

# *Disseny i implementació d'una base de dades relacional per al manteniment d'aparells d'una empresa*

**Carlos Cabrera Larroya**  
Enginyeria en Informàtica

**Juan Martínez Bolaños**  
15/01/2012



- ❑ Introducción
- ❑ Software necesario
- ❑ Análisis Base de Datos
- ❑ Diseño Base de Datos
- ❑ Implementación Base de Datos
- ❑ Análisis Datawarehouse
- ❑ Diseño Datawarehouse
- ❑ Implementación Datawarehouse
- ❑ Valoración económica del proyecto
- ❑ Conclusiones



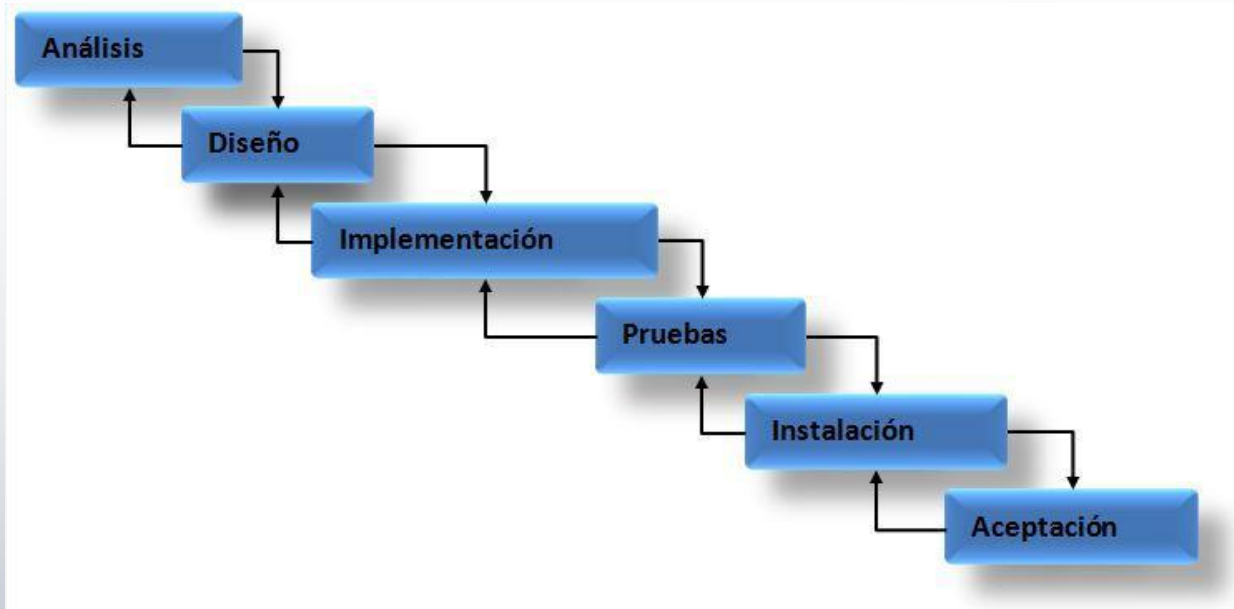


- ❑ El disponer de información organizada, accesible de forma rápida, con históricos, etc., es básico en cualquier organización para la toma de decisiones.
- ❑ Este proyecto se enmarca en el área de las Bases de Datos y pretende dar solución a una problemática planteada, poniendo en práctica los conocimientos adquiridos en las diferentes asignaturas de esta área.
- ❑ Así mismo el proyecto es el punto que culmina el 2º Ciclo de Ingeniería Informática y engloba conocimientos del resto de asignaturas cursadas.
- ❑ El objetivo principal de este proyecto es diseñar e implementar una base de datos relacional para el mantenimiento de los equipos de una empresa, instalados en sus 32 centros de trabajo.
- ❑ La gestión y acceso a la información se deberá hacer mediante procedimientos almacenados.
- ❑ Se deberá diseñar un datawarehouse para obtener estadísticas sobre las incidencias producidas y sobre las empresas que realizan el mantenimiento.





- La metodología de gestión de proyectos que utilizamos es “Ciclo de vida clásico o en cascada”. Esta metodología es sobradamente conocida y fácil de gestionar, y ya disponemos de experiencia en su uso.

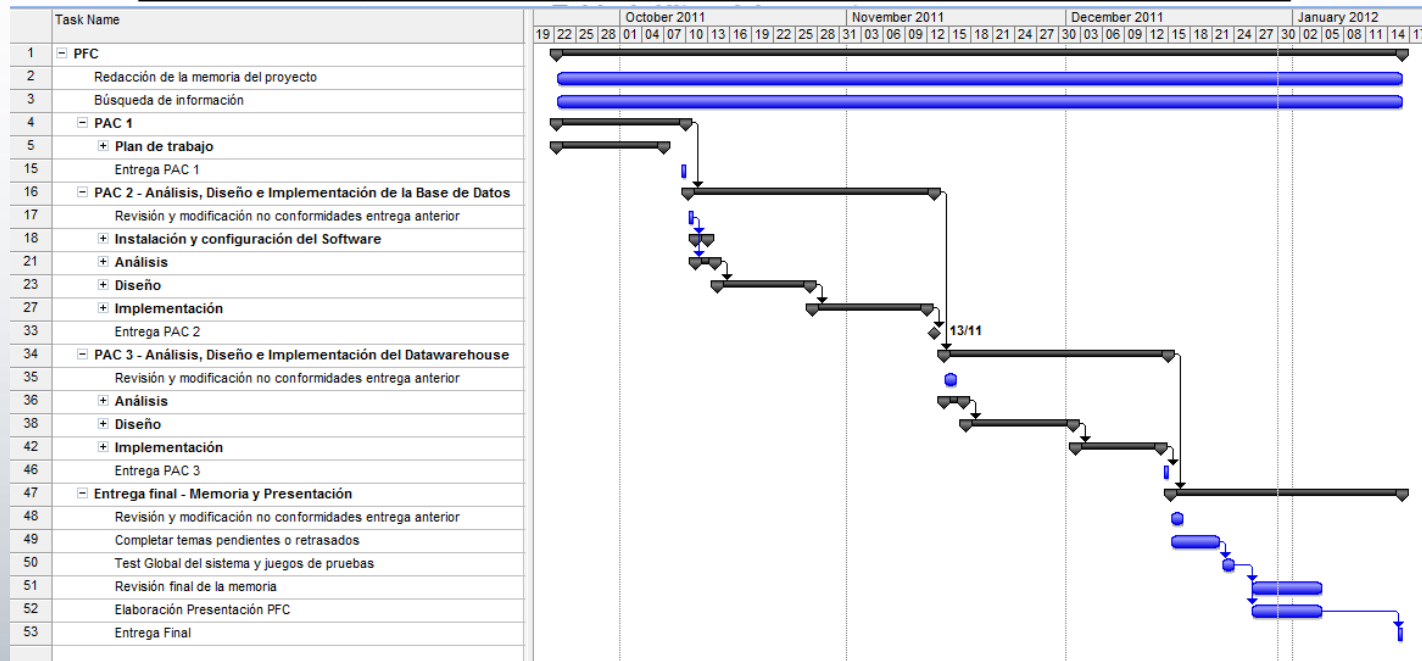




## Hitos del proyecto, Tareas y Diagrama de Gantt



Hitos del Proyecto		
Hito	Detalle	Fecha
Entrega PAC1	Plan de Trabajo	09/10/2011
Entrega PAC2	Análisis, Diseño e Implementación de la Base de Datos	13/11/2011
Entrega PAC3	Análisis, Diseño e Implementación del Datawarehouse	14/12/2011
Entrega final	Memoria + Presentación + Producto	15/01/2011



- Los entregables del proyecto son:
  - Plan de Trabajo
    - planificación de la ejecución del proyecto
  - Producto (trabajo práctico)
    - código fuente desarrollado, scripts para la creación, gestión y chequeo de BBDD y DW
  - Memoria
    - documento en el que se refleja todo el trabajo realizado durante el proyecto
  - Presentación
    - resumen visual del trabajo realizado
- El software utilizado para el proyecto ha sido:
  - MS SQL Server 2005 Express como SGBD
  - MagicDraw, para el diseño de los diferentes modelos





- El análisis de requerimientos se ha dividido en diferentes Sistemas funcionales

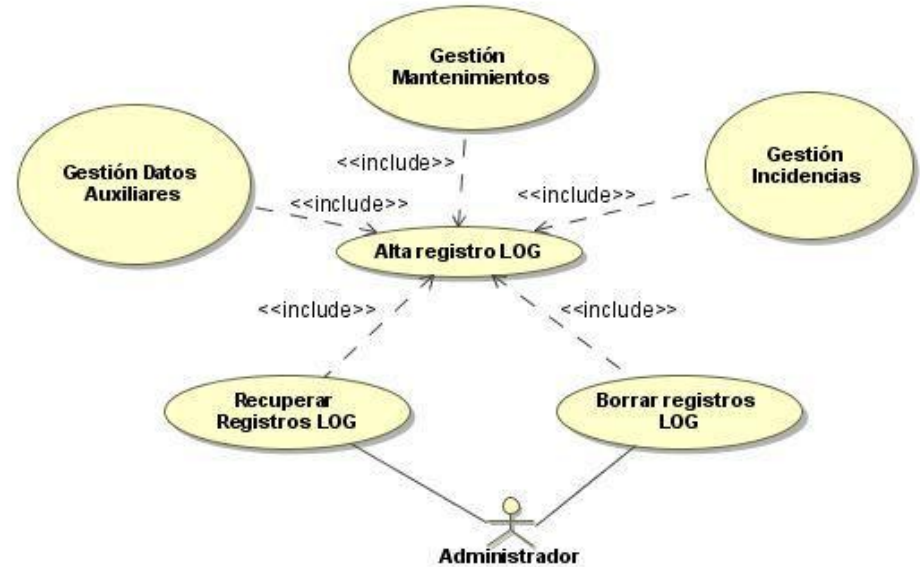


Requerimientos Sistema Gestión Datos Auxiliares - Centros	
Acción	Descripción
Alta	Crea un nuevo Centro. Se deberán indicar todos los datos obligatorios. El Código de Centro no debe existir.
Modificación	Permitirá modificar los datos de un Centro. Se deberá indicar el código de Centro a modificar. El Centro deberá existir.
Eliminar	Permitirá eliminar un Centro del Sistema. Se deberá indicar el código de Centro a eliminar. El Centro deberá existir. El Centro no podrá tener Empleados ni Equipos asociados
Consulta	Permite obtener la información existente de uno o varios Centros. Si no se indica código de Centro, retornará la información de todos los Centros existentes.
En todos los casos se insertará un registro en el LOG, con la acción realizada, los parámetros indicados y el resultado de la acción.	

Requerimientos Sistema Gestión Datos Auxiliares - Tipos de Equipo	
Acción	Descripción
Alta	Crea un nuevo Tipo de Equipo. Se deberán indicar todos los datos obligatorios. El Código de Tipo de Equipo no debe existir.
Modificación	Permitirá modificar los datos de un Tipo de Equipo. Se deberá indicar el código de Tipo de Equipo a modificar. El Tipo de Equipo deberá existir.
Eliminar	Permitirá eliminar un Tipo de Equipo del Sistema. Se deberá indicar el código de Tipo de Equipo a eliminar. El Tipo de Equipo deberá existir. No deberá haber Equipos de ese Tipo dados de alta
Consulta	Permite obtener la información existente de uno o varios Tipos de Equipo. Si no se indica código de Tipo de Equipo, retornará la información de todos los Tipos de Equipo existentes.
En todos los casos se insertará un registro en el LOG, con la acción realizada, los parámetros indicados y el resultado de la acción.	



- Se han identificado y detallado los diferentes casos de uso del sistema para representar las diferentes funcionalidades e interacciones.

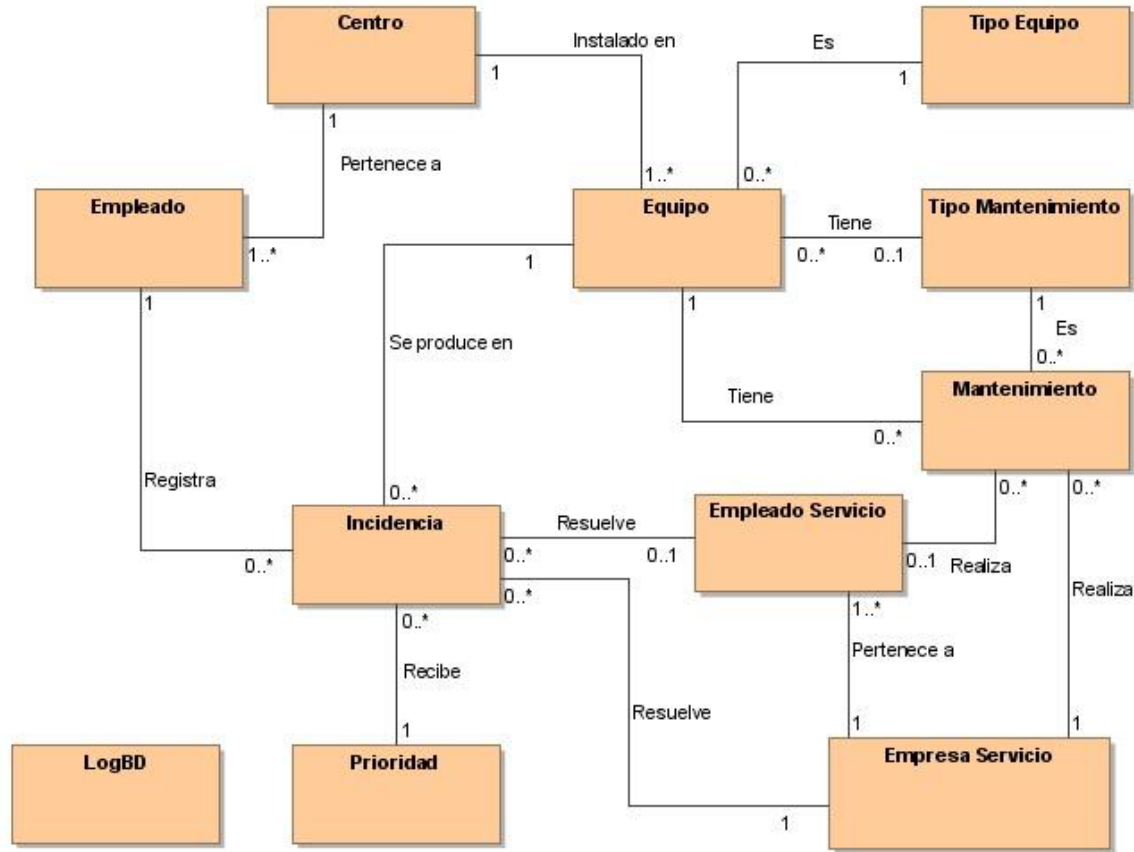


Cabecera	
ID Caso de Uso	CU02
Versión Caso de Uso	V.1.0
Estado	Aprobado
Autor	Carlos Cabrera
Cuerpo	
Título	Sistema LOG
Actores	Administrador
Flujo	El Administrador podrá Recuperar información registrada en el LOG así como eliminarla. Tanto estos dos procesos, como el resto de procesos del sistema, accederán al proceso de Alta registro LOG. Que es el encargado de crear registros en el LOG.



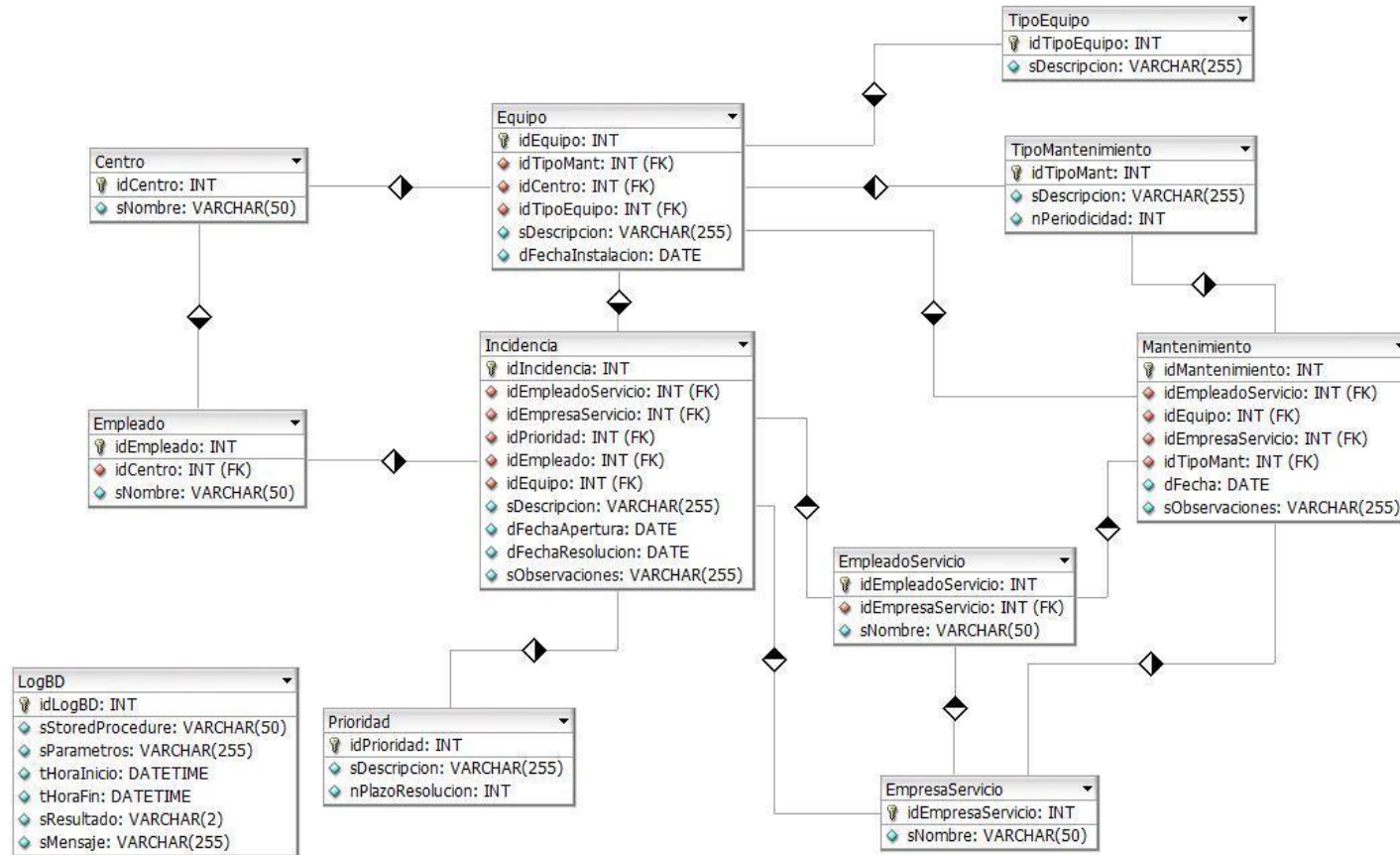


- Mediante un diagrama UML Entidad/Relación, hemos representado las entidades y relaciones identificadas. También se han detallado las entidades y atributos, así como las restricciones de integridad más relevantes.



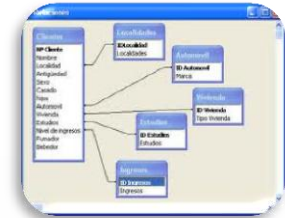


- En este punto pasamos a transformar el conceptual en un modelo lógico, detallando cada entidad, con sus atributos, claves primarias y claves foráneas





- A partir de las entidades y atributos identificados, detallamos el modelo físico de tablas, columnas, índices,...



TipoMantenimiento							
Columna	Tipo	Longitud	Permite Nulos	Indexada	Clave Primaria	Clave Foránea	
						Columna	Tabla
idTipoMant	int			Si	Si		
sDescripcion	varchar	255					
nPeriodicidad	int						

TipoEquipo							
Columna	Tipo	Longitud	Permite Nulos	Indexada	Clave Primaria	Clave Foránea	
						Columna	Tabla
idTipoEquipo	int			Si	Si		
sDescripcion	varchar	255					

Equipo							
Columna	Tipo	Longitud	Permite Nulos	Indexada	Clave Primaria	Clave Foránea	
						Columna	Tabla
idEquipo	int			Si	Si		
idTipoMant	int			Si		idTipoMant	TipoMantenimiento
idCentro	int			Si		idCentro	Centro
idTipoEquipo	int			Si		idTipoEquipo	TipoEquipo
sDescripcion	varchar	255					
dFechaInstalacion	datetime						



```
shalo@gnz-lap:~$ touch script3.sh
shalo@gnz-lap:~$ nano script3.sh
shalo@gnz-lap:~$ chmod +x script3.sh
shalo@gnz-lap:~$ sh script3.sh
A continuación se buscará una cadena en un archivo
no existente y se mostrará
lo que devolvio la variable $?
Comando: grep donde poema.txt
grep: poema.txt: No existe el fichero o el directorio
2
-----Fin del script-----
shalo@gnz-lap:~$
```

- Se han creado una serie de scripts para facilitar la creación de la BD, tablas, inserción de datos, creación de procedimientos almacenados...
  - 1\_Creacion\_BD.sql
  - 2\_Crear\_Tablas.sql
  - 3\_Insercion\_Datos\_Base.sql
  - 4\_Creacion\_SPs\_Borrado.sql
  - 5\_Creacion\_SPs\_Insercion.sql
  - 6\_Creacion\_SPs\_Seleccion.sql
  - 7\_Creacion\_SPs\_Actualizacion.sql

- ❑ Se ha generado una carpeta, Juegos de Pruebas, en la que se detallan todas las pruebas realizadas y sus resultados.
- ❑ Se ha realizado la validación de todos los procedimientos almacenados creados.
- ❑ Los resultados se han agrupado por tabla.



```
-- Ejecución de procedimiento de borrado
```

```
exec sp_d_Incidencia_Mantenimiento
```

```
-- RESULTADO
```

```
(4 row(s) affected)
```

```
(4 row(s) affected)
```

```
SELECT * FROM INCIDENCIA
```

idIncidencia	idEmpresaServicio	idEmpleadoServicio	idPrioridad	idEmpleado	idEquipo	sDescripcion
--------------	-------------------	--------------------	-------------	------------	----------	--------------

```
SELECT * FROM MANTENIMIENTO
```

idMantenimiento	idEquipo	idEmpresaServicio	idEmpleadoServicio	idTipoMant	dFecha	sobservacion
-----------------	----------	-------------------	--------------------	------------	--------	--------------

```
SELECT * FROM LOGBD
```

idLogBD	sStoredProcedure	sParametros	tHoraInicio	tHoraFin	sResultado	sMensaje
597	sp_d_Incidencia_Mantenimiento		13/11/2011 18:42:33	13/11/2011 18:42:33	OK	NULL



□ De los requerimientos iniciales proporcionados por el cliente y las conversaciones mantenidas con él a posteriori, obtenemos una lista de resultados que nos debe proporcionar el sistema:

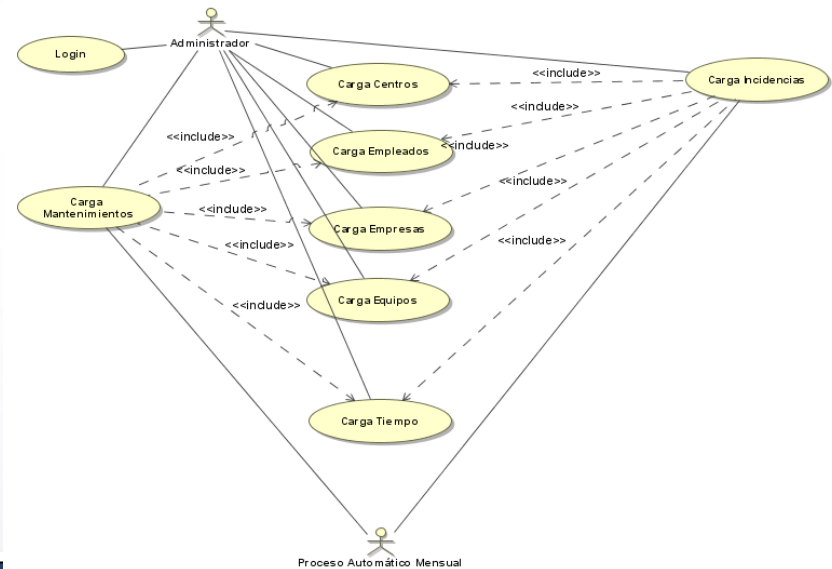


- Número de incidencias por Equipo y Centro
- Número de incidencias solucionadas a tiempo y fuera de plazo, por Empresa de Servicio
- Número de Incidencias detectadas por Empleado
- Número de Mantenimientos realizados por Empresa de Servicio, Equipo y Centro
- Número de Incidencias de los Equipos con más de X Tiempo de antigüedad.
- En todos los casos aplicará que aparezca el componente Tiempo.

# Análisis Datawarehouse



- Se crea una BD para el Datawarehouse y las tablas necesarias para almacenar los datos que nos permitan obtener la información requerida por el cliente.
- Se crea una serie de procedimientos almacenados que extraigan la información de la BD Relacional, la transformen y la inserten en las tablas del DW con el formato requerido.
- El proyecto no contempla el realizar el diseño de las consultas posteriores que permitan explotar la información almacenada en el DW.

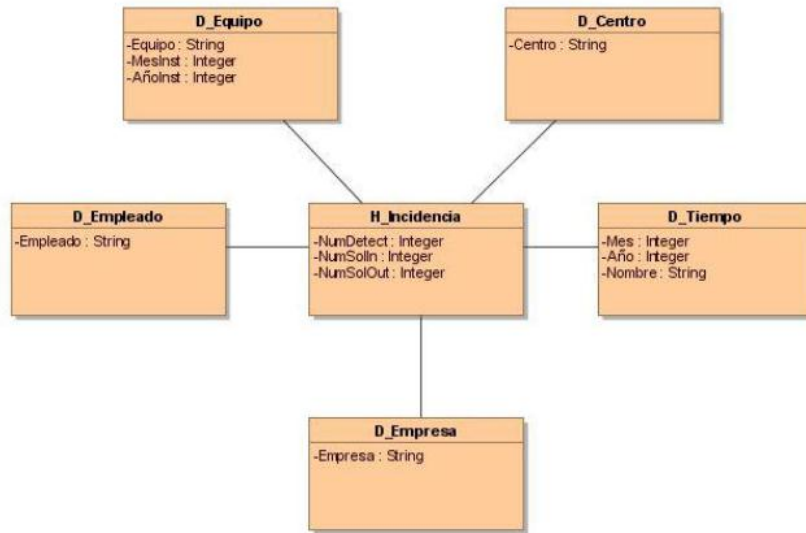


Cabecera	
ID Caso de Uso	CU06
Versión Caso de Uso	V.1.0
Estado	Aprobado
Autor	Carlos Cabrera
Cuerpo	
Título	Carga Datos en DW
Actores	Administrador, Proceso Automático Mensual
Flujo	El Administrador del sistema podrá lanzar individualmente cada uno de los procesos de carga de datos. Se podría definir en el SGBD un proceso automático que mensualmente realice la carga de las Incidencias y Mantenimientos del mes anterior. Los procesos de carga de Incidencias y Mantenimientos realizarán una llamada al resto de procesos de carga.



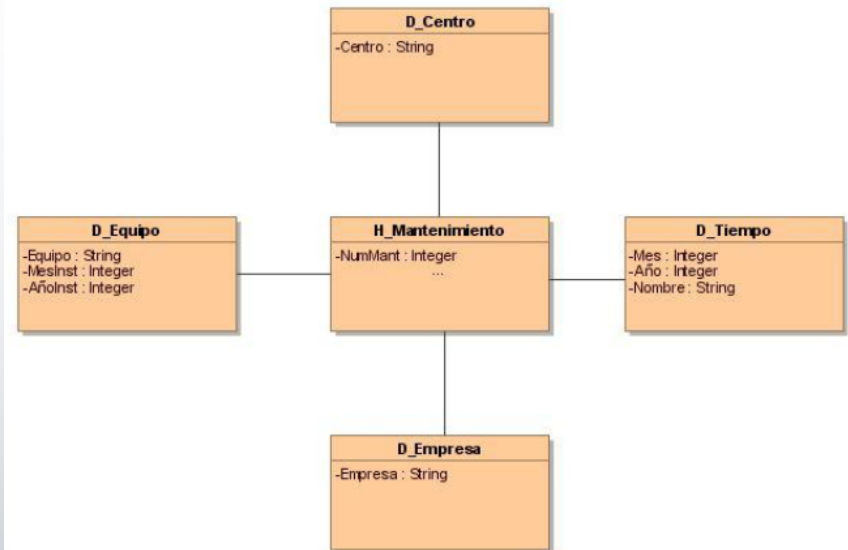
## Estadísticas de Incidencias

- Obtenemos el siguiente Diseño Conceptual, donde identificamos un Hecho (H\_Incidencia) y 5 Dimensiones (D\_Tiempo, D\_Equipo, D\_Centro, D\_Empresa, D\_Empleado).



## Estadísticas de Mantenimientos

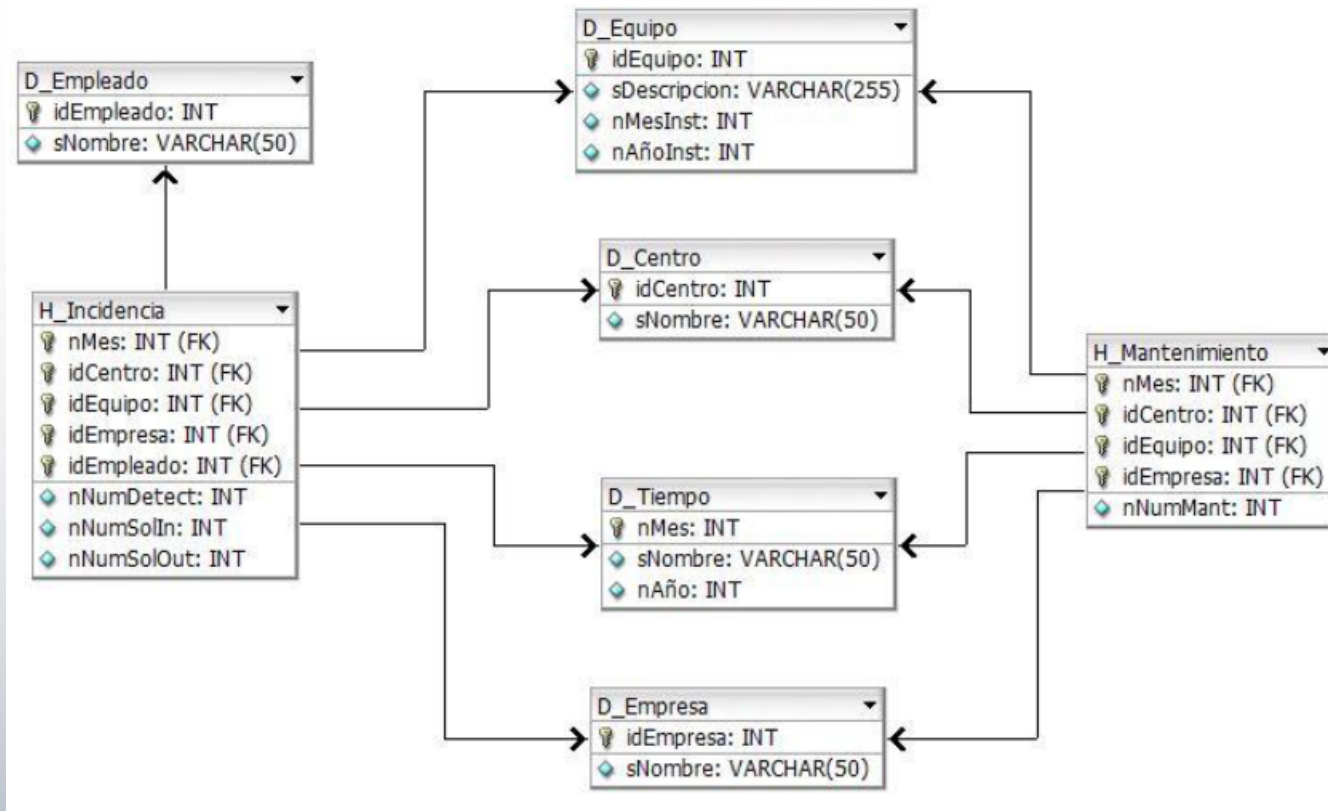
- Obtenemos el siguiente Diseño Conceptual, donde identificamos un Hecho (H\_Mantenimiento) y 4 Dimensiones (D\_Tiempo, D\_Equipo, D\_Centro, D\_Empresa).





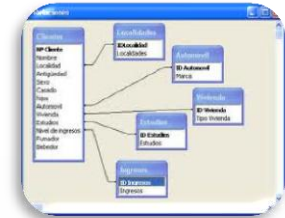


- En este punto pasamos a transformar el conceptual en un modelo lógico, detallando cada entidad, con sus atributos, claves primarias y claves foráneas





- A partir de las entidades y atributos identificados, detallamos el modelo físico de tablas, columnas, índices,...



H_Mantenimiento							
Columna	Tipo	Longitud	Permite Nulos	Indexada	Clave Primaria	Clave Foránea	
						Columna	Tabla
nMes	int		No	Si	Si	nMes	D_Tiempo
idCentro	int		No	Si	Si	idCentro	D_Centro
idEquipo	int		No	Si	Si	idEquipo	D_Equipo
idEmpresa	int		No	Si	Si	idEmpresa	D_Empresa
nNumMant	int		No				

\* nNumMant: número de mantenimientos realizados por mes, centro, equipo y empresa

H_Incidencia							
Columna	Tipo	Longitud	Permite Nulos	Indexada	Clave Primaria	Clave Foránea	
						Columna	Tabla
nMes	int		No	Si	Si	nMes	D_Tiempo
idCentro	int		No	Si	Si	idCentro	D_Centro
idEquipo	int		No	Si	Si	idEquipo	D_Equipo
idEmpresa	int		No	Si	Si	idEmpresa	D_Empresa
idEmpleado	Int		No	Si	Si	idEmpleado	D_Empleado
nNumDetect	Int		No				
nNumSolIn	Int		No				
nNumSolOut	Int		No				

\* nNumDetect: número de Incidencias detectadas por mes, centro, equipo, empresa y empleado

\* nNumSolIn: número de Incidencias solucionadas a tiempo por mes, centro, equipo, empresa y empleado

\* nNumSolOut: número de Incidencias solucionadas fuera de tiempo por mes, centro, equipo, empresa y empleado



```
shalo@gnz-lap:~$ touch script3.sh
shalo@gnz-lap:~$ nano script3.sh
shalo@gnz-lap:~$ chmod +x script3.sh
shalo@gnz-lap:~$ sh script3.sh
A continuación se buscará una cadena en un archivo
no existente y se mostrará
lo que devolvió la variable $?
Comando: grep donde poema.txt
grep: poema.txt: No existe el fichero o el directorio
2
-----Fin del script.-----
shalo@gnz-lap:~$
```

- Al igual que para la BD Relacional, se han creado una serie de scripts para facilitar la creación de los distintos objetos:
  - 8\_Creacion\_BD\_DW.sql
  - 9\_Crear\_Tablas\_DW.sql
  - 10\_Creacion\_SPs\_ETL\_DW.sql
  - 11\_Borrado\_Tablas\_DW.sql



- ❑ En la carpeta “Juegos de Pruebas ETL” se encuentran todos los juegos de pruebas realizados en los test de verificación de los procedimientos almacenados de ETL.
- ❑ Hay un fichero por cada una de las tablas correspondientes a Hechos y Dimensiones.
- ❑ Dentro de cada fichero se indica el procedimiento utilizado y los resultados obtenidos.

```

Eliminamos manualmente el Centro 1 y modificamos el Nombre del Centro 2
-----
D_CENTRO
-----
idCentro      sNombre
-----
2          xxxxxx
3          Almería
4          Ávila
5          Badajoz
...
-----
exec sp_ETL_D_Centro
-----
LogBD
-----
idLogBD  sStoredProcedure      sParametros      tHoraInicio      tHoraFin      sResultado
-----
691      sp_ETL_D_Centro      NULL              30/11/2011 17:11:17      30/11/2011 17:11:17      OK
-----
D_CENTRO
-----
idCentro      sNombre
-----
1          Albacete
2          Alicante
3          Almería
4          Ávila
5          Badajoz
    
```



- ❑ El coste del proyecto ha sido calculado en base al total de días de dedicación resultantes del detalle de tareas programadas.
- ❑ Asumimos la participación de varios roles diferentes dentro del proyecto.
- ❑ El precio estimado para los recursos se basa en una media en la zona de Lleida.
- ❑ Los diferentes roles, actividades y su coste son los siguientes:



Rol	Actividad	Días	Horas/día	% Dedicación	Horas	Precio/Hora	Coste total
Jefe Proyecto	Coordinación proyecto	84	2	10%	16,8	60 €	1.008 €
Analista	Análisis, Diseño y Documentación	84	2	60%	100,8	50 €	5.040 €
Desarrollador	Implementación código y Tests	84	2	30%	50,4	40 €	2.016 €
						<b>Total</b>	<b>8.064 €</b>

# Conclusiones



- ❑ Se ha seguido el Plan de Trabajo definido al inicio y que se han completado las tareas planificadas, cumpliendo bastante fielmente la planificación temporal realizada.
- ❑ Los objetivos asumidos al inicio del proyecto se han completado con éxito.
- ❑ Personalmente me ha permitido profundizar en el mundo del Transact-SQL, el cual apenas había utilizado.
- ❑ Es un aprendizaje más, ya que en mi experiencia laboral no llegas a realizar un proyecto por completo, de principio a fin, y en este caso así ha sido.

Fin

JUOC

