera que así nos aseguramos de que siempre tendrá suministro.

Una central de distribución distribuye a muchos contadores.

Por lo tanto tenemos una relación M:M que hay que solventar con una entidad CONTADOR_CENTRAL que las relacione, y así asegure la alta disponibilidad de energía.

PARTE 5:

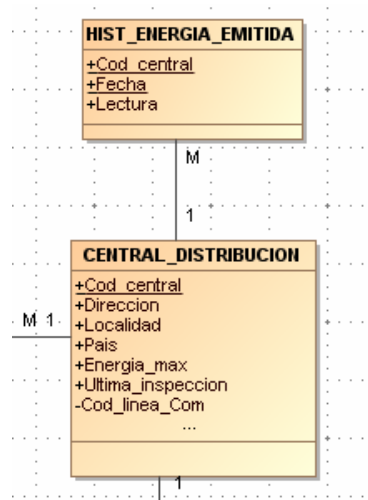


Fig.8.- Relación Parte 5

Una central de distribución emite energía hacia los contadores, de manera que mensualmente se lee la energía que cada central ha emitido y se apunta en la entidad HIST_ENERGIA_EMITIDA.

Así una central de distribución tendrá M lecturas efectuadas según sea la durabilidad de su vida.

PARTE 6:

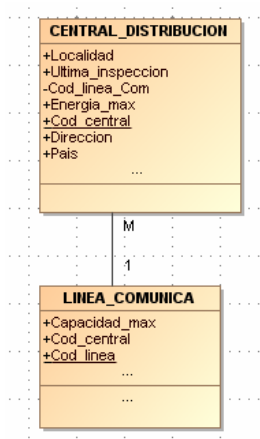


Fig.9.- Relación Parte 6

Una central de distribución está conectada a una línea de comunicación.

Una línea de comunicación tiene conectadas muchas centrales de distribución.

PARTE 7:

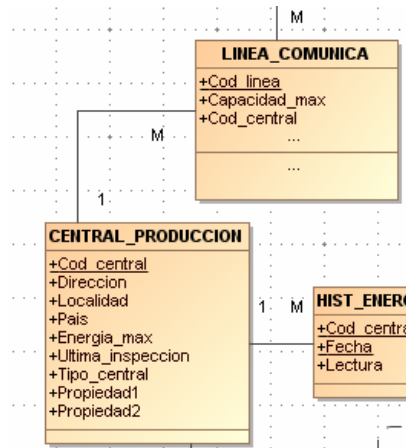


Fig.10.- Relación Parte 7

Una línea de comunicación es alimentada por una central de producción.

Una central de producción puede alimentar a varias líneas de comunicación.

PARTE 8:

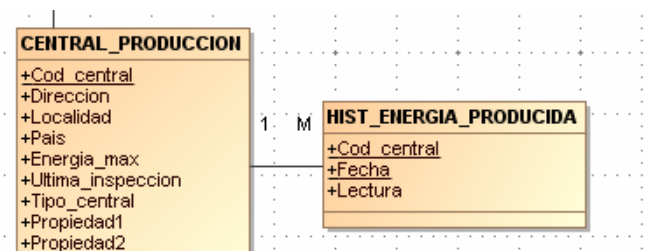


Fig.11.- Relación Parte 8

Una central de producción tendrá tantas lecturas mensuales de la producción realizada como meses de vida tenga la central.

PARTE 9:

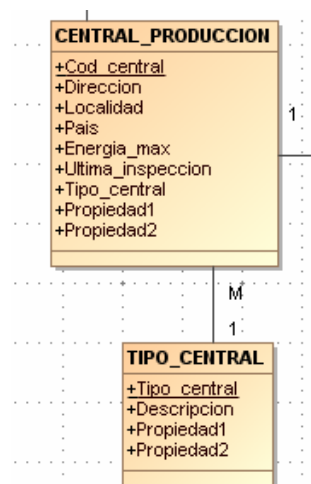


Fig.12.- Relación Parte 9

Una central de producción es de un tipo determinado. Y un tipo de central puede tener muchas centrales de producción dadas de alta.

2.3.- Diseño lógico

La etapa del diseño lógico toma como punto de partida el esquema del diseño conceptual y su finalidad es transformarlo en otro modelo que se adapte a la tecnología que se empleará posteriormente en la fase de implementación de la BBDD (modelo relacional).

En esta fase se tiene en cuenta el tipo de base de datos que se empleará (relacional, jerárquica, transaccionales...) y el sistema gestor de base de datos (SGBD) que se utilizará.

En este proyecto el modelo es el relacional y el SGBD será Oracle.

El modelo relacional está basado en la lógica de predicados y en la teoría de conjuntos. Es el más cómodo para modelar problemas reales y administrar datos dinámicamente. Proporciona una estructura de datos para representar la información del mundo real a través de un conjunto de relaciones.

2.3.1.- Modelo relacional

Hay un método que permite transformar el modelo E/R en un modelo relacional mediante la aplicación de unas reglas definidas. Las reglas básicas son:

RELACION	
MODELO E/R	MODELO RELACIONAL
Entidad	Relación
Relación 1:1	Clave foránea
Relación 1:M	Clave foránea
Relación M:M	Relación

Fig.12.- Relación entre modelos
Modelo relacional del sistema de control energético:

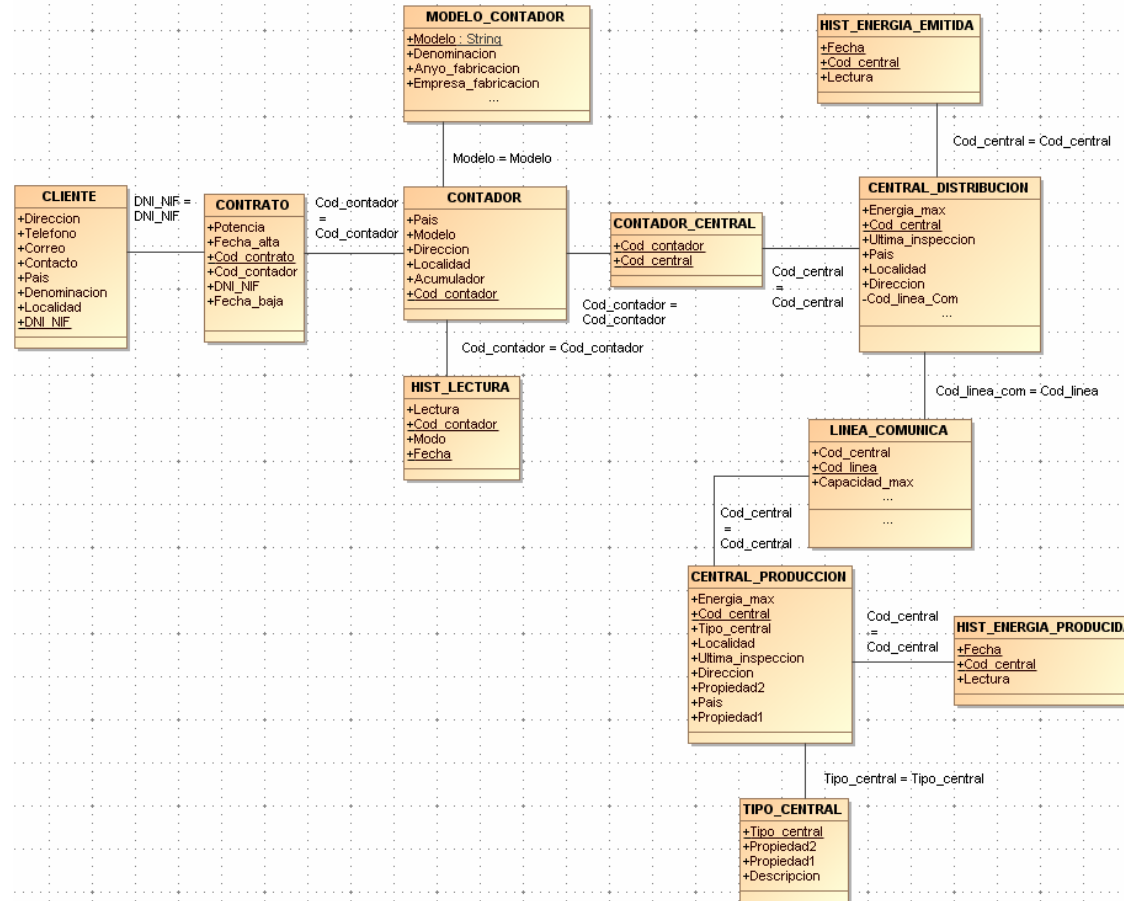


Fig.13.- Modelo relacional

2.3.2.- Explicación del modelo relacional

Las tablas que surgen del modelo relacional las hemos dividido en principales, auxiliares, interrelacionales y estadísticas. Según su labor en nuestro esquema.

2.3.2.1.- Tablas principales

A continuación se describen las tablas principales del modelo relacional expuesto:

CONTADOR				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_contador	PK	NUMBER(12)	Identificador único que distingue inequívocamente a un contador (num_serie)	NOT NULL
Direccion	-	VARCHAR2(40)	Dirección física de la localización del contador	NULL
Localidad	-	VARCHAR2(15)	Localidad de la dirección física	NULL
Pais	-	VARCHAR2(15)	País de la dirección física	NULL
Acumulador	-	NUMBER(38,2)	Acumulador del consumo total	NULL
Modelo	AK	VARCHAR2(20)	Código de modelo de contador	NOT NULL

Fig.14.- Tabla contador

Ejemplo del registro de la tabla:

Cod_contador	Direccion	Localidad	Pais	Acumulador	Modelo
253	Avda/Estación 1	Madrid	España	0	F1-9i8

Fig.15.- Ejemplo tabla contador

NOTA:

1.- Este contador tendría en el acumulador un 0 porque todavía no tendría asociado un contrato, y por lo tanto un cliente que consuma.

2.- El modelo F1-9i8 estaría asociado y por lo tanto tendría que existir previamente, con la tabla de modelos. Es clave foránea.

FINALIDAD: Cumplir el R1 de los requisitos funcionales descritos por el cliente, almacenando los datos principales de los contadores. No se han cumplido todos los requisitos del R1, las excepciones son:

- Tanto el contrato asociado (si existiera) como la potencia se almacenan en otra tabla de contratos que relaciona a los clientes con los contadores. (contrato)
- El consumo eléctrico mensual se almacenará en una tabla histórica, al igual que la fecha de última lectura (hist_lectura).
- Las centrales de distribución a las que se conecta el contador se almacena en una tabla secundaria (cont_central).
- Los datos propios del modelo de contador tienen una tabla propia (modelo_contador).

CENTRAL PRODUCCION				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_central	PK	NUMBER(12)	Identificador único que distingue inequívocamente a una central	NOT NULL
Direccion	-	VARCHAR2(40)	Dirección física de la localización de la central	NULL

Localidad	-	VARCHAR2(15)	Localidad de la dirección física	NULL
Pais	-	VARCHAR2(15)	País de la dirección física	NULL
Energia_max	-	NUMBER(20,2)	Energía máxima que puede producir en un mes	NOT NULL
Ultima_inspeccion	-	DATE	Fecha de la última inspección técnica	NULL
Tipo_central	AK	NUMBER(1)	Identificador del tipo de central que es (CARBON, EOLICA..)	NOT NULL
Propiedad1	-	NUMBER(10)	Propiedad variable según el tipo de central	NULL
Propiedad2	-	NUMBER(10)	Propiedad variable según el tipo de central	NULL

Fig.16.- Tabla Central_produccion

Ejemplo del registro de la tabla:

Cod_central	Dirección	Localidad	País	Energia_max	Ultima_inspeccion	Tipo_central	Propiedad1	Propiedad2
587	Pza/España	Madrid	España	85000	02/02/2012	1	10	0

Fig.17.- Ejemplo tabla central_produccion

NOTA:

1.- Esta central no tendría propiedad2 porque el requisito indica que para las centrales de tipo eólica (1) almacenarán el número de molinos (propiedad1), no especifica una segunda característica.

2.- El campo tipo de central se obtendría de un maestro de tipos de centrales y según éste valor se almacenarán otros en los dos atributos propiedad1 y propiedad2. Es premisa que exista el tipo de central dado de alta antes de la inserción de la central.

FINALIDAD: Cumplir el R2 de los requisitos funcionales descritos por el cliente, almacenando los datos principales de las centrales de producción. No se han cumplido todos los requisitos del R2 con esta tabla, las excepciones son:

- Las centrales de distribución alimentadas se obtendrán a partir de la tabla de las líneas de comunicación (linea_comunica).
- Histórico de la energía producida mensualmente se almacenará en una tabla aparte (hist_ener_producida).

LINEA COMUNICA				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_linea	PK	NUMBER(12)	Identificador único que distingue inequívocamente a una línea de comunicación	NOT NULL
Capacidad_max	-	NUMBER(20,2)	Capacidad máxima que puede transportar	NOT NULL
Cod_central	AK	NUMBER(12)	Código de central de producción de la línea de comunicación	NOT NULL

Fig.18.- Tabla linea_comunica

Ejemplo del registro de la tabla:

Cod_linea	Capacidad_max	Cod_central
7	8542,45	587

Fig.19.- Ejemplo tabla linea_comunica

NOTA:

1.- El código de central de producción ha de existir en la tabla central_produccion siendo clave foránea de ésta.

FINALIDAD: Cumplir el R3 de los requisitos funcionales descritos por el cliente, almacenando la relación entre las centrales de producción y las de distribución a través de sus líneas de comunicación. El requisito pide que una línea de comunicación sólo se puede conectar a una central de producción, pero puede estar conectada a muchas centrales de distribución, por lo tanto en la entidad de centrales de distribución guardaremos la información de su línea de comunicación.

CLIENTE				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
DNI_NIF	PK	VARCHAR22(12)	Identificador único que distingue inequívocamente a un cliente	NOT NULL
Direccion	-	VARCHAR22(40)	Dirección de contacto del cliente	NULL
Localidad	-	VARCHAR21(15)	Localidad de contacto del cliente	NULL
Pais	-	VARCHAR22(15)	Pais de contacto del cliente	NULL
Correo	-	VARCHAR22(30)	Dirección de correo electrónico de contacto del cliente	NULL
Telefono	-	VARCHAR22(9)	Teléfono de contacto con el cliente	NULL
Denominación	-	VARCHAR22(50)	Nombre y apellidos del cliente o su nombre de empresa	NOT NULL
Contacto	-	VARCHAR22(20)	Nombre de la persona de contacto	NULL

Fig.20.- Tabla cliente

Ejemplo del registro de la tabla:

DNI_NIF	Dirección	Localidad	Pais	Correo	Telefono	Denominación	Contacto
12345678X	Pza/Mayor	Obón	España	a.martinez@hotmail.com	974123456	Alfredo Martinez	
A98765432	c/Einstein	Tordesillas	España	Hierros_torres@cepymen te.com	976123456	Hierros Torres	Mari Carmen

Fig.21.- Ejemplo tabla cliente

NOTA:

1.- En el primer registro vemos la información almacenada para un cliente físico, que no tiene persona de contacto pues es él mismo.

2.- En el segundo registro vemos la información almacenada para una empresa, Hierros Torres, que por el contrario, tiene persona de contacto. (Mari Carmen).

FINALIDAD: Cumplir el R4 de los requisitos funcionales descritos por el cliente, almacenando los datos de los clientes. Además de la información necesaria descrita en R4, se ha considerado el almacenar el correo electrónico, el teléfono y la persona de contacto en el caso de empresas.

CENTRAL_DISTRIBUCION				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_central	PK	NUMBER(12)	Identificador único que distingue inequívocamente a una central de distribución	NOT NULL
Direccion	-	VARCHAR2(40)	Dirección física de la localización de la central	NULL
Localidad	-	VARCHAR2(15)	Localidad de la dirección física	NULL
Pais	-	VARCHAR2(15)	País de la dirección física	NULL
Cod_linea	AK	NUMBER(12)	Código de línea de comunicación a la que se conecta la central de	NOT NULL

			distribución	
Energia_max	-	NUMBER(20,2)	Energía máxima que puede emitir	NOT NULL
Ultima_inspeccion	-	DATE	Fecha de la última inspección técnica	NULL

Fig.22.- Tabla central_distribucion

Ejemplo del registro de la tabla:

Cod_centra l	Direccion	Localida d	Pais	Cod_line a	Energia_ma x	Ultima_inspeccio n
112	Ctra.Aeropuert o 12	Sevilla	España	7	789514,98	01/02/2012

Fig.23.- Ejemplo tabla central_distribucion

NOTA:

1.- Esta central está asociada a la línea de comunicación 7, que tendría la obligación de existir en la tabla linea_comunica, que funcionaría como su maestro.

FINALIDAD: Almacenar las centrales de distribución, que relacionan los contadores con las centrales de producción. En esta entidad, la energía no se produce, sino que se distribuye, se emite.

- Las centrales de producción que alimentan a estas centrales de distribución se obtendrían a través de las líneas de comunicación. (linea_comunica).
- Histórico de la energía emitida mensualmente se almacenará en una tabla aparte (hist_ener_emitida).

2.3.2.2.- Tablas auxiliares

A continuación se describen las tablas auxiliares necesarias en el modelo relacional expuesto.

MODELO_CONTADOR				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Modelo	PK	VARCHAR22(20)	Identifica unívocamente a los modelos de contador	NOT NULL
Empresa_fabricacion	-	VARCHAR22(20)	Empresa que fabrica el modelo de contador	NOT NULL
Anyo_fabricacion	-	NUMBER(4)	Año en el que se ha fabricado por primera vez este modelo de contador	NOT NULL
Denominación	-	VARCHAR22(50)	Denominación del modelo	NULL

Fig.24.- Tabla modelo_contador

Ejemplo del registro de la tabla:

Modelo	Empresa_fabricacion	Anyo_fabricacion	Denominación
F1-9i8	Fernandez y Gonzalez	2010	Progresivo

Fig.25.- Ejemplo tabla modelo_contador

FINALIDAD: Almacenar los modelos de contadores existentes, así como su fabricante y año en el que se empezó a fabricar. Para dar de alta un nuevo contador, tiene que existir su modelo previamente en esta tabla. Completaríamos el R1 de los requisitos funcionales indicados por el cliente.

HIST_LECTURA				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Fecha	PK	DATE	Fecha en la que se realiza la lectura del	NOT NULL

Cod_contador	PK	NUMBER(12)	contador	NOT NULL
Lectura	-	NUMBER(12,2)	Identificativo del contador del que se realiza la lectura	NOT NULL
Modo	-	VARCHAR22(1)	Cantidad de KWh que marca el contador	NOT NULL
			Modo en el que se realiza la lectura (P o T) de Presencial o Telemático	NOT NULL

Fig.26.- Tabla hist_lectura

Ejemplo del registro de la tabla:

Fecha	Cod_contador	Lectura	Modo
28/02/2012	235	1257,45	P

Fig.27.- Ejemplo tabla hist_lectura

NOTA: En la fecha 28/02/2012 se leyó el contador 235 de modo presencial y dio una lectura de 1257.45.

FINALIDAD: Tabla que almacena las lecturas de los contadores que se hacen mensualmente. Como premisa ha de existir el contador en la tabla de contadores. Es uno de los requisitos que se indica en R1 y específicamente en el R5. Se ha de controlar que ese mes no se realicen mas de una lectura contador (sea presencial o telemática)
Los modos de realizar las lecturas actualmente sólo son dos, o Presencial (P) o Telepáticamente (T).

HIST_ENERGIA_EMITIDA				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_central	PK	NUMBER(12)	Código identificativo de la central de distribución	NOT NULL
Fecha	PK	DATE	Fecha en la que se mide la energía que se ha emitido ese mes	NOT NULL
Lectura	-	NUMBER(12,2)	Cantidad de KWh que ha emitido la central en el mes	NOT NULL

Fig.28.- Tabla hist_energia_emitida

Ejemplo del registro de la tabla:

Cod_central	Fecha	Lectura
112	30/01/2012	1257,45

Fig.29.- Ejemplo tabla hist_energia_emitida

NOTA: La central de distribución 112 emitió en el mes de Enero de 2012 una cantidad de 1257,45 KWh.

FINALIDAD: Almacenar la cantidad de KWh que una central de distribución emite cada mes. El mes será el almacenado en el atributo "Fecha", es decir, en el ejemplo, la lectura del mes de Enero es de 1257,45. Como premisa ha de existir la central en la tabla de central_distribucion.

HIST_ENERGIA_PRODUCIDA				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_central	PK	NUMBER(12)	Código identificativo de la central de producción	NOT NULL
Fecha	PK	DATE	Fecha en la que se mide la energía que se ha producido en ese mes	NOT NULL
Lectura	-	NUMBER(12,2)	Cantidad de KWh que se ha producido en la central en ese mes	NOT NULL

Fig.30.- Tabla hist_energia_producida

Ejemplo del registro de la tabla:

Cod_central	Fecha	Lectura
587	30/01/2012	1257,45

Fig.31.- Ejemplo tabla hist_energia_producida

NOTA: La central de producción 587 produjo 1257,45 KWh en el mes de Enero.

FINALIDAD: Almacenar la cantidad de KWh que una central de producción fabrica cada mes. El mes será el almacenado en el atributo "Fecha", es decir, en el ejemplo, la lectura del mes de Enero para la central 587 es de 1257,45. La entidad se requiere en R2.

TIPO Central				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Tipo_central	PK	NUMBER(1)	Código identificativo del tipo de central de producción	NOT NULL
Descripción	-	VARCHAR22(20)	Descripción del tipo de central	NOT NULL
Propiedad1	-	VARCHAR22(20)	Valor que almacenará la central en su campo Propiedad1	NULL
Propiedad2	-	VARCHAR22(20)	Valor que almacenará la central en su campo Propiedad2	NULL

Fig.32.- Tabla tipo_central

Ejemplo del registro de la tabla:

Tipo_central	Descripción	Propiedad1	Propiedad2
1	EOLICA	Número de molinos	

Fig.33.- Ejemplo tabla tipo_central

NOTA: El tipo de central 1 corresponde con las Eólicas que almacenarán en su Propiedad1 el número de molinos y en su propiedad2 nada.

FINALIDAD: Maestro de tipos de central de producción que nos informarán del significado de los atributos "Propiedad1" y "Propiedad2" de las centrales de producción.

El requisito del cliente para contemplar esta entidad es el R2.

LOG				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Fecha	PK	DATE	Fecha con segundos en el que se ejecuta el procedimiento	NOT NULL
Procedimiento	-	VARCHAR22(50)	Procedimiento ejecutado	NOT NULL
Param_entrada	-	VARCHAR22(100)	Parámetros de entrada separados por " "	NULL
Param_salida	-	VARCHAR22(100)	Parámetros de salida separados por " "	NULL
Usuario	PK	VARCHAR22(25)	Usuario de la BD que ha ejecutado el parámetro	NOT NULL

Fig.34.- Tabla log

Ejemplo del registro de la tabla:

Fecha	Procedimiento	Param_entrada	Param_salida	Param_salida
12/02/2012 20:30:12	Alta_tipo_central	2 SOLAR NUM_PANELES	OK	ORACLE

Fig.35.- Ejemplo tabla log

NOTA: En el registro de ejemplo podemos observar que el día 12 de febrero el usuario "ORACLE" llamó al procedimiento Alta_tipo_central() con los parámetros de entrada 2|SOLAR|NUM_PANELES| y dio como resultado un OK. Esto significa que se dio de alta un nuevo tipo de central de producción, la "SOLAR" con una propiedad que es el número de paneles.

FINALIDAD: Mantener un log de la actividad de la BBDD. Es un requerimiento de metodología pedido por el cliente.

2.3.2.3.- Tablas interrelacionales

A continuación se detallan las tablas necesarias para generar una relación entre entidades principales:

CONTRATO				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_contrato	PK	NUMBER(12)	Código identificativo de los contratos	NOT NULL
DNI_NIF	AK	VARCHAR22(12)	Código identificativo del cliente	NOT NULL
Cod_contador	AK	NUMBER(12)	Código identificativo del contador	NOT NULL
Fecha_alta	-	DATE	Fecha de alta de la relación contador-cliente	NOT NULL
Fecha_baja	-	DATE	Fecha de baja de la relación contador-cliente	NULL
Potencia	-	NUMBER(12)	Potencia contratada por el cliente medida en KW	NOT NULL

Fig.36.- Tabla contrato

Ejemplo del registro de la tabla:

Cod_contrato	DNI_NIF	Cod_contador	Fecha_alta	Fecha_baja	Potencia
20090002	12345678X	253	01/10/2009		10
200100128	5987555z	154	01/02/2001	30/08/2010	25

Fig.37.- Ejemplo tabla contrato

NOTA: En el primer registro se observa un contrato que relaciona el cliente 12345678X con el contador 253 desde la fecha 01/10/2009.

En el segundo registro está el contrato que relaciona el cliente 5987555Z con el contador 154. Este contrato ya está finalizado desde la fecha 30/08/2009.

Para dar de alta un contrato se tiene que tener dado de alta el cliente y el contador a los que se asocia.

FUNCIONALIDAD: Relacionar los clientes con los contadores y establecer un inicio y fin (si procede) a la relación.

CONTADOR_CENTRAL				
Nombre del atributo	Tipo de atributo	Tipo de dato	Descripción	Restricción
Cod_contador	PK	NUMBER(12)	Código identificativo del contador	NOT NULL
Cod_central	PK	NUMBER(12)	Código identificativo de la central	NOT NULL

Fig.38.- Tabla contador_central

Ejemplo del registro de la tabla:

Cod_contador	Cod_central
253	112

Fig.39.- Ejemplo tabla contador_central

NOTA: Este registro relaciona el contador 253 con la central 112.
 Como premisas básicas han de existir ambos códigos en sus maestros.

FUNCIONALIDAD: Los contadores pueden estar unidos a varias centrales de distribución y las centrales de distribución pueden suministrar a varios contadores, se necesita una tabla que relacione estas dos entidades.

2.3.2.4.- Tablas estadísticas

A continuación se detallan las tablas necesarias para el módulo estadístico descrito en el R7. Estas tablas se actualizarán cada vez que se cambie una entidad relacionada con algún origen de sus datos mediante procedimiento almacenado.

En las estadísticas que se requiera para un dato en concreto, se hará una select de la tabla (E1, E2...) y en la cláusula where se especificará la condición.

Para cada una de las entidades se ha incluido el punto "ENTIDADES IMPLICADAS" donde se detallan las entidades desde las que se obtiene el dato almacenado. Información base desde la que partir para regenerar la información estadística, es decir, para cada cambio de entidad implicada se ha de regenerar la información de las entidades estadísticas mediante un trigger.

E1			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Cod_contador	NUMBER(12)	Identificativo de contador	NOT NULL
Acumulador	NUMBER(12,2)	Consumo del contador	NOT NULL
Cod_central	NUMBER(12)	Identificativo de la central de producción	NOT NULL

Fig.40.- Tabla E1

Ejemplo:

Cod_contador	Acumulador	Cod_central
457	7841,45	112
458	985,45	112
967	2227,94	110

Fig.41.- Ejemplo tabla E1

NOTA: Para una central de producción, el consumo de los contadores asociados.

La tabla almacenará los datos de todas las centrales de producción sumando el consumo de cada contador agrupado. Así cuando el usuario quiera ver los contadores de una central en concreto, realizará una select sobre E1 e indicará en la cláusula WHERE la central de la que quiere ver sus contadores.

En el ejemplo podemos ver que la central 112 tiene asociado dos contadores (457 y 458) con su acumulador de consumo. Y vemos la central 110 con su único contador 967 y el consumo asociado a éste.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 1 del R7 del cliente.

ENTIDADES IMPLICADAS:

HIST_LLECTURA
 LINEA_COMUNICA
 CENTRAL_DISTRIBUCION
 CONTADOR_CENTRAL

E2			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Valor_medio	NUMBER(12,2)	Valor medio de la energía consumida de una línea en un año concreto	NOT NULL
Cod_linea	NUMBER(12)	Identificativo de la línea de comunicación	NOT NULL
Anyo	NUMBER(4)	Año	NOT NULL

Fig.42.- Tabla E2

Ejemplo:

Valor_medio	Cod_linea	Anyo
1257,94	547	2011
2685,98	547	2012
985	548	2012

Fig.43.- Ejemplo tabla E2

NOTA: Para una línea de comunicación y un año, el valor medio de la energía consumida .

La tabla almacenará los datos de todas las líneas y años en los que éstas líneas han consumido junto con el valor medio de la energía consumida. Así cuando el usuario quiera ver los datos de una línea de comunicación y un año, realizará una select sobre E2 especificando en la cláusula WHERE el año y la línea.

En el ejemplo podemos ver que la línea 547 tiene energía consumida para el 2011 y para el 2012. Y la línea 548 tiene energía consumida tan solo para el 2012.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 2 del R7 del cliente.

ENTIDADES IMPLICADAS: CENTRAL_DISTRIBUCION
 CONTADOR_CENTRAL
 HIST_LLECTURA

E3			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Cod_linea	NUMBER(12)	Línea mas cargada de energía consumida	NOT NULL

Fig.44.- Tabla E3

Ejemplo:

Cod_linea
547

Fig.45.- Ejemplo tabla E3

NOTA: Línea mas cargada de energía consumida.

La tabla almacenará el código identificativo de la línea de comunicación que esté mas cargada de todas las existentes de energía que se haya consumido. La tabla tan sólo tendrá un registro.

En el ejemplo vemos que la línea 547 es la mas cargada en nuestro entorno.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 3 del R7 del cliente.

ENTIDADES IMPLICADAS

HIST_LLECTURAS
 CONTADOR_CENTRAL
 CENTRAL_DISTRIBUCION

E4			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Porcentaje_linea	NUMBER(2)	Porcentaje de las líneas que superen el 50% de la energía consumida	NOT NULL
Anyo	NUMBER(4)	Año	NOT NULL

Fig.46.- Tabla E4

Ejemplo:

Porcentaje_linea	Anyo
20	2011
15	2010

Fig.47.- Ejemplo tabla E4

NOTA: Para un año concreto, el tanto por ciento de las líneas que superan el 50% de la energía consumida.

La tabla almacenará el porcentaje de las líneas que superan el 50% de la energía consumida para todos los años. Así cuando el usuario quiera ver un año en concreto tan sólo tiene que filtrar con la cláusula WHERE la entidad E4 para el año que desee.

En el ejemplo tenemos el año 2010 y 2011. En el 2011 fueron el 20% de las líneas y en el 2010 el 15%.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 4 del R7 del cliente.

ENTIDADES IMPLICADAS

HIST_LLECTURAS
 CONTADOR_CENTRAL
 LINEA_COMUNICA
 CENTRAL_PRODUCCION
 HIST_ENERGIA_PRODUCIDA

E5			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Num_centrales_produccion	NUMBER(4)	Número de centrales de producción que generan menos del 30% de ésta	NOT NULL
Anyo	NUMBER(4)	Año	NOT NULL

Fig.48.- Tabla E5

Ejemplo:

Num_centrales_produccion	Anyo
20	2011
15	2010

Fig.49.- Ejemplo tabla E5

NOTA: Para un año concreto, número de centrales de producción que generan menos del 30% de la producción.

Rosana Marín Berraondo

La tabla almacenará el número de centrales de producción que generan menos del 30% de la producción para todos los años. Así, cuando el usuario quiera ver un año en concreto,, tan sólo tendrá que filtrar con la cláusula WHERE la entidad E5 para el año que desee.

En el ejemplo podemos ver que en el año 2011 fueron 20 centrales las que no llegaron al 30% de la producción, y en el año 2010 fueron 15.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 5 del R7 del cliente.

ENTIDADES IMPLICADAS

HIST_ENERGIA_PRODUCIDA

E6			
Nombre del atributo	Tipo de dato	Descripción	Restricción
Cod_contador	NUMBER(12)	Identificativo de contador	NOT NULL

Fig.50.- Tabla E6

Ejemplo:

Cod_contador
589
547
548
521
654
487
987
544
145

Fig.51.- Ejemplo tabla E6

NOTA: Los 10 contadores que históricamente han tenido mas consumo.

La tabla almacenará siempre diez registros que coincidirán con las diez claves identificativas de los contadores que a lo largo de toda la vida del sistema han tenido mas consumo.

En el ejemplo podemos ver las claves de estos diez contadores. Hay que tener cuidado porque al comienzo de la vida del sistema no habrá diez contadores dados de alta. Por lo tanto, esta entidad no tendrá diez registros hasta que el sistema esté en pleno funcionamiento.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 6 del R7 del cliente.

ENTIDADES IMPLICADAS

HIST_LECTURA

E7			
Nombre del atributo	Tipo de dato	Descripción	Restricción
DNI_NIF	VARCHAR22(12)	Identificativo del cliente	NOT NULL
Consumo_medio	NUMBER(12,2)	Consumo medio del cliente	NOT NULL

Fig.52.- Tabla E7

Ejemplo:

DNI_NIF	Consumo_medio
---------	---------------

1234567Z	458,22
9999999B	745,12

Fig.53.- Ejemplo tabla E7

NOTA: Listado de los clientes con su consumo medio.

La tabla tendrá tantos registros como clientes existan en la entidad CLIENTE.

En el ejemplo podemos ver que sólo hay dos clientes y sus consumo.

FINALIDAD: Almacenar los requisitos necesarios para el cumplimiento del punto 7 del R7 del cliente.

ENTIDADES IMPLICADAS

CONTRATO
HIST_LECTURAS

3.- IMPLEMENTACIÓN DE LA BBDD

Definido completamente el diseño de la BD, pasaremos a la implementación.

Antes de tratar los aspectos técnicos y especificar las soluciones, describiremos la metodología de trabajo empleada.

La fase de implementación la completaremos siguiendo los siguientes pasos:

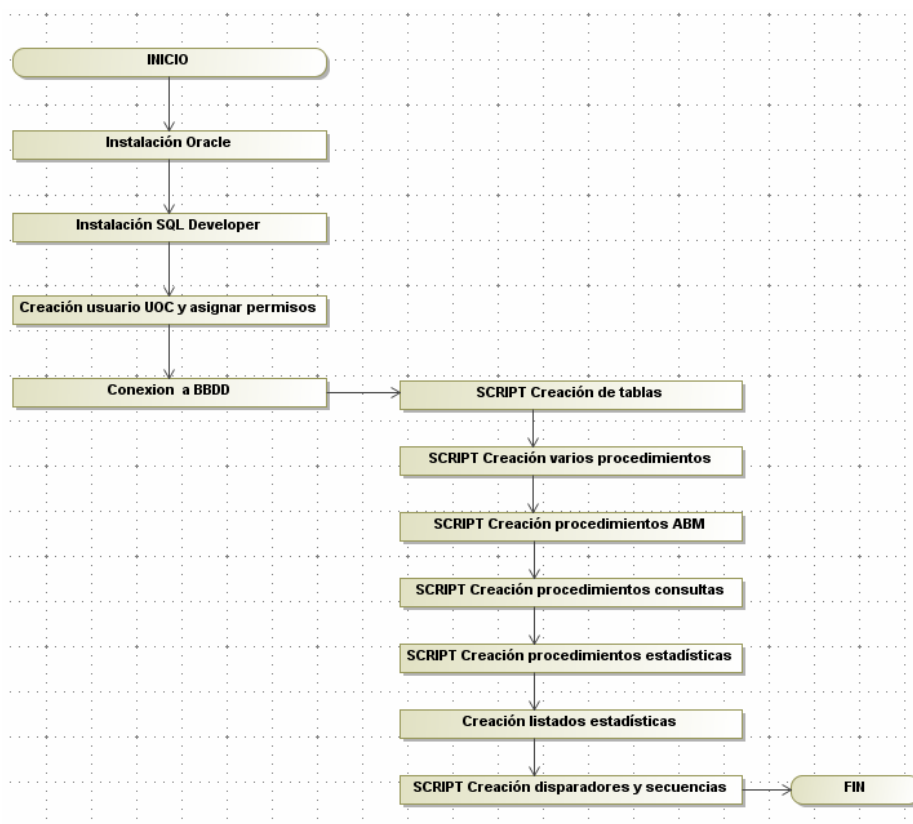


Fig.54.- Pasos fase de implementación

3.1.- Aspectos generales de los scripts

Todos los scripts se han desarrollado siguiendo los mismos patrones, así se consigue homogeneizar totalmente el sistema, permitiendo que la aplicación sea mantenida y mejora posteriormente con facilidad.

A continuación, explicaremos los aspectos generales de los scripts:

3.1.1.- Nomenclatura

Para que todos los scripts sigan la misma nomenclatura, se han puesto estas reglas:

NOMENCLATURA	
Codificación	Descripción
P_nombretabla_delete	Procedimiento que realiza la baja de un registro de la tabla
P_nombretabla_insert	Procedimiento que realiza el insert de un registro de la tabla
P_nombretabla_update	Procedimiento que realiza la modificación de un registro de la tabla
P_estadistica_num	Procedimiento que genera la estadística indicada en el num.
Tr_nombretabla_delete	Trigger que se ejecuta al realizar un delete de la tabla
Tr_nombretabla_update	Trigger que se ejecuta al realizar una actualización de la tabla
Tr_nombretabla_insert	Trigger que se ejecuta al realizar una inserción en la tabla
P_nombre	Procedimiento general
SEQ_nombreCampo	Secuencia para el campo indicado en el nombreCampo

Fig.55.- Nomenclaturas para los objetos

3.1.2.- Cabecera

En la parte superior de cada procedimiento encontraremos la cabecera del mismo con la información relativa al fragmento de código que habrá inmediatamente debajo de la misma.

La cabecera describe lo que hace el procedimiento o fragmento de código, especifica y explica los tipos y valores posibles de cada parámetro de entrada y salida, indicando cuales son los valores de retorno que se esperan después de su ejecución, y por último un ejemplo de la llamada del mismo procedimiento.

Hay un apartado en la cabecera que permite añadir un registro de fecha/usuario/acción cada vez que un usuario cambie el procedimiento, de manera que sea posible un control de versiones por fecha. Así, si el procedimiento falla, se verá quien fue el último que lo modificó y el motivo.

De manera gráfica, una cabecera tendría este aspecto:

```

/*****
Fecha de creación:          06/04/2012

Usuario de la creación:    UOC

Nombre del procedimiento:  p_contador_insert

Funcionalidad:            Inserta un registro en la tabla CONTADOR

Parámetros de entrada:    (tabla)
-----
NOMBRE          TIPO          VALOR EJEMPLO
DIRECCION      STRING        'Pza.España'
LOCALIDAD      STRING        'Madrid'
PAIS           STRING        'España'
ACUMULADOR     NUMBER        0
MODELO         STRING        'R6ym'

Parámetros de salida:     (rsp)
-----
NOMBRE          TIPO          VALOR EJEMPLO
rsp1           STRING        OK

Valor de retorno:        La salida será OK si la actualización del log se ha realizado correctamente.
En caso de error se retorna: 'ERROR: número del error'

Valores de los errores:
1-- Campos nulos
2-- Clave de la tabla duplicada
3-- Tipo de dato no esperado
4-- Error desconocido
5-- Modelo de contador no encontrado

Ejemplo de uso:         p_contador_insert('Pza.España','Madrid','España',0, 'R6ym', rsp1);

MODIFICACIONES POSTERIORES:
*****/
```

Fig.56.- Cabecera de un procedimiento

3.1.3.- Tratamiento excepciones

El tratamiento de excepciones es un factor fundamental para mantener la integridad de los datos del sistema. También permite saber si una transacción o conjunto de instrucciones ha dado un resultado esperado o por el contrario, si ha producido algún error, sabiendo la causa.

Las excepciones se pueden definir o bien de forma manual o automáticas.

- De forma manual, es el programador el que crea una regla específica para determinadas circunstancias que desee controlar.
- De forma automática son las reglas comunes heredadas de toda base de datos relacional. Por ejemplo, duplicidad de clave primaria, la existencia de las claves foráneas...

Estas excepciones al producirse, se puede mostrar el error específico gracias a la función SQLERRM, que retorna un mensaje de error asociado a la excepción producida más recientemente.

El sistema de control energético cuando ejecuta una determinada instancia que pasa todos los posibles filtros de error sin generar ninguna excepción, realiza un COMMIT, que confirma la transacción dejando los cambios de manera persistente.

Por el contrario, si se produce algún error, se puede tratar en el apartado de EXCEPTION del procedimiento y realizar un ROLLBACK deshaciendo los cambios producidos desde la última transacción.

Ejemplo de detección de errores de forma manual:

```
-- Comprobación de los posibles errores
-- Atributos que no pueden ser nulos, que no lo sean
IF (COD_CONTADOR IS NULL) OR (MODELO IS NULL) THEN
    RAISE campos_nulos;
END IF

-- Atributos que son clave foránea existan en la maestra
SELECT COUNT(*)
INTO num_modelos
FROM MODELO_CONTADOR
WHERE (Modelo = MODELO);

IF (num_modelos=0) THEN
    RAISE noexistemodelo;
END IF;
```

Fig.57.- Tratamiento de excepciones de forma manual

El tratamiento de excepciones para la detección de errores podría ser la siguiente:

```
EXCEPTION
-- Errores debidos a los atributos NOT NULL
WHEN campos_nulos THEN
    -- Se desechan los cambios que se hayan podido producir
    ROLLBACK;
    -- Se registra en la tabla de LOG
    rsp:= 'ERROR: Hay campos con valor NULL que no se pueden almacenar.';
    param_entrada:= NOMBRE|CONTADOR|DIRECCION|LOCALIDAD|PAIS|ACUMULADOR|MODELO
    -- Llamamos al procedimiento que inserta en la tabla de LOG
    p_log_insert('P_contador_insert', param_entrada, rsp);
    DBMS_OUTPUT.PUT_LINE(rsp);
    COMMIT;
-- Error debido a que no existe el modelo en la tabla de tipos de modelos de contadores
WHEN noexstemodelo THEN
    -- Se desechan los cambios que se hayan podido producir
    ROLLBACK;
    -- Se registra en la tabla de LOG
    rsp:= 'ERROR: No existe el modelo en la tabla modelo_contador.';
    param_entrada:= NOMBRE|CONTADOR|DIRECCION|LOCALIDAD|PAIS|ACUMULADOR|MODELO
    -- Llamamos al procedimiento que inserta en la tabla de LOG
    p_log_insert('P_contador_insert', param_entrada, rsp);
    DBMS_OUTPUT.PUT_LINE(rsp);
    COMMIT;
-- Errores genéricos
WHEN OTHERS THEN
    -- Se desechan los cambios que se hayan podido producir
    ROLLBACK;
    -- Se registra en la tabla de LOG
    rsp:= 'ERROR: 'N|| SUBSTR(SQLERRM, 1, 200);
    param_entrada:= NOMBRE|CONTADOR|DIRECCION|LOCALIDAD|PAIS|ACUMULADOR|MODELO
    -- Llamamos al procedimiento que inserta en la tabla de LOG
    p_log_insert('P_contador_insert', param_entrada, rsp);
    DBMS_OUTPUT.PUT_LINE(rsp);
    COMMIT;
```

Fig.58.- Control de las excepciones

3.1.4.- Tratamiento de datos de entrada

Todos los procedimientos que trabajan con parámetros de entrada de tipo cadena de caracteres disponen de un tratamiento que convierten a mayúsculas todos los datos y elimina los espacios en blanco de inicio o final de la cadena.

Aunque los requerimientos no lo especificaban, este tratamiento de datos tiene la ventaja de cuando se realiza una consulta referenciada a estos atributos de tipo string no hay problemas de tipo case sensitive. Así, en todo el entorno del proyecto se recomienda utilizar mayúsculas para todo tratamiento de datos de tipo string.

Un ejemplo del tratamiento de datos de entrada sería:

-- Los datos de entrada le quitamos los espacios en blanco y convertimos a mayúsculas

```
DIRECCION := UPPER(TRIM(DIRECCION));
PAIS      := UPPER(TRIM(PAIS));
LOCALIDAD := UPPER(TRIM(LOCALIDAD));
MODELO   := UPPER(TRIM(MODELO));
```


3.1.5.- Almacenamiento en el log

Uno de los requerimientos de la metodología solicitados por el cliente consistía en almacenar todas las llamadas a procedimientos que se realizaran junto con los parámetros que se utilizaran de entrada y de salida.

Para implementar este requerimiento se ha optado por insertar un registro en la tabla LOG.

Antes de registrar la llamada al procedimiento hay que tener en cuenta que si esta llamada ha sido errónea, habrá que realizar un ROLLBACK de la transacción para posteriormente poder realizar un COMMIT de la tabla LOG.

Además, hay que preparar los datos para insertar en el LOG: los parámetros de entrada del procediendo y el resultado de salida en la variable rsp.

El resto de atributos a insertar son el nombre del procedimiento (depende del procedimiento que se haya ejecutado), la fecha y la hora (función sysdate) y el usuario que haya realizado la acción.

3.1.6.- Optimización de consultas

Se ha tenido en cuenta el álgebra relacional (unión, diferencia, intersección, producto cartesiano, proyección, selección y combinación natural) para elaborar las consultas de una manera mas óptima y minimizando el coste total, en tiempo, de la consulta SQL.

Para obtener este coste mínimo se ha seguido la estrategia siguiente:

- Ejecutar las operaciones de álgebra relacional que reducen la cardinalidad de los resultados intermedios.
- Atrasar al máximo la ejecución de las operaciones que aumenten la cardinalidad de los resultados intermedios.

3.2.- Aspectos específicos de los scripts

3.2.1.- Descripción de los scripts específicos

Este apartado da respuesta a qué hace cada ficheros y como lo hace, pero sin entrar en detalles de programación excesivamente concretos, pues están comentados en los mismos archivos.

1.- En el script "1_creacion_tablas_secuencias.sql" se generarán todas las tablas y secuencias vinculadas a las claves empleadas para estas tablas. Este proceso consiste en coger las tablas descritas en el diseño anterior y codificarlas mediante el lenguaje de programación de Oracle.

2.- En el script "2_creacion_procedimientos_ABM.sql" se han incluido los procedimientos ABM de cada una de las tablas creadas en el punto 1.

3.- En el script "3_creacion_vistas_materializadas.sql" se han incluido las vistas materializadas que son objetos necesarios para evitar errores generados por oracle,

Rosana Marín Berraondo

al saltar un trigger y seleccionar información de la misma tabla que hace saltar el trigger en este mismo (o en un procedimiento llamado desde él).

4.- En el script "4_creacion_procedimientos_estadisticas.sql" se han incluido los procedimientos que actualizan las tablas que almacenan las estadísticas del requisito número 7.

5.- En el script "5_creacion_procedimiento_estadisticas.sql" se han incluido los procedimientos para la realización de las estadísticas. R7.

6.- En el script "6_creacion_triggers.sql" se han incluido los triggers de la aplicación necesarios para la actualización de los datos estadísticos.

7.- En el script "7_creacion_procedimientos_consultas.sql" se han incluido los procedimientos que crean las consultas del requisito 6.

8.- En el script "8_creacion_procedimientos_listados_estadisticas.sql" se han incluido los listados que muestran los datos de las estadísticas.

9.- A continuación se ha creado el script "9_relleno_tablas.sql" para rellenar las tablas y poder realizar pruebas. Este fichero contiene principalmente llamadas a los procedimientos ABM creados en el punto 2.

10.- En el script "10_borrado_datos.sql" se incluyen las sentencias para borrar todos los datos de las tablas e inicializar de nuevo las secuencias.

11.- En el script "11_inicializa_bd.sql" se han incluido el borrado total de todos los objetos creados en la base de datos.

12.- En el script "12_test_listados_consultas.sql" se hacen las pruebas de los listados de las consultas.

13.- En el script "13_test_listados_estadisticas.sql" se hacen las pruebas de los listados de las estadísticas.

3.2.2.- Script de creación de tablas

El archivo 1_creacion_tablas_secuencias.sql es el encargado de crear las tablas y las secuencias vinculadas a las claves empleadas en estas tablas. Todas las tablas siguen exactamente la misma estructura que la que especifica el modelo lógico.

Se ha optado por utilizar una clave artificial autoincrementada para los campos PK identificativos de tipo numérico que puedan existir en las tablas. Se ha determinado por comodidad, y así evitar posibles futuros problemas.

Este fichero también incluye las instrucciones para la creación de la tabla log y de las tablas del módulo estadístico. En resumen, este fichero crea 20 tablas que serán todas las necesarias para almacenar la información del sistema de control energético.

Ejemplo de creación de tabla de contadores:

```

1 CREATE TABLE CONTADOR
2 (
3   COD_CONTADOR NUMBER(12, 0) NOT NULL
4   , DIRECCION VARCHAR2(40)
5   , LOCALIDAD VARCHAR2(15)
6   , PAIS VARCHAR2(15)
7   , ACUMULADOR NUMBER(38, 2)
8   , MODELO VARCHAR2(20) NOT NULL
9   , CONSTRAINT CONTADOR_PK PRIMARY KEY
10  (
11    COD_CONTADOR
12  )
13  ENABLE
14 );
15

```

Fig.59.- Script de creación de tabla CONTADOR

Ejemplo de creación de la secuencia del código de contador:

```
CREATE SEQUENCE SEQ_COD_CONTADOR INCREMENT BY 1 START WITH 1 MAXVALUE 999999999999 MINVALUE 1;
```

Fig.60.- Script de creación de secuencia SEQ_COD_CONTADOR

Resumen de los objetos creados en este script:

Objetos de 1_creacion_tablas_secuencias.sql		
TIPO	OBJETO	DESCRIPCION
Tabla	CONTADOR	Almacena los contadores
Tabla	CENTRAL_PRODUCCION	Almacena las centrales de producción
Tabla	LINEA_COMUNICA	Almacena las líneas de comunicación
Tabla	CLIENTE	Almacena los clientes
Tabla	CENTRAL_DISTRIBUCION	Almacena las centrales de distribución
Tabla	MODELO_CONTADOR	Almacena los modelos de contador
Tabla	HIST_LECTURA	Almacena el histórico de lecturas de los contadores
Tabla	HIST_ENERGIA_EMITIDA	Almacena el histórico de la energía emitida por las centrales de distribución
Tabla	HIST_ENERGIA_PRODUCIDA	Almacena el histórico de la energía producida por las centrales de producción
Tabla	TIPO_CENTRAL	Almacena los tipos de centrales de producción existentes
Tabla	LOG	Almacena el log de los procedimientos
Tabla	CONTRATO	Almacena los contratos, relacionando los clientes con los contadores
Tabla	CONTADOR_CENTRAL	Almacena la relación entre los contadores y las centrales de distribución
Tabla	E1	Almacena los requisitos de las estadísticas, punto 1
Tabla	E2	Almacena los requisitos de las estadísticas, punto 2
Tabla	E3	Almacena los requisitos de las estadísticas, punto 3
Tabla	E4	Almacena los requisitos de las estadísticas, punto 4
Tabla	E5	Almacena los requisitos de las estadísticas, punto 5
Tabla	E6	Almacena los requisitos de las estadísticas, punto 6
Tabla	E7	Almacena los requisitos de las estadísticas, punto 7
Secuencia	SEQ_COD_CONTADOR	Secuencia para la PK de la tabla que almacena los contadores. Campo COD_CONTADOR.
Secuencia	SEQ_COD_CENTRAL_PROD	Secuencia para la PK de la tabla CENTRAL_PRODUCCION. (COD_CENTRAL)
Secuencia	SEQ_COD_LINEA	Secuencia para la PK de la tabla LINEA_COMUNICA. (COD_LINEA)
Secuencia	SEQ_COD_CENTRAL_DIST	Secuencia para la PK de la tabla CENTRAL_DISTRIBUCION (COD_CENTRAL)
Secuencia	SEQ_TIPO_CENTRAL	Secuencia para la PK de la tabla TIPO_CENTRAL (TIPO_CENTRAL)
Secuencia	SEQ_COD_CONTRATO	Secuencia para la PK de la tabla CONTRATO (COD_CONTRATO)

Fig.61.- Objetos de 1_creacion_tablas_secuencias.sql

3.2.3.- Script de procedimientos ABM

Rosana Marín Berraondo

En este script se crean los procedimientos de Alta, Baja y Modificación de las tablas del proyecto. (el archivo se llama "2_creacion_procedimientos_ABM.sql")

Tomaremos como ejemplo el ABM de la tabla contadores, para describir la funcionalidad de estos procedimientos, ya que para todas las demás tablas el funcionamiento es idéntico.

Alta:

Cuando se da de alta un registro en una de las tablas, los pasos a seguir son:

- Comprobación de datos de entrada
- Comprobación de nulos
- Comprobación de claves foráneas
- Generación de la clave primaria (si esta es autonumérica creada por una secuencia)
- Inserción en la tabla
- Control de excepciones

Ejemplo de alta de un contador:

```
|/*****
Fecha de creación:          06/04/2012
Usuario de la creación:    UOC
Nombre del procedimiento:  p_contador_insert
Funcionalidad:             Inserta un registro en la tabla CONTADOR
Parámetros de entrada:    (tabla)
-----
NOMBRE      TIPO      VALOR EJEMPLO
DIRECCION   STRING    'Pza.España'
LOCALIDAD   STRING    'Madrid'
PAIS        STRING    'España'
ACUMULADOR  NUMBER    0
MODELO      STRING    'R6ym'
-----
Parámetros de salida:     (rsp)
-----
NOMBRE      TIPO      VALOR EJEMPLO
rsp1        STRING    OK
Valor de retorno:         La salida será OK si la actualización del log se ha realizado corre
                          En caso de error se retorna: 'ERROR: número del error'
Valores de los errores:
1-- Campos nulos
2-- Clave de la tabla duplicada
3-- Tipo de dato no esperado
4-- Error desconocido
5-- Modelo de contador no encontrado
Ejemplo de uso:           p_contador_insert('Pza.España','Madrid','España',0, 'R6ym', rsp1);
```

Fig.62.- Procedimiento de alta de un contador

Baja:

La baja de determinados registros de las tablas no suelen ser triviales, especialmente si la clave de ese registro se utiliza como clave foránea en otra tabla. Por lo tanto, hay que tener especial cuidado con las bajas.

Los pasos a seguir serían:

- Comprobación de la existencia del registro a borrar
- Comprobación de dependencias de la clave principal
- Delete en la tabla

- Control de excepciones

Ejemplo de baja de un contador:

```

/*****
Fecha de creación:      06/04/2012

Usuario de la creación:  UOC

Nombre del procedimiento:  p_modelo_contador_delete

Funcionalidad:          Borra un registro en la tabla MODELO_CONTADOR

Parámetros de entrada:  (tabla)
-----
NOMBRE      TIPO      VALOR EJEMPLO
MODELO      STRING     'R7g86'

Parámetros de salida:   (rsp)
-----
NOMBRE      TIPO      VALOR EJEMPLO
rsp1        STRING     OK

Valor de retorno:       La salida será OK si la actualización del log se ha realizado correctamente.
                        En caso de error se retorna: 'ERROR: número del error'

Valores de los errores:
2-- Tipo de dato no esperado
4-- Error desconocido
1-- Campos nulos
3-- Modelo encontrado en la tabla contadores

Ejemplo de uso:         p_modelo_contador_delete('R7g86');

MODIFICACIONES POSTERIORES:

*****/
CREATE OR REPLACE PROCEDURE P_MODELO_CONTADOR_DELETE

```

Fig.63.- Procedimiento de baja de un modelo de contador

Modificación:

La modificación de un registro en la tabla supone el cambio de alguno de sus datos. Los únicos datos que no se pueden modificar son las claves primarias. Para las modificaciones hay que tener especial cuidado con los nulos y con las claves foráneas.

Los pasos a seguir serán:

- Comprobación de la existencia del registro a modificar
- Comprobación de que si se quiere modificar algún atributo foráneo, éste exista en su tabla maestra.
- Modificación del registro
- Control de excepciones

Ejemplo de la modificación de un contador:

```

/*****
Fecha de creación:      06/04/2012
Usuario de la creación:  UOC
Nombre del procedimiento: p_contador_update
Funcionalidad:          Actualiza un registro en la tabla CONTADOR

Parámetros de entrada:  (tabla)
-----
NOMBRE      TIPO      VALOR EJEMPLO
COD_CONTADOR  NUMBER  1
DIRECCION   STRING  'Pza.España'
LOCALIDAD   STRING  'Madrid'
PAIS        STRING  'España'
ACUMULADOR  NUMBER  0
MODELO      STRING  'R6ym'

Parámetros de salida:   (rsp)
-----
NOMBRE      TIPO      VALOR EJEMPLO
rsp1        STRING    OK

Valor de retorno:       La salida será OK si la actualización del log se ha realizado correctamente.
                        En caso de error se retorna: 'ERROR: número del error'

Valores de los errores:
5-- Campos nulos
1-- Clave de la tabla duplicada
2-- Tipo de dato no esperado
4-- Error desconocido
3-- contador no encontrado

Ejemplo de uso:         p_contador_update(1, 'Pza.España','Madrid','España',0, 'R6ym', rsp1);

MODIFICACIONES POSTERIORES:

```

Fig.64.- Procedimiento de modificación de un contador

Resumen de los objetos creados en este script:

2_creacion_procedimientos_ABM.sql				
Nombre	Parámetros de entrada	Parámetros de salida	Descripción	Otros procedimientos que ejecuta
P_central_distribucion_insert	Direccion Localidad Pais Cod_linea Energia_max Ultima_inspeccion	RSP	Alta de una central de distribución	P_log_insert
P_central_distribucion_update	Cod_central Direccion Localidad Pais Cod_linea Energia_max Ultima_inspeccion	RSP	Actualización de una central de distribución	P_log_insert
P_central_distribucion_delete	Cod_central	RSP	Borrado de una central de distribución	P_log_insert
P_central_produccion_insert	Direccion Localidad Pais Energia_max Ultima_inspeccion Tipo_central Propiedad1 Propiedad2	RSP	Alta de una central de producción	P_log_insert
P_central_produccion_update	Cod_central Direccion Localidad Pais	RSP	Modificación de los datos de una central de	P_log_insert

UOC – Ingeniería técnica informática de Sistemas
TFC- Bases de datos relacionales

	Energia_max Ultima_inspeccion Tipo_central Propiedad1 Propiedad2		producción	
P_central_produccion_delete	Cod_central	RSP	Borrado de una central de producción	P_log_insert
P_cliente_insert	DNI_NIF Dirección Localidad Pais Correo Telefono Denominación Contacto	RSP	Alta de un cliente	P_log_insert
P_cliente_update	DNI_NIF Dirección Localidad Pais Correo Telefono Denominación Contacto	RSP	Modificación de los datos de un cliente	P_log_insert
P_cliente_delete	DNI_NIF	RSP	Borrado de un cliente	P_log_insert
P_contador_insert	Direccion Localidad Pais Acumulador Modelo	RSP	Inserción de un nuevo contador	P_log_insert
P_contador_update	Cod_contador Direccion Localidad Pais Acumulador Modelo	RSP	Modificación de los datos de un contador	P_log_insert
P_contador_delete	Cod_contador	RSP	Baja de un contador	P_log_insert
P_contador_central_insert	Cod_contador Cod_central	RSP	Alta de una relación entre un contador y una central de distribución	P_log_insert
P_contador_central_delete	Cod_contador Cod_central	RSP	Baja de una relación entre un contador y una central de distribución	P_log_insert
P_contrato_insert	DNI_NIF Cod_contador Fecha_alta Potencia	RSP	Alta de un contrato	P_log_insert
P_contrato_update	Cod_contrato DNI_NIF Cod_contador Fecha_alta Fecha_baja Potencia	RSP	Actualización de los datos de un contrato	P_log_insert
P_contrato_delete	Cod_contrato	RSP	Baja del registro del contrato	P_log_insert
P_hist_energia_emitida_insert	Cod_central Fecha Lectura	RSP	Alta de la energía emitida por una central de distribución en una fecha	P_log_insert

Rosana Marín Berraondo

P_hist_energia_emitida_update	Cod_central Fecha Lectura	RSP	Modificación de la lectura de una central en una fecha	P_log_insert
P_hist_energia_emitida_delete	Cod_central Fecha	RSP	Borrado de la lectura de una central en una fecha	P_log_insert
P_hist_energia_prod_insert	Cod_central Fecha Lectura	RSP	Alta de la lectura de energía producida en un mes por una central de producción	P_log_insert
P_hist_energia_prod_update	Cod_central Fecha Lectura	RSP	Modificación de la lectura de energía producida por una central de producción en un mes	P_log_insert
P_hist_energia_prod_delete	Cod_central Fecha	RSP	Borrado de la lectura de una central en una fecha	P_log_insert
P_hist_lectura_insert	Cod_contador Fecha Lectura	RSP	Inserción del consumo realizado por un contador en una fecha	P_log_insert
P_hist_lectura_update	Cod_contador Fecha Lectura	RSP	Modificación de la lectura de un contador en una fecha	P_log_insert
P_hist_lectura_delete	Cod_contador Fecha	RSP	Borrado de la lectura de un contador en una fecha	P_log_insert
P_linea_comunica_insert	Capacidad_max Cod_central	RSP	Alta de una línea de comunicación	P_log_insert
P_linea_comunica_update	Cod_linea Capacidad_max Cod_central	RSP	Modificación de una línea de comunicación	P_log_insert
P_linea_comunica_delete	Cod_linea	RSP	Borrado de una línea de comunicación	P_log_insert
P_log_insert	Procedimiento Param_entrada Param_salida	RSP	Alta de un registro en el LOG	
P_modelo_contador_insert	Empresa_fabricacion Anyo_fabricacion Denominación	RSP	Alta de un nuevo modelo de contador	P_log_insert
P_modelo_contador_update	Modelo Empresa_fabricacion Anyo_fabricacion Denominación	RSP	Modificación de los datos de un modelo de contador	P_log_insert
P_modelo_contador_delete	Modelo	RSP	Borrado de los datos de un modelo de contador	P_log_insert
P_tipo_central_insert	Descripción Propiedad1 Propiedad2	RSP	Alta de un nuevo tipo de central de producción	P_log_insert
P_tipo_central_update	Tipo_central Descripción	RSP	Modificación de un tipo de	P_log_insert

	Propiedad1 Propiedad2		central de producción	
P_tipo_central_delete	Tipo_central	RSP	Borrado de un tipo de central de producción	P_log_insert

Fig.65.- Objetos de 2_creacion_procedimientos_ABM.sql

NOTAS:

- La tabla CONTADOR_CENTRAL no tiene procedimiento de actualización porque tan solo tiene los campos COD_CONTADOR y COD_CENTRAL, ya que muestra la relación entre ambas. Si actualizamos la tabla, sólo se pueden actualizar las PKs, por lo tanto, se ve mas coherente realizar un delete y un insert.

3.2.4.- Script de creación de procedimientos estadísticos

En el archivo 5_creacion_procedimiento_estadisticas.sql se crea el procedimiento P_ESTADISTICAS que según el parámetro de entrada que recibe (nombre de tabla desde el cual su trigger ha llamado al procedimiento) llama a otros procedimientos que actualizan las tablas estadísticas.

El funcionamiento lo podemos ver con el ejemplo de la estadística 1:

Esta pide que dada una central de producción, nos devuelva el consumo de los contadores que dependen de la central. Por lo tanto, la tabla que almacena esta información (E1) se actualizará cuando el campo ACUMULADOR de la tabla CONTADOR se inserte / actualice / borre. El flujo es el siguiente:

- INSERT/ UPDATE/ DELETE en el campo ACUMULADOR de la tabla CONTADOR hace que salte el disparador TR_CONTADOR.
- El disparador TR_CONTADOR llama al procedimiento P_ESTADISTICAS informándole que la tabla que ha efectuado un cambio es CONTADOR.
- El procedimiento P_ESTADISTICAS llama al procedimiento P_E1_ACTUALIZA que es el que realiza la inserción en E1.

Ejemplo del procedimiento P_ESTADISTICAS:

```

PROCEDURE P_ESTADISTICAS
(
  TABLA IN VARCHAR2,
  PARAM IN VARCHAR2,
  RESP OUT VARCHAR2)
IS
  rsp VARCHAR2(25);
  resp_log VARCHAR2(25);
  cod_contador number;
  acumulador number;
  ll_pos number;
  fecha date;
  anio number;
BEGIN
  -- SEGUN EL VALOR QUE SE LE PASE AL PROCEDIMIENTO, ES DECIR, LA TABLA
  -- EL PROCEDIMIENTO LLAMARÁ A UNA U OTRA FUNCIÓN, PARA ACTUALIZAR CA:
  -- QUE DEPENDAN DE ESA TABLA

  CASE TABLA
  WHEN 'HIST_LECTURA' THEN
    ll_pos := INSTR(param, '|');

    anio:= to_number(substr(param, 1, ll_pos -1));
    cod_contador:= to_number(substr(param, ll_pos + 2, length(pa

    P_E2_ACTUALIZA(anio, cod_contador, rsp);
    P_E3_ACTUALIZA(rsp);
    P_E4_ACTUALIZA(anio, rsp);
    P_E6_ACTUALIZA(rsp);
    P_E7_ACTUALIZA(rsp);

  WHEN 'HIST_ENERGIA_PRODUCIDA' THEN
    anio := to_number(param);
    P_E5_ACTUALIZA(anio, rsp);

  WHEN 'CONTADOR' THEN
    ll_pos := INSTR(param, '|');

    cod_contador:= to_number(substr(param, 1, ll_pos -1));
    acumulador := to_number(substr(param, ll_pos + 2, length(par

    P_E1_ACTUALIZA(cod_contador, acumulador, rsp);

  ELSE resp:= 'ERROR:1- La tabla no genera actualización de estadis
END CASE;
  
```

Fig.66.- Procedimiento P_ESTADISTICAS

3.2.5.- Script de creación de triggers

En el script "6_creacion_triggers.sql" se crean los triggers necesarios para la carga de datos de las tablas de las estadísticas.

Así, los triggers que se generan son:

6_creacion_triggers.sql		
Nombre	Descripción	Otros procedimientos que ejecuta
TR_CONTADOR	Trigger que salta al modificar/borrar/insertar el campo acumulador de la tabla contador	P_log_insert
TR_HIST_LECTURA	Trigger que salta al modificar/borrar/insertar algún registro de la tabla hist_lectura	P_log_insert
TR_HIST_ENER_PRODUCIDA	Trigger que salta al modificar/borrar/insertar algún registro de la tabla hist_energia_producida	P_log_insert

Fig.67.- Objetos de 6_creacion_triggers.sql

3.2.6.- Script de creación de vistas materializadas

En el script "3_creacion_vistas_materializadas.sql" se crean las vistas materializadas necesarias para la selección de información en la carga de datos de las estadísticas.

Estos objetos son necesarios porque oracle lanza un error al intentar leer de las tablas que lanzan los triggers y creando vistas que lean de las tablas se obtienen los datos evitando el error de oracle.

3_creacion_vistas_materializadas.sql	
Nombre	Descripción
hist_lectura_copy	Vista materializada sobre la tabla hist_lectura
HIST_ENERGIA_PROD_copy	Vista materializada sobre la tabla hist_energia_producida

Fig.68.- Objetos de 3_creacion_vistas_materializadas.sql

3.2.7.- Script de creación de los procedimientos de las consultas

En el script "7_creacion_procedimientos_consultas.sql" se crean los procedimientos para generar los listados necesarios en el requisito 6 de los requerimientos.

Los procedimientos generados son:

7_creacion_procedimientos_consultas.sql		
Nombre	Descripción	Requisito que cumple
p_contadores_80_porcentaje_listado	Dada una ciudad y una fecha lista contadores cuyo consumo mensual supera 80% media.	R6 punto A
p_centrales_mas_energia	Lista las 10 centrales de distribución que distribuyen mas energía.	R6 punto B
p_lineas_mas_cargadas	Lista las 10 líneas de comunicación mas cargadas con relación a su capacidad.	R6 punto C
p_clientes_alta_disponibilidad	Lista los clientes con servicio de alta disponibilidad	R6 punto D
p_central_produccion_consumo	Dada una central de producción y un periodo de fechas da el consumo producido por sus contadores	R6 punto E
p_porcentaje_lecturas	Porcentaje de lecturas presenciales y telemáticas	R6 punto F
p_contadores_antiguedad	Listado de contadores con una antigüedad	R6 punto G

Fig.69.- Objetos de 7_creacion_procedimientos_consultas.sql

3.2.8.- Script de creación de los procedimientos de estadísticas

En el script "4_creacion_procedimientos_estadisticas.sql" se crean los procedimientos necesarios para la actualización de las tablas de las estadísticas. A estos procedimientos se les llama desde el procedimiento p_estadísticas según las tablas que haya que actualizar dependiendo del trigger que haya llamado a p_estadísticas.

4_creacion_procedimientos_estadisticas.sql		
Nombre	Descripción	Requisito que cumple
p_e1_actualiza	Inserta / Actualiza datos de la tabla E1	R7 punto A

p_e2_actualiza	Inserta / Actualiza datos de la tabla E2	R7 punto B
p_e3_actualiza	Inserta / Actualiza datos de la tabla E3	R7 punto C
p_e4_actualiza	Inserta / Actualiza datos de la tabla E4	R7 punto D
p_e5_actualiza	Inserta / Actualiza datos de la tabla E5	R7 punto E
p_e6_actualiza	Inserta / Actualiza datos de la tabla E6	R7 punto F
p_e7_actualiza	Inserta / Actualiza datos de la tabla E7	R7 punto G

Fig.70.- Objetos de 4_creacion_procedimientos_estadisticas.sql

3.2.9.- Script de listado de las estadísticas

En el script "8_creacion_procedimientos_listados_estadisticas.sql" se crean los procedimientos necesarios para listar el contenido de las tablas estadísticas (E1, E2...).

8_creacion_procedimientos_listados_estadisticas.sql		
Nombre	Descripción	Requisito que cumple
p_e1_listado	Lista los datos de la tabla E1	R7 punto A
p_e1_listado	Lista los datos de la tabla E2	R7 punto B
p_e1_listado	Lista los datos de la tabla E3	R7 punto C
p_e1_listado	Lista los datos de la tabla E4	R7 punto D
p_e1_listado	Lista los datos de la tabla E5	R7 punto E
p_e1_listado	Lista los datos de la tabla E6	R7 punto F
p_e1_listado	Lista los datos de la tabla E7	R7 punto G

Fig.71.- Objetos de 9_creacion_procedimientos_estadisticas.sql

3.2.10.- Script de carga de datos

En el archivo "9_relleno_tablas.sql" sirve para rellenar todas las tablas con un juego de pruebas.

Este fichero llama a los procedimientos de alta, con el objetivo de insertar los registros en cada una de las tablas. Si no existieran esos procedimientos, la inserción se haría con la sentencia INSERT de oracle.

Se recomienda que tras la ejecución de "9_relleno_tablas.sql" no se modifique la estructura de los objetos creados.

3.2.11.- Script de borrado de información

En el script "10_borrado_datos.sql" se borra el contenido de las tablas de la BBDD y se inicializan las secuencias.

3.2.12.- Script de borrado de objetos

En el script "11_inicializa_bd.sql" se borran todos los objetos creados para este proyecto.

3.2.13.- Script prueba listados consultas

En el script "12_test_listados_consultas.sql" se testean los listados de las consultas solicitadas en el requisito 6.

3.2.14.- Script prueba estadísticas

En el script "13_test_listados_estadisticas.sql" se testean los listados solicitados en el requisito 7.

4.- PLAN DE CONTINGENCIAS

En este apartado trataremos los principales problemas o hechos que podrían producirse a lo largo de la realización del proyecto y que impedirían la obtención de los objetivos o el cumplimiento de la planificación inicial. Exponemos además, la medida de prevención que se ha determinar para evitarlo o en el caso de que surja, la solución prevista.

Lista de riesgos y medidas asociadas:

- **Riesgo de pérdida de datos:** El equipo donde se está desarrollando el proyecto podría dejar de funcionar, y por lo tanto no poder trabajar alguna jornada o incluso no poder volver a trabajar en ese PC.

Medida: después de cada jornada diaria de trabajo se hace una copia de seguridad en un pen-drive o memoria externa. Así, los archivos no se pierden. En el caso de que el equipo no se pudiera recuperar hay un segundo equipo de backup, mas lento y menos productivo... pero permitiría seguir trabajando hasta solucionar el problema del principal.

- **Motivos personales:** podría suceder que el estudiante se pusiera enfermo o le surgiera algún imprevisto personal, laboral o familiar que le impidiera destinar tantas jornadas o todas las horas de la jornada previstos al proyecto.

Medida: la planificación del proyecto se ha realizado teniendo en cuenta éstos factores. Así, se ha indicado que se trabajará de lunes a viernes, teniendo el fin de semana de descanso. Si ocurriera alguna de éstas situaciones que ralentizara el desarrollo establecido del proyecto, el estudiante podría aprovechar éstos días de descanso para recuperar estas jornadas u horas perdidas, y así no dejar de cumplir con la planificación.

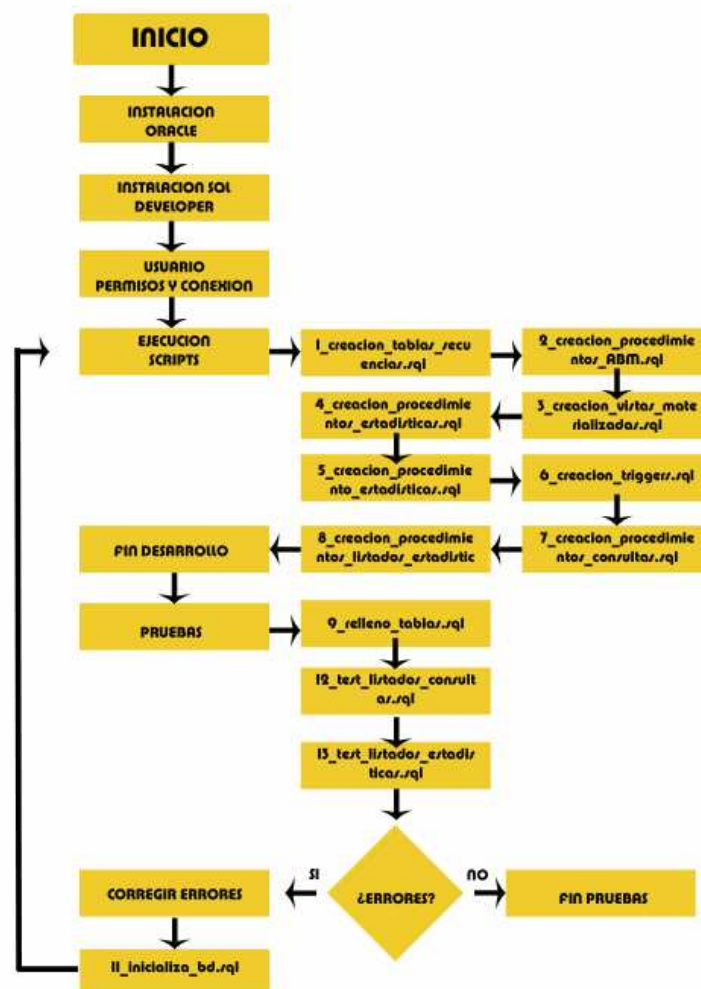
Además, existe la posibilidad de trabajar mas horas en cada jornada, no es un tiempo cerrado, sino flexible y abierto a posibles compensaciones entre jornadas. Así, si un día se termina la tarea prevista en una hora, en vez de tres, el estudiante puede optar por adelantar la tarea de la jornada siguiente, o bien, aprovechar esas dos horas en una tarea de otro día en la que rinda menos.

- **Riesgo de no cumplir la planificación:** si aún trabajando los festivos no se cumpliera con la planificación se debería de hablar con el cliente para buscar una solución conjunta.

5.- PLAN DE PRUEBAS

5.1.- Metodología a seguir

En el siguiente esquema se pueden ver los pasos a seguir:



5.2.- Orden de los scripts

En el documento anexo /producto/instrucciones.pdf se explica cómo proceder partiendo de cero a instalar y configurar esta base de datos de manera textual.

5.3.- Test de generación de objetos

1_creacion_tablas_secuencias.sql

Tras la ejecución de este primer script en SQL Developer se obtiene este log:

```
table E1 creado.  
table E2 creado.  
table E3 creado.  
table E4 creado.  
table E5 creado.  
table E6 creado.  
table E7 creado.  
table HIST_ENERGIA_EMITIDA creado.  
table HIST_ENERGIA_PRODUCIDA creado.  
table HIST_LECTURA creado.  
table LINEA_COMUNICA creado.  
table LOG creado.  
table MODELO_CONTADOR creado.  
table TIPO_CENTRAL creado.  
sequence SEQ_COD_CENTRAL_DIST creado.  
sequence SEQ_COD_CENTRAL_PROD creado.  
sequence SEQ_COD_CONTADOR creado.  
sequence SEQ_COD_CONTRATO creado.  
sequence SEQ_COD_LINEA creado.  
sequence SEQ_TIPO_CENTRAL creado.
```

Fig.72.- Log de ejecución de creación_tablas_secuencias

En el árbol de objetos aparecerán las tablas y secuencias generadas:

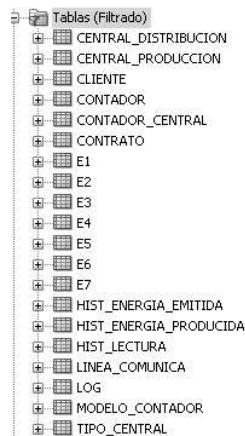


Fig.72.- Tablas generadas



Fig.73.- Secuencias generadas

También se puede comprobar consultando el diccionario de datos:

```
SELECT TABLE_NAME  
FROM USER_TABLES;
```

```
SELECT SEQUENCE_NAME  
FROM ALL_SEQUENCES  
where SEQUENCE_OWNER = 'UOC';
```

Ambas sentencias nos devolverán los nombres de los objetos creados.

2_creacion_procedimientos_ABM.sql

Tras la ejecución de este script, donde se han generado todos los procedimientos de manipulación de datos en las tablas, se obtiene el siguiente log en sqldeveloper:

```
PROCEDURE P_TIPO_CENTRAL_UPDATE compilado
PROCEDURE P_TIPO_CENTRAL_DELETE compilado
PROCEDURE P_MODELO_CONTADOR_UPDATE compilado
PROCEDURE P_MODELO_CONTADOR_DELETE compilado
PROCEDURE P_LINEA_COMUNICA_UPDATE compilado
PROCEDURE P_LINEA_COMUNICA_DELETE compilado
PROCEDURE P_HIST_LECTURA_UPDATE compilado
PROCEDURE P_HIST_LECTURA_DELETE compilado
PROCEDURE P_CENTRAL_DISTRIBUCION_INSERT compilado
PROCEDURE P_CENTRAL_DISTRIBUCION_UPDATE compilado
PROCEDURE P_CENTRAL_DISTRIBUCION_DELETE compilado
PROCEDURE P_CENTRAL_PRODUCCION_UPDATE compilado
PROCEDURE P_CENTRAL_PRODUCCION_DELETE compilado
PROCEDURE P_CLIENTE_UPDATE compilado
PROCEDURE P_CLIENTE_DELETE compilado
PROCEDURE P_CONTADOR_UPDATE compilado
PROCEDURE P_CONTADOR_DELETE compilado
PROCEDURE P_CONTADOR_CENTRAL_DELETE compilado
PROCEDURE P_CONTRATO_UPDATE compilado
PROCEDURE P_CONTRATO_DELETE compilado
PROCEDURE P_HIST_ENERGIA_EMITIDA_UPDATE compilado
PROCEDURE P_HIST_ENERGIA_EMITIDA_DELETE compilado
PROCEDURE P_HIST_ENERGIA_PROD_UPDATE compilado
PROCEDURE P_HIST_ENERGIA_PROD_DELETE compilado
```

Fig.74.- Log de creación de los procedimientos ABM

En el árbol de objetos aparecerán los procedimientos generados:

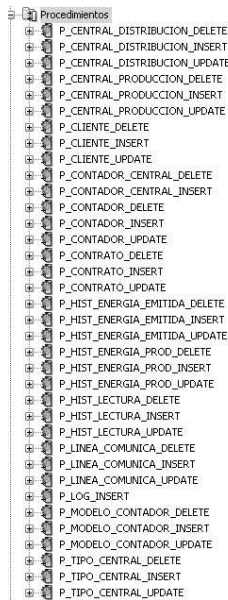


Fig.75.- Procedimientos generados

Consultado el diccionario de datos, obtenemos también el nombre de los objetos generados:

```
SELECT OBJECT_NAME
FROM ALL PROCEDURES
where OWNER = 'UOC';
```

3_creacion_vistas_materializadas.sql

Tras la ejecución de este script se obtiene el siguiente log en sqldeveloper:

```
materialized view HIST_LECTURA_COPY creado.
materialized view HIST_ENERGIA_PROD_COPY creado.
```

Fig.76.- Log de creación de vistas materializadas

En el árbol de objetos del SQL Developer éstos objetos aparecerán:

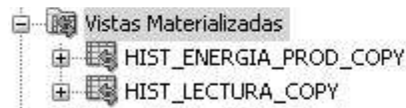


Fig.77.- Vistas materializadas generadas

La consulta del diccionario de datos:

```
select TABLE_NAME  
from USER_TABLES;
```

Para el caso de las vistas materializadas, en el diccionario de datos se ven como una tabla mas.

4_creacion_procedimientos_estadisticas.sql

Tras la ejecución de este script se obtiene el siguiente log en sqldeveloper:

```
'PROCEDURE p_e1_actualiza compilado  
'PROCEDURE p_e2_actualiza compilado  
'PROCEDURE p_e3_actualiza compilado  
'PROCEDURE p_e4_actualiza compilado  
'PROCEDURE p_e5_actualiza compilado  
'PROCEDURE p_e6_actualiza compilado  
'PROCEDURE p_e7_actualiza compilado
```

Fig.78.- Log de creación de los procedimientos de actualización de estadísticas

En el árbol de objetos del SQL Developer, en la sección de procedimientos, se pueden observar los nuevos objetos creados:

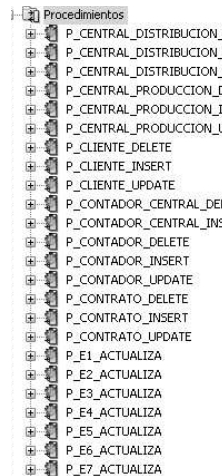


Fig.79.- Procedimientos generados

La consulta del diccionario de datos:

```
SELECT OBJECT_NAME  
FROM ALL PROCEDURES  
where OWNER = 'UOC'  
and object_name like 'P_E%';
```

5_creacion_procedimiento_estadisticas.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
PROCEDURE P_ESTADISTICAS compilado
```

Fig.80.- Log de creación del procedimiento de estadísticas

En el árbol de los objetos podemos observar que se ha creado nuestro nuevo procedimiento:

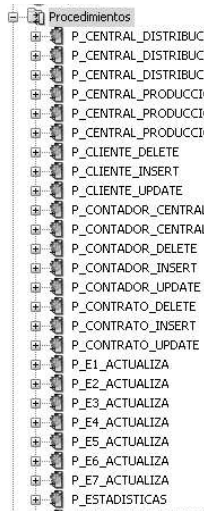


Fig.81.- Procedimiento P_ESTADISTICAS generado

6_creacion_triggers.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
TRIGGER TR_CONTADOR compilado  
TRIGGER TR_HIST_LECTURA compilado  
TRIGGER TR_HIST_ENER_PRODUCIDA compilado
```

Fig.82.- Log de creación de disparadores

En el árbol de objetos podemos ver los triggers creados:

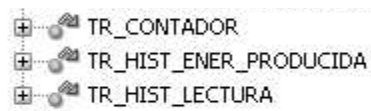


Fig.83.- Disparadores generados con el script

La consulta del diccionario de datos es:

```
SELECT *  
FROM ALL_TRIGGERS  
where OWNER = 'UOC';
```

7_creacion_procedimientos_consultas.sql

Tras la ejecución de este script el log del SQL Developer nos muestra:

```
PROCEDURE p_contadores_80_porcentaje_listado compilado  
PROCEDURE p_centrales_mas_energia compilado  
PROCEDURE p_lineas_mas_cargadas compilado  
PROCEDURE p_clientes_alta_disponibilidad compilado  
PROCEDURE p_central_produccion_consumo compilado  
PROCEDURE p_porcentaje_lecturas compilado  
PROCEDURE p_contadores_antiguedad compilado
```

Fig.84.- Log de creación de procedimientos de consulta

En el árbol de los procedimientos se puede observar que han aparecido estos nuevos objetos.

8_creacion_procedimientos_listados_estadisticas.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
PROCEDURE p_e1_listado compilado
PROCEDURE p_e2_listado compilado
PROCEDURE p_e3_listado compilado
PROCEDURE p_e4_listado compilado
PROCEDURE p_e5_listado compilado
PROCEDURE p_e6_listado compilado
PROCEDURE p_e7_listado compilado
```

Fig.85.- Log de creación de procedimientos de listado de estadísticas

En el árbol de los procedimientos se puede observar que han aparecido estos nuevos objetos.

9_relleno_tablas.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
bloque anónimo terminado
```

Fig.86.- Log de relleno de datos de las tablas

Tras la ejecución de éste script, podemos ver que ha funcionado correctamente, porque cada una de las tablas contiene información. Así, realizando un *SELECT * FROM tabla* veremos los datos que se han insertado en cada una de las tablas.

12_test_listados_consultas.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
-----
dentificador de la línea: 1
arga de la línea: 10428
nergía a ampliar: 46222
-----
dentificador de la línea: 9
arga de la línea: 2533,3333333333333333333333333333333333333333333
nergía a ampliar: 44183
-----
dentificador de la línea: 6
arga de la línea: 1532
nergía a ampliar: 56779
-----
dentificador de la línea: 5
arga de la línea: 1514,4
nergía a ampliar: 62194
-----
dentificador de la línea: 8
arga de la línea: 1336,5
nergía a ampliar: 58548
-----
```

Fig.87.- Log del listado de consultas

El contenido completo de la salida por pantalla se puede ver en el anexo situado al final de la memoria.

13_test_listados_estadisticas.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
-----  
Dni/Nif del cliente: B5555555  
Consumo medio 50,56  
-----  
Dni/Nif del cliente: B6666666  
Consumo medio 26,5  
-----  
Dni/Nif del cliente: 11111111A  
Consumo medio 26,28  
-----  
Dni/Nif del cliente: 4444444Z  
Consumo medio 37  
-----  
Dni/Nif del cliente: 7777777E  
Consumo medio 31,79  
-----
```

Fig.88.- Log del listado de estadísticas

El contenido completo de la salida por pantalla se puede leer en el anexo situado al final de la memoria.

10_borrado_datos.sql

Tras la ejecución de este script el log del SQL Developer nos devuelve:

```
0 filas eliminado  
.5 filas eliminado  
0 filas eliminado  
0 filas eliminado  
0 filas eliminado  
20 filas eliminado  
40 filas eliminado  
20 filas eliminado  
0 filas eliminado  
5 filas eliminado  
6 filas eliminado  
0 filas eliminado  
sequence SEQ_COD_CENTRAL_DIST borrado.  
sequence SEQ_COD_CENTRAL_PROD borrado.  
sequence SEQ_COD_CONTADOR borrado.  
sequence SEQ_COD_CONTRATO borrado.  
sequence SEQ_COD_LINEA borrado.  
sequence SEQ_TIPO_CENTRAL borrado.  
sequence SEQ_COD_CENTRAL_DIST creado.  
sequence SEQ_COD_CENTRAL_PROD creado.  
sequence SEQ_COD_CONTADOR creado.  
sequence SEQ_COD_CONTRATO creado.  
sequence SEQ_COD_LINEA creado.  
sequence SEQ_TIPO_CENTRAL creado.
```

Fig.89.- Log del borrado de datos

El contenido completo de la salida por pantalla se puede leer en el documento anexo al final de la memoria.

11_inicializa_bd.sql

Tras la ejecución de este script se borran todos los objetos de la base de datos.

Podemos verlo en este pantallazo:

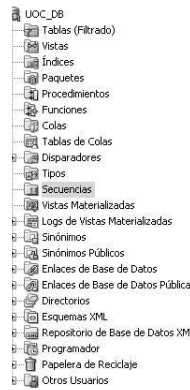


Fig.90.- Ausencia de objetos en la base de datos

5.4.- Plan de pruebas con ERROR

A lo largo de la implementación del producto nos hemos encontrado con situaciones en las que las pruebas devolvían error, pasamos a comentar alguna de ellas:

- En la creación del primer procedimiento de inserción, lo generamos en la base de datos, pero a la hora de insertar un registro en la tabla (TIPO_CENTRAL era nuestro caso), no lo insertaba. Consultando el log, teníamos un registro que indicaba: ERROR: Error no controlado. Tras estudiar posibles problemas, el error lo encontramos en que la variable que se utilizaba para almacenar los campos de la tabla se llamaban igual que los campos. Por lo tanto, la sentencia insert no funcionaba correctamente. La solución fue llamar a la variable con otro nombre.
- Intento de insertar un registro en la tabla de contratos, sin existir el cliente en la tabla de clientes:

Al realizar el insert, oracle informa de que el bloque anónimo ha terminado, pero realizamos un select de la tabla de contratos, y nos encontramos con que el registro no está. El paso siguiente es consultar la tabla de log y comprobamos el error:

ICHA	PROCEDIMIENTO	PARAM_ENTRADA	PARAM_SALIDA
5/12	P_CONTRATO_INSERT	99999999R 1 01/01/12 10	ERROR: NO SE HA ENCONTRADO CLAVE FORÁNEA I

6.- VALORACIÓN ECONÓMICA Y RECURSOS NECESARIOS

6.1.- Recursos Humanos

Los recursos humanos están constituidos por todas las personas necesarias para la realización del proyecto.

Los diferentes roles necesarios para la realización del proyecto son:

Rosana Marín Berraondo

- Jefe de proyecto: se encargará de la planificación de las tareas, del cumplimiento de las entregas acordadas, de la comunicación con el cliente y la coordinación del resto de roles.
- Analista: sus principales tareas serán las de analizar los requerimientos que ha de cumplir la programación y crear los diseños que den una solución informática a las necesidades de información.
- Programador: Codifica los diseños o algoritmos que le da el analista.
- Becario: Se encargará de testear el software y de añadir datos a las tablas.
- Gestión: en toda empresa hay unas tareas de gestión, realizadas por el gerente, comercial, departamento de recursos humanos y de nóminas que hay que incluir en cada uno de los proyectos realizados.

En éste proyecto todos los roles serán ejecutados por la alumna Rosana Marín Berraondo.

Además, el papel de cliente lo efectúa el consultor Manel Rella Ruiz.

6.2.- Recursos Técnicos y Materiales

Los recursos técnicos están compuestos por las técnicas y métodos empleados para la realización del trabajo.

Hardware:

- PC AMD Athlon 64 procesador 3200+ a 2GHZ con 1 GB de RAM y Windows XP Profesional con Service Pack 3 como sistema operativo.
- Pen-drive USB Lacie de 8GB para guardar la documentación.

Software:

- Microsoft Word 2003 para redactar la documentación.
- Microsoft PowerPoint 2003 para crear la presentación del proyecto.
- Oracle 10g Express como motor de la BD.
- SQL Developer como herramienta para acceder a la BD.
- MagicDraw y GanttProject para realizar diagramas.
- Notepad++ y mspaint para codificar scripts y editar imágenes.

6.3.- Valoración económica

La valoración económica se divide en el costo de cada uno de los recursos humanos y materiales necesarios para su ejecución.

Así, en la siguiente tabla tenemos la valoración económica de los **recursos humanos** para cada una de las tareas expuestas:

Rol	Tarea	Jornadas	Horas	Sueldo (€)	anual €/hora	Importe (€)	tarea	
Jefe de proyecto	PAC 1 (plan de trabajo)	12	36	30.000	16,67	600,00		
	Entrega de la PAC2 y revisión	2	6	30.000	16,67	100,00		
	Entrega de la PAC3 y revisión	2	6	30.000	16,67	100,00		
	Entrega del TFC	1	3	30.000	16,67	50,00		
Analista	Análisis de los requisitos funcionales	5	15	25.000	13,89	208,33		
	Diseño conceptual de la BBDD	4	12	25.000	13,89	166,67		
	Crear diagrama E/R	1	3	25.000	13,89	41,67		
	Diseño lógico	4	12	25.000	13,89	166,67		
	Crear diagrama modelo lógico	1	3	25.000	13,89	41,67		
	Redacción de parte de la memoria PEC2	1	3	25.000	13,89	41,67		
	Creación juego de pruebas	2	6	25.000	13,89	83,33		
	Redacción de parte de la memoria PEC3	2	6	25.000	13,89	83,33		
	Presentación en PowerPoint	2	6	25.000	13,89	83,33		
	Redactar la memoria final	4	12	25.000	13,89	166,67		
	Programador	Scripts creación de tablas	1	3	20.000	11,11	33,33	
		Índices de las tablas	1	3	20.000	11,11	33,33	
		Procedimientos almacenados	3	9	20.000	11,11	100,00	
Disparadores		3	9	20.000	11,11	100,00		
Desarrollo módulo estadístico		6	18	20.000	11,11	200,00		
Corrección tras juegos de pruebas		4	12	20.000	11,11	133,33		
Preparación del código y anexos		3	9	20.000	11,11	100,00		
Posible implementación de mejoras del sistema		4	12	20.000	11,11	133,33		
Becario		Población de las tablas	1	3	15.000	8,33	25,00	
		Testeo juego pruebas	2	6	15.000	8,33	50,00	
Gestión			40	23.000	12,78	511,11		
		71	253			3352,78		

Rosana Marín Berraondo

En el cuadro anterior se puede observar la división de tareas por rol y su estimación en jornadas de tres horas cada una.

El cálculo de sueldo de cada rol es aproximado y el precio a la hora se ha calculado dividiendo el sueldo anual entre 1800 horas al año de un calendario laboral.

Así se obtiene que la valoración de los recursos humanos asciende a 3355 euros.

A la valoración total del proyecto se ha de sumar los **recursos materiales**:

- Hardware: se valorará en unos 500€
- Software: las licencias de los programas necesarios son gratuitas o bien las proporciona la UOC.

En resumen, la valoración estimada del proyecto asciende a unos **3900 euros**.

7.- MEJORAS PROPUESTAS

A continuación proponemos mejoras para el sistema:

- Almacenar un histórico de inspecciones para las centrales de producción y distribución.
- Almacenar una potencia máxima permitida para los modelos de contador o, en su caso, para los contadores.
- Almacenar si un contador está activo o no, bien dado de baja guardando su fecha de baja o bien con un check que lo indicara.
- Almacenar la fecha de alta y baja para las centrales de producción y distribución, para futuras estadísticas.
- Creación de usuarios para la gestión del log.
- Posibilidad de borrado de los históricos por un solo atributo de sus PKs. Por ejemplo, en hist_lectura, pk formada por fecha y cod_contador, posibilidad de borrar todas las lecturas de una fecha, o todas las lecturas de un contador.
- En el diseño de las tablas, añadir de forma ordinaria a todas y cada una de ellas los tres campos siguientes:
 - o FACT: fecha junto con hora de la última actualización del registro.
 - o UACT: usuario que ha realizado la última actualización del registro.
 - o TACT: campo que almacena "B", "I", "A" para informar de la última acción realizada con el registro. BORRADO / INSERCIÓN / ACTUALIZACIÓN.

Así tendremos información de quien y cuándo se ha realizado la última modificación de cada uno de los registros. Este cambio implica un mantenimiento de usuarios para poder realizar el seguimiento.

8.- CONCLUSIONES

El desarrollo de este proyecto ha resultado una experiencia positiva en cuanto a que he podido poner en práctica gran parte de los conocimientos adquiridos a lo largo de la carrera de ingeniería informática de sistemas.

El esfuerzo y la dedicación en el transcurso del desarrollo del producto se ha visto recompensado con la obtención de un proyecto de dimensiones reales y calidad elevada.

9.- BIBLIOGRAFIA

LIBROS DE CONSULTA:

Materiales didácticos de las asignaturas: Base de datos 1, Base de datos II, Estructura de la Información, Técnicas de Desarrollo de Software e Informática aplicada a la Gestión.

FUENTES DE INTERNET:

Documentación de Oracle 10: www.oracle.com
<http://www.wikioracle.es/>

10.- ANEXOS

10.1.- Instalación del producto

Los pasos a seguir para instalar la base de datos implementada en el PFC de bases de datos relacionales son:

- 1.- Instalar Oracle Express v10.2.0.1
- 2.- Instalar SQL Developer v1.0.0.15
- 3.- Conectar con el usuario SYSTEM a la BD a través de la página de inicio de Oracle Express.
Se recomienda crear un nuevo usuario, por ejemplo UOC con los permisos correspondientes y el tablespace USERS.
- 4.- Ejecutar SQL Developer
- 5.- Crear una nueva conexión, por ejemplo UOC_DB, con el usuario SYSTEM.
Role: default / Hostname: localhost / port: 1521 / SID: xe
- 6.- En la pestaña "DBMS Output" activar "set serveroutput on"
- 7.- Orden de los scripts a ejecutar:
 - 1_creacion_tablas_secuencias.sql
 - 2_creacion_procedimientos_ABM.sql
 - 3_creacion_vistas_materializadas.sql
 - 4_creacion_procedimientos_estadisticas.sql
 - 5_creacion_procedimiento_estadisticas.sql
 - 6_creacion_triggers.sql
 - 7_creacion_procedimientos_consultas.sql
 - 8_creacion_procedimientos_listados_estadisticas.sql
 - 9_relleno_tablas.sql
- 8.- Se adjunta un script 10_borrado_datos.sql para inicializar los datos de la base de datos y dejarla partiendo de cero.
- 9.- El script 11_inicializa_bd.sql borra todos los objetos creados de la base de datos.
- 10.- Los listados por pantalla, tanto de consultas, como de estadísticas se pueden testear

con los scripts:

12_test_listados_consultas.sql
13_test_listados_estadisticas.sql

10.2.- Resultado plan de pruebas consultas

Tras la ejecución del script de las consultas, el resultado que vemos en la consola es:

bloque anónimo terminado

Listado de contadores que superan el 80%:

Localidad: MADRID
Fecha: 02/01/2012
Contadores

Código de contrato: 5
Potencia: 15
Tanto por cien: 479,63

Código de contrato: 9
Potencia: 7
Tanto por cien: 151,3

Código de contrato: 2
Potencia: 15
Tanto por cien: 343,28

Código de contrato: 8
Potencia: 15
Tanto por cien: 350,77

Código de contrato: 10
Potencia: 10
Tanto por cien: 146,8

Código de contrato: 9
Potencia: 7
Tanto por cien: 146,3

Código de contrato: 5
Potencia: 15
Tanto por cien: 303,31

Código de contrato: 8
Potencia: 15
Tanto por cien: 231,95

Código de contrato: 2
Potencia: 15
Tanto por cien: 463,89

Listado de centrales de distribución que distribuyen mas energia:

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Dirección: AVDA. DISTRIBUCIÓN 23
Energía emitida: 452200
Energía máxima: 2000

Listado de las líneas de comunicación mas cargadas:

Identificador de la línea: 2
Carga de la línea: 272988
Energía a ampliar: 60186

Identificador de la línea: 3
Carga de la línea: 17640
Energía a ampliar: 46657

Identificador de la línea: 7
Carga de la línea: 16035,4285714285714285714285714286
Energía a ampliar: 56018

Identificador de la línea: 10
Carga de la línea: 15792
Energía a ampliar: 44498

Identificador de la línea: 1
Carga de la línea: 10428
Energía a ampliar: 46222

Identificador de la línea: 9
Carga de la línea: 2533,33333333333333333333333333333333
Energía a ampliar: 44183

Identificador de la línea: 6
Carga de la línea: 1532
Energía a ampliar: 56779

Identificador de la línea: 5

Rosana Marín Berraondo

Carga de la línea: 1514,4
Energía a ampliar: 62194

Identificador de la línea: 8
Carga de la línea: 1336,5
Energía a ampliar: 58548

Identificador de la línea: 4
Carga de la línea: 1083,6
Energía a ampliar: 80465

Listado de los clientes con alta disponibilidad:

NIF/DNI del cliente: 99999999R
Contrato: 1
Modelo del contador: 718

NIF/DNI del cliente: 88888888R
Contrato: 2
Modelo del contador: 703

NIF/DNI del cliente: B11111111
Contrato: 3
Modelo del contador: 718

NIF/DNI del cliente: 11111111A
Contrato: 4
Modelo del contador: 705

NIF/DNI del cliente: 2222222B
Contrato: 5
Modelo del contador: 707

NIF/DNI del cliente: 33333333C
Contrato: 6
Modelo del contador: 709

NIF/DNI del cliente: 4444444Z
Contrato: 7
Modelo del contador: 711

NIF/DNI del cliente: 5555555Q
Contrato: 8
Modelo del contador: 713

NIF/DNI del cliente: 6666666W
Contrato: 9
Modelo del contador: 714

NIF/DNI del cliente: 7777777E
Contrato: 10
Modelo del contador: 715

NIF/DNI del cliente: B2222222
Contrato: 11
Modelo del contador: 718

NIF/DNI del cliente: B3333333
Contrato: 12
Modelo del contador: 703

NIF/DNI del cliente: B3333333
Contrato: 13
Modelo del contador: 718

NIF/DNI del cliente: B5555555
Contrato: 14
Modelo del contador: 705

NIF/DNI del cliente: B6666666
Contrato: 15
Modelo del contador: 707

Produccion y consumo para la central

Central de producción: 3
Fecha inicial: 01/01/2011
Fecha final: 31/12/2011
Consumo de los contadores asociados: 709,2
Producción: 67048

Porcentaje de lecturas presenciales y telemáticas en un periodo de tiempo:

Fecha inicial: 01/01/2011
Fecha final: 31/12/2011
Lecturas presenciales: 87
Lecturas telemáticas: 13

Listado de contadores con una antigüedad de:

Años: 12
Contador: 1

Contador: 3

Contador: 11

Contador: 13

Contador: 29

10.3.- Resultado plan de pruebas estadísticas

Tras la ejecución del script de las estadísticas, el resultado que vemos por pantalla es:

bloque anónimo terminado

Listado E1 - Contadores y consumo de central de producción

Código contador: 1
Acumulador: 94,84

Código contador: 2
Acumulador: 1156,36

Código contador: 3
Acumulador: 112

Código contador: 4
Acumulador: 96,2

Código contador: 5
Acumulador: 507,22

Rosana Marín Berraondo

Código contador: 6
Acumulador: 94

Código contador: 7
Acumulador: 96

Código contador: 8
Acumulador: 378,6

Código contador: 9
Acumulador: 222,8

Código contador: 10
Acumulador: 154,28

Código contador: 11
Acumulador: 168

Código contador: 12
Acumulador: 166,68

Código contador: 13
Acumulador: 160,02

Código contador: 14
Acumulador: 224,44

Código contador: 15
Acumulador: 112

Código contador: 16
Acumulador: 50,44

Código contador: 17
Acumulador: 266,36

Código contador: 18
Acumulador: 392

Código contador: 19
Acumulador: 322,2

Código contador: 20
Acumulador: 265,22

Código contador: 21
Acumulador: 348

Código contador: 22
Acumulador: 496

Código contador: 23
Acumulador: 354,6

Código contador: 24
Acumulador: 422,8

Código contador: 25
Acumulador: 354,28

Código contador: 26
Acumulador: 148

Código contador: 27

Acumulador: 1166,68

Código contador: 28
Acumulador: 380,02

Código contador: 29
Acumulador: 584,44

Código contador: 30
Acumulador: 312

Listado E2 - Valor medio de energia consumida de una línea en un año

Línea: 38
Año: 2012
Valor medio: 0

Listado E3 - Línea mas cargada.

Línea: 27

Listado E4 - Porcentaje de líneas que superan el 50% de su capacidad

Año: 2012
Porcentaje de lineas: 0

Listado E5 - Número de centrales de producción que generan menos del 30% de la producción total

Año: 2012
Número de centrales: 20

Listado E6 - Top 10 de contadores con mas consumo:

Código contador: 27

Código contador: 29

Código contador: 5

Código contador: 2

Código contador: 22

Código contador: 24

Código contador: 18

Código contador: 28

Código contador: 8

Código contador: 23

Listado E7 - Consumo medio de todos los clientes:

Dni/Nif del cliente: 8888888R
Consumo medio 48,3

Dni/Nif del cliente: 2222222B
Consumo medio 48,28

Dni/Nif del cliente: 9999999R
Consumo medio 7,68

Rosana Marín Berraondo

Dni/Nif del cliente: B1111111
Consumo medio 31,5

Dni/Nif del cliente: B2222222
Consumo medio 19,75

Dni/Nif del cliente: 33333333C
Consumo medio 27,63

Dni/Nif del cliente: 5555555Q
Consumo medio 45,83

Dni/Nif del cliente: 6666666W
Consumo medio 40,35

Dni/Nif del cliente: B5555555
Consumo medio 50,56

Dni/Nif del cliente: B6666666
Consumo medio 26,5

Dni/Nif del cliente: 11111111A
Consumo medio 26,28

Dni/Nif del cliente: 4444444Z
Consumo medio 37

Dni/Nif del cliente: 7777777E
Consumo medio 31,79

Dni/Nif del cliente: B3333333
Consumo medio 58,54

10.4.- Resultado plan de pruebas borrado de datos

Tras la ejecución de este script se borra el contenido de todas las tablas y se reinician las secuencias.

Podremos comprobarlo realizando una sentencia select sobre cualquier tabla:

```
SELECT *  
FROM LOG;
```

La salida por pantalla me informa de todos los registros borrados.

```
60 filas eliminado  
40 filas eliminado  
1 filas eliminado  
4 filas eliminado  
3 filas eliminado  
10 filas eliminado  
14 filas eliminado  
60 filas eliminado  
20 filas eliminado  
15 filas eliminado  
30 filas eliminado  
60 filas eliminado  
30 filas eliminado  
720 filas eliminado  
240 filas eliminado
```


120 filas eliminado
40 filas eliminado
25 filas eliminado
16 filas eliminado
5 filas eliminado
sequence SEQ_COD_CENTRAL_DIST borrado.
sequence SEQ_COD_CENTRAL_PROD borrado.
sequence SEQ_COD_CONTADOR borrado.
sequence SEQ_COD_CONTRATO borrado.
sequence SEQ_COD_LINEA borrado.
sequence SEQ_TIPO_CENTRAL borrado.
sequence SEQ_COD_CENTRAL_DIST creado.
sequence SEQ_COD_CENTRAL_PROD creado.
sequence SEQ_COD_CONTADOR creado.
sequence SEQ_COD_CONTRATO creado.
sequence SEQ_COD_LINEA creado.
sequence SEQ_TIPO_CENTRAL creado.