

Diseño e implementación de la base de datos de un sistema centralizado de control del gasto público de los parlamentos europeos



Autor: Josep Xicart Carrencá

Titulación: Ingeniería técnica informática de sistemas

Consultor: Manel Rella Ruiz

Fecha: 12 de enero de 2014

Dedicatoria

A mi mujer y mis hijos, por la paciencia y el apoyo

Resumen

En esta memoria se describe el desarrollo de un proyecto software, consistente en una base de datos para el control del gasto público de los parlamentos europeos. Entendemos por desarrollo, el conjunto de etapas que constituyen el ciclo de vida del software, desde el análisis de requisitos hasta la instalación del producto, pasando por el diseño, la codificación y las pruebas.

Todo proyecto software va acompañado de un contacto previo con el cliente, en el que se sientan las bases del desarrollo. Esta etapa previa se refleja en el primer capítulo de la memoria. Allí es donde se han establecido los objetivos que se pretende cubrir, es decir, el alcance del proyecto. También se ha establecido una planificación, una valoración de recursos necesarios y un análisis de riesgos.

En los capítulos siguientes, se han ido cubriendo, una tras otra, todas las etapas del desarrollo del proyecto. Se ha intentado proporcionar abundantes explicaciones, acompañadas de figuras que complementan la exposición y ayudan a su comprensión. A menudo se han presentado varios modos de resolver una situación. Cuando esto ha ocurrido, se han analizado y discutido las diferentes posibilidades que se presentaban, para finalmente justificar la decisión tomada.

Se ha dado especial importancia a la etapa de pruebas. A este respecto, ha parecido importante justificar que los informes y estadísticas que se solicitan, ofrecen información fiable. En consecuencia, las pruebas no se limitan a presentar los resultados, sino que se intenta también demostrar que dichos resultados son correctos. Ello se ha hecho obteniendo la misma información por otros procedimientos.

En los capítulos finales se ofrece una estimación del coste económico del proyecto, y a continuación, a modo de conclusión, algunos comentarios sobre el desarrollo y la experiencia que éste ha aportado.

Índice de contenidos

1.	Introducción	7
1.1.	Justificación	7
1.2.	Objetivos del TFC.....	7
1.2.1.	Objetivo funcional del problema.....	7
1.2.2.	Objetivos de diseño y documentación.....	8
1.2.3.	Pruebas.....	8
1.2.4.	Memoria y presentación	8
1.3.	Enfoque y método seguido	8
1.3.1.	Ciclo de vida en cascada.....	8
1.3.2.	Adaptaciones sobre el ciclo de vida en cascada.....	9
1.3.3.	Valoración de los recursos necesarios	10
1.4.	Planificación del trabajo.....	10
1.4.1.	Disponibilidad de tiempo	10
1.4.2.	Diagrama de Gantt de la planificación	11
1.4.3.	Análisis de riesgos y planes de contingencias	12
1.5.	Productos obtenidos	12
1.6.	Breve descripción del desarrollo del proyecto.....	12
2.	Análisis de los requisitos funcionales.....	14
2.1.	Almacenamiento de datos	14
2.2.	Mantenimiento de los datos	14
2.3.	Procedimientos de consulta sobre los gastos de los parlamentos	14
2.4.	Registro de llamadas a los procedimientos.....	15
2.5.	Módulo estadístico.....	15
2.6.	Diseño de un juego de pruebas.....	16
2.7.	Consecuencias del análisis de requisitos.....	17
3.	Diseño de la base de datos.....	18
3.1.	Modelo UML.....	18
3.2.	Comentarios sobre el modelo.....	18
3.2.1.	Aspectos no representados en el modelo.....	18
3.2.2.	Otros comentarios.....	19
3.3.	Definición de la estructura de la BBDD relacional	19
3.3.1.	Supresión de la herencia	19
3.3.2.	Conversión del modelo estático en un modelo relacional.....	21
3.4.	Definición de las tablas	22

3.4.1.	Estructura de las tablas	22
3.4.2.	Otros comentarios sobre las tablas.....	25
3.4.3.	Restricciones de integridad	25
3.4.4.	Tabla de Log	26
3.4.5.	Claves primarias y uso de secuencias.....	26
4.	Procedimientos para el mantenimiento de las tablas maestras.....	27
4.1.	Llamadas y parámetros	27
4.2.	Decisiones de diseño sobre el mantenimiento de las tablas	28
4.3.	Control de errores.....	28
4.4.	Dependencias entre tablas y triggers.....	30
5.	Procedimientos de consulta.....	31
5.1.	Llamadas y parámetros	31
5.2.	Salida de los procedimientos de consulta.....	31
5.3.	Control de errores.....	31
5.4.	Comentarios sobre los procedimientos de consulta.....	32
6.	Registro de llamadas a los procedimientos.....	33
6.1.	Inserción de registros en la tabla de log	33
6.2.	Consulta de la tabla de Log	33
7.	Módulo estadístico.....	34
7.1.	Diseño de las tablas del módulo estadístico	34
7.2.	Implementación del procedimiento que calcula las estadísticas.....	37
7.2.1.	Parámetros de entrada y salida	37
7.2.2.	Control de errores	37
7.2.3.	Llamada al procedimiento.....	37
7.3.	Otros comentarios sobre el módulo estadístico	38
8.	Diseño de los datos de prueba.....	39
8.1.	Datos de parlamentos y parlamentarios.....	39
8.2.	Datos de histórico de gastos	39
8.3.	Tablas auxiliares	40
9.	Descripción y carga del producto software.....	42
9.1.	Descripción del producto	42
9.2.	Carga del producto y los datos.....	43
10.	Prueba del producto.....	44
10.1.	Pruebas de alta, baja, modificación de tablas maestras.....	44
10.1.1.	Diseño de las pruebas de ABM.....	44
10.1.2.	Ejemplo de ejecución de las pruebas ABM	44

10.2.	Pruebas sobre los procedimientos de consulta	46
10.2.1.	Diseño de las pruebas de los procedimientos de consulta	46
10.2.2.	Contraste de los resultados obtenidos	47
10.3.	Pruebas sobre el módulo estadístico	48
10.3.1.	Diseño de las pruebas del módulo estadístico	49
10.3.2.	Ejemplos del resultado de las pruebas.....	49
11.	Valoración económica	54
11.1.	Cálculo del coste económico.....	54
11.2.	Comentarios sobre la valoración económica del proyecto.....	54
12.	Conclusiones.....	55

Índice de figuras

Figura 1-Planificación	11
Figura 2 -Diagrama UML del problema	18
Figura 3- Supresión de la herencia	20
Figura 4. Eliminación de la herencia-1	20
Figura 5.Base de datos relacional.....	22
Figura 6. Tabla de Log.....	26
Figura 7. Secuencias	26
Figura 8. Mantenimiento de las tablas maestras.....	27
Figura 9: Trigger	27
Figura 10. Trigger para actualizar parlamentarios	30
Figura 11. Procedimientos de consulta.....	31
Figura 12. Script "Ultima Linea Log"	33
Figura 13. Cuerpo del procedimiento "Ultima_linea_log"	33
Figura 14. Tablas del módulo estadístico	34
Figura 15. Procedimiento para el módulo estadístico	34
Figura 16. Carga de tablas maestras	43
Figura 17. Pruebas ABM	44
Figura 18. Ejemplo pruebas ABM.....	44
Figura 19. Pruebas de los procedimientos de consulta	46
Figura 20. Ejecución proc. consulta B.....	47
Figura 21. Ejecución proc. F	47
Figura 22. Verificación resultados del proc. B.....	48
Figura 23. Verificación resultados proc. F.....	48
Figura 24. Pruebas módulo estadístico	48
Figura 25. Pruebas ME-1	49
Figura 26. Pruebas contraste ME-1	50
Figura 27. Pruebas ME-5	50
Figura 28. Gastos de los parlamentarios en 2011.....	51
Figura 29. Parlamentarios con gasto en 2011.....	51
Figura 30. Pruebas ME-6	52
Figura 31. Parlamentario con mayor gasto en 2012	52
Figura 32. Parlamentario con menor gasto en 2012.....	53

1. Introducción

1.1. Justificación

La realización de un trabajo con los requerimientos que se han solicitado, es un excelente ejercicio para desarrollar habilidades en varias vertientes distintas:

- *Planificación y organización.* El volumen del trabajo a realizar es considerable; el tiempo de que se dispone, limitado, y la fecha de entrega, impostergable. Todo ello obliga a planificar el desarrollo por etapas y tiempos, a priorizar y a valorar qué aspectos son más importantes.
- *Integración de diversos conocimientos.* En este trabajo, es necesario emplear e integrar, conocimientos tales como: ingeniería del software, programación, consultas SQL, manejo de un SGBD, etc., pertenecientes a diversas asignaturas.
- *Búsqueda de información.* No es suficiente la información que se encuentra en los materiales docentes; a menudo es necesario saber encontrar soluciones a problemas muy concretos
- *Comunicación escrita.* No es suficiente haber desarrollado un determinado proyecto, Hay que saber comunicar y justificar lo que se ha realizado. Es más, no basta con que esta comunicación sea correcta desde el punto de vista de su contenido; la forma y la presentación son igualmente importantes si queremos que nuestro interlocutor quede satisfecho y convencido de que se trata de un buen trabajo.

En resumen, podríamos decir que este trabajo es el “prototipo” de una situación a la que probablemente nos enfrentaremos en el ámbito profesional.

Desde el punto de vista académico, podríamos añadir además el acierto que constituye incluir un trabajo de este tipo en el plan de formación. A menudo se ha hablado de la distancia existente entre el mundo académico y el profesional. Este trabajo, sin duda, contribuye a acortar dicha distancia.

1.2. Objetivos del TFC

Se describen a continuación, los objetivos de forma general.

1.2.1. Objetivo funcional del problema

Se nos pide desarrollar una base de datos adecuada para controlar el gasto de los parlamentos y parlamentarios de la comunidad económica europea.

La base de datos debe incorporar:

- Datos acerca de 3 entidades principales: parlamentos europeos, parlamentarios, e histórico de gastos. Los datos específicos que se desea guardar, se encuentran bien detallados. Del análisis del enunciado, se desprende también la necesidad de creación de varias tablas auxiliares.
- Procedimientos que permitan el mantenimiento de las tres entidades mencionadas (parlamentos, parlamentarios, histórico de gastos)
- Procedimientos de consulta que devuelvan los gastos de unos determinados parlamentos/parlamentarios/años.

- Un registro histórico de las llamadas a todos los procedimientos que se invoquen.
- Un “módulo estadístico” que almacenará valores de gastos “precalculados”. Su finalidad es poder proporcionar, en muy poco tiempo, unas consultas muy específicas y a menudo complejas. El “módulo estadístico” debe estar “siempre al día”. En consecuencia, cada gasto que se registre debe venir acompañado de la correspondiente actualización en el módulo estadístico.

1.2.2. Objetivos de diseño y documentación

Además de los requisitos puramente funcionales, deben cubrirse también los siguientes objetivos formales y de diseño:

- Se debe realizar un diagrama UML o E/R que permita elaborar un modelo conceptual de la realidad que se pretende describir, con sus correspondientes restricciones de integridad. Dicho modelo servirá para el posterior diseño de la base de datos.
- Los procedimientos deben estar adecuadamente documentados. La documentación debe incluir, como mínimo, una descripción de lo que hace cada procedimiento, así como la descripción de los parámetros de entrada y salida

1.2.3. Pruebas

El producto debe venir acompañado de un juego de pruebas para demostrar que se cumplen los requisitos funcionales. Dicho juego de pruebas debe incluir:

- *Un juego de datos suficiente* para cubrir las distintas casuísticas
- *Llamadas a los procedimientos realizados*. Con dichas llamadas se debe demostrar que los procedimientos funcionan correctamente.
- Demostración de que el módulo estadístico funciona correctamente.

1.2.4. Memoria y presentación

Todo el trabajo realizado debe presentarse acompañado de la correspondiente memoria, así como de una breve presentación/resumen

1.3. Enfoque y método seguido

Para el desarrollo del proyecto, inicialmente se ha intentado aplicar el enfoque del *ciclo de vida en cascada*. Este planteamiento, sin embargo, ha demostrado ser poco realista, obligándonos a replantearnos etapas anteriores para poder continuar con el desarrollo.

1.3.1. Ciclo de vida en cascada

Este modelo de desarrollo ha sido útil en la medida en que ha permitido empezar a organizar y planificar el trabajo. Sus etapas y las actividades que se previó realizar en cada una de ellas, se resumen en la siguiente tabla:

Etapa del ciclo de vida en cascada	Actividades a realizar
Análisis previo:	Lectura del documento que especifica el proyecto. Valorar qué software/hardware necesitamos para realizar el proyecto
Análisis de requisitos	Esta etapa se nos proporciona prácticamente terminada. Puede verse matizada en el transcurso del desarrollo debido a las consultas y comentarios en el aula.
Diseño	Realización de un diagrama UML que represente el modelo conceptual de la realidad a describir
Implementación	Preparación de los scripts de creación de tablas y procedimientos almacenados capaces de cumplir con los requisitos del proyecto.
Pruebas	Creación de un conjunto de datos de pruebas.
Entrega del producto	Preparación de toda la documentación y entrega final

1.3.2. Adaptaciones sobre el ciclo de vida en cascada

A medida que se avanzaba en el desarrollo del producto, se ha hecho evidente la necesidad de replantear etapas que en un principio se daban por superadas. En consecuencia, el enfoque del ciclo de vida en cascada quedó rápidamente superado por un enfoque iterativo.

A modo de ejemplo, y *asumiendo el riesgo de anticipar información que más adelante comentará en detalle*, nos parece adecuado mencionar aquí algunas de las problemáticas que nos han obligado a “dar marcha atrás” en el ciclo de desarrollo.

Problemática	Evolución de la solución	
Diseño del modelo E/R en cuanto a los gastos	Planteamiento inicial	Se previeron la creación de 3 tablas “gastos de los parlamentarios”, “gastos de los parlamentos”, y una tercera tabla con campos comunes a ambos conceptos
	Problema no detectado previamente	El diseño propuesto complicaba excesivamente el desarrollo de los procedimientos de mantenimiento de gastos y consulta de gastos
	Replanteamiento	Se decidió utilizar una única tabla de gastos, con todos los campos reflejados en ella.
Procedimientos para rellenar las tablas de parlamentos y parlamentarios	Planteamiento inicial	Los dos procedimientos se podrían escribir de forma independiente el uno del otro.
	Problema no detectado previamente	Cada vez que se produzca una alta o baja de un parlamentario, es necesario actualizar el “número de parlamentarios de ese parlamento” (el cual se almacena precisamente en la tabla de parlamentos)
	Replanteamiento	Implementar un <i>Trigger</i> que actualizara el número de parlamentarios en la tabla de parlamentos, y que se ejecutará cuando se produzcan altas o bajas de parlamentarios
Carga de los datos de prueba	Planteamiento inicial	El usuario que llama al procedimiento de alta de gasto, conoce el valor de todos los datos a insertar.

Problemática	Evolución de la solución	
	Problema no detectado previamente	En el diseño de la base de datos, se concluyó que las tablas de parlamentarios y de gastos, utilizarían una clave primara artificial (número secuencial). El usuario que invoca al procedimiento de alta, no conoce el valor que debe asignarse a la clave artificial
	Replanteamiento	Se utilizara el mecanismo de la base de datos conocido como secuencia ("SEQUENCE") que el SGBD pone a nuestra disposición
Hay que realizar varios cálculos que implican medias anuales de gastos con respecto al número de parlamentarios de cada parlamento	Planteamiento inicial	Utilizar en los cálculos, el valor del "número de parlamentarios" que se almacena en la tabla de parlamentos
	Problema no detectado previamente	En la tabla de parlamentos solo se almacenará el número actual de parlamentarios. Es perfectamente posible que el número varíe de año en año
	Replanteamiento	Utilizar a efectos del cálculo de las medias, el número de parlamentarios que han registrado algún gasto en el período y parlamento considerados

1.3.3. Valoración de los recursos necesarios

Recursos de software:

- Gestor de BBDD Oracle Database Express Edition 11.2
- Entorno gráfico Oracle SQL Developer
- Programa para el dibujo de graficos E/R: MagicDraw UML Personal Edition 17.0
- Herramienta para dibujo de diagramas de Gannt: Microsoft Project 2010

Recursos de hardware:

- Ordenador portátil Hewlett-Packard (HP Elitebook Folio 9470m) con S.O. Windows 7 y 8 GB RAM. Las especificaciones son adecuadas para el desarrollo del producto

Recursos humanos:

- Las diferentes figuras implicadas en un proyecto real (analista, programador, técnico en BBDD) quedan asumidos por el autor del TFC.
- El consultor asume la figura de cliente
- Deben mencionarse también los compañeros del aula por sus valiosas aportaciones

1.4. Planificación del trabajo

1.4.1. Disponibilidad de tiempo

Se estima que la disponibilidad de tiempo es:

- En días laborables: 1,5 horas/3 días por semana
- Durante los fines de semana y festivos: 3 horas diarias

Inicialmente, las tareas se han planificado por semanas, según se muestra en la tabla adjunta

Entrega	Tareas previstas	Tiempo previsto
PAC1 7 octubre 2013	Lectura y comprensión del proyecto. Instalación entorno BBDD Oracle y software complementario. Familiarización con el software	Semanas del 23 de septiembre al 6 de octubre
PAC2 11 noviembre 2013	Diseño BBDD -UML	Semana del 7 al 13 de octubre
	Aprendizaje ORACLE	Semana del 14 al 20 de octubre
	Scripts tablas maestras	Semana del 21 al 27 de octubre
	Procs. alta/baja/modificación	Semana del 28 oct. al 3 noviembre
	Procs. consultas ordinarias	Semana del 4 al 10 de noviembre
	Redacción PAC2	9/10 noviembre
PAC3 11 diciembre 2013	Diseño M.E.	Semana del 11 al 17 de noviembre
	Scripts Tablas M.E.	Semana del 18 al 24 de noviembre
	Procs. alimentar M.E.	Semana del 25 nov al 1 de diciembre
	Procs. consulta M.E.	Semana del 2 al 8 de diciembre
	Redacción PAC3	9/10 diciembre
Entrega final 13 enero 2013	Preparación de la memoria, el resumen, realizar correcciones	Del 12 de diciembre al 12 de enero

1.4.2. Diagrama de Gantt de la planificación

El diagrama de Gantt correspondiente a esta planificación, se muestra a continuación

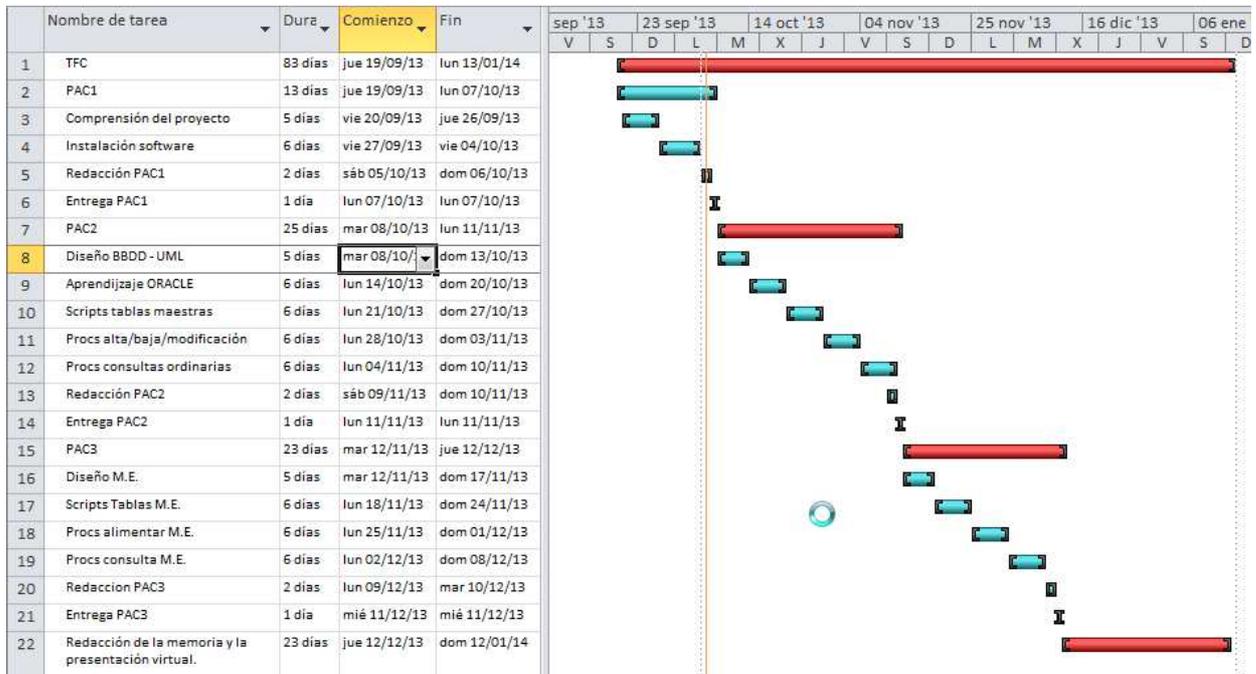


Figura 1-Planificación

1.4.3. Análisis de riesgos y planes de contingencias

A continuación se comentan los diferentes tipos de incidencias que presumiblemente se pueden presentar, junto con el plan de contingencia correspondiente.

Incidencia	Plan de contingencia
Imprevistos de origen profesional o familiar Imprevistos debidos a la complejidad del propio proyecto (Es posible que algún aspecto particularmente complejo, obligue a emplear mucho más tiempo del previsto)	En principio se cuenta con las vacaciones navideñas, que constituyen un recurso extra de tiempo.
Avería de hardware	Se dispone en el domicilio de otros ordenadores con los que se podría continuar el trabajo
Pérdida de datos por avería, virus. etc	Se dispone de una copia de seguridad remota mediante la herramienta "Dropbox", accesible desde cualquier ordenador conectado a internet.

En un proyecto real con un cliente real, se podrían presentar muchos otros tipos de contingencia. Probablemente la más importante es que, *el cliente no tenga claros los requisitos funcionales y los modifique con frecuencia*, con el consecuente impacto en la fecha de entrega del proyecto y/o el coste económico

1.5. Productos obtenidos

Los productos resultados del presente proyecto han sido

- La solución de software solicitada. Esta consiste en un conjunto de scripts SQL que se ejecutan en el entorno ORACLE, y que permiten crear las tablas, procedimientos almacenados y otros elementos del entorno de la BBDD (disparadores, secuencias)
- Un juego de pruebas exhaustivo.
- La presente memoria, donde se documenta en detalle el producto software y el proceso de desarrollo
- Una breve presentación-resumen de todo el trabajo realizado

1.6. Breve descripción del desarrollo del proyecto

Dedicaremos los próximos capítulos, a exponer como se ha desarrollado el proyecto software. Su contenido será el siguiente:

Capítulo	Resumen del contenido
2.	Análisis de los requisitos: Se analizarán en profundidad los requisitos del proyecto. De allí se deducirán los siguientes pasos a seguir
3.	Diseño de la base de datos. Partiendo de los requisitos, se modelará en lenguaje UML la situación que se pretende informatizar. De allí se acabará deduciendo la estructura de las tablas que necesitaremos

Capítulo	Resumen del contenido
4.	Procedimientos para el mantenimiento de las tablas maestras: Se describen los procedimientos que se han desarrollado para realizar altas/bajas/modificaciones de parlamentos, parlamentarios e histórico de gastos, así como los errores que devuelven
5.	Procedimientos de consulta: Se describen los procedimientos que realizan las diferentes consultas solicitadas en el proyecto
6.	Registro de llamadas a procedimientos. El proyecto solicita que se guarde en una tabla de log, el resultado de las llamadas a los procedimientos. Es este apartado se describe dicha tabla
7.	Módulo estadístico. Aquí se describen las tablas que albergan los datos solicitados en el módulo estadístico, así como el procedimiento que realiza los cálculos de dicho módulo.
8.	Diseño de los datos de prueba. Se explica cómo se han diseñado los datos de prueba para poder realizar con ellos las pruebas de los diferentes aspectos del producto software.
9.	Descripción y carga del producto software. Se listan todos los archivos que forman parte del producto obtenido, y cómo deben cargarse en la base de datos
10.	Pruebas del producto. Se describen las diferentes pruebas realizadas, y se muestra de qué manera cubren los requisitos.

2. Análisis de los requisitos funcionales

Recordemos que el objetivo fundamental del problema, consiste en diseñar un entorno adecuado para el mantenimiento y la consulta de gastos de los parlamentos y parlamentarios europeos. Del análisis del problema, se deducen 6 tipos de requisitos a cubrir, que se detallan en los sub-apartados siguientes:

2.1. Almacenamiento de datos

Se especifica en los apartados [R1], [R2], [R3] del enunciado. No parece necesario repetirlos aquí. Se dará respuesta a este requisito mediante la técnica del modelado de la realidad a describir mediante un diagrama UML, y su posterior transformación en una estructura de BBDD relacional, que se describen en el capítulo 3.

Vale la pena comentar el siguiente aspecto: el enunciado es muy claro al diferenciar los *gastos imputables directamente a los parlamentarios* de los *gastos generales de los parlamentos*. La BBDD debe tener mecanismos no solo para poder diferenciarlos de forma correcta, sino también para poderlos recuperar de forma diferenciada con facilidad.

2.2. Mantenimiento de los datos

Con la finalidad de facilitar el mantenimiento de los parlamentos, parlamentarios y el histórico de gastos, se nos pide desarrollar los procedimientos almacenados que se detallan a continuación:

- Alta parlamentos
- Baja parlamentos
- Modificación (edición) de parlamentos
- Alta parlamentarios
- Baja parlamentarios
- Modificación (edición) de parlamentarios
- Alta de un gasto
- Baja de un gasto
- Modificación (edición) de un gasto.

2.3. Procedimientos de consulta sobre los gastos de los parlamentos

Se solicita la elaboración de procedimientos almacenados para el mantenimiento de las tablas principales. A pesar de que en el enunciado quedan suficientemente claros dichos procedimientos, nos parece adecuado resumirlos a continuación.

Procedimiento	Datos de entrada	Datos de salida
A	Parlamento, año, parlamentario	Listado de gastos asociados al parlamentario durante ese año, por orden descendiente del gasto
B	Parlamento, año	Listado de parlamentarios con gasto directo, incluyendo nombre, apellidos, partido político en el momento de registrar el gasto, suma de los gastos directos de cada parlamentario
C	Año	Listado que incluya, para cada parlamento: nº de parlamentarios, gasto medio de los parlamentarios, gasto imputable a los parlamentarios, gasto imputable solo al parlamentos, gasto total del parlamento
D	Año, parlamento	Listado del gasto por categorías (código, descripción de la categoría, y suma del gasto imputable a esa categoría)
E	Año, Parlamento	Listado de parlamentarios que superen la media del gasto de los parlamentarios para ese parlamento
F	Año	Listado que represente el gasto contable de los parlamentos, incluyendo: Parlamento, número de parlamentarios, suma de cada uno de los siguientes conceptos: gasto pendiente de aprobar, gasto aprobado, gasto pendiente de abonar, gasto abonado

Se solicita también un control de errores. Es decir, si el procedimiento fracasa por alguna razón, terminará de forma controlada. Además, se almacenará un resultado del error producido (ver 2.4)

Por último, se solicita que todos los procedimientos estén debidamente documentados. La documentación debe cubrir 2 propósitos:

- Para el programador que posteriormente los utilice: la documentación debe ser suficiente para saber qué realiza el procedimiento, así como conocer el número y tipo de parámetros de entrada y salida, sin necesidad de mirar el código del procedimiento
- Debe facilitar su mantenimiento posterior, es decir, ser suficiente para permitir entender a un programador que quizá no fue el autor inicial del código, cómo funciona el procedimiento

2.4. Registro de llamadas a los procedimientos

Se solicita disponer de un registro de todas las llamadas a los procedimientos. Ello implica la creación de una tabla especial a tal efecto (a la que llamaremos *tabla de log*). Los datos mínimos a almacenar son: el nombre del procedimiento ejecutado, los parámetros de entrada y los de salida.

Las llamadas a los procedimientos almacenados, pueden acabar con resultado correcto, o bien acabar con error (ver punto 2.3). Ambas situaciones se reflejarán también en la tabla de log.

2.5. Módulo estadístico

Se solicita disponer de informes pre-calculados de cierta complejidad. Se trata de informes que en la mayor parte de los casos, se podrían obtenerse mediante una sentencia SELECT, pero con un coste en tiempo superior. La ventaja de tenerlos pre-calculados, es que la recuperación de los datos solicitados no tendrá apenas penalización en tiempo. Dichos informes quedan

suficientemente claros en el apartado [R7] del enunciado, sin embargo nos parece adecuado resumirlos también a continuación

Estadística	Ámbito de aplicación	Informe
1	Para cada parlamento	Suma de sus gastos totales, durante los 4 últimos años (contados a partir del último año en el que ese parlamento ha registrado algún gasto).
2	Para cada parlamento y año	Diferencia en euros, entre el parlamentario que ha registrado más gastos asociados, y el que ha gastado menos, en un parlamento y año determinados.
3	Para cada año	El parlamento que ha tenido el mayor importe de gastos durante ese año.
4	Para cada parlamentario	Suma de sus gastos totales, durante los 4 últimos años (contados a partir del último año en el que ese parlamentario ha registrado algún gasto).
5	Para cada año	El gasto medio de la figura del parlamentario (teniendo en cuenta todos los parlamentarios de todos los parlamentos).
6	Para cada año	El parlamentario con el máximo importe de gastos asociados, y el que ha tenido el mínimo importe de gastos asociados.
7	Para cada año	El parlamento cuya media de gasto por parlamentario, ha sido mayor durante ese año, así como el valor de dicha media.
8	Para cada año	La suma de todos los gastos de todos los parlamentos durante ese año.

Del análisis de requisitos, se desprende que será necesario:

- Crear tablas específicas para tener siempre disponibles estos informes.
- Recalcular cada uno de ellos, cada vez que se produzca el alta, baja o modificación de un gasto, de manera que permanezcan siempre actualizados.

2.6. Diseño de un juego de pruebas

Con tal de poder valorar que el producto se ha desarrollado de forma correcta, se solicita un juego de pruebas, el cual debe comprender

- Diseño de un conjunto de datos de prueba
- Llamadas a los procedimientos de alta/baja/modificación (punto 2.2 y punto 10.1)
- Llamadas a los procedimientos de consulta sobre los gastos (punto 2.3 y punto 10.2)
- Análisis de los resultados del módulo estadístico (punto 2.5 y punto 10.3)

2.7. Consecuencias del análisis de requisitos

En los capítulos siguientes se expondrán las soluciones desarrolladas para cada uno de los requisitos comentados. A continuación se muestran los capítulos de la memoria donde se desarrolla cada punto.

Requisito	Descripción	Desarrollo
2.1	Almacenamiento de datos	Capítulo 3
2.2	Mantenimiento de datos	Capítulo 4
2.3	Procedimientos de consulta sobre los gastos	Capítulo 5
2.4	Registro de llamadas a los procedimientos	Capítulo 6
2.5	Módulo estadístico	Capítulo 7
2.6	Diseño del juego de pruebas	Capítulos 8 y 10

3. Diseño de la base de datos

Para el diseño de la base de datos se seguirán los siguientes pasos:

- Representación de la realidad que se pretende describir, mediante un diagrama UML
- Obtención de la estructura de BBDD relacional a partir de dicho modelo
- Detalle de las tablas y campos a crear, con sus claves primarias y restricciones de integridad

Estos pasos se especifican en los siguientes apartados

3.1. Modelo UML

La realidad que se describe en el enunciado del proyecto (requisitos [R1], [R2], [R3]) puede modelarse a través del siguiente diagrama UML

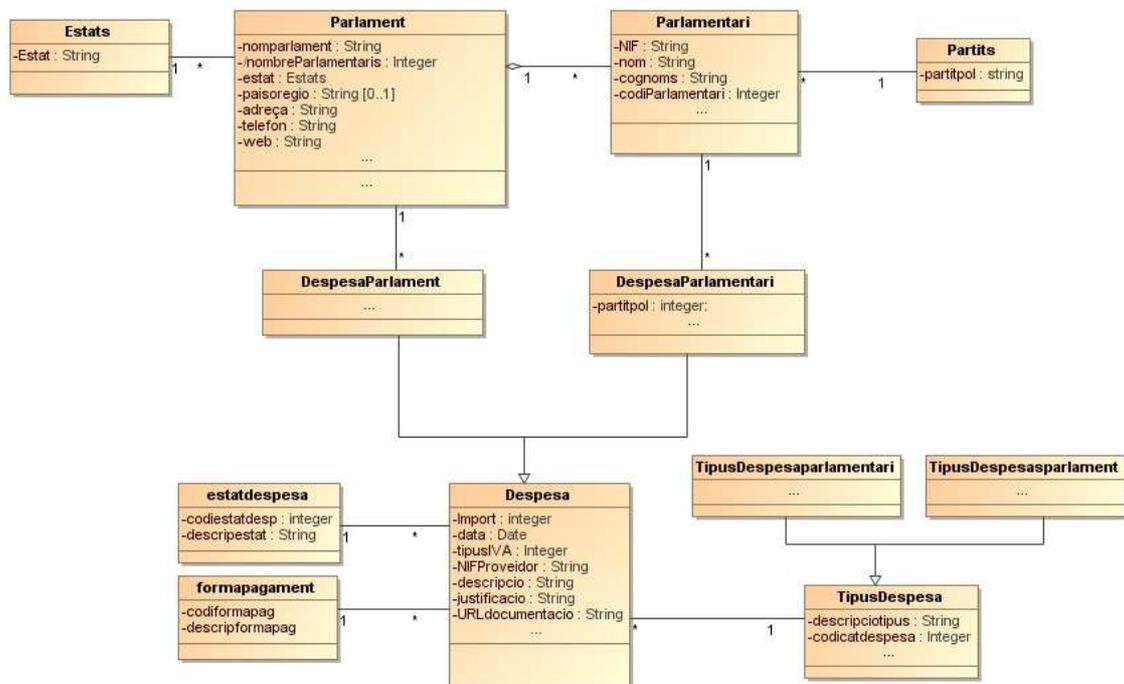


Figura 2 -Diagrama UML del problema

3.2. Comentarios sobre el modelo

Es conveniente ampliar la información que ofrece el diagrama UML con la siguiente información:

3.2.1. Aspectos no representados en el modelo

- El NIF del proveedor solo es necesario cuando el gasto es superior a 30 EURO.

3.2.2. Otros comentarios

- El modelo se ha diseñado de modo que un parlamentario solamente puede pertenecer a un único parlamento (no se deduce del enunciado, sino que se considera una suposición razonable)
- Hemos considerado “Estat” (estado de la comunidad europea al que pertenece el parlamento) como un atributo obligatorio, mientras que “país/región” lo hemos considerado optativo.
- Dado la diversidad de regiones que pueden existir en la Comunidad Europea, y considerando que en una región difícilmente existirá más de un parlamento, no ha parecido necesario introducir una tabla de regiones.
- Una persona física podría formar parte de diversos parlamentos en el transcurso del tiempo. Esto nos llevaría a definir otra entidad (“Personas”) con el NIF y nombre y apellidos de la persona física que en un momento dado forma parte de un parlamento. Se ha valorado la posibilidad de introducir esta nueva entidad, pero se han visto los siguientes inconvenientes
 - Su introducción nos obliga a algunas complicaciones en el mantenimiento de las tablas (por ejemplo, antes de dar de alta un parlamentario, habría que verificar si su NIF existe en la tabla de personas, y en caso contrario crearlo).
 - Esa entidad “Personas”, no tiene sentido en ningún otro lugar del modelo (es decir, no hay ninguna entidad distinta de “Parlamentaris”, que pueda también hacer uso o relacionarse con la entidad “Personas”), por tanto no se le puede obtener mayor utilidad que la de relacionarse con “Parlamentaris”

En consecuencia, finalmente se ha descartado la introducción de la entidad “Personas”

3.3. Definición de la estructura de la BBDD relacional

Para obtener la estructura de la BBDD relacional debemos realizar los siguientes pasos:

- Supresión de la herencia: Tenemos en nuestro proyecto 2 casos de herencia que es necesario eliminar
- Conversión del modelo estático en un modelo relacional.

3.3.1. Supresión de la herencia

En el diagrama UML obtenido, vemos que tenemos dos casos en los que es necesario suprimir la herencia:

El primero de ellos, está constituido por la superclase “Despesa” y las subclases “Despesaparlament” y “DespesaParlamentari”.

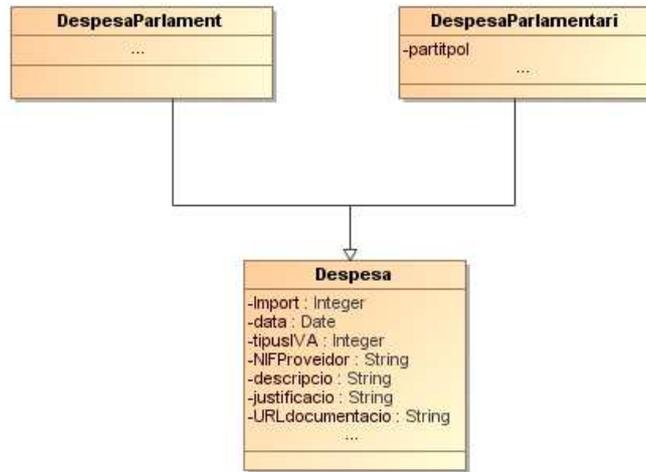


Figura 3- Supresión de la herencia

Analizamos las diversas maneras de suprimir la herencia en este caso:

Modo de supresión de la herencia	Ventajas/inconvenientes
1. Crear una tabla para "DespesaParlament" y otra para "DespesaParlamentari", conteniendo cada una de ellas tanto sus propios atributos como los atributos heredados de la superclase "Despesa"	<ul style="list-style-type: none"> No parece la organización más adecuada ya que se repiten en tablas diferentes atributos comunes Tiene problemas de escalabilidad (por ejemplo la creación de futuros atributos comunes, debería realizarse en las dos tablas)
2. Crear una tabla para la superclase "Despesa" y a la vez una para cada una de las subclases "Parlament" y "parlamentari"	<ul style="list-style-type: none"> Desde el punto de vista de los datos es una solución elegante. Al estar los datos repartidos en 3 tablas, complica mucho el diseño de los procedimientos de mantenimiento, de consulta, y el módulo estadístico
3. Crear una tabla única con toda la jerarquía de clases	<ul style="list-style-type: none"> Todos los datos relativos a gastos, pasan a estar en una única tabla

Decidimos adoptar la tercera solución: consolidar todos los datos relativos a gastos en una única tabla (ver apartado 3.3.2)

- También tenemos la superclase "TipusDespesa" y las subclases "TipusDespesaParlament" y "TipusDespesaParlamentari"

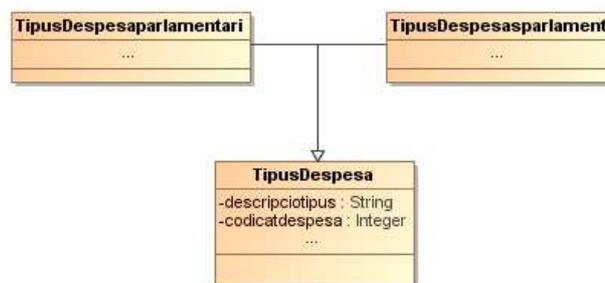


Figura 4. Eliminación de la herencia-1

“TipusDespesaParlament” y “Tipusdespesa parlamentari” no tienen en nuestro modelo atributos propios, por tanto unificaremos las 3 clases en una única tabla y *añadiremos un identificador que nos indicará de cuál de los dos “tipus de despesa” se trata.* (Como veremos posteriormente, dicho identificador se convertirá en un campo de nombre TIPUSDESPESA, y podrá adoptar los valores PARLAMENT o PARLAMENTARI)

3.3.2. Conversión del modelo estático en un modelo relacional

Una vez suprimida la herencia, obtenemos el modelo relacional de nuestro problema. En la siguiente tabla, se analizan los pasos necesarios para la transformación del modelo estático en un modelo relacional.

Conversión del modelo estático en relacional	Comentarios
Convertir cada entidad del modelo relacional en una tabla, y cada atributo en un campo. Identificar claves primarias para cada tabla.	En la tabla “despesas” será necesario añadir una clave artificial (no existe ninguna combinación de campos que nos aseguren una clave primaria. Por ejemplo: en la misma fecha, un mismo parlamentario podría realizar el mismo gasto por el mismo concepto e importe exactamente, ambos gastos serían indistinguibles)
Las relaciones 1..* deben tratarse, añadiendo en el lado “*”, los atributos que forman parte de la clave primaria del lado “1”	Existen varias relaciones de este tipo, no presentan mayores complicaciones
Tratamiento de las relaciones 1..0 o 1..1	Estas relaciones existen entre Despesa<->DesperaParlament Despesa<->Despesaparlamentari Se tratarán añadiendo en las tablas “DespesaParlament” y “DespesaParlamentari”, la clave primaria de “Despesa”
Tratamiento de las relaciones *.* y de asociaciones-nárias	No existen en nuestro modelo, por tanto no aplican

Finalmente, el modelo relacional obtenido, se representa en la siguiente figura

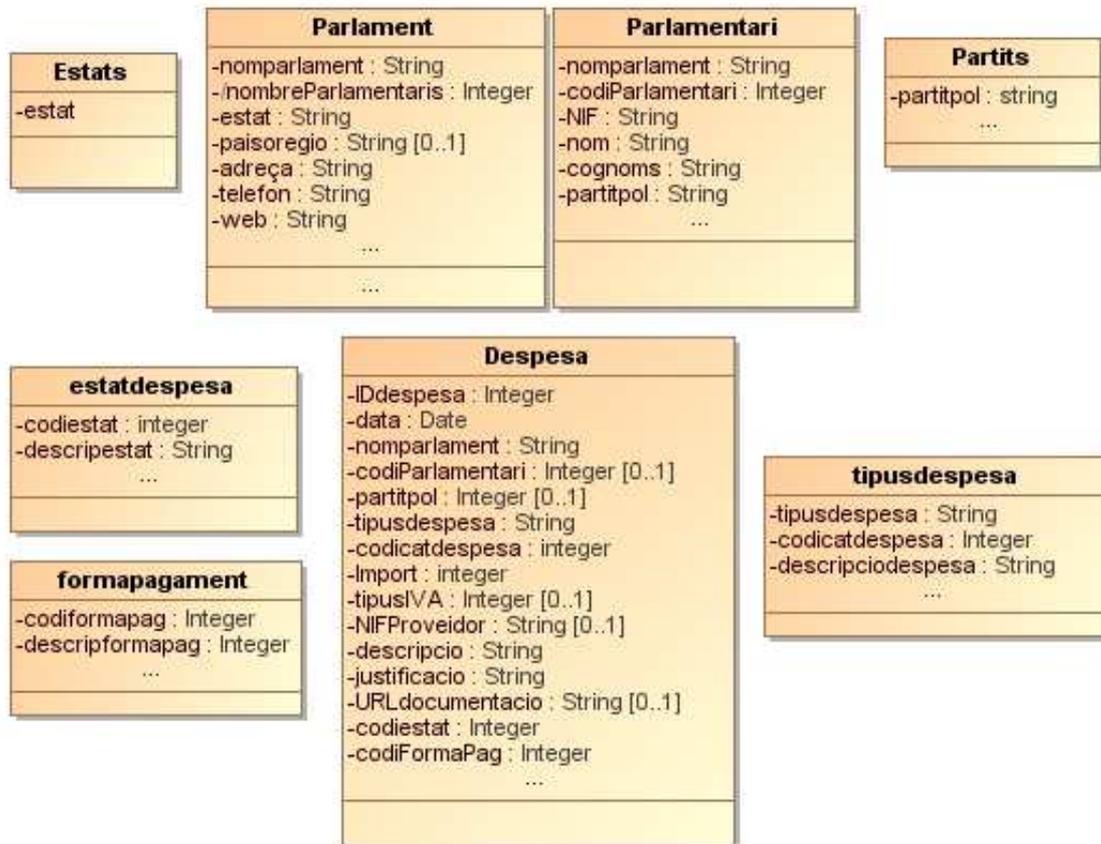


Figura 5. Base de datos relacional

3.4. Definición de las tablas

Una vez obtenido el modelo relacional, podemos pasar ya a concretar las tablas y campos necesarios

3.4.1. Estructura de las tablas

A continuación se representa el diseño de las tablas indicando:

- Nombre y significado de cada tabla
- Nombre y significado de cada campo
- Tipo de dato que se empleará en la base de datos
- Claves primarias, claves foráneas

Tabla: ESTATS		Significado: Estados de la CEE	
Campo	Significado	Tipo de dato	Comentarios
ESTAT	Estado de la CEE	VARCHAR2(40)	Clave primaria
Clave primaria: [ESTAT]			

Tabla: PARTITS		Significado: Partidos políticos de los parlamentarios	
Campo	Significado	Tipo de dato	Comentarios
PARTITPOL	Partido político al que pueden pertenecer un parlamentario	VARCHAR2(40)	No nulo (es clave primaria)
Clave primaria: [PARTITPOL]			

Tabla: PARLAMENTS		Significado: Cámaras parlamentarias de la CEE	
Campo	Significado	Tipo de dato	Comentarios
NOMPARLAMENT	Nombre del parlamento	VARCHAR2(40)	No nulo (es clave primaria)
NOMBREPARLAMENTARIS	Número de parlamentarios actual	NUMBER	No nulo
ESTAT	Estado al que pertenece el parlamento	VARCHAR2(40)	No nulo Clave foránea al campo ESTAT de la tabla ESTATS
PAISOREGIO	Región a la que pertenece el parlamento	VARCHAR2(20)	
ADRECA	Dirección del parlamento	VARCHAR2(100)	No nulo
TELEFON	Teléfono del parlamento	VARCHAR2(40)	No nulo
WEB	Página web del parlamento	VARCHAR2(40)	No nulo
Clave primaria: [NOMPARLAMENT]			

Tabla: PARLAMENTARIS		Significado: Parlamentarios de los diferentes parlamentos de la CEE	
Campo	Significado	Tipo de dato	Comentarios
NOMPARLAMENT	Nombre del parlamento al que pertenece el parlamentario	VARCHAR2(40)	Clave foránea al campo NOMPARLAMENT de la tabla PARLAMENTS No nulo (forma parte de la clave primaria)
CODIPARLAMENTARI	Código del parlamentario dentro de su parlamento	NUMBER	No nulo (forma parte de la clave primaria)
NIF	NIF del parlamentario	VARCHAR2(15)	No nulo
NOM	Nombre del parlamentario	VARCHAR2(40)	No nulo
COGNOMS	Apellidos del parlamentario	VARCHAR2(40)	No nulo
PARTITPOL	Partido político al que pertenece el parlamentario	VARCHAR2(40)	Clave foránea al campo PARTITPOL de la tabla PARTITS
Clave primaria: [NOMPARLAMENT, CODIPARLAMENTARI]			

Tabla: ESTATDESPESA		Significado: Situación en la que se encuentra un gasto (pendiente de aprobar, aprobado, pendiente de abonar, abonado)	
Campo	Significado	Tipo de dato	Comentarios
CODIESTAT	Código de la situación del gasto	INTEGER	No nulo (es clave primaria)
DESCRIPSTAT	Descripción de la situación del gasto	VARCHAR2 (40)	No nulo
Clave primaria: [CodiESTAT]			

Tabla: TIPUSDESPESA		Significado: Tipo de gasto: Se utiliza para categorizar cada gasto (por ejemplo puede ser: sueldo, dietas, transporte, etc).	
Campo	Significado	Tipo de dato	Comentarios
TIPUSDESPESA	Indica si ese tipo de gasto, se considera como GASTO GENERAL DE PARLAMENTO, o bien como GASTO PROPIO DE PARLAMENTARIO	VARCHAR2 (15)	No nulo (forma parte de la clave primaria)
CODICATDESPESA	Código de categoría del gasto	INTEGER	No nulo (forma parte de la clave primaria)
DESCRIPCIODESPESA	Descripción del tipo de gasto	VARCHAR2 (50)	No nulo
Clave primaria: [TIPUSDESPESA, CODICATDESPESA]			

Tabla: FORMAPAGAMENT		Significado: Forma en la que se abonará el gasto (VISA del parlamentario, transferencia bancaria efectuada por el parlamento)	
Campo	Significado	Tipo de dato	Comentarios
CODIFORMAPAG	Código de la forma de pago	INTEGER	No nulo (es clave primaria)
DESCRIFORMAPAG	Descripción de la forma de pago	VARCHAR2 (40)	No nulo
Clave primaria: [CODIFORMAPAG]			

Tabla: DESPESES		Significado: Gastos en que incurren los parlamentos y parlamentarios	
Campo	Significado	Tipo de dato	Comentarios
ID_DESPESA	Identificador único de cara registro de gasto (clave artificial)	NUMBER	No nulo (es clave primaria)
DATA_DESPESA	Fecha en la que se realiza el gasto	DATE	No Nulo
NOMPARLAMENT	Nombre del parlamento que realiza el gasto	VARCHAR2(40)	No nulo
CODIPARLAMENTARI	Código del parlamentario	NUMBER	
PARTITPOL	Nombre del partido político al que pertenece el parlamentario al registrar el gasto	VARCHAR2(40)	
TIPUSDESPESA	Indica si un gasto en concreto se considera GASTO GENERAL DE PARLAMENTO, o bien GASTO PROPIO DE PARLAMENTARIO	VARCHAR2 (15)	La combinación de estos dos campos, es clave foránea hacia los campos de mismo nombre, de la tabla TIPUSDESPESA
CODICATDESPESA	Código de categoría a la que pertenece el gasto	INTEGER	
IMPORT_DESPESA	Importe del gasto	NUMBER	No nulo
TIPUSIVA	Tipo de IVA	NUMBER	
NIFPROVEIDOR	Nif del proveedor, si existe	VARCHAR2 (15)	
DESCRIPCIO	Descripción textual del gasto	VARCHAR2 (40)	No nulo
JUSTIFICACIO	Justificación del gasto	VARCHAR2 (40)	No nulo
URLDOCUMENTACIO	URL con información escaneada de la documentación	VARCHAR2 (100)	
CODIESTAT	Código del estado en el que se encuentra el gasto	INTEGER	No nulo
CODIFORMAPAG	Código de la forma de pago con la que se abonará el gasto	INTEGER	No nulo
Clave primaria: [ID_DESPESA]			

3.4.2. Otros comentarios sobre las tablas

Podría haberse definido otras tablas auxiliares (por ejemplo, tabla con los posibles tipos de IVA, tabla de regiones). Se han descartado porque no constituían ninguna aportación al objetivo principal, que es el control de gastos de los parlamentos europeos.

3.4.3. Restricciones de integridad

Los campos sujetos a restricciones de integridad aparecen indicados como "CLAVE FORÁNEA" en la columna de COMENTARIOS (ver 3.4.1)

Al crear dichos campos en la BBDD, se utilizará el mecanismo *FOREING KEY* del SGBD; de este modo la integridad de la BBDD queda garantizada.

Adicionalmente, en las tablas maestras se controlará *-mediante los procedimientos de ALTA-* un posible intento de violación de las claves foráneas, y se devolverá el correspondiente error. Ello permitirá al usuario de esos procedimientos, comprender la causa por la cual no se ha podido insertar el registro y actuar en consecuencia.

Hay, dos situaciones de integridad referencial que solamente se controlarán en los procedimientos de ALTA o MODIFICACIÓN -y no a través del mecanismo *FOREING KEY-*. Afectan a la tabla de DESPESES

- La primera de ellas afecta al campo TIPUSDESPESA, y es la siguiente:

Cada GASTO, puede ser, o bien de tipo PARLAMENT, o bien de tipo PARLAMENTARI. Para controlar en cuál de esas dos situaciones nos encontramos se utiliza el campo TIPUSDESPESA, que podrá adoptar, por tanto, 2 valores posibles: "PARLAMENT" y "PARLAMENTARI".

Esta situación podría haberse controlado creando una nueva tabla con un campo para informar esos dos valores, y obligando a que el TIPUSDESPESA fuera clave foránea con respecto a ese campo. Se consideró, sin embargo, que existiendo solamente dos TIPUSDESPESA posibles, no merecía la pena crear una tabla solamente para esta finalidad.

- La segunda situación afecta al campo CODIPARLAMENTARI y se describe a continuación

En el caso de que un GASTO sea de tipo PARLAMENTARI, es necesario informar en el campo CODIPARLAMENTARI, un código de parlamentario que pertenezca al parlamento en cuestión. Si por el contrario, el GASTO es de tipo PARLAMENT, el CODIPARLAMENTARI debe ser NULO, porque se trata de un gasto general del parlamento, y no de parlamentario.

Como ya se ha indicado, controlaremos estas dos situaciones mediante el código adecuado los procedimientos de alta/modificación de la tabla DESPESES.

3.4.4. Tabla de Log

Como se ha visto en el apartado 2.1.4, es necesario almacenar el resultado de las llamadas a los procedimientos en una tabla de log

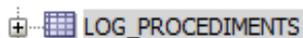


Figura 6. Tabla de Log

Además de los datos mínimos requeridos (nombre del procedimiento ejecutado, los parámetros de entrada y los de salida), se ha decidido incorporar también la fecha/hora de ejecución. En un entorno real sería aconsejable almacenar también el usuario que ejecuta el procedimiento.

La estructura de la tabla de log es la siguiente:

Tabla: LOG_PROCEDIMENTS		Significado: Parlamentarios de los diferentes parlamentos de la CEE	
Campo	Significado	Tipo de dato	Comentarios
ID_LOG	Identificador único del número de registro (clave artificial)	NUMBER	No nulo (es clave primaria)
DATA_PROCES	Fecha y hora de la ejecución	VARCHAR2(20)	No nulo
PROC_EXECUTAT	Nombre del procedimiento ejecutado	VARCHAR2(50)	No nulo
PARAM_ENTRADA	Nombre de las variables con las que se invoca el procedimiento, así como el valor con el que se han invocado	CLOB	
PARAM_SORTIDA	Breve información textual sobre el resultado de la ejecución. NULL si la ejecución ha sido errónea	CLOB	
RESULTAT_EXEC	OK si la ejecución ha sido correcta. En caso contrario contendrá el código de error y una descripción	CLOB	
Clave primaria: [ID_LOG]			

3.4.5. Claves primarias y uso de secuencias

Existen dos tablas para las que se ha previsto como clave primaria, un número secuencial creciente (*Clave artificial*). Se trata de las tablas de DESPESES y la de LOG_PROCEDIMENTS.

Para informar adecuadamente esos campos de clave primaria, se ha utilizado la funcionalidad de SECUENCIAS (“SEQUENCES”) del SGBD. A tal efecto, se han creado dos secuencias de la imagen (SEQ_ID_DESPESA Y SEQ_ID_LOG).

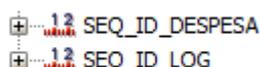


Figura 7. Secuencias

4. Procedimientos para el mantenimiento de las tablas maestras

Las 3 tablas maestras (PARLAMENT, PARLAMENTARI y DESPESA), deben mantenerse mediante procedimientos de alta/baja/modificación, tal como se comentó en el apartado 2.2

Los procedimientos para mantener las tablas maestras se encuentran en 6 archivos de script SQL, cuyo nombre coincide con el del propio procedimiento

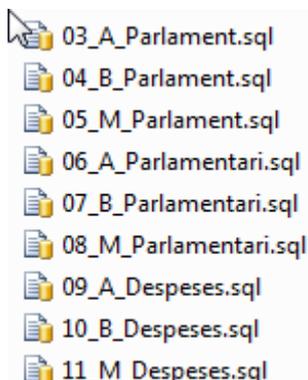


Figura 8. Mantenimiento de las tablas maestras

Asimismo, se ha implementado un procedimiento que se ejecuta como *trigger* y cuya utilidad se comenta en detalle en el apartado 4.3

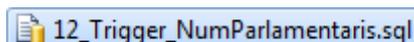


Figura 9: Trigger

4.1. Llamadas y parámetros

Los nombres de los procedimientos, así como los parámetros de entrada, se muestran en la siguiente tabla.

Función	Parámetros de entrada
A_PARLAMENT	NOMPARLAMENT, ESTAT, PAISOREGIO, ADRECA, TELEFON, WEB
B_PARLAMENT	NOMPARLAMENT
M_PARLAMENT	NOMPARLAMENT, PAISOREGIO, ADREGA, TELEFON,WEB
A_PARLAMENTARI	NOMPARLAMENT, CODIPARLAMENTARI, NIF, NOM, COGNOMS, PARTITPOL
B_PARLAMENTARI	NOMPARLAMENT, CODIPARLAMENTARI
M_PARLAMENTARI	NOMPARLAMENT, CODIPARLAMENTARI, NIF, NOM, COGNOMS,PARTITPOL
A_DESPESA	DATADESPESA, NOMPARLAMENT,CODIPARLAMENTARI, TIPUSDESPESA, CODICATDESPESA, IMPORT_DESPESA, TIPUSIVA, NIFPROVEIDOR, DESCRIPCIO, JUSTIFICACIO, URLDOCUMENTACIO, CODIESTAT, CODIFORMAPAG
B_DEPESA	ID_DESPESA
M_DESPESA	ID_DESPESA, IMPORT_DESPESA, TIPUSIVA, NIFPROVEIDOR, DESCRIPCIO, JUSTIFICACIO, URLDOCUMENTACIO, CODIESTAT, CODIFORMAPAG

Todos los procedimientos toman como parámetro de salida, un campo llamado RSP, en el cual se informará del resultado de la ejecución. Ello permitirá a quien lo utilice, comprobar si la ejecución ha sido o no correcta, y tomar decisiones al respecto

4.2. Decisiones de diseño sobre el mantenimiento de las tablas

Se han tomado las siguientes decisiones de diseño sobre el mantenimiento de las tablas maestras

Decisión	Justificación
Un parlamentario no se puede dar de baja si tiene gastos informados	Recuérdese que el propósito del proyecto, es un control histórico del mantenimiento de gastos; no una gestión completa de los parlamentos. El hecho de que un parlamentario con gastos registrados cause baja en un parlamento, es irrelevante a efectos del control de gastos que ya han ocurrido.
A un GASTO ya informado, no se puede cambiar el valor de TIPUSDESPESA (PARLAMENT->PARLAMENTARI, o viceversa)	No existe motivo alguno para tener que cambiar el campo TIPUSDESPESA de un gasto ya registrado, exceptuando la posibilidad de que el registro inicial hubiera sido erróneo. Si efectivamente hubiera ocurrido ese error, existe la posibilidad de borrar el gasto y volverlo a crear correctamente. Permitir la modificación de este cambio, complica de manera injustificada el mantenimiento del módulo estadístico

4.3. Control de errores

A continuación se muestran los códigos devueltos por los procedimientos en caso de error.

PROCEDIMIENTO: A_PARLAMENT	
CODIGO DE ERROR	SIGNIFICADO
ERR_ALTA_PARLAMENT_01	Ya existe el parlamento que se pretende dar de alta
ERR_ALTA_PARLAMENT_02	El parlamento que se quiere dar de alta es nulo
ERR_ALTA_PARLAMENT_03	El estado debe existir en la tabla de estados
ERR_ALTA_PARLAMENT_04	La dirección del parlamento no puede ser nula
ERR_ALTA_PARLAMENT_05	El teléfono del parlamento no puede ser nulo
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: B_PARLAMENT	
CODIGO DE ERROR	SIGNIFICADO
ERR_BAJA_PARLAMENT_01	No existe el parlamento que se quiere dar de baja
ERR_BAJA_PARLAMENT_02	El parlamento que se quiere dar de baja, tiene parlamentarios
ERR_BAJA_PARLAMENT_03	El parlamento que se quiere dar de baja, tiene gastos registrados (NOTA: este error solo se produce si un parlamento NO tiene gastos de parlamentarios , y en cambio SI tiene gastos propios de parlamento)
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: M_PARLAMENT	
CODIGO DE ERROR	SIGNIFICADO
ERR_MOD_PARLAMENT_01	No existe el parlamento que se quiere modificar
ERR_MOD_PARLAMENT_02	La dirección del parlamento no puede ser nula
ERR_MOD_PARLAMENT_03	El teléfono del parlamento no puede ser nulo
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: A_PARLAMENTARI	
CODIGO DE ERROR	SIGNIFICADO
ERR_ALTA_PARLAMENTARI_01	El parlamento no existe
ERR_ALTA_PARLAMENTARI_02	El código del parlamentario ya existe para ese parlamento, o bien es nulo
ERR_ALTA_PARLAMENTARI_03	El partido político no existe
ERR_ALTA_PARLAMENTARI_04	El NIF del parlamentario no puede ser nulo
ERR_ALTA_PARLAMENTARI_05	El nombre del parlamentario no puede ser nulo
ERR_ALTA_PARLAMENTARI_06	Los apellidos del parlamentario no pueden ser nulos
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: B_PARLAMENTARI	
CODIGO DE ERROR	SIGNIFICADO
ERR_BAJA_PARLAMENTARI_01	No existe el parlamentario que se quiere dar de baja
ERR_BAJA_PARLAMENTARI_02	El parlamentario tiene gastos asociados
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: M_PARLAMENTARI	
CODIGO DE ERROR	SIGNIFICADO
ERR_MOD_PARLAMENTARI_01	El parlamentario no existe
ERR_MOD_PARLAMENTARI_02	El partido político no existe o es nulo
ERR_MOD_PARLAMENTARI_03	El NIF del parlamentario no puede ser nulo
ERR_MOD_PARLAMENTARI_04	El nombre del parlamentario no puede ser nulo
ERR_MOD_PARLAMENTARI_05	Los apellidos del parlamentario no pueden ser nulos
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: A_DESPESA	
CODIGO DE ERROR	SIGNIFICADO
ERR_ALTA_DESP_01	El parlamento no existe
ERR_ALTA_DESP_02	El tipo de gasto debe ser *PARLAMENT* o *PARLAMENTARI* obligatoriamente
ERR_ALTA_DESP_03	Se está informando un tipo de gasto *PARLAMENTARI*, pero el código de parlamentario no existe o no está informado
ERR_ALTA_DESP_04	No existe en código de categoría de gasto en la tabla TIPUSDESPESA
ERR_ALTA_DESP_05	El importe del gasto es nulo
ERR_ALTA_DESP_06	La descripción del gasto es nula
ERR_ALTA_DESP_07	La justificación del gasto es nula
ERR_ALTA_DESP_08	No existe el código de gasto en la tabla ESTATDESPESA, o bien es nulo
ERR_ALTA_DESP_09	No existe el código de la forma de pago en la tabla FORMAPAGAMENT, o bien es nulo
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO: B_DESPESA	
CODIGO DE ERROR	SIGNIFICADO
ERR_BAJA_DESP_01	No existe el gasto que se quiere dar de baja
Error emitido por el SGBD	Ha ocurrido a otro tipo de error

PROCEDIMIENTO: M_DESPESA	
CODIGO DE ERROR	SIGNIFICADO
ERR_MOD_DESP_01	El gasto no existe
ERR_MOD_DESP_02	Importe del gasto no informado
ERR_MOD_DESP_03	Descripción del gasto no informada
ERR_MOD_DESP_04	Justificación del gasto no informada
ERR_MOD_DESP_05	No existe el código de gasto en la tabla ESTATDESPESA, o bien es nulo
ERR_MOD_DESP_06	No existe el código de la forma de pago en la tabla FORMAPAGAMENT, o bien es nulo
Error emitido por el SGBD	Ha ocurrido otro tipo de error

4.4. Dependencias entre tablas y triggers

Según los requisitos del proyecto, la tabla PARLAMENTARIS debe almacenar el número actual de parlamentarios. Ello implica aumentar o disminuir el número de parlamentarios cuando se produzcan altas o bajas de éstos.

Para conseguir esto de forma eficaz, y tal como ya se adelantaba en la introducción del capítulo 4, se ha implementado un TRIGGER que realizará este proceso (ver imagen).

```
CREATE OR REPLACE
TRIGGER TRIG_ACTUALITZA_NUMPAR AFTER INSERT OR DELETE ON PARLAMENTARIS
```

Figura 10. Trigger para actualizar parlamentarios

Como puede comprobarse, el *Trigger* se desencadena cada vez que se inserta o se borra un parlamentario. El código del *trigger* se ocupa de hacer un *UPDATE* contra el campo PARLAMENT.NUMPARLAMENTARIS.

Podría haberse este incremento/decremento del número de parlamentarios, mediante código en los procedimientos de alta/baja de parlamentarios. Sin embargo es más adecuado utilizar un TRIGGER. Si éste se ha implementado de forma correcta, la gestión es realizada por el SGBD y es muy fiable.

5. Procedimientos de consulta

Tal como se comentaba anteriormente, uno de los requisitos consistía en implementar procedimientos de consulta que mostraran unas informaciones específicas (la cuales pueden verse en la tabla del apartado 2.3). A tal efecto, se han implementado 6 procedimientos de consulta:

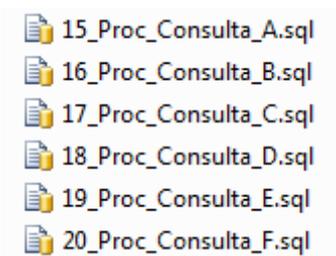


Figura 11. Procedimientos de consulta

5.1. Llamadas y parámetros

Los nombres de los procedimientos de consulta, así como los parámetros de entrada, se muestran en la siguiente tabla.

Consulta	Procedimiento	Datos de entrada
A	proc_DespesesAnyParlamentari	PARLAMENT, ANY, CODIPARLAMENTARI
B	proc_ParlamentarisDespDir	PARLAMENT, ANY
C	proc_DespTotalTotsParlam	ANY
D	proc_DespPerCategorias	NOMPARLAMENT, ANY
E	proc_ParlamDespejaSupAMitjana	NOMPARLAMENT, ANY
F	proc_EstatContableParlaments	ANY

5.2. Salida de los procedimientos de consulta

En principio, los procedimientos deberían permitir que un programador de una capa superior del software, obtuviera los resultados y los muestre al usuario a través de alguna interface adecuada. *En consecuencia, la salida debería obtenerse mediante un parámetro.* Sin embargo, *necesitábamos mostrar los resultados de algún modo para el juego de pruebas.* Por ello, se ha decidido mostrar la salida en la propia consola del entorno SQL, -mediante la instrucción DBMS_OUTPUT.PUT_LINE-, elaborando así una sencilla interface de salida.

En resumen, *la salida de los procedimientos de consulta se obtiene únicamente por pantalla, no por parámetros de salida.*

5.3. Control de errores

A continuación se muestran los posibles errores devueltos por los procedimientos de consulta

PROCEDIMIENTO A: proc_DespesesAnyParlamentari	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_A_01	El parlamento no existe
ERROR_PROC_A_02	El parlamentario no existe en ese parlamento
ERROR_PROC_A_03	No hay gastos para ese parlamentario en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO B: proc_ParlamentarisDespDir	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_B_01	El parlamento no existe
ERROR_PROC_B_02	No hay gastos para ese parlamento en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO C: proc_DespTotalTotsParlam	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_C_01	No se registraron gastos en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO D: proc_DespPerCategorias	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_D_01	El parlamento no existe
ERROR_PROC_D_02	No hay gastos para ese parlamento en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO E: proc_ParlamDespejaSupAMitjana	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_E_01	El parlamento no existe
ERROR_PROC_E_02	No hay gastos para ese parlamento en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

PROCEDIMIENTO F: proc_EstatContableParlaments	
CODIGO DE ERROR	SIGNIFICADO
ERROR_PROC_F_01	No se registraron gastos en el año solicitado
Error emitido por el SGBD	Ha ocurrido otro tipo de error

5.4. Comentarios sobre los procedimientos de consulta

Merece la pena hacer algunas aclaraciones sobre los datos que se obtienen en los procedimientos de consulta

- Los gastos que se obtienen, son siempre importes sin IVA. El motivo es que en teoría el IVA puede ser distinto para distintos estados de la comunidad europea y para distintos conceptos. Para que los importes de los gastos puedan compararse entre los distintos parlamentos de forma representativa, no deberían incluir el IVA
- A efectos de los totales, se tienen en cuenta todos los gastos, con independencia de la situación en la que se encuentren (pendiente de aprobar, aprobado, etc.). Ello es así en todos los procedimientos con la excepción del procedimiento F, en el cual se pide precisamente el desglose de gastos según su situación contable.
- Los procedimientos c y e calculan medias del gasto de los parlamentarios. Se ha decidido utilizar como número de parlamentarios, *el número de parlamentarios que tienen algún gastos registrado en el período que se trata* (y no el número actual de parlamentarios, que podría ser distinto)

6. Registro de llamadas a los procedimientos

Como se ha comentado anteriormente (véase apartado 2.4), los requerimientos exigen registrar en una tabla de *log*, (ver apartado 3.4.4), el resultado de las llamadas a los procedimientos.

6.1. Inserción de registros en la tabla de log

El código para la inserción de los registros en la tabla de log, se localiza en el cuerpo de los mismos procedimientos cuya llamada va a registrarse. Si el procedimiento tiene éxito, normalmente el código se ejecuta justo antes del fragmento de código encargado del control de errores; en caso contrario, el código de control de error (en la zona de *EXCEPTION* del procedimiento) es el que efectúa la inserción.

La siguiente imagen muestra el contenido de la tabla de Log en un cierto momento

ID_LOG	DATA_PROCES	PROC_EXECUTAT	PARAM_ENTRADA	PARAM_SORTIDA	RESULTAT_EXEC
94	20959 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:02/05/...	(null)	OK
95	20960 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:14/04/...	(null)	OK
96	20961 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:01/03/...	(null)	OK
97	20962 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:20/08/...	(null)	OK
98	20963 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:19/05/...	(null)	OK
99	20964 26-12-2013 19:10:33	proc_A_DESPESA	DATA_DESPESA:13/08/...	(null)	OK
100	20965 26-12-2013 19:10:33	proc A DESPESA	DATA DESPESA:30/09/...	(null)	OK

6.2. Consulta de la tabla de Log

Típicamente, la consulta de la tabla de Log se realizaría mediante sentencias SELECT. Sin embargo, durante la realización de las pruebas, se ha visto que era conveniente disponer de algún medio sencillo para comprobar la traza dejada en el log por el último procedimiento ejecutado (especialmente en lo que se refiere a los procedimientos de alta/baja/modificación de las tablas maestras). Por ello, se ha creado un procedimiento llamado *Ultima_linea_log*, contenido en el script de la imagen.

 21_Ultima_Linea_Log.sql

Figura 12. Script "Ultima Linea Log"

Este procedimiento, escribe por pantalla algunos campos del último registro insertado en la tabla de log, y se emplea precisamente para hacer una comprobación del resultado de los procedimientos de alta/baja/modificación (ver 10.1.1)

```
CREATE OR REPLACE PROCEDURE ULTIMA_LINEA_LOG as
C_LOG LOG_PROCEDIMENTS%ROWTYPE;
BEGIN
SELECT * INTO C_LOG FROM LOG_PROCEDIMENTS WHERE ROWNUM=1 ORDER BY ID_LOG DESC;
DBMS_OUTPUT.PUT_LINE('*****');
DBMS_OUTPUT.PUT_LINE('EJECUTADO: '||C_LOG.PROC_EXECUTAT);
DBMS_OUTPUT.PUT_LINE('ENTRADA : '||C_LOG.PARAM_ENTRADA);
DBMS_OUTPUT.PUT_LINE('RESULTADO: '||C_LOG.RESULTAT_EXEC);
END ULTIMA_LINEA_LOG;
```

Figura 13. Cuerpo del procedimiento "Ultima_linea_log"

7. Módulo estadístico

Recordemos que la finalidad del módulo estadístico (ver apartado 2.5) es tener pre-calculados una serie de informes sobre los gastos de los parlamentos y parlamentarios.

Para dar respuesta a este requisito, se han implementado los siguientes elementos:

- Una serie de tablas (tantas como estadísticas diferentes hay que calcular), con los campos necesarios para albergar el resultado de las estadísticas

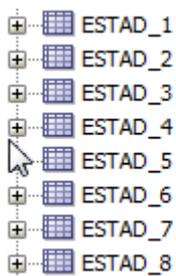


Figura 14. Tablas del módulo estadístico

- Un único procedimiento, que será responsable de recalculando todas las estadísticas cada vez que se produzca una alta, baja o modificación de un gasto (hay que tener en cuenta que una modificación en un único gasto, puede tener consecuencias en todas las estadísticas)

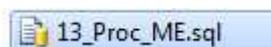


Figura 15. Procedimiento para el módulo estadístico

7.1. Diseño de las tablas del módulo estadístico

A continuación se detalla el diseño de las tablas del módulo estadístico

Tabla: ESTAD_1		Para cada parlamento: Suma de sus gastos totales, durante los 4 últimos años (contados a partir del último año en el que ese parlamento ha registrado algún gasto)	
Campo	Significado	Tipo de dato	Comentarios
NOMPARLAMENT	Parlamento	VARCHAR2(40)	No nulo (es clave primaria)
IMPORT_DESPESA	Suma de los gastos de los 4 últimos años del parlamento	NUMBER	
PERIODE	Información textual del período al que nos referimos	VARCHAR2(40)	
Clave primaria: [NOMPARLAMENT]			

Tabla: ESTAD_2		Para cada parlamento y año: Diferencia en euros, entre el parlamentario que ha registrado más gastos asociados, y el que ha gastado menos, en ese parlamento y durante ese año	
Campo	Significado	Tipo de dato	Comentarios
NOMPARLAMENT	Parlamento	VARCHAR2(40)	No nulo (forma parte de la clave primaria)
ANY_ESTAD	Año	INTEGER	No nulo (forma parte de la clave primaria)
IMPORT_DIFERENCIA	Diferencia solicitada	NUMBER	
Clave primaria: [NOMPARLAMENT, ANY_ESTAD]			

Tabla: ESTAD_3		Para cada año: El parlamento que ha tenido el mayor importe de gastos durante ese año	
Campo	Significado	Tipo de dato	Comentarios
ANY_ESTAD	Año	INTEGER	No nulo (es clave primaria)
NOMPARLAMENT	Parlamento	VARCHAR2(40)	
MAXIM_DESPESA	Gasto de ese parlamento	NUMBER	
Clave primaria: [ANY_ESTAD]			

Tabla: ESTAD_4		Para cada parlamentario: Suma de sus gastos totales, durante los 4 últimos años (contados a partir del último año en el que ese parlamentario ha registrado algún gasto)	
Campo	Significado	Tipo de dato	Comentarios
CODIPARLAMENTARI	Código del parlamentario	NUMBER	No nulo (forma parte de la clave primaria)
NOMPARLAMENT	Nombre del parlamento al que pertenece ese parlamentario	VARCHAR2(40)	No nulo (forma parte de la clave primaria)
NOM	Nombre del parlamentario	VARCHAR2(40)	
COGNOMS	Apellidos del parlamentario	VARCHAR2(40)	
PERIODE	Información textual del período al que nos referimos	VARCHAR2(40)	
DESPESA_PERIODE	Suma de los gastos de los 4 últimos años del parlamento	NUMBER	
Clave primaria: [CODIPARLAMENTARI, NOMPARLAMENT]			

Tabla: ESTAD_5		Para cada año: El gasto medio de la figura del parlamentario (teniendo en cuenta todos los parlamentarios de todos los parlamentos)	
Campo	Significado	Tipo de dato	Comentarios
ANY_ESTAD	Año	INTEGER	No nulo (es clave primaria)
DESPESA_MITJA	Media del gasto del parlamentario	NUMBER	
Clave primaria: [ANY_ESTAD]			

Tabla: ESTAD_6		Para cada año: El parlamentario con el máximo importe de gastos asociados, y el que ha tenido el mínimo importe de gastos asociados	
Campo	Significado	Tipo de dato	Comentarios
ANY_ESTAD	Año	NUMBER	No nulo (forma parte de la clave primaria)
CODIPARLAMENTARI_MAX	Código del parlamentario con el máximo gasto	NUMBER	
NOMPARLAMENT_MAX	Nombre del parlamento al que pertenece el parlamentario de máximo gasto	VARCHAR2(40)	
NOM_MAX	Nombre del parlamentario de máximo gasto	VARCHAR2(40)	
COGNOMS_MAX	Apellidos del parlamentario de máximo gasto	VARCHAR2(40)	
DESPEsa_MAX	Gasto del parlamentario de máximo gasto	NUMBER	
CODIPARLAMENTARI_MIN	Código del parlamentario con el mínimo gasto	NUMBER	
NOMPARLAMENT_MIN	Nombre del parlamento al que pertenece el parlamentario de mínimo gasto	VARCHAR2(40)	
NOM_MIN	Nombre del parlamentario de mínimo gasto	VARCHAR2(40)	
COGNOMS_MIN	Apellidos del parlamentario de mínimo gasto	VARCHAR2(40)	
DESPEsa_MIN	Gasto del parlamentario de mínimo gasto	NUMBER	
Clave primaria: [ANY_ESTAD]			

Tabla: ESTAD_7		Para cada año: El parlamento cuya media de gasto por parlamentario, ha sido mayor durante ese año, así como el valor de dicha media	
Campo	Significado	Tipo de dato	Comentarios
ANY_ESTAD	Año	INTEGER	No nulo (es clave primaria)
NOMPARLAMENT	Parlamento con mayor media de gasto por parlamentario	VARCHAR2(40)	
MITJA_DESPESA	Gasto medio de ese parlamento	NUMBER	
Clave primaria: [ANY_ESTAD]			

Tabla: ESTAD_8		Para cada año: La suma de todos los gastos de todos los parlamentos durante ese año (incluyendo tanto gastos de parlamentos como de parlamentarios)	
Campo	Significado	Tipo de dato	Comentarios
ANY_ESTAD	Año	INTEGER	No nulo (es clave primaria)
DESPEsa	Suma del gasto de	NUMBER	
Clave primaria: [ANY_ESTAD]			

7.2. Implementación del procedimiento que calcula las estadísticas

El procedimiento que realiza los cálculos del módulo estadístico recibe como nombre *ESTADISTIQUES*.

Se comentan a continuación algunos aspectos acerca de cómo se ha implementado el procedimiento del módulo estadístico

7.2.1. Parámetros de entrada y salida

- Los parámetros de entrada que toma son:

Parámetros procedimiento ESTADISTIQUES	Significado
DATA_DESPESA	Fecha del gasto
NOMPARLAMENT	Parlamento afectado
CODIPARLAMENTARI	Código de parlamentario (puede ser nulo si se trata de un gasto que afecta solo al parlamento)
IMPORT_DESPESA	Importe del gasto

- El único parámetro de salida que toma, es el campo RSP que se utilizará para el control de errores.

7.2.2. Control de errores

El procedimiento deberá ejecutarse justo después de que se ha registrado un nuevo gasto. Previamente los datos del gasto se han validado (ver apartado 4.3). En consecuencia, no es necesario implementar nuevamente un control de errores en el procedimiento del módulo estadístico, para comprobar la validez de los parámetros de entrada.

De todas formas, y dado que en teoría pueden suceder otro tipo de errores, se han tomado las siguientes decisiones

- Se implementará un control de errores genérico (“*EXCEPTION-WHEN OTHERS*”)
- Se registrará en la tabla de LOG_PROCEDIMENTS, el resultado de la llamada al módulo estadístico, *solamente en el caso de que haya sucedido algún error*.

7.2.3. Llamada al procedimiento

Se han valorado 2 posibilidades para actualizar las estadísticas

- Invocar al procedimiento dentro del cuerpo de los procedimientos que realizan altas/bajas/modificaciones de gastos
- Implementarlo como un trigger a ejecutarse cada vez que se produzca un INSERT, DELETE o UPDATE en la tabla ESTADISTIQUES

Se ha visto que técnicamente la vía del trigger presentaba complicaciones. El motivo es el siguiente:

- El procedimiento de cálculo de las estadísticas, *necesita hacer uno de la propia tabla ESTADISTIQUES para poder hacer sus cálculos*. Además, debe invocarse *cuando los nuevos datos están ya grabados, es decir, cuando ha terminado la transacción*. (Técnicamente significa que el recalcular debe hacerse AFTER INSERT, DELETE OR UPDATE)
- Resulta que, cuando se desencadena un trigger AFTER INSERT, DELETE OR UPDATE, *el trigger forma parte de la propia transacción. la transacción no acaba hasta que el propio trigger no acaba. Eso bloquea la propia tabla ESTADISTIQUES que necesitamos consultar*.

En resumen, implementar el trigger AFTER INSERT, DELETE OR UPDATE, *bloquea la propia tabla ESTADISTIQUES que necesitamos consultar para recalcular las estadísticas*.

Por tanto se ha decidido utilizar la primera opción mencionada: *Invocar al procedimiento de cálculo de las estadísticas, directamente dentro del cuerpo de los procedimientos que realizan altas / bajas / modificaciones de gastos, justo después de que se haya modificado la tabla ESTADISTIQUES*.

7.3. Otros comentarios sobre el módulo estadístico

Las estadísticas 5 y 7 calculan medias del gasto de los parlamentarios. Al igual que en el caso de los procedimientos de consulta (ver 5.4), las medias se han calculado en base *al número de parlamentarios que han registrado algún gasto en el período implicado* (y no en base al número actual de los parlamentarios guardado en la tabla de parlamentos, el cual podría ser distinto si con el transcurso del tiempo se han producido altas o bajas de parlamentarios).

8. Diseño de los datos de prueba

Los requerimientos exigen (ver apartado 2.6) la creación de un juego de datos suficientemente extenso como para poder poblar todas las tablas implicadas, y probar de forma satisfactoria todos los aspectos del producto software. Además ha parecido interesante dotar a los datos, en la medida de lo posible, de un cierto realismo.

8.1. Datos de parlamentos y parlamentarios

Se han creado datos correspondientes a 5 cámaras parlamentarias. Para cada una, se han creado una representación parlamentaria reducida pero suficiente para realizar las pruebas, tal como se muestra en la siguiente tabla

Cámaras parlamentarias	Número de parlamentarios
PARLAMENT DE CATALUNYA	12
CONGRESO DE LOS DIPUTADOS	13
PARLAMENTO VASCO	9
ASSEMBLEE NATIONALE	11
DEUSTCHER BUNDESTAG	15

Vemos que tenemos, por tanto, un total de 60 parlamentarios.

8.2. Datos de histórico de gastos

De los requerimientos (ver 2.3 y 2.5) se observa que, tanto los procedimientos de consulta como las estadísticas, hacen referencia a períodos anuales de gastos. Algunas de las estadísticas hacen referencia a períodos de los últimos 4 años. En consecuencia, se ha considerado suficiente diseñar un conjunto de gastos que abarque un período de 5 años.

Era necesario crear gastos tanto de PARLAMENTS como de PARLAMENTARIS. Un gasto propio de los parlamentarios, -que además se menciona en la definición del proyecto-, es el SUELDO. Por tanto, se han creado gastos de sueldos mensuales para los 60 diputados a lo largo de 5 años (en total 3.600 registros). A ello se han añadido de forma aleatoria otros gastos correspondientes tanto a parlamentos como a parlamentarios, hasta un total de 4.110 registros. La distribución por categorías se muestra en la siguiente tabla.

Tipo de gasto	Concepto (descripción textual de la tabla TIPUS_DESPESA)	Número de gastos en la tabla de histórico de gastos
PARLAMENTARI	Sou mensual	3600
PARLAMENTARI	Dinar fora del parlament	50
PARLAMENTARI	Mobil o tablet	40
PARLAMENTARI	Ordinador	15
PARLAMENTARI	Transport public	35
PARLAMENTARI	Dieta commissio parlamentaria	15
PARLAMENTARI	Escorta o xofer parlamentari	5
PARLAMENTARI	Lloguer cotxe parlamentari	70
PARLAMENTARI	Altres despeses parlamentari	50
PARLAMENT	Escorta o xofer parlament	70
PARLAMENT	Lloguer cotxe parlament	45
PARLAMENT	Seguretat parlament	40
PARLAMENT	Material oficina parlament	40
PARLAMENT	Altres despeses parlament	35

8.3. Tablas auxiliares

El resto de las tablas implicadas (ESTATS, PARTITS, TIPUSDESPESA, ESTATDESPESA, FORMAPAGAMENT) se han poblado con los datos necesarios para poder alimentar correctamente las tres tablas maestras. Se muestran a continuación los datos cargados en cada una de ellas.

Tabla	Datos cargados																																													
ESTATS	<table border="1"> <tr><td>ESTAT</td></tr> <tr><td>ALEMANIA</td></tr> <tr><td>ESPAÑA</td></tr> <tr><td>FRANCIA</td></tr> </table>	ESTAT	ALEMANIA	ESPAÑA	FRANCIA																																									
ESTAT																																														
ALEMANIA																																														
ESPAÑA																																														
FRANCIA																																														
FORMAPAGAMENT	<table border="1"> <tr> <td>CODIFORMAPAG</td> <td>DESCRIFORMAPAG</td> </tr> <tr> <td>1</td> <td>TRANSF. BANCARIA PARLAMENT</td> </tr> <tr> <td>2</td> <td>VISA PARLAMENTARI</td> </tr> </table>	CODIFORMAPAG	DESCRIFORMAPAG	1	TRANSF. BANCARIA PARLAMENT	2	VISA PARLAMENTARI																																							
CODIFORMAPAG	DESCRIFORMAPAG																																													
1	TRANSF. BANCARIA PARLAMENT																																													
2	VISA PARLAMENTARI																																													
TIPUSDESPESA	<table border="1"> <thead> <tr> <th>TIPUSDESPESA</th> <th>CODICATDESPESA</th> <th>DESCRIPCIODESPESA</th> </tr> </thead> <tbody> <tr><td>PARLAMENTARI</td><td>1</td><td>Sou mensual</td></tr> <tr><td>PARLAMENTARI</td><td>2</td><td>Dinar fora del parlament</td></tr> <tr><td>PARLAMENTARI</td><td>3</td><td>Mobil o tablet</td></tr> <tr><td>PARLAMENTARI</td><td>4</td><td>Ordinador</td></tr> <tr><td>PARLAMENTARI</td><td>5</td><td>Transport public</td></tr> <tr><td>PARLAMENTARI</td><td>6</td><td>Dieta commissio parlamentaria</td></tr> <tr><td>PARLAMENTARI</td><td>7</td><td>Escorta o xofer parlamentari</td></tr> <tr><td>PARLAMENTARI</td><td>8</td><td>Lloguer cotxe parlamentari</td></tr> <tr><td>PARLAMENTARI</td><td>9</td><td>Altres despeses parlamentari</td></tr> <tr><td>PARLAMENT</td><td>1</td><td>Escorta o xofer parlament</td></tr> <tr><td>PARLAMENT</td><td>2</td><td>Lloguer cotxe parlament</td></tr> <tr><td>PARLAMENT</td><td>3</td><td>Seguretat parlament</td></tr> <tr><td>PARLAMENT</td><td>4</td><td>Material oficina parlament</td></tr> <tr><td>PARLAMENT</td><td>5</td><td>Altres despeses parlament</td></tr> </tbody> </table>	TIPUSDESPESA	CODICATDESPESA	DESCRIPCIODESPESA	PARLAMENTARI	1	Sou mensual	PARLAMENTARI	2	Dinar fora del parlament	PARLAMENTARI	3	Mobil o tablet	PARLAMENTARI	4	Ordinador	PARLAMENTARI	5	Transport public	PARLAMENTARI	6	Dieta commissio parlamentaria	PARLAMENTARI	7	Escorta o xofer parlamentari	PARLAMENTARI	8	Lloguer cotxe parlamentari	PARLAMENTARI	9	Altres despeses parlamentari	PARLAMENT	1	Escorta o xofer parlament	PARLAMENT	2	Lloguer cotxe parlament	PARLAMENT	3	Seguretat parlament	PARLAMENT	4	Material oficina parlament	PARLAMENT	5	Altres despeses parlament
TIPUSDESPESA	CODICATDESPESA	DESCRIPCIODESPESA																																												
PARLAMENTARI	1	Sou mensual																																												
PARLAMENTARI	2	Dinar fora del parlament																																												
PARLAMENTARI	3	Mobil o tablet																																												
PARLAMENTARI	4	Ordinador																																												
PARLAMENTARI	5	Transport public																																												
PARLAMENTARI	6	Dieta commissio parlamentaria																																												
PARLAMENTARI	7	Escorta o xofer parlamentari																																												
PARLAMENTARI	8	Lloguer cotxe parlamentari																																												
PARLAMENTARI	9	Altres despeses parlamentari																																												
PARLAMENT	1	Escorta o xofer parlament																																												
PARLAMENT	2	Lloguer cotxe parlament																																												
PARLAMENT	3	Seguretat parlament																																												
PARLAMENT	4	Material oficina parlament																																												
PARLAMENT	5	Altres despeses parlament																																												

Tabla	Datos cargados																	
PARTITS	<table border="1"> <thead> <tr> <th data-bbox="802 230 1219 259">PARTITPOL</th> </tr> </thead> <tbody> <tr><td data-bbox="802 259 1219 288">CDU/CSU</td></tr> <tr><td data-bbox="802 288 1219 318">CONVERGENCIA I UNIO</td></tr> <tr><td data-bbox="802 318 1219 347">DIE LINKE</td></tr> <tr><td data-bbox="802 347 1219 376">ECOLOGISTE</td></tr> <tr><td data-bbox="802 376 1219 405">EH BILDU</td></tr> <tr><td data-bbox="802 405 1219 434">ESQUERRA REPUBLICANA DE CATALUNYA</td></tr> <tr><td data-bbox="802 434 1219 463">INICIATIVA PER CAT VERDS</td></tr> <tr><td data-bbox="802 463 1219 492">PARTIDO NACIONALISTA VASCO</td></tr> <tr><td data-bbox="802 492 1219 521">PARTIDO POPULAR</td></tr> <tr><td data-bbox="802 521 1219 551">PARTIDO SOCIALISTA OBRERO ESPAÑOL</td></tr> <tr><td data-bbox="802 551 1219 580">PARTIT DELS SOCIALISTES DE CATALUNYA</td></tr> <tr><td data-bbox="802 580 1219 609">PSOE SOCIALISTAS VASCOS</td></tr> <tr><td data-bbox="802 609 1219 638">SOCIALISTE, REPUBLICAIN ET CITOYEN</td></tr> <tr><td data-bbox="802 638 1219 667">SPD</td></tr> <tr><td data-bbox="802 667 1219 696">UNION DES DEMOCRATES ET INDEPENDANTIS</td></tr> <tr><td data-bbox="802 696 1219 725">UNION POUR UN MOUVEMENT POPULAIRE</td></tr> </tbody> </table>	PARTITPOL	CDU/CSU	CONVERGENCIA I UNIO	DIE LINKE	ECOLOGISTE	EH BILDU	ESQUERRA REPUBLICANA DE CATALUNYA	INICIATIVA PER CAT VERDS	PARTIDO NACIONALISTA VASCO	PARTIDO POPULAR	PARTIDO SOCIALISTA OBRERO ESPAÑOL	PARTIT DELS SOCIALISTES DE CATALUNYA	PSOE SOCIALISTAS VASCOS	SOCIALISTE, REPUBLICAIN ET CITOYEN	SPD	UNION DES DEMOCRATES ET INDEPENDANTIS	UNION POUR UN MOUVEMENT POPULAIRE
PARTITPOL																		
CDU/CSU																		
CONVERGENCIA I UNIO																		
DIE LINKE																		
ECOLOGISTE																		
EH BILDU																		
ESQUERRA REPUBLICANA DE CATALUNYA																		
INICIATIVA PER CAT VERDS																		
PARTIDO NACIONALISTA VASCO																		
PARTIDO POPULAR																		
PARTIDO SOCIALISTA OBRERO ESPAÑOL																		
PARTIT DELS SOCIALISTES DE CATALUNYA																		
PSOE SOCIALISTAS VASCOS																		
SOCIALISTE, REPUBLICAIN ET CITOYEN																		
SPD																		
UNION DES DEMOCRATES ET INDEPENDANTIS																		
UNION POUR UN MOUVEMENT POPULAIRE																		
ESTATDESPESA	<table border="1"> <thead> <tr> <th data-bbox="839 786 1177 815">CODIESTAT</th> <th data-bbox="979 786 1177 815">DESCRIPSTAT</th> </tr> </thead> <tbody> <tr><td data-bbox="839 815 1177 844">1</td><td data-bbox="979 815 1177 844">PENDENT APROVACIO</td></tr> <tr><td data-bbox="839 844 1177 873">2</td><td data-bbox="979 844 1177 873">APROVADA</td></tr> <tr><td data-bbox="839 873 1177 902">3</td><td data-bbox="979 873 1177 902">PENDENT ABONAMENT</td></tr> <tr><td data-bbox="839 902 1177 931">4</td><td data-bbox="979 902 1177 931">ABONADA</td></tr> </tbody> </table>	CODIESTAT	DESCRIPSTAT	1	PENDENT APROVACIO	2	APROVADA	3	PENDENT ABONAMENT	4	ABONADA							
CODIESTAT	DESCRIPSTAT																	
1	PENDENT APROVACIO																	
2	APROVADA																	
3	PENDENT ABONAMENT																	
4	ABONADA																	

9. Descripción y carga del producto software

La solución software obtenida, consiste en una serie de scripts SQL que incluyen la creación en el SGBD, de todas las funcionalidades descritas en los apartados anteriores, así como los datos de prueba.

9.1. Descripción del producto

En la siguiente tabla de muestran cada uno de los archivos SQL junto con una breve descripción de su funcionalidad.

Script del producto	Función
00_BorrarTablas.sql	Borra , si la hubiera, las tablas y las secuencias del mismo nombre que las utilizadas por el producto.
01_CrearTablas.sql	Crea todas las tablas necesarias.
02_Cargar_Tablas_Auxiliares.sql	Puebla con datos mediante instrucción INSERT INTO, todas las tablas excepto las 3 tablas maestras.
03_A_Parlament.sql	Script que realiza el alta de parlamentos.
04_B_Parlament.sql	Script que realiza la baja de parlamentos.
05_M_Parlament.sql	Script que realiza la modificación de parlamentos.
06_A_Parlamentari.sql	Script que realiza el alta de parlamentarios.
07_B_Parlamentari.sql	Script que realiza la baja de parlamentarios.
08_M_Parlamentari.sql	Script que realiza la modificación de parlamentarios.
09_A_Despeses.sql	Script que realiza el alta de gastos.
10_B_Despeses.sql	Script que realiza la baja de gastos.
11_M_Despeses.sql	Script que realiza la modificación de gastos.
12_Trigger_NumParlamentaris.sql	<i>Trigger</i> que recalcula el número de parlamentarios de un parlamento cuando se produce una alta o baja de un parlamentario.
13_Proc_ME.sql	Script que recalcula las tablas del módulo estadístico cada vez que se produce una alta, baja o modificación de un gasto.
14_Cargar_Las3TablasMaestras.sql	Se pueblan las 3 tablas maestras mediante los scripts de alta
15_Proc_Consulta_A.sql	Implementa el procedimiento de consulta A
16_Proc_Consulta_B.sql	Implementa el procedimiento de consulta B
17_Proc_Consulta_C.sql	Implementa el procedimiento de consulta C
18_Proc_Consulta_D.sql	Implementa el procedimiento de consulta D
19_Proc_Consulta_E.sql	Implementa el procedimiento de consulta E
20_Proc_Consulta_F.sql	Implementa el procedimiento de consulta F
21_Ultima_Linea_Log.sql	Muestra datos del último registro insertado en la tabla de Log
22_Pruebas_ABM.sql	Pruebas de altas, bajas, modificaciones de las tres tablas maestras
23_Pruebas_Proc_Consulta.sql	Pruebas de los procedimientos de consulta
24_Pruebas_Contraste_Proc_Consulta.sql	Mismas consultas que en el caso anterior pero mediante sentencias SQL, para contrastar resultados
25_Pruebas_ME.sql	Pruebas del módulo estadístico

9.2. Carga del producto y los datos

Los scripts están pensados para ejecutarse en el mismo orden mencionado en la tabla anterior. No se ha contemplado la creación de un *TABLESPACE* particular para el producto; por tanto sirve cualquiera con los permisos adecuados.

Merece la pena comentar que, si bien la mayoría de los scripts se ejecuta en un tiempo muy breve, el script que carga las tablas maestras puede llegar tardar bastante tiempo en ejecutarse (cerca de 1 minuto en una de las pruebas realizadas en el ordenador donde se ha realizado el proyecto, ver imagen). Ello es debido a la gran cantidad de registros de gastos, y a que después de la inserción de cada gasto, se recalculan todas las estadísticas.

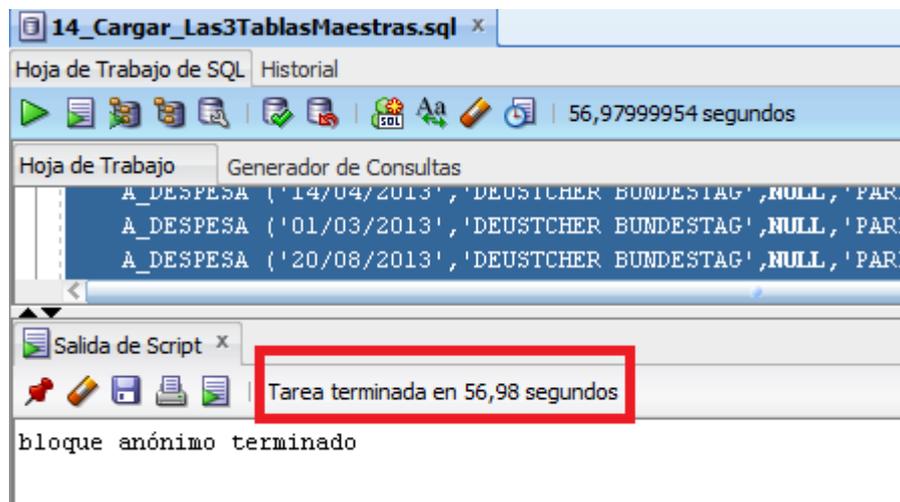


Figura 16. Carga de tablas maestras

10. Prueba del producto

Se ha creado una batería de pruebas, basadas todas ellas en el conjunto de datos previamente cargado. A continuación se explica su diseño y la manera de llevarlas a cabo

10.1. Pruebas de alta, baja, modificación de tablas maestras

Las pruebas de los scripts de alta/baja/modificación (a las que llamaremos *pruebas de ABM*) de las tablas maestras, se realizan ejecutando los scripts que se muestran en la imagen:

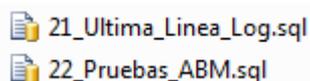


Figura 17. Pruebas ABM

10.1.1. Diseño de las pruebas de ABM

En el script 22_Pruebas_ABM.sql, se realizan llamadas a cada uno de los procedimientos de alta/baja/modificación de parlamentos, parlamentarios, e histórico de gastos. Un fragmento de dicho script se muestra en la siguiente imagen

```
-- ALTA ERRONEA PARLAMENTO - PARLAMENTO EXISTENTE
A_PARLAMENT ('PARLAMENT DE CATALUNYA','FRANCIA','CATALUNYA','Parc de la Ciut
ULTIMA_LINEA_LOG;

-- ALTA ERRONEA PARLAMENTO - PARLAMENTO NULO
A_PARLAMENT ('','FRANCIA','CATALUNYA','Parc de la Ciutadella, s/n, 08003 Bar
ULTIMA_LINEA_LOG;

-- ALTA ERRONEA PARLAMENTO - ESTADO INEXISTENTE
A_PARLAMENT ('PARLAMENTO ITALIANO','ITALIA','','---','---','---',RSP);
ULTIMA_LINEA_LOG;
```

Figura 18. Ejemplo pruebas ABM

Como puede verse, cada de las llamadas a los procedimientos ABM va seguida de una llamada al procedimiento llamado *ULTIMA_LINEA_LOG* (cuyo código se encuentra en el script *21_Ultima_Linea_Log.sql*). Este script, como se comentaba anteriormente (ver 6.2) tiene como misión mostrar por *pantalla el resultado del procedimiento ABM que se acaba de ejecutar*. De este modo, podemos ir comprobando el resultado de los procedimientos AMB a medida que se ejecutan.

10.1.2. Ejemplo de ejecución de las pruebas ABM

Se muestra a continuación, un fragmento de la salida obtenida después de ejecutar el script *22_Pruebas_ABM.sql*

```
*****
```

```

EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ-ESTAT:ESPAÑA-PAISOREGIO:ANDALUCccIA-ADRECA:c\
San Juan de Ribera, s/n 41009 Sevilla.-TELEFON:933046500-
WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: OK
*****
EJECUTADO: M_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ-PAISOREGIO:ANDALUCIA-ADRECA:c\ San Juan de
Ribera, s/n 41009 Sevilla.-TELEFON:933046500-WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: OK
*****
EJECUTADO: B_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ
RESULTADO: OK
*****
EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENT DE CATALUNYA-ESTAT:FRANCIA-PAISOREGIO:CATALUNYA-
ADRECA:Parc de la Ciutadella, s/n, 08003 Barcelona-TELEFON:933046500-
WEB:HTTP://www.parlament.cat/-
RESULTADO: ERR_ALTA_PARLAMENT_01: El parlamento PARLAMENT DE CATALUNYA ya existe
*****
EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:-ESTAT:FRANCIA-PAISOREGIO:CATALUNYA-ADRECA:Parc de la
Ciutadella, s/n, 08003 Barcelona-TELEFON:933046500-WEB:HTTP://www.parlament.cat/-
RESULTADO: ERR_ALTA_PARLAMENT_02: El parlamento que se quiere dar de alta es nulo
*****
EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ITALIANO-ESTAT:ITALIA-PAISOREGIO:-ADRECA:----
TELEFON:----WEB:----
RESULTADO: ERR_ALTA_PARLAMENT_03: El estado debe existir en la tala de estados
*****
EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ-ESTAT:ESPAÑA-PAISOREGIO:ANDALUCIA-ADRECA:-
TELEFON:933046500-WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: ERR_ALTA_PARLAMENT_04: La direccion del parlamento no puede ser nula
*****
EJECUTADO: A_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ-ESTAT:ESPAÑA-PAISOREGIO:ANDALUCIA-ADRECA:---
TELEFON:-WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: ERR_ALTA_PARLAMENT_05: El telefono del parlamento no puede ser nulo
*****
EJECUTADO: B_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ
RESULTADO: ERR_BAJA_PARLAMENT_01: El parlamento PARLAMENTO ANDALUZ no existe
*****
EJECUTADO: B_PARLAMENT
ENTRADA : NOMPARELAMENT:CONGRESO DE LOS DIPUTADOS
RESULTADO: ERR_BAJA_PARLAMENT_02: El parlamento CONGRESO DE LOS DIPUTADOS tiene
parlamentarios
*****
EJECUTADO: M_PARLAMENT
ENTRADA : NOMPARELAMENT:PARLAMENTO ANDALUZ-PAISOREGIO:ANDALUCIA-ADRECA:c\ San Juan de
Ribera, s/n 41009 Sevilla.-TELEFON:933046500-WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: ERR_MOD_PARLAMENT_01: El parlamento PARLAMENTO ANDALUZ no existe
*****
EJECUTADO: M_PARLAMENT
ENTRADA : NOMPARELAMENT:CONGRESO DE LOS DIPUTADOS-PAISOREGIO:ESPAÑA-ADRECA:-
TELEFON:9145587458-WEB:http://www.congreso.es/-
RESULTADO: ERR_MOD_PARLAMENT_02: La direccion del parlamento no puede ser nula
*****
EJECUTADO: M_PARLAMENT
ENTRADA : NOMPARELAMENT:CONGRESO DE LOS DIPUTADOS-PAISOREGIO:ESPAÑA-
ADRECA:C/Floridablanca s/n - 28071 - MADRID-TELEFON:-
WEB:HTTP://www.parlamentodeandalucia.es/-
RESULTADO: ERR_MOD_PARLAMENT_03: El telefono del parlamento no puede ser nulo
*****
EJECUTADO: A_PARLAMENTARI
ENTRADA : NOMPARELAMENT:CONGRESO DE LOS DIPUTADOS-CODIPARLAMENTARI:14-NIF:19084893E-
NOM:Pedro-COGNOMS:Perez-PARTITPOL:PARTIDO POPULAR-
RESULTADO: OK
*****
EJECUTADO: M_PARLAMENTARI
ENTRADA : NOMPARELAMENT:CONGRESO DE LOS DIPUTADOS-CODIPARLAMENTARI:14-NIF:19084893E-
NOM:JUAN LOPEZ-COGNOMS:Perez-PARTITPOL:PARTIDO POPULAR-
RESULTADO: OK

```

10.2. Pruebas sobre los procedimientos de consulta

Para probar los procedimientos de consulta, se han creado los scripts que se muestran en la imagen

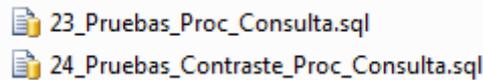


Figura 19. Pruebas de los procedimientos de consulta

A continuación se explica el propósito de cada uno de estos scripts

10.2.1. Diseño de las pruebas de los procedimientos de consulta

El procedimiento `23_Pruebas_Proc_Consulta.sql`, invoca a los diferentes procedimientos de consulta. Su código se muestra a continuación.

```
SET SERVEROUTPUT ON SIZE UNLIMITED
SET DEFINE OFF

DECLARE
    RSP    CLOB;

BEGIN

    /*****
    /***** PROCEDIMIENTO A *****/
    /*****

    PROC_DESPESESANYPARLAMENTARI('PARLAMENT DE CATALUNYA', 2009,1, RSP ) ;

    /*****
    /***** PROCEDIMIENTO B *****/
    /*****

    proc_ParlamentarisDespDir ('DEUSTCHER BUNDESTAG',2011, RSP );

    /*****
    /***** PROCEDIMIENTO C *****/
    /*****

    PROC_DESPTOTALTOTSPARLAM (2012 , RSP) ;

    /*****
    /***** PROCEDIMIENTO D *****/
    /*****

    proc_DespPerCategorias ('ASSEMBLEE NATIONALE',2013,RSP);

    /*****
    /***** PROCEDIMIENTO E *****/
    /*****

    PROC_PARLAMDESPEJASUPAMITJANA ('DEUSTCHER BUNDESTAG',2012, RSP);

    /*****
    /***** PROCEDIMIENTO F *****/
    /*****

    proc_EstatContableParlaments (2013, RSP);
END;
```

A continuación se muestran algunos ejemplos de su ejecución:

Después de la llamada al procedimiento B (Listado de los parlamentarios del *DEUSTCHER BUNDESTAG* en el año 2011 con el total de sus gastos directos), se obtiene el siguiente resultado:

Salida de Script x | Tarea terminada en 0,09 segundos

```

***** SUMA DE LOS GASTOS DIRECTOS DE LOS PARLAMENTARIOS
***** PERTENECIENTES AL PARLAMENTO: DEUSTCHER BUNDESTAG
***** DURANTE EL AÑO: 2011

```

PARLAMENTA	NOMBRE	APELLIDOS	PARTIDO	GASTO
1	Merkel	Angela	CDU/CSU	822072
2	Christian	Lange	SPD	804048
7	Maria	Böhmer	CDU/CSU	774139
14	Stefan	Liebich	DIE LINKE	769200
6	Rainer	Arnold	SPD	759000
5	Wolfgang	Gunkel	SPD	749484
4	Klaus-Dieter	Gröhler	CDU/CSU	732719
13	Gregor	Gysi	DIE LINKE	723600
15	Diana	Golze	DIE LINKE	712800
9	Mark	Hauptmann	CDU/CSU	712800
3	Günter	Lach	CDU/CSU	705253
10	Hubertus	Heil	SPD	700513
12	Klaus	Barthel	SPD	691289
11	Sebastian	Edathy	SPD	652244
8	Sybille	Benning	CDU/CSU	630000

Figura 20. Ejecución proc. consulta B

Después de la llamada al procedimiento F (Listado del estado de los gastos de todos los parlamentos, desglosados según su estado contable), se obtiene el resultado de la imagen:

```

***** ESTADO CONTABLE DE TODOS LOS PARLAMENTOS:
***** EN EL AÑO: 2013

```

PARLAMENTO	NUM. PARLAMENTARIOS	PEND. APROBAR	APROBADAS	PEND. ABONAR	ABONADAS
DEUSTCHER BUNDESTAG	15	1058	2062	1048	11478919
CONGRESO DE LOS DIPUTADOS	13	0	0	0	9325417
PARLAMENT DE CATALUNYA	12	1537	1654	1838	8379922
ASSEMBLEE NATIONALE	11	0	0	0	8199780
PARLAMENTO VASCO	9	184	94	3960	6675642

Figura 21. Ejecución proc. F

10.2.2. Contraste de los resultados obtenidos

Para verificar que los resultados obtenidos son válidos, se ha creado el script *24_Pruebas_Contraste_Proc_Consulta.sql*. Este script contiene sentencias SQL que general listados análogos a los obtenidos por los procedimientos de consulta a-f. En dichas SELECT se han empleado los mismos datos que se han pasado como parámetros en los procedimientos de consulta. Por tanto, los resultados deberían ser coincidentes.

Por ejemplo, la *SELECT* para el procedimiento de consulta B, proporciona el siguiente resultado (idéntico a los resultados obtenidos en el apartado 10.2.1)

	CODIPARLAMENTARI	NOM	COGNOMS	PARTITPOL	AÑO	TOTAL
1		1Merkel	Angela	CDU/CSU	2011	822072
2		2 Christian	Lange	SPD	2011	804048
3		7 Maria	Böhmer	CDU/CSU	2011	774139
4		14 Stefan	Liebich	DIE LINKE	2011	769200
5		6 Rainer	Arnold	SPD	2011	759000
6		5 Wolfgang	Gunkel	SPD	2011	749484
7		4 Klaus-Dieter	Gröhler	CDU/CSU	2011	732719
8		13 Gregor	Gysi	DIE LINKE	2011	723600
9		15 Diana	Golze	DIE LINKE	2011	712800
10		9 Mark	Hauptmann	CDU/CSU	2011	712800
11		3 Günter	Lach	CDU/CSU	2011	705253
12		10 Hubertus	Heil	SPD	2011	700513
13		12 Klaus	Barthel	SPD	2011	691289
14		11 Sebastian	Edathy	SPD	2011	652244
15		8 Sybille	Benning	CDU/CSU	2011	630000

Figura 22. Verificación resultados del proc. B

En el caso del procedimiento de consulta F, el resultado obtenido mediante la *SELECT* de comprobación es también idéntico al del ejemplo 10.2.1:

	NOMPARLAMENT	NOMBREPARLAMENTARIS	SUM(PENDENT_APROVACIO)	SUM(APROVADA)	SUM(PENDENT_ABONAMENT)	SUM(ABONADA)
1	DEUTSCHER BUNDESTAG	15	1058	2062	1048	11478919
2	CONGRESO DE LOS DIPUTADOS	13	0	0	0	9325417
3	PARLAMENT DE CATALUNYA	12	1537	1654	1838	8379922
4	ASSEMBLEE NATIONALE	11	0	0	0	8199780
5	PARLAMENTO VASCO	9	184	94	3960	6675642

Figura 23. Verificación resultados proc. F

Se puede comprobar fácilmente que se obtienen resultados análogos en todos los scripts.

10.3. Pruebas sobre el módulo estadístico

Como parte de la entrega, se proporciona el siguiente script SQL que realiza las pruebas del módulo estadístico

25_Pruebas_ME.sql

Figura 24. Pruebas módulo estadístico

Para asegurarnos de que los resultados que ofrece el módulo estadístico son fiables, ha parecido conveniente compararlos con los mismos resultados, pero obtenidos por otros medios. Ello se explica en los siguientes apartados.

10.3.1. Diseño de las pruebas del módulo estadístico

En el script de pruebas se han realizado dos tipos de consultas:

- SELECTS directas a las tablas del módulo estadístico (es decir, los resultados que proporciona el módulo estadístico propiamente dicho)
- Paralelamente, se muestran los mismos resultados obtenidos mediante SELECTS directas a la tabla de DESPESES.

De este modo se puede comprobar la confianza que nos merece el módulo estadístico.

10.3.2. Ejemplos del resultado de las pruebas

Se ofrecen a continuación algunos ejemplos de los resultados proporcionados por el módulo estadístico (las pruebas completas pueden realizarse ejecutando el archivo "23_Pruebas_ME.sql")

- ESTADÍSTICA-1

En la siguiente imagen se muestra los resultados del punto 1 del módulo estadístico

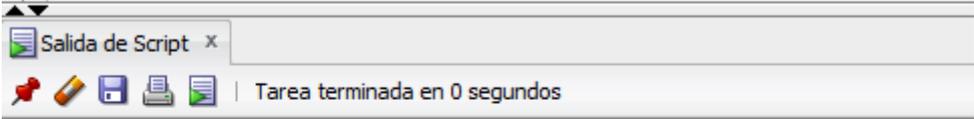
```

/*****
/***** ESTADISTICA 1 *****/
*****/

-- PARA CADA PARLAMENTO: LA SUMA DE TODOS LOS GASTOS
-- DURANTE LOS 4 ULTIMOS AÑOS
-- INCLUYENDO TANTO LOS GASTOS DE LOS PARLAMENTOS
-- COMO LOS DE LOS PARLAMENTARIOS

-- RESULTADO OBTENIDO DE LA TABLA DEL MODULO ESTADISTICO
SELECT * FROM ESTAD_1 order by NOMPARLAMENT;

```

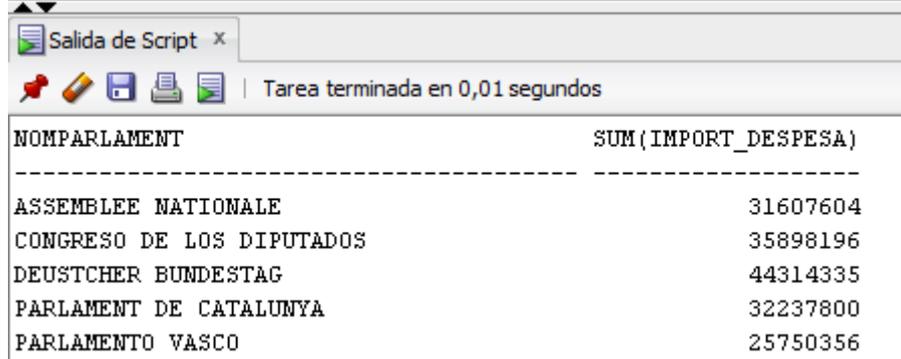


NOMPARLAMENT	IMPORT_DESPESA	PERIODE
ASSEMBLEE NATIONALE	31607604	2010-2013
CONGRESO DE LOS DIPUTADOS	35898196	2010-2013
DEUSTCHER BUNDESTAG	44314335	2010-2013
PARLAMENT DE CATALUNYA	32237800	2010-2013
PARLAMENTO VASCO	25750356	2010-2013

Figura 25. Pruebas ME-1

Los mismos resultados obtenidos mediante una SELECT directa a la tabla ESTADISTIQUE, ofrecen los mismos resultados

```
-- RESULTADO OBTENIDO REALIZANDO UNA SELECT
-- CONTRA LA TABLA DESPESES
SELECT NOMPARLAMENT, SUM (IMPORT_DESPESA) from DESPESES
WHERE (
EXTRACT (YEAR FROM DATA_DESPESA)= 2010 OR
EXTRACT (YEAR FROM DATA_DESPESA)= 2011 OR
EXTRACT (YEAR FROM DATA_DESPESA)= 2012 OR
EXTRACT (YEAR FROM DATA_DESPESA)= 2013)
GROUP BY NOMPARLAMENT ORDER BY NOMPARLAMENT;
```



NOMPARLAMENT	SUM(IMPORT_DESPESA)
ASSEMBLEE NATIONALE	31607604
CONGRESO DE LOS DIPUTADOS	35898196
DEUSTCHER BUNDESTAG	44314335
PARLAMENT DE CATALUNYA	32237800
PARLAMENTO VASCO	25750356

Figura 26. Pruebas contraste ME-1

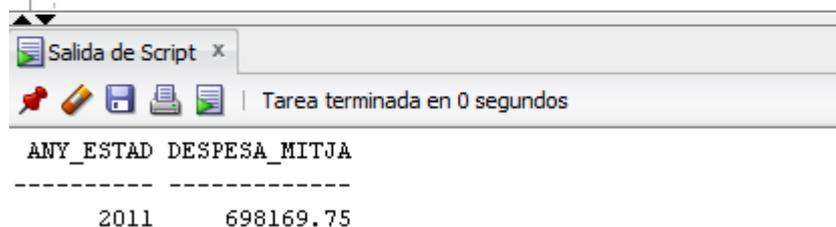
- ESTADISTICA-5

Resultados del punto 5 del módulo estadístico, para el año 2011. Vemos que la media de los gastos de todos los parlamentarios durante ese año fue de 698169,75

```
/*-----*/
/*----- ESTADISTICA 5 -----*/
/*-----*/

--MEDIA DEL GASTO DE TODOS LOS PARLAMENTARIOS EN UN AÑO

SELECT * FROM ESTAD_5 WHERE ANY_ESTAD=2011;
-- RESULTADO: 698169.75
```



ANY_ESTAD	DESPESA_MITJA
2011	698169.75

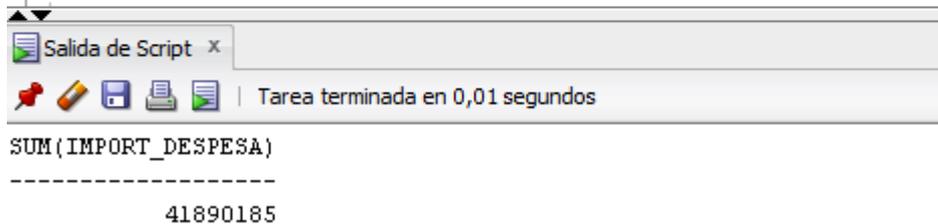
Figura 27. Pruebas ME-5

Para ver si el resultado es correcto, calculamos la media del gasto de los parlamentos por otros medios. En primer lugar calculamos la suma de los gastos de los parlamentarios para ese año, tal como se muestra en la siguiente imagen

```

-- SUMA DE LOS GASTOS TOTALES DE PARLAMENTARIOS EN EL AÑO 2011
SELECT SUM (IMPORT_DESPESA) FROM DESPESES
WHERE
    EXTRACT (YEAR FROM DATA_DESPESA)= 2011 AND
    TIPUSDESPESA='PARLAMENTARI';
-- Resultado: 41890185

```



Salida de Script x | Tarea terminada en 0,01 segundos

```

SUM(IMPORT_DESPESA)
-----
41890185

```

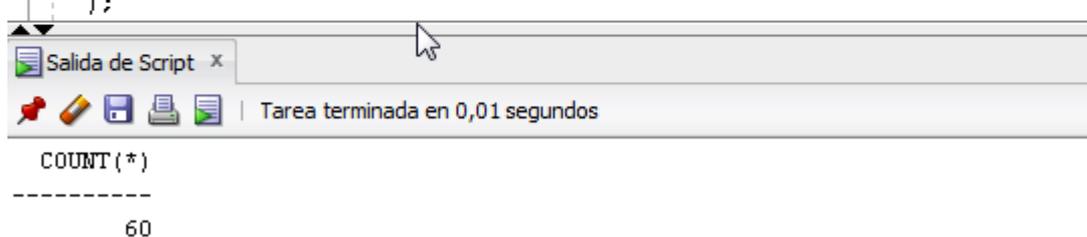
Figura 28. Gastos de los parlamentarios en 2011

Para ese mismo año, el número de parlamentarios con gasto fue de 60:

```

-- NUMERO DE PARLAMENTARIOS QUE HAN REGISTRADO ALGUN GASTO EN EL AÑO 2011
SELECT COUNT (*) FROM(
SELECT DISTINCT NOMPARELAMENT, CODIPARELAMENTARI
FROM DESPESES
WHERE
    EXTRACT (YEAR FROM DATA_DESPESA)= 2011 AND
    TIPUSDESPESA='PARLAMENTARI'
);

```



Salida de Script x | Tarea terminada en 0,01 segundos

```

COUNT (*)
-----
60

```

Figura 29. Parlamentarios con gasto en 2011

Finalmente, tenemos que la media de los gastos de los parlamentarios en el año 2011, fue de: $41890185 / 60 = 698169,75$, es decir, el mismo resultado que ofrece el módulo estadístico.

- ESTADISTICA-6

Según el módulo estadístico, los diputados con el gasto máximo y el mínimo durante el año 2012, fueron el número 1 del DEUSTCHER BUNDESTAG y el número 7 de la ASSEMBLEE NATIONALE.

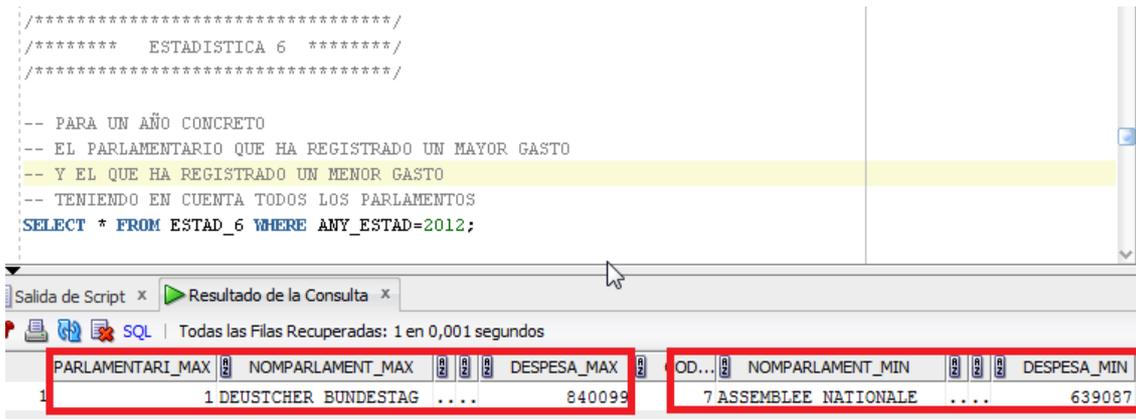


Figura 30. Pruebas ME-6

En la siguiente imagen vemos quien es el diputado que ha tenido mayor gasto en 2012, mirando directamente en DESPESES

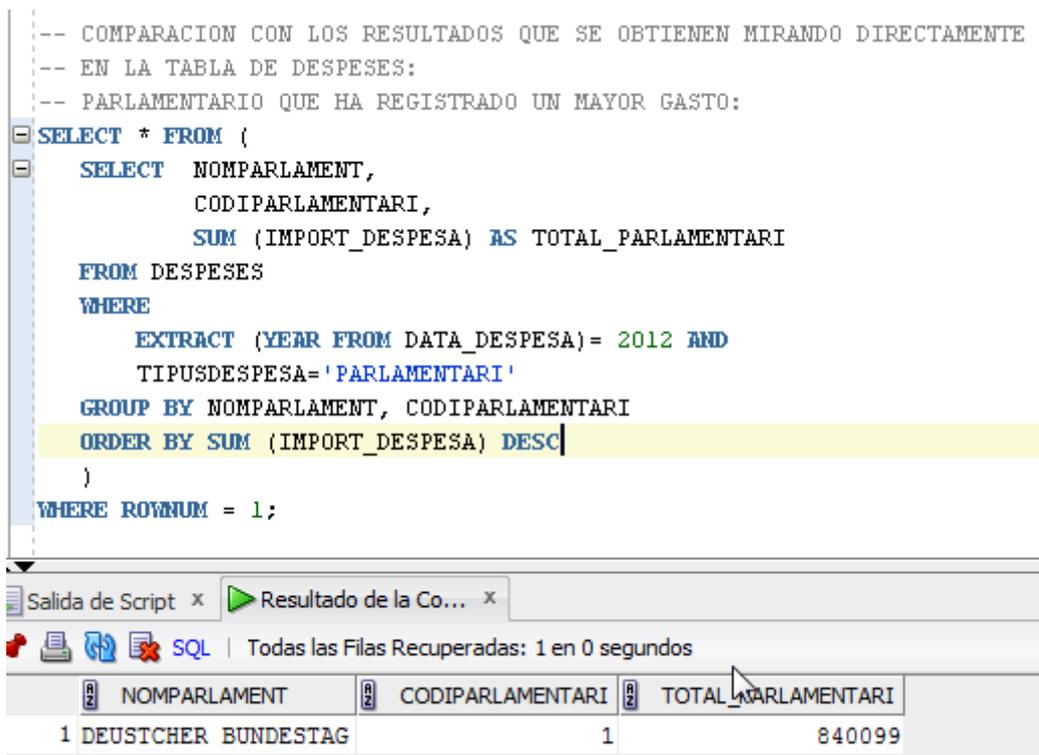
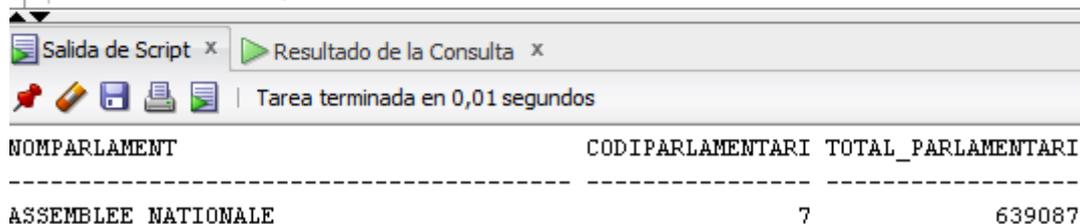


Figura 31. Parlamentario con mayor gasto en 2012

Diputado que ha tenido menos gastos en 2012:

```
-- PARLAMENTARIO QUE HA REGISTRADO UN MENOR GASTO:
SELECT * FROM (
  SELECT NOMPARLAMENT,
         CODIPARLAMENTARI,
         SUM (IMPORT_DESPESA) AS TOTAL_PARLAMENTARI
  FROM DESPESES
  WHERE
    EXTRACT (YEAR FROM DATA_DESPESA)= 2012 AND
    TIPUSDESPESA=' PARLAMENTARI '
  GROUP BY NOMPARLAMENT, CODIPARLAMENTARI
  ORDER BY SUM (IMPORT_DESPESA) ASC
)
WHERE ROWNUM = 1;
```



NOMPARLAMENT	CODIPARLAMENTARI	TOTAL_PARLAMENTARI
ASSEMBLEE NATIONALE	7	639087

Figura 32. Parlamentario con menor gasto en 2012

Vemos, por tanto, que los resultados son coincidentes.

Mediante el script 23_Pruebas_ME.sql puede verse que se han realizado pruebas parecidas para absolutamente todos los puntos del módulo estadístico y que los resultados son siempre coincidentes.

11. Valoración económica

Se realiza una valoración económica en función de las horas de dedicación.

Se han imaginado 3 roles en el desarrollo del proyecto: el de jefe de proyecto, el de analista y el de programador.

11.1. Cálculo del coste económico

ROL	Coste (euros/hora)
Jefe de proyecto	65
Analista	40
Programador	35

Tarea	Responsable	Horas de dedicación	Importe
Establecer los objetivos y alcance del proyecto. Planificación. Análisis de recursos	Jefe de proyecto	8	520
Análisis de requisitos. Diseño de la base de datos	Analista	20	800
Instalación del software. Creación de la BBDD. Desarrollo de todo el pto. software (creación de tablas, procedimientos, realización de las pruebas)	Programador	110	3850
Elaboración de la documentación (Memoria, presentación)	Analista	45	2925

Por tanto el coste total del proyecto sería de 8095 Euros.

11.2. Comentarios sobre la valoración económica del proyecto

Parece importante comentar que el número de horas de dedicación previstas en la planificación inicial, no han sido suficientes para el desarrollo del proyecto. La desviación mayor con respecto a las previsiones iniciales, corresponde a las horas que se han dedicado al producto software. El diseño de la base de datos también ha llevado un tiempo superior al esperado, sobre todo debido a un cambio que se decidió a posteriori en el diseño (ver 1.3.2.)

12. Conclusiones

Después de reconsiderar en perspectiva todo el trabajo realizado, es oportuno hacer algunos comentarios:

En lo que respecta al desarrollo del producto: se han alcanzado los objetivos propuestos, y se han satisfecho tanto los requerimientos funcionales como los de documentación. Merecen especial mención las pruebas realizadas sobre los procedimientos de consulta/módulo estadístico, que demuestran su correcto funcionamiento al obtener los mismos resultados por dos vías distintas.

En cuanto a la planificación, se ha constatado lo siguiente:

- El avance del proyecto sufrió retrasos con respecto al calendario que se propuso inicialmente. Ello ha sido debido, sobre todo, a que la disponibilidad de tiempo en días laborales fue menor que la esperada.
- No es realista una planificación de tipo “ciclo de vida en cascada” estricta, ni siquiera en un proyecto con los requisitos funcionales tan bien definidos como los que teníamos.

Un aspecto que se valora de forma especialmente positiva, es que ha permitido profundizar en conocimientos previos. Merece la pena destacar los siguientes puntos:

- El proyecto ha requerido de un uso intensivo de consultas SQL. En particular, las *sentencias SQL anidadas* han resultado ser una poderosa herramienta para obtener resultados complejos. SQL, a diferencia de otros lenguajes de tipo *imperativo* -como Java o C++-, es un ejemplo de *lenguaje declarativo*, con el que, a menudo, expresar lo que se desea está lejos de ser trivial. La necesidad de “pensar en SQL” ha resultado ser no solo un reto sino también un ejercicio muy interesante y práctico.
- Se desconocía la potencia del lenguaje de programación propio de Oracle (PL/SQL). El proyecto ha permitido adquirir un cierto conocimiento de dicho lenguaje. Ha resultado muy interesante la facilidad con que se combina el acceso a los datos mediante SQL, con su posterior tratamiento en un lenguaje muy parecido a Pascal.

Por último, el proyecto ha aportado una valiosa experiencia en algunos aspectos prácticos:

- No se puede menospreciar la complejidad de un proyecto. A menudo surgen complicaciones que obligan a dar marcha atrás y repensar etapas que se consideraban solucionadas. Es conveniente, por tanto, disponer de un “colchón de tiempo” para afrontar este tipo de situaciones
- Siempre hay más de una solución para un mismo problema. Los criterios para decidir el tratamiento que daremos a este dilema no son únicos. En el ámbito profesional a menudo nos decantaremos por el camino más económico en tiempo. En un trabajo de tipo académico, si el tiempo lo permite, nos podemos permitir el uso de soluciones más elegantes. Hay que tener en cuenta también, que las soluciones rápidas no siempre son escalables. Por tanto es necesario tener en cuenta, antes de escoger una solución a un problema, si el desarrollo es susceptible o no de necesitar ampliaciones en un futuro.

Glosario

Secuencia (*Sequence*). Funcionalidad de un sistema de gestión de base de datos, que cada vez que se invoca, proporciona un nuevo número de forma automática, de acuerdo a ciertas reglas.

SGBD. Iniciales de “Sistema de gestión de base de datos”. Se trata de un conjunto de programas –normalmente proporcionado por el mismo fabricante y comercializados como un producto único- capaces de gestionar grandes conjuntos de datos distribuidos en tablas y campos. Los SGBD proporcionan herramientas para definir la estructura de los datos, permitir el acceso y la manipulación de los datos, y garantizar la integridad de los mismos, entre otras funciones

BBDD. Iniciales de “Base de datos”. Conjunto de tablas. A menudo se utiliza de manera informal como sinónimo de SGBD

Diagrama de Gantt. Herramienta gráfica que muestra el tiempo previsto para diferentes tareas a lo largo de un tiempo determinado. Fue desarrollada por Henry Laurence Gantt a principios del siglo XX

Disparador (*Trigger*): Funcionalidad propia de los sistemas de gestión de base de datos, que permite realizar de forma automática un proceso cuando se producen modificaciones específicas en algunos campos o registros.

Diagrama UML. Herramienta grafica que pretende representar y describir de manera formal una cierta realidad formada por distintas entidades con características determinadas y que mantienen entre sí relaciones específicas

Bibliografía

- Steven Feuerstein, "Oracle PL/SQL Programming" (5ª edición), Ed. O'Reilly, Sebastopol, 2009
- Chuck Murray, "Oracle Database SQL Developer User's Guide, Release 1.5", (editado por Oracle en formato electrónico), 2009
- "Oracle® Database SQL Reference" (publicación de Oracle en línea) <http://docs.oracle.com/>
- Sistac Planas, J. y otros. "Bases de Dades II" (material docente de la Universitat Oberta de Catalunya). Barcelona, 2004
- Sistac Planas, J. y otros. "Sistemes de gestió de bases de dades" (material docente de la Universitat Oberta de Catalunya). Barcelona, 2004
- Campderrich Falgueras, B "Enginyerya del programari" (material docente de la Universitat Oberta de Catalunya). Barcelona, 2004.

Anexos

Jxicart_presentacio.pdf

Jxicart_producte.zip