



**Universidad
Zaragoza**

Proyecto Fin de Carrera

DESARROLLO E IMPLEMENTACIÓN DE UNA SOLUCIÓN PARA EL ANÁLISIS, EXPLOTACIÓN Y VISUALIZACIÓN DE DATOS DE PACIENTES RECOGIDOS EN UNA UNIDAD DE CUIDADOS INTENSIVOS

Autor:

Jesús Echeverría Hita

Director:

Álvaro Alesanco Iglesias

Ingeniería de Telecomunicación

Departamento de Ingeniería Electrónica y Comunicaciones

Escuela de Ingeniería y Arquitectura

Zaragoza, junio de 2016

Agradecimientos

Este proyecto fin de carrera supone la culminación de una etapa muy larga e importante de mi vida. Durante este tiempo mucha gente ha estado a mi lado y este es un pequeño agradecimiento a todo lo que me han aportado.

*Quiero dar las gracias en primer lugar a **Álvaro** por su apoyo incondicional en la elaboración de este proyecto, por haber sabido comprender a la perfección mis condiciones familiares, laborales o geográficas, y haberme permitido realizar un proyecto de business intelligence en un ámbito funcional, el de eHealth, que yo desconocía.*

*Sin duda, a mi madre, **Julia**, quien con una delicada situación familiar y su enorme esfuerzo pagó mis estudios. Siento el que la espera haya tenido que ser tan larga pero, si a alguien debo lo que tengo, es a ella.*

*A mis hermanas, **Itxaso** y **Cristina**, quienes sin duda también tuvieron que arrimar su hombro para que pudiese optar a unos estudios universitarios, aunque eso pudiese significar privarse de los suyos propios.*

*A aquellos compañeros de trabajo, de estudios y amigos que han apoyado sin cansarse la finalización de mis estudios, por muy larga que pudiera parecer la espera. Este agradecimiento es para **Carlos, Antonio, Juan Antonio (Barri), Lander, Guillermo (Willy), María** o **Ana**. Su apoyo me ha dado ánimos para mirar hacia adelante y lograr este objetivo.*

*A mis distintos responsables de Banco de España: **Asunción (Chon), Susana** y **Caridad**, quienes han insistido tanto en que no abandonase mis estudios por difícil que pudiera ser retomarlos tan tardíamente. Su seguimiento y ánimo me han permitido continuar sin decaer.*

*Y, por último, a la gente que más quiero: mi familia. Mi mujer **Ana Fuencisla**, quien tanto me ha apoyado y que me ha aportado la ayuda necesaria para llegar hasta aquí, y mis tres hijos **Arkaitz, Irati** y **Haizea**, a quienes sin duda he tenido que robar tiempo para poder compaginar trabajo y estudios. Sólo espero poder recuperar con creces este tiempo parcialmente perdido para poder recompensaros como es debido. Os quiero mucho.*

Y a todas las personas que, de una u otra manera, me han ayudado a lo largo de estos años. A todos, simplemente, gracias.

RESUMEN

El apoyo de las tecnologías de información a los diferentes ámbitos de la vida cotidiana es, cada día, mayor que el día anterior, a un ritmo ni mucho menos lineal. El del Business Intelligence es un área de conocimiento que no descansa, que aporta mucho valor a cualquier sector y ello hace que, incluso en las mayores debacles económicas, se apueste por dicha tecnología al reconocérsele su fundamental labor de permitir deducir conclusiones y, por lo tanto, evitar repeticiones indeseadas a futuro. Se trata de un sector en plena expansión y que es aplicable a una multitud de actividades.

En el caso concreto de este proyecto, su ámbito de aplicación serán los datos existentes en unos ficheros proporcionados voluntariamente por una organización, Physionet, cuyo contenido consiste en el registro continuado de las admisiones de pacientes en hospitales, en una larga serie histórica, con una abundante muestra de distintas tipologías de pacientes, así como con distintas patologías y evoluciones. Entre estos datos existirán colecciones de pacientes, de sanitarios, de admisiones hospitalarias, de servicios prestados en las diferentes secciones del hospital, de estancias en la unidad de cuidados intensivos (UCI), de las distintas medidas de signos vitales tomadas, de sus muestras de laboratorio, etc.

Partiendo de esta información en ficheros, se implementará una base de datos relacional Oracle en la que poder hacer un mejor uso de recursos a la hora de consultar dichos datos. Esta base de datos permitirá que los diferentes datos de ficheros tomen entidad propia, normalicen su aspecto, manifiesten características de unicidad, o reflejen relaciones existentes entre ellos, algo no posible en los ficheros originalmente proporcionados.

Pero el manejo de una base de datos no evita una continua sucesión de consultas, de tipo SQL (Structured Query Language) en este caso, que permitan conocer los datos existentes y operar con ellos, lo cual conlleva un continuo desarrollo de programas a medida para cada estudio concreto a realizar. Es por ello que parece razonable optar por una herramienta de análisis y explotación de la información que, a partir del esfuerzo de una etapa inicial de diseño lógico de las estructuras más fundamentales, de modelización del negocio concreto a estudiar, permita posteriormente un mayor dinamismo a la hora de elaborar estudios, o aporte alternativas tecnológicas más potentes y óptimas que un desarrollo por informe deseado.

Para dar solución a este interés, se optará por implementar un proyecto en la herramienta de Datawarehouse por excelencia, MicroStrategy, de manera que el conocimiento funcional de los ficheros proporcionados por Physionet queden plasmados en una serie de objetos inteligentes, que permitan recuperar la información dinámicamente del repositorio más apropiado en cada momento, todo ello con una interfaz visual más liviana y con una operativa mucho más simple, sin necesidad de conocimientos de programación en SQL.

En dicho proyecto se elaborarán tanto dichas estructuras fundamentales como otros objetos más cercanos a un usuario final, que permitan conocer de manera individual o agregada los datos disponibles. De esta manera, la misma información podrá ser analizada desde muchas caras de un mismo prisma, con gran fluidez, permitiendo obtener conclusiones, tomar decisiones o, incluso, anticiparse a comportamientos futuros más o menos predecibles.

Se tratará, sin duda, de una confortable relación de simbiosis entre las tecnologías de la información y la medicina, en el conocido ámbito del eHealth. Personalmente, esperamos que sea de su agrado y disfruten de este proyecto.

Índice general

1	Introducción y objetivos	1
1.1	Introducción	1
1.2	Objetivos	2
1.3	Materiales y herramientas utilizadas	3
1.4	Organización de la memoria	3
2	Primer contacto y descarga de datos	5
2.1	Primer contacto con Physionet	5
2.2	Descarga de los ficheros	7
3	Configuración del software	9
4	Explotación de la información	13
4.1	Creación de objetos de esquema	13
4.1.1	Tablas	14
4.1.2	Atributos	22
4.1.3	Hechos	30
4.1.4	Jerarquías	35
4.2	Creación de objetos públicos	37
4.2.1	Indicadores	37
4.2.2	Filtros	39
4.2.3	Selecciones dinámicas	40
4.2.4	Informes	42
4.2.5	Navegaciones	48

4.2.6	Tableros de control	57
4.3	Creación de objetos del proyecto	58
4.3.1	Filtros de seguridad	58
4.4	Cálculo predictivo	62
4.4.1	Métricas de formación	62
4.5	Ejemplo de aplicación	70
4.5.1	Caso práctico	70
5	Conclusiones	75
6	Planificación del proyecto	79
	Bibliografía	81
A	Descarga e instalación de Oracle	83
A.1	Descarga de Oracle	83
A.2	Instalación de Oracle	89
B	Descarga e instalación de SQL Developer	97
B.1	Descarga de SQL Developer	97
B.2	Instalación de SQL Developer	99
C	Creación de la base de datos	103
C.1	Acceso mediante usuario administrador	103
C.2	Creación del usuario propietario del esquema	105
C.3	Creación de la estructura de tablas	108
C.4	Carga de ficheros a tablas	109
C.5	Ajustes adicionales tras las cargas	113
C.6	Creación de usuarios de metadata y warehouse	113
C.7	Visión general del modelo de datos físico	118
D	Descarga e instalación de MicroStrategy	121
D.1	Descarga de MicroStrategy	121
D.2	Instalación de MicroStrategy	124

D.3 Configuración de MicroStrategy	133
D.4 Creación de un proyecto	152

Índice de figuras

2.1	Certificado de finalización de curso.	6
2.2	Listado de ficheros a descargar.	7
4.1	Menú de Catálogo de Warehouse.	14
4.2	Tablas disponibles en el Catálogo de Warehouse.	15
4.3	Funcionalidades adicionales del Catálogo de Warehouse.	15
4.4	Sentencia SQL contra la tabla física ‘PATIENTS’.	16
4.5	Definición de la tabla lógica ‘LU_ADMITTIME_YYYYMM’.	17
4.6	Sentencia SQL contra la tabla lógica ‘LU_ADMITTIME_YYYYMM’.	18
4.7	Sentencia SQL contra la tabla de look-up física ‘LU_CHARTTIME_YYYYMM’.	19
4.8	Creación de la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’.	20
4.9	Sentencia SQL contra la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’.	21
4.10	Sentencia SQL contra la tabla de hechos ‘CHARTEVENTS’.	21
4.11	Menú contextual de ‘Nuevo – Atributo’.	23
4.12	Definición del atributo ‘Paciente’ sobre la tabla ‘PATIENTS’.	24
4.13	Definición completa del identificador único del atributo ‘Paciente’.	25
4.14	Representaciones adicionales no únicas del atributo ‘Paciente’.	26
4.15	Definición completa del atributo ‘Paciente’.	26
4.16	Acceso a la funcionalidad de ‘Actualizar Esquema’.	27
4.17	Opciones de parametrización de la funcionalidad de ‘Actualizar Esquema’.	28
4.18	Definición de padres del atributo ‘Paciente’.	28

4.19	Definición de hijos del atributo ‘Paciente’.	29
4.20	Vista lógica de la tabla ‘PATIENTS’.	30
4.21	Definición de un hecho sobre la tabla ‘PATIENTS’.	32
4.22	Expresión del hecho sobre la tabla ‘AGG_MEDIDA_ETNIA_DATO’.	33
4.23	Expresión del hecho sobre la tabla ‘CHARTEVENTS’.	33
4.24	Consulta SQL de un informe que utiliza el hecho.	34
4.25	Definición de la jerarquía ‘Pacientes’.	36
4.26	Vista global de las jerarquías del sistema.	37
4.27	Definición de un indicador como promedio de un hecho.	38
4.28	Definición del filtro ‘Pacientes fallecidos’.	39
4.29	Definición de un filtro dinámico sobre el año de nacimiento.	40
4.30	Selección dinámica para solicitar el año de nacimiento.	41
4.31	Información general a definir en la selección dinámica.	41
4.32	Definición de un informe de edad media de fallecimiento.	42
4.33	Solicitud de años de nacimiento de los pacientes a consultar.	43
4.34	Aspecto de la resolución del informe en modo cuadrícula.	44
4.35	Aspecto de la resolución del informe en modo gráfico.	44
4.36	Sintaxis de la sentencia SQL de resolución del informe.	45
4.37	Informe gráfico del porcentaje de pacientes fallecidos por género.	46
4.38	Evolución temporal de pacientes y estancias en UCI de pacientes fallecidos.	46
4.39	Admisiones por seguro médico, estado civil y etnia del paciente.	47
4.40	Admisiones por etnia y seguro médico del paciente.	47
4.41	Muestras de laboratorio por género y tipo de muestra.	48
4.42	Medidas tomadas a pacientes fallecidos por última unidad de UCI.	48
4.43	Tiempo de permanencia en UCI según el flujo entrada-salida.	49
4.44	Filtro dinámico por tipo de seguro médico en el informe.	50
4.45	Aspecto del informe una vez resuelto.	50
4.46	Alternativas de navegación en el informe.	51
4.47	Informe tras navegar a ‘Procedencia del paciente’.	51
4.48	Navegación para llegar al nivel de detalle de ‘Paciente’.	51

4.49	Resultado del informe ya navegado a ‘Paciente’.	52
4.50	Modificación del mapa de navegación del atributo ‘Paciente’.	53
4.51	Informe de los cinco pacientes con más admisiones en web.	53
4.52	Resultado de navegación por defecto a plantilla desde web.	53
4.53	Elevado peso lógico de la tabla ‘CHARTEVENTS’.	54
4.54	Menor peso lógico de la tabla ‘AGG_MEDIDA_ETNIA_DATO’.	55
4.55	Informe resuelto contra la tabla agregada.	55
4.56	Consulta SQL empleada contra la tabla agregada.	56
4.57	Selección de un destino de navegación desde el informe.	56
4.58	Dimensionalidad de informe disconforme con la tabla agregada.	57
4.59	Consulta SQL empleada contra la tabla de hechos original.	57
4.60	Composición de visualizaciones en un tablero de control.	58
4.61	Administración de usuarios del proyecto en MicroStrategy.	59
4.62	Mantenimiento del filtro de seguridad de un usuario concreto.	60
4.63	Selección y asociación de un filtro de seguridad concreto.	60
4.64	Informe resuelto sin restricciones con usuario ‘Administrator’.	61
4.65	Consulta SQL del informe resuelto sin filtro de seguridad.	61
4.66	Informe resuelto con filtro de seguridad con usuario ‘Sanitario’.	62
4.67	Condicionamiento implícito debido al filtro de seguridad.	62
4.68	Comprobación de datos mediante el uso de navegación.	62
4.69	Comienzo del asistente de indicador de formación.	63
4.70	Selección del tipo de análisis a realizar.	64
4.71	Selección de variables dependientes e independientes para el análisis.	65
4.72	Especificación de objetos resultantes del indicador de formación.	65
4.73	Resumen de la definición del indicador de formación.	66
4.74	Contenido interno del indicador probabilístico automáticamente creado.	66
4.75	Cubo de coeficientes resultantes de la regresión.	67
4.76	Visor de la calidad de los coeficientes obtenidos.	67
4.77	Estándar PMML de exportación del modelo predictivo.	68
4.78	Contraste gráfico de datos reales frente a datos predictivos.	68

4.79	Definición de un umbral sobre el porcentaje de desviación.	69
4.80	Informe resuelto contemplando el umbral de aviso por desviación. . .	69
4.81	Indicador de distintos pacientes tratados.	70
4.82	Indicador de distintos pacientes fallecidos.	71
4.83	Indicador de tasa de Pacientes fallecidos vs. tratados.	71
4.84	Definición del informe de interés.	72
4.85	Resultados de ejecución del informe.	72
4.86	Comprobación de resultados en Oracle.	73
5.1	Costes de licencias de MicroStrategy.	76
6.1	Diagrama de Gantt del proyecto.	79
A.1	Formulario de alta en la web de Oracle.	83
A.2	Sitio web de descargas de Oracle.	84
A.3	Descargas disponibles de Oracle Database 11g Release 2.	85
A.4	Especificaciones del puesto de trabajo.	85
A.5	Accesos directos a las descargas de archivos.	86
A.6	Aspecto durante la descarga de archivos.	86
A.7	Archivos de instalación de Oracle descargados.	86
A.8	Versión de evaluación de Winzip.	87
A.9	Configuración de la descompresión de los archivos.	87
A.10	Resultado de la descompresión del primer archivo.	88
A.11	Resultado de la descompresión del segundo archivo.	88
A.12	Situación final tras la correcta descompresión.	89
A.13	Localización del archivo de instalación.	89
A.14	Comienzo del proceso de instalación.	90
A.15	Advertencia de incompatibilidad del sistema.	90
A.16	Configuración de actualizaciones de seguridad.	91
A.17	Selección de instalación y creación de una base de datos.	92
A.18	Selección de la versión de escritorio para la instalación.	92
A.19	Configuración de parámetros esenciales de la base de datos.	93

A.20	Resumen final de la instalación de Oracle.	94
A.21	Respaldo del archivo de resumen de la instalación.	94
A.22	Progreso de la instalación de Oracle.	95
A.23	Progreso de la creación de la base de datos.	95
A.24	Finalización de la creación de la base de datos.	96
A.25	Mensaje de terminación y URL de Enterprise Manager.	96
B.1	Sitio de descarga de SQL Developer.	98
B.2	Progreso de la descarga de SQL Developer.	98
B.3	Archivo instalador comprimido de SQL Developer.	99
B.4	Descompresión del instalador de SQL Developer.	100
B.5	Aspecto portable de SQL Developer en el puesto de trabajo.	100
C.1	Conexión para el usuario administrador ‘SYSTEM’.	104
C.2	Ping a la instancia ‘MIMIC3EE’.	104
C.3	Relación de tamaños de bloque y de datafile.	105
C.4	Consulta SQL para conocer el tamaño de bloque.	106
C.5	Creación de un tablespace mediante cuatro datafiles.	106
C.6	Aspecto de los cuatro datafiles en el puesto de trabajo.	106
C.7	Creación del usuario propietario del esquema ‘MIMIC3_OWNER’.	107
C.8	Concesión de privilegios al usuario ‘MIMIC3_OWNER’.	107
C.9	Conexión para el usuario propietario ‘MIMIC3_OWNER’.	108
C.10	Opción ‘Importar Datos’ de SQL Developer.	109
C.11	Selección del archivo de datos a importar.	109
C.12	Selección del método de carga de datos a utilizar.	110
C.13	Elección de columnas concretas a importar.	110
C.14	Corrección de las máscaras por defecto para tipos ‘DATE’.	111
C.15	Cálculo de estadísticas a todo el esquema Oracle.	114
C.16	Creación del usuario de Metadatos ‘MIMIC3_METADATA’.	115
C.17	Creación del usuario de conexión ‘MIMIC3_WAREHOUSE’.	115
C.18	Concesión de privilegios al usuario ‘MIMIC3_METADATA’.	116
C.19	Concesión de privilegios al usuario ‘MIMIC3_WAREHOUSE’.	117

C.20 Conexión para el usuario de Metadatos ‘MIMIC3_METADATA’. . . 117

C.21 Conexión para el usuario de conexión ‘MIMIC3_WAREHOUSE’. . . 117

C.22 Modelo de datos físico de MIMICIII. 119

D.1 Sitio web de descargas de MicroStrategy. 122

D.2 Descargas disponibles tras la identificación. 122

D.3 Versión 10.2.0 de MicroStrategy para Windows. 123

D.4 Archivo instalador comprimido tras la descarga. 123

D.5 Archivos de instalación ya descomprimidos. 123

D.6 Pantalla de bienvenida del proceso de instalación. 124

D.7 Aviso de la necesidad de una clave de activación. 124

D.8 Selección de idioma de MicroStrategy. 125

D.9 Progreso de la preparación de la instalación. 125

D.10 Arranque del recorrido de instalación. 125

D.11 Aviso de detención necesaria de servicios Windows. 126

D.12 Aceptación de las condiciones de la licencia. 126

D.13 Aportación de datos personales y clave de licencia. 126

D.14 Selección de rutas de instalación del software. 127

D.15 Selección de componentes integrantes de la instalación. 127

D.16 Primera sección de componentes seleccionados. 127

D.17 Segunda sección de componentes seleccionados. 128

D.18 Advertencia de necesidad de software adicional. 128

D.19 Limitaciones de uso de la versión de sistema operativo. 128

D.20 Código de acceso al módulo Health Center. 129

D.21 Recordatorio de uso de License Manager. 129

D.22 Primera sección de la activación del servidor. 129

D.23 Segunda sección de la activación del servidor. 130

D.24 Solicitud de un código de activación del servidor. 130

D.25 Credenciales de arranque de Intelligence Server. 130

D.26 URL de acceso a Web Services. 131

D.27 Recordatorio para la posterior definición de la URL. 131

D.28 Elección de asociaciones entre MicroStrategy y Microsoft Office. . .	132
D.29 Resumen de la parametrización de copia de archivos.	132
D.30 Comienzo de la instalación de MicroStrategy.	132
D.31 Finalización de la instalación.	133
D.32 Selección de creación de repositorios.	133
D.33 Definición de repositorios concretos a crear.	134
D.34 Especificación del DSN de Metadata.	135
D.35 Acceso al asistente de conectividad de MicroStrategy.	135
D.36 Bienvenida al asistente de conectividad.	135
D.37 Tipo de base de datos que contendrá el repositorio.	136
D.38 Ruta de conexión a emplear en este repositorio.	136
D.39 Prueba de conectividad mediante usuario y contraseña.	137
D.40 Resultado satisfactorio de la prueba de conectividad.	137
D.41 Mensaje de DSN creado correctamente.	137
D.42 Tipo de base de datos del segundo repositorio.	138
D.43 Ruta de conexión al segundo repositorio.	138
D.44 Prueba de conectividad del segundo repositorio.	139
D.45 Resultado correcto de la segunda prueba de conectividad.	139
D.46 Mensaje de segundo DSN creado.	139
D.47 Selección del DSN de Metadata recién definido.	140
D.48 Vista preliminar del SQL de creación de la Metadata.	140
D.49 Selección del DSN de Historial.	140
D.50 Selección del DSN de Estadísticas.	141
D.51 Resumen de repositorios recién declarados.	141
D.52 Arranque de la creación de los distintos repositorios.	142
D.53 Resumen de la creación de los repositorios.	142
D.54 Regreso al menú principal de configuración de la instalación.	142
D.55 Avance hacia la configuración de Intelligence Server.	143
D.56 Selección de la conexión válida para el repositorio de Metadata. . .	143
D.57 Alta de usuario y contraseña administrador.	144
D.58 Intento de conexión con el repositorio de Metadata.	144

D.59 Desbloqueo de características del Firewall de Windows.	144
D.60 Definición de servidores de Intelligence Server.	144
D.61 Puerto de comunicaciones de Intelligence Server.	145
D.62 Posibilidad de empleo de SSL en las comunicaciones.	145
D.63 Selección de la conexión válida para el repositorio de Estadísticas.	146
D.64 Resumen de la configuración de Intelligence Server.	146
D.65 Ejecución de esta configuración.	147
D.66 Resumen de la ejecución de la configuración.	147
D.67 Regreso al menú principal del asistente de configuración.	147
D.68 Nombre del nuevo origen de proyecto a crear.	148
D.69 Resumen de la Metadata a emplear por el origen de proyecto.	148
D.70 Definición del tipo de seguridad a emplear.	149
D.71 Resumen previo a la ejecución del origen de proyecto.	149
D.72 Ejecución de la creación del origen de proyecto.	150
D.73 Regreso al menú principal del asistente de configuración.	150
D.74 Acceso a License Manager.	151
D.75 Introducción de la clave de activación del producto.	151
D.76 Mensaje satisfactorio de la activación del producto.	152
D.77 Estado de la licencia en License Manager.	152
D.78 Acceso a la creación de un nuevo proyecto.	153
D.79 Asistente de creación de proyecto.	153
D.80 Definición de parámetros generales del proyecto.	154
D.81 Creación del nuevo proyecto.	154
D.82 Asignación de roles al usuario actual en el proyecto.	155
D.83 Acceso al catálogo de Warehouse del proyecto.	155
D.84 Selección de instancia asociada al Warehouse.	156
D.85 Instancia de Warehouse no disponible en la selección.	156
D.86 Asistente de creación de la instancia de Warehouse.	156
D.87 Tipo de base de datos de la instancia de Warehouse.	156
D.88 Selección del ODBC correspondiente al Warehouse.	157
D.89 Resumen de parámetros de la instancia de Warehouse.	157

D.90 Selección de la instancia de Warehouse recién creada.	157
D.91 Advertencia de catálogo de warehouse vacío en el proyecto.	158
D.92 Concesión de permisos al usuario de conexión de MicroStrategy. . .	158
D.93 Creación de sinónimos hacia objetos del usuario propietario.	159
D.94 El catálogo de warehouse continúa vacío.	159
D.95 Formulario de opciones del catálogo de warehouse.	160
D.96 Configuración de sentencia SQL de carga del catálogo de warehouse.	160
D.97 Catálogo de warehouse correctamente cargado con objetos disponibles.	161
D.98 Selección de objetos utilizables en el proyecto.	161
D.99 Comentarios tras los cambios en el catálogo de warehouse.	162
D.100 Respaldo del catálogo de warehouse al repositorio de Metadata. . .	162
D.101 Posibilidad de creación de objetos de esquema del proyecto.	162
D.102 Finalización del asistente de creación de proyectos.	163

Capítulo 1

Introducción y objetivos

1.1 Introducción

El avance de las tecnologías de la información y comunicaciones permite, día a día, su utilización en un mayor abanico de áreas de aplicación. No hay tipología de negocio donde no se vea una posibilidad de utilización de las mismas con garantías de éxito en sus resultados.

Desde el ámbito concreto de eHealth, los logros tecnológicos más recientes son siempre bien recibidos o, incluso, previamente esperados. Es deseable que, dichos avances tecnológicos, sean un buen apoyo a partir del cual mejorar sustancialmente aspectos como la esperanza de vida de las personas, localizar precozmente enfermedades, gestionar de una mejor manera los recursos disponibles o relacionar factores potencialmente causantes de ciertas dolencias, por poner varios ejemplos.

Para ello, es primordial disponer de una gran cantidad de datos reales con la que lograr una observación lo suficientemente generalizada como para no incurrir en errores de extrapolación, así como utilizar esta información de la manera más eficaz posible con el fin de no desperdiciar el menor de los valores añadidos que dicha información pueda aportar.

Por lo tanto, parece lógica una simbiosis entre eHealth y Business Intelligence con el fin de poder proporcionar facilidades al análisis de esta información,

posibilitar la explotación de grandes volúmenes de datos relacionados entre sí, proporcionar resultados directos y con gran dinamismo, así como poder llegar a apoyar la toma de decisiones o la obtención de conclusiones. Los entornos Datawarehouse dotan de gran agilidad la obtención de estos resultados, siendo una gran solución de tratamiento ante la avalancha de datos digitales que, cada día más, envuelve cada situación cotidiana.

1.2 **Objetivos**

El objetivo de este Proyecto Fin de Carrera es disponer de una plataforma tecnológica que permita la explotación de datos médicos tomados a una gran muestra de pacientes durante sus estancias en hospitales (con sus distintas admisiones, sus estancias o no en la Unidad de Cuidados Intensivos, sus mediciones de determinados indicadores médicos, sus analíticas, etc.) proporcionando mecanismos tecnológicos que, cómodamente, permitan al usuario (bien sea éste un gerente de hospital, un auxiliar médico, un doctor, un estadista...) conocer la situación de todos y cada uno de estos datos, con el fin de poder estudiarlos de la manera más acorde a sus necesidades y de una manera sencilla, dinámica e intuitiva que evite desarrollar programas a la medida de cada estudio concreto, de trasladar estos datos a otras herramientas más específicas de análisis, sin conocer lenguajes de programación o de explotación de datos.

Para ello, se dispondrá de los datos originales, proporcionados mediante descargas de ficheros por una organización denominada Physionet. Esta información será almacenada apropiadamente en una base de datos relacional, con el fin de gestionar apropiadamente sus consultas, así como proceder a un más eficaz almacenamiento de los datos. Asimismo, se dispondrá de una suite de herramientas de organización, modelizado y consulta de estos datos, de manera que no sea necesario el conocimiento informático de más bajo nivel por parte de los usuarios que quieran analizar esta información sino que, conociendo el negocio (estructura funcional y relaciones) se pueda hacer un uso casi ofimático de estas herramientas.

1.3 Materiales y herramientas utilizadas

El material necesario (en este caso la información base a explotar) consistirá en una serie de ficheros con formato CSV que serán puestos a disposición de ser descargados por parte de la organización Physionet. Estos datos constituyen el sistema MIMIC3, un amplio conjunto de datos relacionados entre sí acerca de pacientes que han atravesado estancias en una unidad de cuidados intensivos (UCI).

Con el fin de almacenar correctamente esta información, de manera que posteriormente pueda ser analizada más fácilmente y con mejores rendimientos mediante consultas SQL, se dispondrá de un gestor de base de datos, Oracle concretamente, donde estos ficheros descargados poblarán una serie de tablas.

Para poder gestionar apropiadamente esta base de datos, se dispondrá de la herramienta SQL Developer, propiedad también del fabricante Oracle. Con ella será más sencillo gestionar la creación de las tablas, sus cargas, alta y configuración de usuarios, etc., proporcionando un entorno más amigable que no haga necesario optar por sentencias invocadas a través de línea de comandos.

La etapa de modelización y explotación de la información correrá a cargo de una suite de aplicaciones denominada MicroStrategy. Apoyándose en un servidor de aplicaciones Apache Tomcat funcionará su aplicación MicroStrategy Web mientras que, desde la aplicación MicroStrategy Developer, será posible, adicionalmente, diseñar objetos más avanzados o más relacionados con el propio esquema físico de la base de datos.

1.4 Organización de la memoria

En el capítulo 1 se ha desarrollado una breve introducción al PFC y los objetivos principales que se han perseguido.

El capítulo 2 analiza el modo en el que se ha llegado a acceder a la información de la plataforma MIMIC3 procedente de la organización Physionet, así como el procedimiento empleado para la descarga de esta información en forma de ficheros.

En el capítulo 3 se analiza de manera bastante general la diversidad de componentes tecnológicos necesarios para la realización del proyecto, redirigiendo a los diferentes anexos existentes con el fin de obtener una información mucho más detallada.

El capítulo 4 detalla muy explícitamente todos los pasos que deben seguirse en la herramienta MicroStrategy con el fin de crear todos los objetos necesarios para poder configurar apropiadamente un proyecto de Business Intelligence que sea capaz de explotar la información de MIMIC3. Es, sin duda, la parte principal del Proyecto Fin de Carrera, de manera que es distribuido de manera secuencial en la creación de objetos de esquema, creación de objetos públicos, definición de objetos del proyecto y, por último, se hace referencia al cálculo predictivo.

El capítulo 5 trata de reflejar las conclusiones principales a las que se ha llegado tras la elaboración de este proyecto, así como cuáles podrían ser las líneas siguientes de actuación en las que futuros proyectos podrían profundizar, intentando sacar un mayor partido a este tipo de tecnologías de análisis tanto estadístico, como de seguimiento y predictivo.

El capítulo 6 representa un diagrama temporal aproximado de la organización de tareas del proyecto.

En el anexo A se dispone de una detallada guía de la descarga e instalación de Oracle como gestor de base de datos empleado en el proyecto.

En el anexo B se dispone de otra detallada guía, en este caso de descarga e instalación de SQL Developer como herramienta de administración de la base de datos Oracle.

En el anexo C se describen paso a paso los detalles de creación de la base de datos Oracle que albergará los datos de MIMIC3 (acceso mediante usuario administrador, creación de diferentes usuarios, creación de tablas, ajustes diversos), así como las actuaciones realizadas con el fin de trasladar la información residente en ficheros a las propias tablas.

En el anexo D, por último, se dispone de una detallada guía de descarga e instalación de la suite de MicroStrategy, con una amplia guía de configuración y creación del proyecto concreto destinado a gestionar la información de MIMIC3.

Capítulo 2

Primer contacto y descarga de datos

2.1 Primer contacto con Physionet

En primer lugar, se establece una primera toma de contacto con la información médica de pacientes que sería objetivo de análisis a través de la siguiente web:

<http://mimic.physionet.org/>

En dicha web reside mucha información acerca de los datos existentes en su repositorio de ficheros, cómo poder obtenerlos, o sugerencias de explotación de los mismos en distintos gestores.

Tal y como describe esta página web, con el fin de poder descargar libremente los ficheros con la información deseada, previamente se debe superar una aceptación por parte de su organización Physionet. Y, dicha aceptación, requiere de la previa realización de un curso que acredite que, la persona que descargará la información (para su uso personal, nunca transferible), es conocedora de los correctos procedimientos con los que dicha información debe ser tratada (al disponer ésta de datos sensibles de pacientes, por mucho que dicha información se encuentre muy despersonalizada, habiéndose tratado de enmascararla al máximo posible).

Por tanto, un primer paso consistirá en obtener un usuario y contraseña con el que poder acceder a la zona de esta web que requiere de una identificación, al no tratarse ya de una zona de libre acceso. Para ello, el alta de usuario se gestionará a través del siguiente enlace:

<https://physionet.org/pnw/login>

A continuación, una vez obtenido este primer acceso a la organización, se debe gestionar la realización de un curso on-line de formación sobre tratamiento de datos sensibles (salvo que ya se disponga del mismo con anterioridad). Para ello, se accederá a este enlace:

<https://phrp.nihtraining.com/users/login.php>

y, tras crear también una cuenta de acceso, se procederá a realizar el correspondiente curso de formación. Una vez finalizado éste, se obtendrá una acreditación con el aspecto mostrado en la Figura 2.1.



Figura 2.1: Certificado de finalización de curso.

Con este documento y la aportación de una serie de datos adicionales acerca del objetivo último de la descarga de los ficheros, se solicitará a Physionet el acceso libre a la descarga de ficheros. Tras realizar dicha solicitud y verificar Physionet la información acreditada, se recibirá el correspondiente correo electrónico desde el que obtener un certificado a través de la siguiente URL:

<http://mimic.mit.edu/certs/lcpca.crt>

que, una vez instalado en el puesto desde el que realizar las descargas, permitirá obtener los ficheros deseados.

2.2 Descarga de los ficheros

Una vez ubicado el certificado en el puesto de descargas, el acceso a los diferentes ficheros en los que reside la información se realiza a través del siguiente enlace:

<https://physionet.org/works/MIMICIIIClinicalDatabase/files/>

Se trata de 26 ficheros de datos (y 2 ficheros adicionales de verificación de integridad), comprimidos para reducir el volumen de la descarga, que pueden resumirse con la Figura 2.2.

1. [checksum_md5_zipped.txt](#) (MD5 checksum for zipped files)
2. [checksum_md5_unzipped.txt](#) (MD5 checksum for unzipped files)
3. [ADMISSIONS.csv.gz](#) (2.4M compressed, 12M decompressed)
4. [CALLOUT.csv.gz](#) (1.2M compressed, 6.1M decompressed)
5. [CAREGIVERS.csv.gz](#) (49K compressed, 204K decompressed)
6. [CHARTEVENTS.csv.gz](#) (3.2G compressed, 27G decompressed)
7. [CPTEVENTS.csv.gz](#) (4.8M compressed, 57M decompressed)
8. [DATETIMEEVENTS.csv.gz](#) (43M compressed, 515M decompressed)
9. [DIAGNOSES_ICD.csv.gz](#) (4.6M compressed, 18M decompressed)
10. [DRGCODES.csv.gz](#) (1.7M compressed, 10M decompressed)
11. [D_CPT.csv.gz](#) (4.0K compressed, 14K decompressed)
12. [D_ICD_DIAGNOSES.csv.gz](#) (281K compressed, 1.3M decompressed)
13. [D_ICD_PROCEDURES.csv.gz](#) (75K compressed, 304K decompressed)
14. [D_ITEMS.csv.gz](#) (185K compressed, 989K decompressed)
15. [D_LABITEMS.csv.gz](#) (11K compressed, 43K decompressed)
16. [ICUSTAYS.csv.gz](#) (1.9M compressed, 6.1M decompressed)
17. [INPUTEVENTS_CV.csv.gz](#) (414M compressed, 2.4G decompressed)
18. [INPUTEVENTS_MV.csv.gz](#) (144M compressed, 949M decompressed)
19. [LABEVENTS.csv.gz](#) (321M compressed, 1.8G decompressed)
20. [MICROBIOLOGYEVENTS.csv.gz](#) (3.8M compressed, 44M decompressed)
21. [NOTEVENTS.csv.gz](#) (1.2G compressed, 3.7G decompressed)
22. [OUTPUTEVENTS.csv.gz](#) (56M compressed, 386M decompressed)
23. [PATIENTS.csv.gz](#) (559K compressed, 2.5M decompressed)
24. [PRESCRIPTIONS.csv.gz](#) (100M compressed, 742M decompressed)
25. [PROCEDUREEVENTS_MV.csv.gz](#) (7.6M compressed, 49M decompressed)
26. [PROCEDURES_ICD.csv.gz](#) (1.8M compressed, 6.5M decompressed)
27. [SERVICES.csv.gz](#) (1.2M compressed, 3.4M decompressed)
28. [TRANSFERS.csv.gz](#) (5.3M compressed, 25M decompressed)

Figura 2.2: Listado de ficheros a descargar.

Por lo tanto, se realiza la descarga de cada uno de los ficheros involucrados, momento a partir del cual el acceso a la información de Physionet puede seguir realizándose de manera anónima, sin necesidad de acreditar datos de usuario (ya con el único fin de obtener información aclarativa acerca de los datos).

Para tener una primera visión de los datos existentes en los ficheros, desde la web de la organización se facilita acceso a un entorno simplificado de consultas de dichos datos, mediante el siguiente enlace:

<https://mimic2app.csail.mit.edu/querybuilder/>

Igualmente, se facilita una URL desde la que poder obtener una primera tentativa de estructura DDL (Data Definition Language) de tablas Oracle en las que almacenar la información:

<https://github.com/MIT-LCP/mimic-code/tree/master/buildmimic/oracle>

Siendo un buen punto de partida (para no tener que consultar, columna a columna de cada fichero, cuál es el tipo de dato y longitud máxima presente en los distintos ficheros), es necesario modificar en más de una ocasión esta estructura, bien debido a las modificaciones sufridas por MIMIC a lo largo de sus diferentes versiones, bien debido a diferencias de criterios de almacenaje en tablespaces, particionamiento de tablas voluminosas, creación de índices o existencia de restricciones de integridad (constraints), por citar solo algunos ejemplos.

Por lo tanto, una vez obtenidos los ficheros con la información a explotar, y una vez desarrollados también unos scripts iniciales de creación de las tablas Oracle que albergarán dicha información, se procede a descargar e instalar el gestor de base de datos Oracle.

Capítulo 3

Configuración del software

Para poder disponer de toda la infraestructura tecnológica necesaria para acometer este proyecto, se requiere de unos cuantos componentes software que deben ser aprovisionados, instalados y configurados.

Se opta por trasladar este contenido de descargas, instalaciones y configuraciones a una sección de anexos del documento principal, al tratarse de un punto fundamental para poder proseguir con el proyecto (y que, por tanto, no puede obviarse de esta memoria) pero que, sin embargo, no es de especial interés en relación al objetivo principal del proyecto, que es explotar la información del repositorio proporcionado por Physionet.

En primer lugar, la descarga del software de Oracle requerido para disponer de un gestor de base de datos queda reflejado en el contenido del ‘Anexo A: descarga e instalación de Oracle’, en su primera parte correspondiente a ‘Descarga de Oracle11g’.

Adicionalmente, tras la descarga de los archivos correspondientes al software de gestión de la base de datos, será necesario realizar el proceso de instalación de dicho software. Este paso queda igualmente reflejado en el contenido del ‘Anexo A: descarga e instalación de Oracle’, concretamente en su segunda parte correspondiente a ‘Instalación de Oracle11g’.

Una vez realizada la instalación del gestor de base de datos Oracle, es recomendable disponer de un entorno amigable de administración y consulta de

dicha base de datos, siendo otra opción (menos flexible) el empleo de sentencias mediante línea de comandos. Siendo así, se opta por disponer del software SQL Developer, también proporcionado por el proveedor Oracle. Su descarga queda documentada en el contenido del ‘Anexo B: descarga e instalación de SQL Developer’, concretamente en su primera parte denominada ‘Descarga de SQL Developer’.

Tras haber practicado la descarga del software SQL Developer, se procederá a instalarlo. Para ello, se seguirán los pasos descritos en el mismo anexo llamado ‘Anexo B: descarga e instalación de SQL Developer’, pero en esta ocasión reflejados en el segundo apartado ‘Instalación de SQL Developer’.

De igual manera, una vez descargado e instalado el gestor de la base de datos (Oracle), y también una vez descargado e instalado un software gráfico de administración de dicho gestor de base de datos (SQL Developer), es momento de definir la estructura necesaria en dicho gestor para que éste dé soporte a las pretensiones de explotación de los datos del Warehouse (datos proporcionados por Physionet).

Este proceso de ajuste o parametrización de la base de datos queda descrito en el anexo llamado ‘Anexo C: creación de la base de datos’, donde quedan reflejados los procesos de carga de los ficheros a tablas Oracle, donde queda expuesto el entramado de usuarios necesarios para gestionar los distintos accesos a la base de datos desde uno u otro aplicativo, etc.

Por último, será igualmente necesaria la descarga e instalación del software que desempeñará la función de explotar la información disponible. Este paso queda descrito en el apartado llamado ‘Anexo D: descarga e instalación de MicroStrategy’, describiéndose por separado los pasos de descarga de dicha suite (en el apartado llamado ‘Descarga de MicroStrategy’), los pasos requeridos para la instalación de dicho software (en el apartado ‘Instalación de MicroStrategy’), sus pasos más iniciales de parametrización de cadenas de conexión y orígenes de proyecto (concretamente en el apartado llamado ‘Configuración de MicroStrategy’) o incluso la definición de un proyecto concreto que permita la explotación definitiva de los datos de interés (en el apartado llamado ‘Creación de un proyecto’).

Con la descarga, instalación y parametrización de estas herramientas fundamentales, se da cobertura a la práctica totalidad de la arquitectura de la plataforma:

- una base de datos para albergar datos de la aplicación, metadatos de la herramienta de explotación, estadísticas de la herramienta de explotación o historial de la herramienta de explotación.

- una herramienta de fácil manejo para administrar dicha base de datos sin limitar esta administración a la consola con línea de comandos.

- una suite compuesta por las herramientas necesarias para poder modelizar el comportamiento funcional de los datos y explotar éstos sin necesidad de conocimiento de lenguajes SQL de acceso a datos.

Sin duda, existirán otros componentes necesarios o convenientes en base a las necesidades de despliegue de la aplicación en entornos web (como sería el caso de un servidor de aplicaciones como Apache Tomcat), o de necesidades de movilidad (como sería el caso de MicroStrategy Mobile, disponible tanto para iOS como para Android).

Capítulo 4

Explotación de la información

4.1 Creación de objetos de esquema

Llegado este punto, comienza la tarea de modelización del proyecto en MicroStrategy. Se entiende por modelización el traslado de la información funcional (concerniente al negocio que va a analizarse, en este caso las estancias hospitalarias de pacientes) a un repositorio lógico que proporcione a la herramienta de explotación la inteligencia suficiente para conocer cómo resolver cada solicitud de información recibida. Por ejemplo, se trata de indicar de qué tablas del warehouse (así como de qué columnas) podrá llegar a obtener una determinada información, bien sea un atributo (algo que califica un hecho, una propiedad del hecho que permitirá clasificarlo, filtrarlo, consolidarlo o desglosarlo) o bien sea un hecho en sí mismo (un dato que se conoce, que generalmente cuantifica y que puede ser calificado mediante uno o varios atributos).

La denominación de objetos de esquema y su diferencia principal con respecto a los objetos públicos es su directa dependencia de la estructura física del esquema de base de datos del que se esté recuperando la información. La simple red denominación de una tabla, la aparición o desaparición de éstas, el cambio de una columna de dichas tablas (bien aparición, desaparición, nombre, formato físico, necesidad de un diferente tratamiento de su contenido al ser consultado...) forzará la adaptación de estos pilares fundamentales del proyecto para que todo siga funcionando de la

manera esperada. El resto de elementos de la aplicación recibirán el nombre de objetos públicos debido a que se asientan sobre objetos de esquema previamente definidos en la propia herramienta y, por tanto, se desvinculan (con esta capa de abstracción intermedia) de los detalles más físicos del esquema de base de datos.

Por lo tanto, se considerarán objetos de esquema las tablas, los hechos, los atributos, las jerarquías... Buena parte de ellos serán utilizados a lo largo de este proyecto, por lo que se procede a detallar la forma de definirlos y utilizarlos apropiadamente.

4.1.1 Tablas

Conviene revisar, en primer lugar, que se dispone de todas y cada una de las tablas esperadas (las presentes en el warehouse de Oracle, es decir, las asociadas a los ficheros proporcionados por Physionet) dentro del catálogo de warehouse de la herramienta. Para ello, a través de la herramienta Developer, una vez conectados al proyecto ‘PROYECTO_MIMIC3’ existente en el origen de proyecto ‘MIMIC3_ORIGEN_PROYECTO_3LEVELS’, bastará con acudir al menú ‘Esquema – Catálogo de Warehouse’ mostrado en la Figura 4.1.

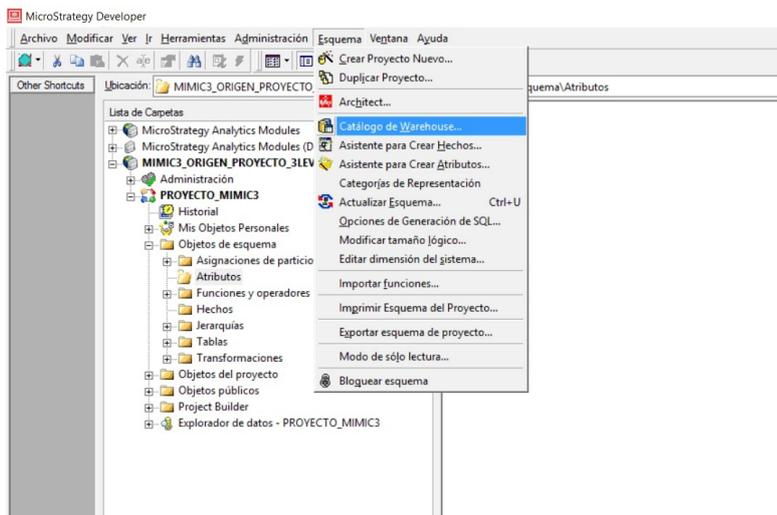


Figura 4.1: Menú de Catálogo de Warehouse.

Se mostrará un listado de todas las tablas presentes en la cadena de conexión definida para el warehouse (suma de los paneles de ambos laterales), así como

cuáles de ellas han sido finalmente seleccionadas para su empleo en este proyecto (únicamente las del panel de la derecha), como muestra la Figura 4.2.

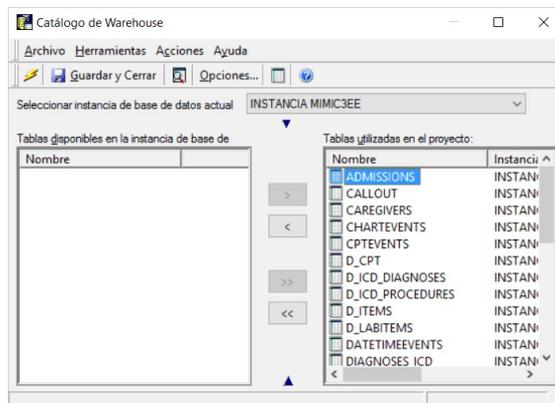


Figura 4.2: Tablas disponibles en el Catálogo de Warehouse.

Si se desea, existen funcionalidades para conocer un conteo de las filas de una determinada tabla, previsualizar un resumen de unas cuantas filas de la tabla, o consultar simplemente la estructura que MicroStrategy ha importado para cada una de sus columnas. Estas funcionalidades quedan agrupadas bajo un menú contextual asociado a las distintas tablas, mostrado en la Figura 4.3.

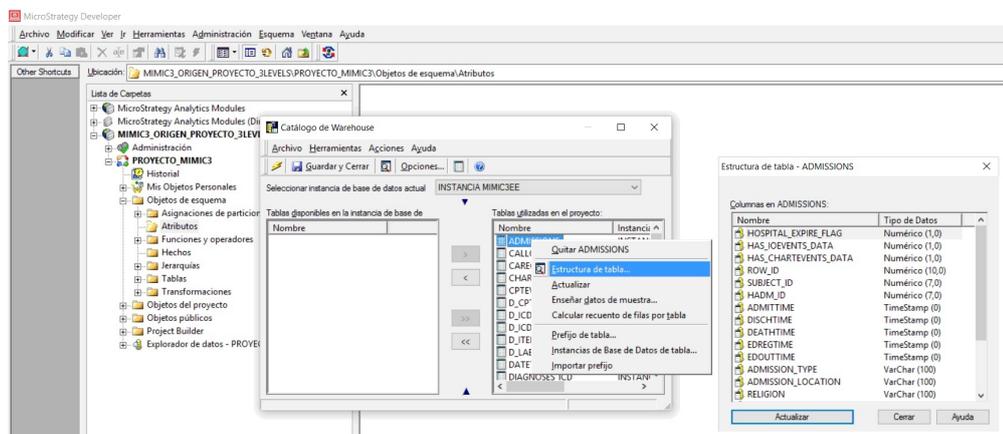


Figura 4.3: Funcionalidades adicionales del Catálogo de Warehouse.

En el caso que nos concierne, todas y cada una de las tablas proporcionadas por Physionet serán añadidas al catálogo de Warehouse del proyecto (tal y como refleja la figura correspondiente al catálogo de Warehouse), de modo que se pueda obtener información de cualquiera de ellas (al tratarse, prácticamente en su totalidad, de

tablas de hechos, con una información de mayor o menor interés de estudio). Esta sería la parte correspondiente a la catalogación de tablas físicas.

Estas tablas serán utilizadas de manera directa por la herramienta a la hora de obtener de ellas ciertos hechos (sobre los que calcular indicadores) o agrupar la información por alguno de los atributos que la utilizan como base durante la composición dinámica de la sentencia SQL de recuperación de un determinado informe. Puede observarse en el ejemplo de la Figura 4.4 en el que se utiliza la tabla ‘PATIENTS’ para obtener de ella ciertos indicadores de una parte de un informe relacionado con pacientes.

```

select  a11.GENDER GENDER,
        count(distinct (Case when a11.DOD is not null then a11.SUBJECT_ID else NULL end)) WJXBFS1,
        max((Case when a11.DOD is not null then 1 else 0 end)) GODWFLAG1_1,
        count(distinct a11.SUBJECT_ID) WJXBFS2
from    PATIENTS      a11
group by a11.GENDER

```

Figura 4.4: Sentencia SQL contra la tabla física ‘PATIENTS’.

Pero, adicionalmente, es posible también definir tablas lógicas en MicroStrategy, es decir, objetos de MicroStrategy que no existen como tal (como tablas, como vistas o como sinónimos hacia ninguno de los anteriores) en la base de datos Oracle. Sería el equivalente a realizar una vista en Oracle (que, en tiempo de consulta, Oracle reemplazaría internamente por el código SQL que proporciona los resultados de la vista), pero sin necesidad de crear un nuevo objeto en Oracle, definiendo directamente en MicroStrategy la sintaxis SQL que será reemplazada en el lugar de la vista en el momento de ir a ser empleada durante la generación dinámica de la sentencia SQL de resolución de un informe. Esta opción es muy práctica cuando no se desea crear un objeto adicional en Oracle, como sería el caso de una tabla (que, además, sería necesario mantener apropiadamente refrescada con la información deseada) o una vista (materializada o no).

En este caso, poniendo como ejemplo la creación de una tabla lógica que sirva de maestro o tabla de look-up de los distintos meses (entendiéndose mes como la unión de año y mes) en los que existen datos de admisiones en la tabla de hechos ‘ADMISSIONS’, se crearía una tabla lógica que, directamente, obtuviese los distintos meses presentes en dicha tabla (de manera que, sin necesidad de definir

un objeto nuevo en Oracle y sin necesidad de mantenerlo constantemente íntegro con la información de la tabla de admisiones, se encuentre perfectamente disponible en MicroStrategy como una tabla más y siempre actualizada). La definición de esta tabla lógica a la que llamaremos ‘LU_ADMITTIME_YYYYMM’ se realiza con la sentencia SQL (similar a la de constitución de una vista en Oracle) y la estructura de columnas (en cuanto a tipo de dato, nombre, longitud, etc.) mostradas en la Figura 4.5.

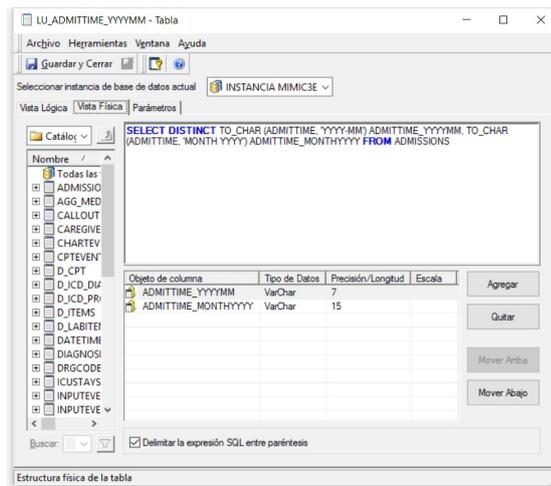


Figura 4.5: Definición de la tabla lógica ‘LU_ADMITTIME_YYYYMM’.

Por tanto, esta tabla es válida como tabla de look-up de un futuro atributo que no represente directamente el momento en el que se practica la admisión (con precisión de día y hora), sino una consolidación del mismo, un nivel inferior (menor) de detalle: el año y mes al que corresponde dicha fecha de admisión (siendo, por tanto, una relación ‘uno a muchos’ la existente entre esta fecha de menor detalle con la fecha detallada en sí de admisión).

La única limitación que puede presentar el uso de estas tablas lógicas es que, al no estar calculadas en disco de la base de datos (sino, realmente, consistir en una subconsultas a calcular en tiempo de consulta del informe), pueden aportar un peor rendimiento a la resolución de informes.

La sintaxis que presentaría la sentencia SQL de una sección de un informe en el que interviniese, por ejemplo, ese futuro atributo que permitiera llegar al detalle del momento de la admisión a un nivel de mes y año, y que dispondría como tabla de

look-up de esta tabla lógica recién creada llamada ‘LU_ADMITTIME_YYYYMM’, se refleja en la Figura 4.6.

```

select pa18.ADMITTIME_YYYYMM ADMITTIME_YYYYMM,
a19.ADMITTIME_YYYYMM ADMITTIME_MONTHYYYY,
pa18.INSURANCE INSURANCE,
pa18.WJXBFS1 WJXBFS1,
pa18.WJXBFS2 WJXBFS2,
pa18.WJXBFS3 WJXBFS3,
pa18.WJXBFS4 WJXBFS4,
pa18.WJXBFS5 WJXBFS5,
pa18.WJXBFS6 WJXBFS6,
pa18.WJXBFS7 WJXBFS7,
pa18.WJXBFS8 WJXBFS8
from T2AKUVPIZMD000pa18
join (SELECT DISTINCT TO_CHAR (ADMITTIME, 'YYYY-MM') ADMITTIME_YYYYMM, TO_CHAR (ADMITTIME, 'MONTH YYYY') ADMITTIME_MONTHYYYY FROM ADMISSIONS) a19
on (pa18.ADMITTIME_YYYYMM = a19.ADMITTIME_YYYYMM)

```

Figura 4.6: Sentencia SQL contra la tabla lógica ‘LU_ADMITTIME_YYYYMM’.

Puede apreciarse cómo, la herramienta de explotación, la utilizará como si de una tabla más se tratase, sólo que utilizando la sentencia SQL indicada en su definición en lugar de utilizar el nombre directo de una tabla física del Warehouse.

La alternativa a esta posible merma de rendimiento por el uso de estas subqueries asociadas a tablas lógicas puede consistir en la creación de tablas de look-up físicas en Oracle que sí dispongan de esta información almacenada, siendo los inconvenientes, en este caso, la necesidad de crear este objeto en Oracle (en lugar de en la herramienta de explotación) y la necesidad de gestionar apropiadamente su carga para que nunca quede desfasada frente a la tabla a la que pretende calificar.

Esta nueva alternativa es la que se adopta, por ejemplo, a la hora de realizar una tabla de look-up que también contemple los diferentes meses, pero en este caso no de las admisiones (tabla de hechos ‘ADMISSIONS’), sino de las tomas de medidas de signo vital (tabla de hechos ‘CHARTEVENTS’). El motivo por el que, en esta ocasión, de opta por una tabla física llamada ‘LU_CHARTTIME_YYYYMM’ (con los datos ya almacenados en la base de datos) y no por una tabla lógica de MicroStrategy (o subconsultas en tiempo de obtención de informe contra la propia tabla de medidas de signo vital) es la diferencia en volumetría: mientras que la tabla ‘ADMISSIONS’ presenta 58.976 filas, la tabla ‘CHARTEVENTS’ presenta 263.201.340 filas (siendo impensable abordarla constantemente en tiempo de resolución de informes, por más que su estructura interna en Oracle haya sido gestionada mediante particiones, índices locales, etc.).

Puesto que, aun siendo tablas de look-up, no dejarían de ser tablas físicas,

residentes en el Warehouse y catalogadas en el catálogo de Warehouse de MicroStrategy al igual que las tablas de hechos proporcionadas por Physionet, el empleo por parte de la herramienta de éstas sería directa, tal y como muestra en la Figura 4.7 la sintaxis SQL redactada en una sección de un informe que trata de describir las medidas de signo vital desglosadas por fecha en formato mes y año, es decir, empleando la mencionada tabla ‘LU_CHARTTIME_YYYYMM’.

```

select  pa13.ADMISSION_LOCATION ADMISSION_LOCATION,
        pa13.INSURANCE INSURANCE,
        pa13.CHARTTIME_YYYYMM CHARTTIME_YYYYMM,
        a14.CHARTTIME_MONTHYYYY CHARTTIME_MONTHYYYY,
        pa13.WJXBFS1 WJXBFS1,
        pa13.WJXBFS2 WJXBFS2,
        pa13.WJXBFS3 WJXBFS3
from    TZPA88SUQMD000 pa13
join    LU_CHARTTIME_YYYYMM a14
on      (pa13.CHARTTIME_YYYYMM = a14.CHARTTIME_YYYYMM)

```

Figura 4.7: Sentencia SQL contra la tabla de look-up física ‘LU_CHARTTIME_YYYYMM’.

Por último, es de vital importancia generar tablas físicas en Oracle que aporten datos precalculados a un nivel de agregación menor (menor detalle o desglose), de manera que permitan a la herramienta de explotación obtener datos a partir de ellas sin la necesidad de escalar, jerárquicamente, muchos niveles de varias tablas relacionamente unidas. Estas tablas son llamadas tablas agregadas. Pueden ser simples tablas físicas en Oracle (que proporcionen redundancia sobre las tablas de hechos inicialmente proporcionadas por Physionet), siendo calculadas en tiempo de ventana batch (carga nocturna) a diferentes niveles de dimensionalidad o agregación, o puede tratarse directamente de vistas materializadas definidas en Oracle que, de manera constante (adecuando la opción de ‘refresh’ de éstas) se encuentren perfectamente sincronizadas con las tablas de hechos sobre las que aportan dicha redundancia.

Estas tablas agregadas son fundamentales en entornos datawarehouse, dado que permiten reducir a niveles ínfimos los tiempos de resolución de informes al disponer, de manera batch, de precálculos de ciertas agregaciones que serán aprovechadas por la herramienta de explotación.

Por ejemplo, en caso de querer disponer sobre la tabla de medidas de signo vital (tabla de hechos ‘CHARTEVENTS’) de una medida agregada (es decir, la suma) del valor medido, pero siendo éste distribuido (dimensionado, agrupado) por

etnia del paciente (no disponible en la propia tabla de hechos ‘CHARTEVENTS’, sino en la tabla de admisiones o tabla de hechos ‘ADMISSIONS’), por año y mes de la toma de medida del signo vital (lo que implica realizar un escalado en la dimensión temporal para pasar del momento exacto en el que se practicó la toma a simplemente el mes y año al que dicho momento corresponde, como hemos visto en el ejemplo de la tabla de look-up física llamada ‘LU_CHARTTIME_YYYYMM’) y por tipo de signo vital medido (estando presente, éste sí, en la propia tabla de hechos ‘ADMISSIONS’), conviene no dejar que la herramienta de explotación genere dicha consulta SQL mezclando todas las fuentes de información necesarias para obtener los datos al nivel de detalle requerido, sino que parece aconsejable que, en tiempo de ventana batch, una tabla precalcule dicha información a ese nivel de detalle y la ponga a disposición de la herramienta de explotación). Esta tabla, que pasará a denominarse ‘AGG_MEDIDA_ETNIA_DATO’, es creada, por tanto, a partir de la tabla ‘CHARTEVENTS’ (y otras tablas de las que obtener el resto de información) mediante la ejecución de la sentencia DDL mostrada en la Figura 4.8.

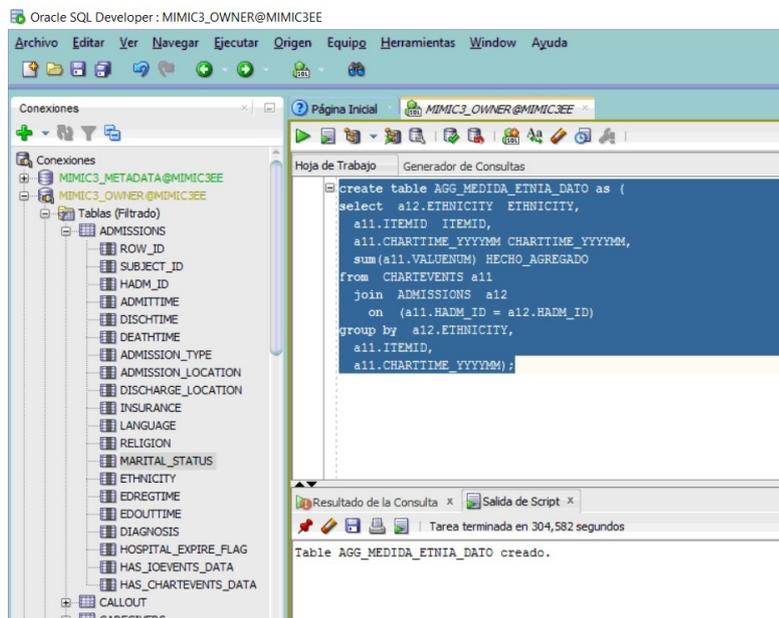


Figura 4.8: Creación de la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’.

Y, por tanto, dispondrá ya de un hecho calculado de manera agregada (pilar fundamental de la tecnología datawarehouse). En base a la dimensionalidad

requerida en un informe, la herramienta de explotación optará por acudir a esta tabla o, directamente (siempre que no disponga de la posibilidad de acudir a la tabla agregada), a la propia tabla de hechos. Esta selección automática de a qué tabla acudir para obtener un hecho será explicada en el propio apartado de Hechos.

Por ejemplo, en la Figura 4.9 puede verse cómo la herramienta de explotación, durante la generación dinámica de la sentencia SQL de elaboración de un informe, dispondría de esta tabla agregada siempre que un hecho estuviese definido sobre ella y percibiese que el nivel de dimensionalidad del informe no desborda la granularidad a la que dicha tabla agregada fue calculada.

```

select  a11.ITEMID ITEMID,
        a12.LABEL LABEL,
        a12.ABBREVIATION ABBREVIATION,
        a11.ETHNICITY ETHNICITY,
        sum(a11.HECHO_AGREGADO) WJXBFS1
from    AGG_MEDIDA_ETNIA_DATO      a11
        join D_ITEMSa12
        on  (a11.ITEMID = a12.ITEMID)
where   a11.CHARTTIME_YYYYMM in (210010, 210011, 210012)
group by a11.ITEMID,
         a12.LABEL,
         a12.ABBREVIATION,
         a11.ETHNICITY

```

Figura 4.9: Sentencia SQL contra la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’.

Esto sustituye el tener que atacar directamente la tabla ‘CHARTEVENTS’ y la tabla ‘ADMISSIONS’, intervinientes en la agregación previa de la tabla ‘AGG_MEDIDA_ETNIA_DATO’ (situación que se daría en caso de no existir dicha tabla agregada y, por tanto, disponer el hecho sólo de estas dos tablas de Physionet como orígenes de su información), reflejado en la Figura 4.10.

```

select  a11.ITEMID ITEMID,
        a13.LABEL LABEL,
        a13.ABBREVIATION ABBREVIATION,
        a12.ETHNICITY ETHNICITY,
        sum(a11.VALUENUM) WJXBFS1
from    CHARTEVENTS      a11
        join ADMISSIONS  a12
        on  (a11.HADM_ID = a12.HADM_ID)
        join D_ITEMSa13
        on  (a11.ITEMID = a13.ITEMID)
where   a11.CHARTTIME_YYYYMM in (210010, 210011, 210012)
group by a11.ITEMID,
         a13.LABEL,
         a13.ABBREVIATION,
         a12.ETHNICITY

```

Figura 4.10: Sentencia SQL contra la tabla de hechos ‘CHARTEVENTS’.

Obviamente, los tiempos de respuesta se verían seriamente afectados.

4.1.2 Atributos

Una vez comprobado el correcto estado en general del catálogo de warehouse, comienza la etapa de definir los atributos (podría también comenzarse por la definición de los hechos, el orden es indistinto).

Un atributo será toda aquella calificación que pueda hacerse de un determinado hecho: propiedades por las que pueda identificarse, filtrarse o agruparse un hecho. Por ejemplo, el paciente asociado a un determinado tratamiento, el hospital en el que un paciente se encuentra hospitalizado, el médico que se está encargando de revisar su evolución... Y, a su vez, estos atributos pueden tener igualmente otros atributos ‘padres’ que les sirvan de calificación (por ejemplo, la edad correspondiente a un determinado paciente, o el tipo de sanitario asociado a un sanitario concreto, etc.).

Por su parte, un atributo se compondrá de una o más componentes (llamadas representaciones de atributo). Una de las componentes será siempre el identificador único (aquello que identifica de manera única un posible elemento de ese atributo de entre sus posibles valores), y este identificador podrá estar compuesto por una o más columnas (dependerá de cómo se organice esa información en el modelo de datos físico). Estas columnas podrán estar localizadas en una o en más tablas del catálogo de Warehouse, incluso podrán localizarse con diferente nomenclatura dependiendo de cada una de estas tablas, o es posible que provengan de una transformación realizada sobre alguna columna... Adicionalmente, un atributo podrá disponer de otras componentes que sean no identificativas, sino descriptivas, es decir, asociadas de manera única a cada identificador. Por último, todo atributo debe tener asociada una tabla maestra o de look-up, que será aquella tabla en la que se encuentran recogidos todos los posibles valores existentes en el warehouse para ese atributo y, además, con una única presencia por cada identificador único del atributo.

Una vez descrito de manera general el significado de un atributo, sus distintas representaciones de atributo, su tabla de look-up..., se comienza con la definición de éstos. Como ejemplo veremos la definición del atributo ‘Paciente’, que podría

tratarse del centro principal de análisis de este proyecto (sin un Paciente, no existe dato alguno que estudiar).

Un paciente será un atributo que vendrá identificado por un identificador único compuesto por una sola columna llamada ‘SUBJECT_ID’, y con una tabla maestra o de look-up asociada llamada ‘PATIENTS’. Su disponibilidad estará ligada, además de a la propia tabla de look-up, a muchas otras tablas del warehouse, dado que se trata de un atributo fundamental en este proyecto.

Para definirlo, dentro de la carpeta Atributos, se accederá al menú contextual de ‘Nuevo – Atributo’ mostrado en la Figura 4.11.

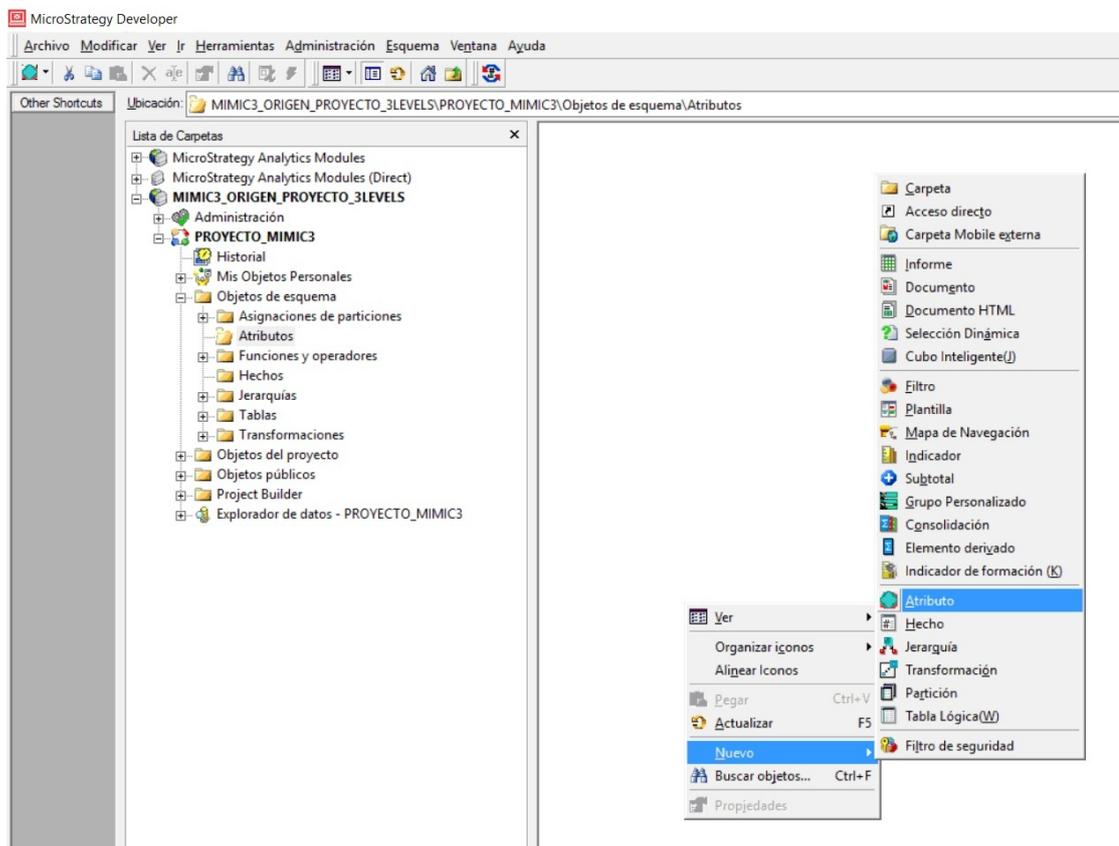


Figura 4.11: Menú contextual de ‘Nuevo – Atributo’.

Se accede a un menú de definición de atributo, que en primer lugar solicitará indicar una columna de una tabla como una de las representaciones del atributo. Suele ser recomendable comenzar con la representación de atributo ligada al identificador único del atributo (lo mejor, comenzar escogiendo la columna ‘SUBJECT_ID’ correspondiente a su identificador único, localizada además en la

tabla de look-up ‘PATIENTS’), como se muestra en la Figura 4.12.

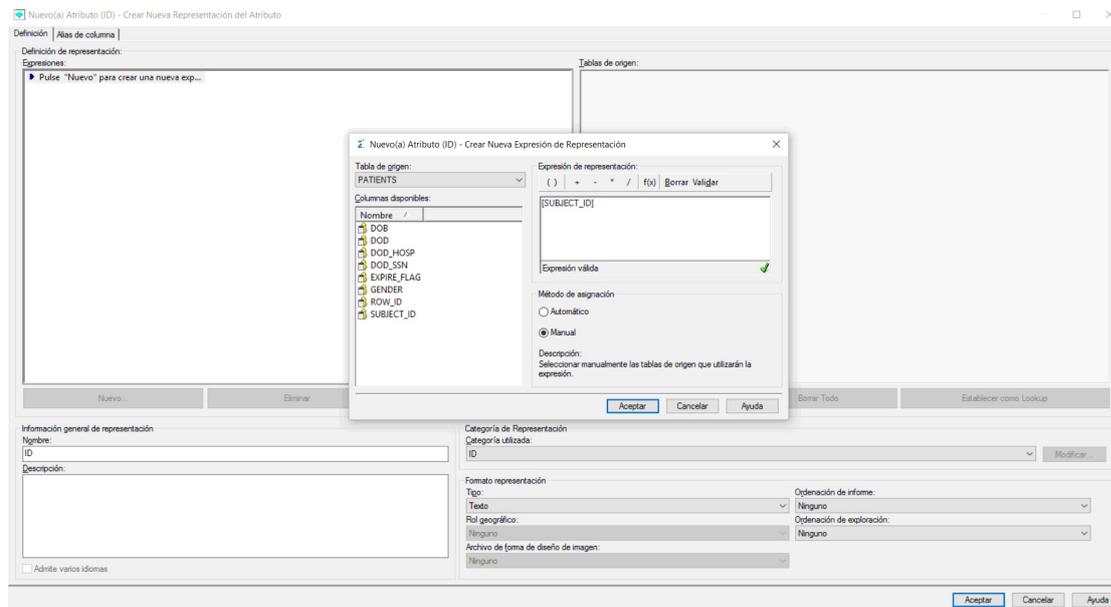


Figura 4.12: Definición del atributo ‘Paciente’ sobre la tabla ‘PATIENTS’.

Se optará por elegir la asignación manual de tablas frente a la asignación automática. De esta manera, será el usuario de MicroStrategy (el arquitecto del proyecto) quien decida qué otras tablas con una columna de igual nombre se asocian a este atributo, dado que la opción automática asociaría todas aquellas tablas que presentasen una columna de igual nombre. Esta base de datos se encuentra correctamente normalizada y podría no ser un problema optar por la asignación automática pero, en ocasiones, una columna con igual nombre en dos tablas no tiene por qué representar lo mismo (se da en este proyecto con varias columnas, aunque son pocas) o, por otra parte, puede quedar fuera una columna de otra tabla sí relacionada pero con un nombre diferente (que la asignación automática, por tanto, no detectaría).

Se informará un nombre de la representación de atributo asociada a la columna ‘SUBJECT_ID’, se seleccionarán todas las demás tablas del warehouse donde dicha información se encuentra presente con este mismo significado, se elegirá una tabla de look-up (‘PATIENTS’ en este caso), se pueden elegir opciones de ordenación de los elementos del atributo tanto en visualización de informes como en consulta de sus valores en el maestro de pacientes, etc., como refleja la Figura 4.13.

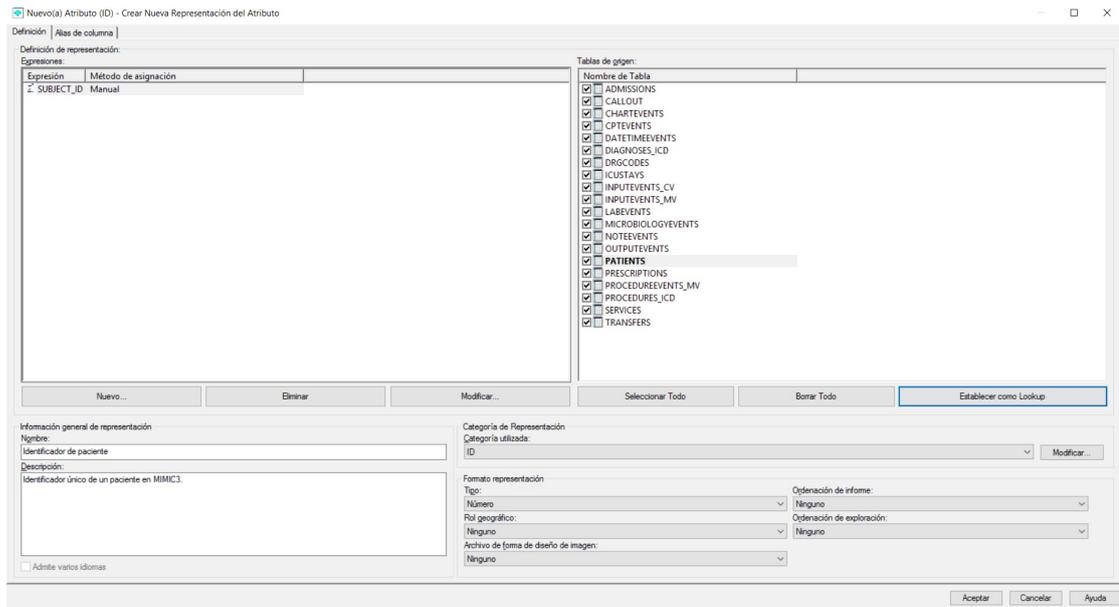


Figura 4.13: Definición completa del identificador único del atributo ‘Paciente’.

A continuación, se completará este atributo con el resto de representaciones que, aun siendo ya distintas de la identificativa, pueden ser empleadas para filtrar una determinada consulta, o para mostrar en la rejilla de un determinado informe. Se muestra en la Figura 4.14, como ejemplo, la manera en la que se definiría la representación de atributo asociada a la fecha de nacimiento del paciente (que, en esta ocasión, sólo se encuentra presente en la tabla de look-up propia de pacientes llamada ‘PATIENTS’, pero podría encontrarse en más tablas en caso de disponer de un modelo de datos físico no normalizado, sino algo más desnormalizado).

Por último, la Figura 4.15 muestra una vista general del atributo una vez que éste ha sido finalmente definido para todas las representaciones de atributo de interés.

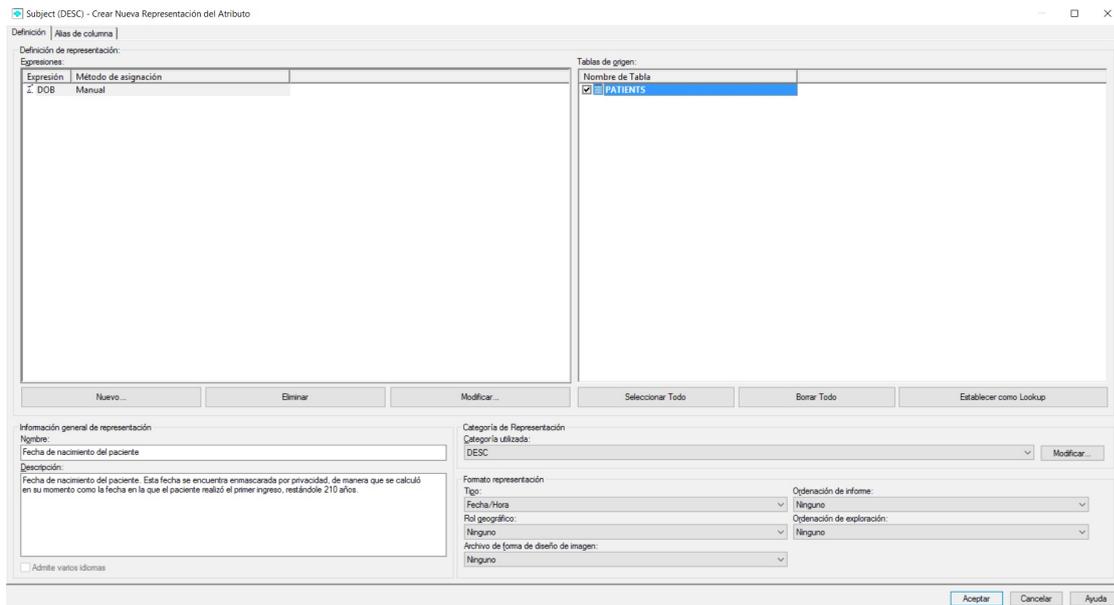


Figura 4.14: Representaciones adicionales no únicas del atributo 'Paciente'.

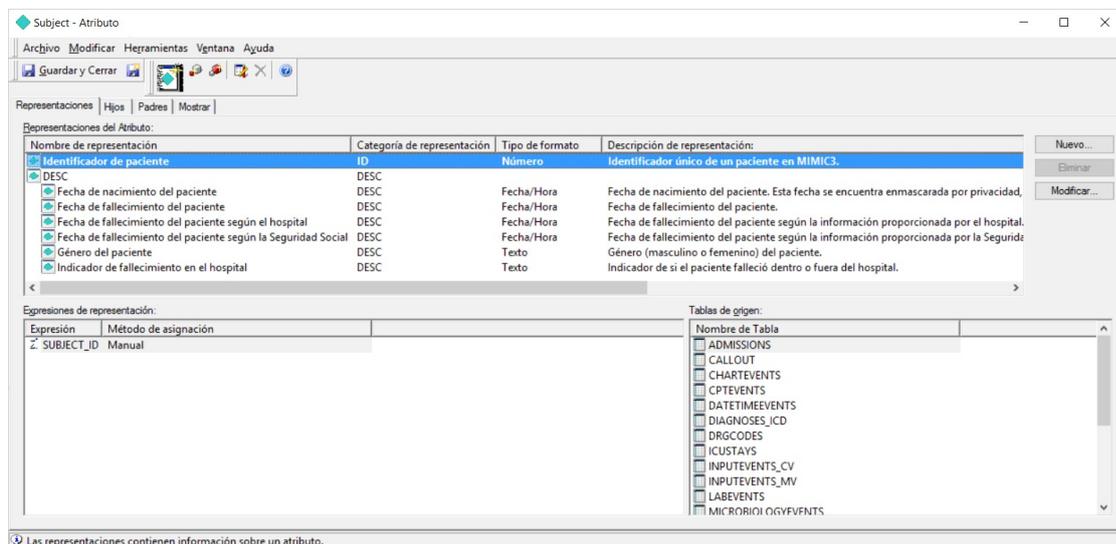


Figura 4.15: Definición completa del atributo 'Paciente'.

Una vez definido uno o varios atributos, conviene realizar la tarea de actualizar esquema, disponible en el menú presente en la Figura 4.16.

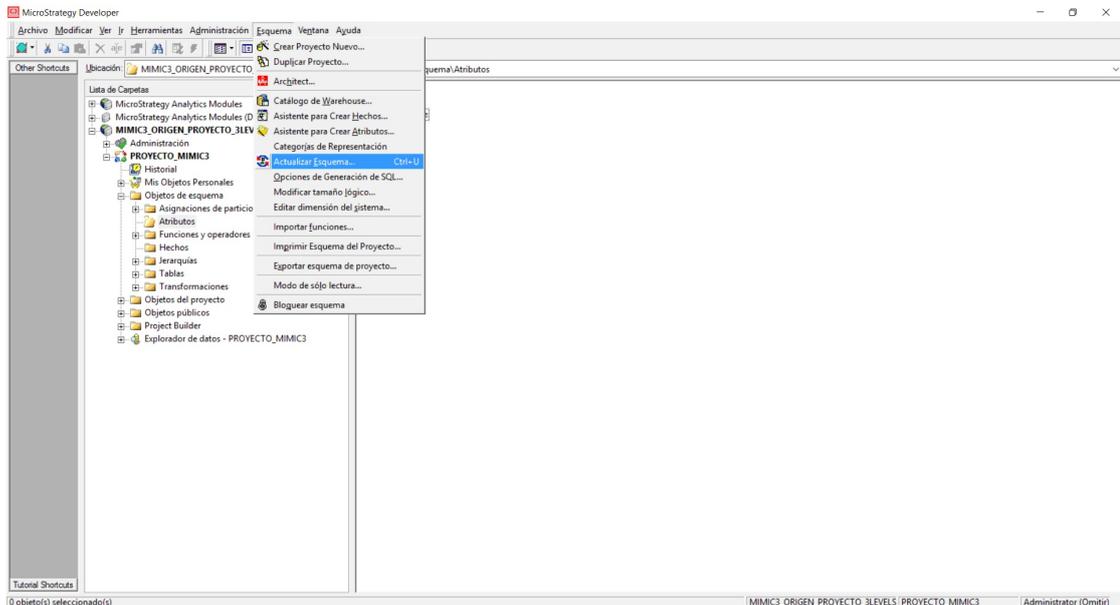


Figura 4.16: Acceso a la funcionalidad de ‘Actualizar Esquema’.

El sentido de esta tarea consiste en aportar un determinado ‘peso lógico’ a cada tabla del catálogo de warehouse (algo así como una catalogación de cómo de voluminosa es esa tabla y, por tanto, cómo de costosa será su consulta, basándose en cuántos atributos de más bajo nivel jerárquico se apoyan en dicha tabla o, dicho de otra manera, analizando cuántos atributos compondrán la clave lógica de cada tabla), además de establecer una integridad entre los distintos metadatos recién definidos (atributos hijos y padres, por ejemplo).

En la Figura 4.17 se muestra el formulario siguiente, en el que es posible parametrizar qué funciones sí y cuáles no se desean efectuar.

Es posible relacionar un atributo con otro (dado que se ha comentado con anterioridad que un atributo como podría ser el ‘Paciente’ podría estar a su vez calificado por otro atributo como podría ser su edad...). Por tanto, en el momento en el que ya se dispone de una buena serie de atributos definidos en la aplicación, y pudiendo ser éstos relacionados, se establecerán las relaciones padre-hijo entre éstos. Si, de nuevo, se emplea el atributo ‘Paciente’ como ejemplo, obtendremos la posibilidad de definir padres suyos (en general, suelen darse relaciones muchos a

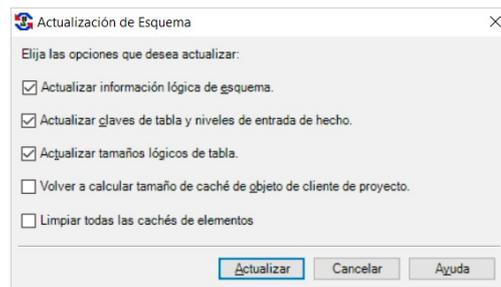


Figura 4.17: Opciones de parametrización de la funcionalidad de ‘Actualizar Esquema’.

uno entre hijos y padre, empleándose en general la tabla de look-up del atributo hijo como tabla de relación, dado que será la tabla que disponga del identificador único, como propiedad adicional, del atributo padre), tal y como muestra la Figura 4.18.

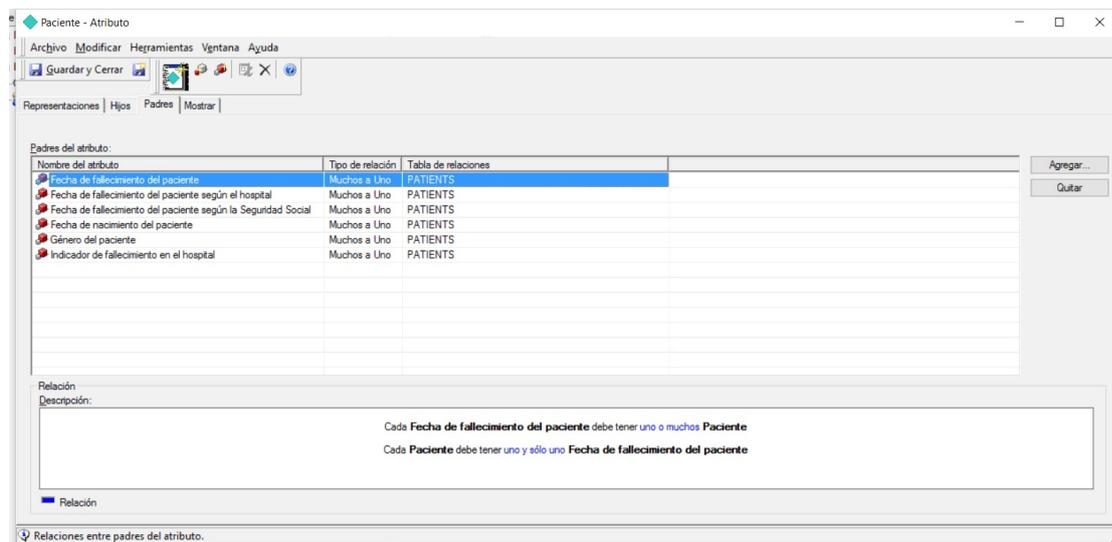


Figura 4.18: Definición de padres del atributo ‘Paciente’.

Podremos definir también hijos suyos (siendo de nuevo, en general, una relación uno a muchos la relación existente entre padre e hijos, y empleándose como tabla de relación la tabla de look-up del atributo hijo, al ser ésta la que contenga como columna no identificativa el identificador único del atributo padre), como ilustra la Figura 4.19.

Esto permitirá a la herramienta conocer las relaciones jerárquicas existentes entre atributos definidos en el esquema, de manera que un mismo hecho pueda ser

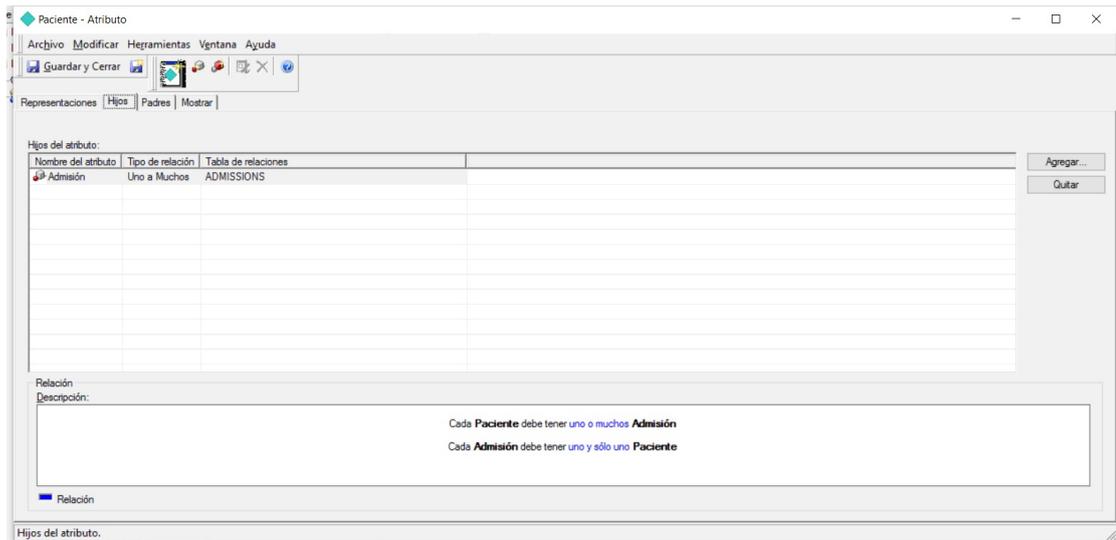


Figura 4.19: Definición de hijos del atributo 'Paciente'.

obtenido frente a un determinado atributo, por mucho que ambos no compartan una misma tabla de obtención de los datos, siempre que ese hecho sí que esté posicionado en alguna de las tablas en las que alguno de los atributos hijos del atributo deseado esté, igualmente, posicionado.

Igualmente, el conocimiento de qué atributos apoyan sus representaciones de columna identificativas (identificador único de éstos) en cada tabla, así como sus relaciones jerárquicas entre sí, permitirán a la herramienta saber qué columnas totales componen la clave lógica de una determinada tabla, qué peso lógico se le puede asociar con el fin de otorgarle una mayor o menor lentitud de respuesta (inversamente proporcional a su volumetría), etc.

La aplicación, en base a qué atributos de más bajo nivel (sin hijos) tengan apoyada su clave identificativa en una determinada tabla, interpretará que se trata de componentes físicos de la clave primaria de la tabla en Oracle. Por tanto, dada una mayor cifra de columnas que conformen la clave primaria de la tabla, se entiende que mayor es el grado de detalle que dicha tabla contiene y, por tanto, se entiende que mayor volumen de información posee, atribuyéndole un mayor peso lógico para su posterior toma de decisiones.

Por ejemplo, al consultar la tabla 'PATIENTS' obtendremos la situación mostrada en la Figura 4.20.

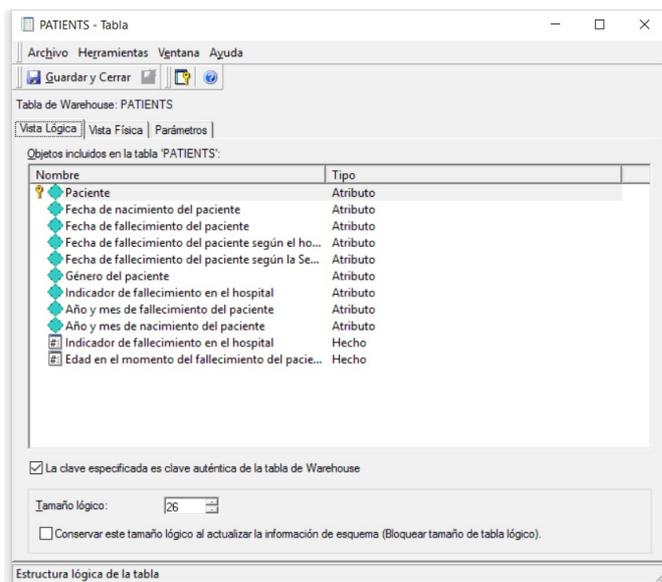


Figura 4.20: Vista lógica de la tabla 'PATIENTS'.

Este formulario indica la lista de atributos que están posicionados sobre esta tabla (es decir, alguna de sus diversas representaciones de atributo señala a alguna de las columnas de esta tabla), cuáles son clave lógica (identificados por un icono de una llave) o el peso lógico que MicroStrategy ha determinado para esta tabla.

4.1.3 Hechos

Una vez ha sido descrito el significado de un atributo, es el momento de describir el significado de un hecho (aunque ya ha sido indicado que el orden de prioridad a la hora de definir atributos o hechos es indiferente).

Un hecho consistirá en toda aquella información residente en el Warehouse con un sentido de medición, con una cierta lógica de agregación o suma, generalmente cuantificable. Por lo tanto, será aquella información que, una vez determinado el grado de agrupación sobre el que desea determinarse, podrá ser sumado, o podrán ser contadas sus componentes, etc.

El motivo de que se denomine 'hecho' es que, realmente, es algo invariable, es algo ya existente previamente a la elaboración del datawarehouse y que, además, no conviene alterar. Es el dato origen que, en caso de desear ser modificado, someterá a la herramienta de explotación (o a los procesos de carga de otras tablas más

elaboradas a partir de las primigenias) a la necesidad de una posterior modificación, o a definir un indicador que realice una determinada operación sobre dicho hecho, o a aplicar un determinado filtro para descartar valores no deseados del hecho.

El principal beneficio de no alterar el hecho tal y como provenga será la directa carga en el datawarehouse de esta información desde el sistema fuente, así como el que, un cambio de criterio en la definición de una métrica, sólo implique el modificar la sintaxis de la fórmula o de los filtros que se aplican sobre ese hecho cargado directamente en la base de datos para obtener la métrica, pero no implicará la actualización del valor del hecho en todos los registros de las tablas afectadas.

Pero no deben hacerse prejuicios sobre qué debe cumplirse para que una determinada información sea un hecho. Un hecho no tiene por qué ser numérico, ni siquiera agregable. Una representación de atributo no identificativa (sino descriptiva) podría verse, perfectamente, como un hecho (sobre el que se aplicase, posteriormente, una función de agregación de Oracle ‘MAX’ en lugar de la clásica función ‘SUM’, al tratarse de un hecho no cuantificable, ni agregable). Parecido, pero de manera inversa, sucede con las métricas o indicadores: no necesariamente deben acudir a operar sobre un hecho, sino que también pueden operar sobre un atributo y no necesariamente con lógica no aritmética, sino realizando operaciones más bien asociadas .^a primera vista.^a un hecho, como podría ser un conteo mediante la función de agregación de Oracle ‘COUNT’ y, por qué no, parametrizada con la cláusula ‘DISTINCT’. Visto de esta manera, la diferenciación entre hecho y atributo puede verse como algo dificultoso (y asustar al recién iniciado) o algo que ofrece flexibilidad (y agrada a otros usuarios). Será en función de los objetivos de los informes que deseen elaborarse cuando se pueda clarificar mejor si una determinada información residente en el Warehouse debe tomarse como un atributo, como un hecho o, por qué no, como ambas (algo perfectamente posible).

Continuando con el proyecto Mimic, se accede al menú de configuración de hechos con el fin de dar un nuevo alta. De nuevo, se empleará el ejemplo de la tabla ‘PATIENTS’, maestra de pacientes del Warehouse, y se tratará de definir un hecho que disponga de la información relativa a la edad del paciente en su momento de fallecimiento.

El aspecto del formulario de mantenimiento de un hecho será el mostrado en la Figura 4.21.

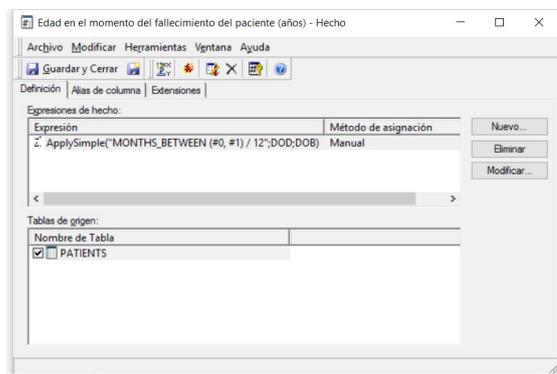


Figura 4.21: Definición de un hecho sobre la tabla ‘PATIENTS’.

En la sección de ‘Expresiones de hecho’, se elegirá una o más formas de recuperar la información correspondiente a este hecho de la base de datos. En este caso de ejemplo, sólo se presenta una expresión, pero podría ser necesario definir más de una expresión con el fin de abarcar todos los posibles orígenes de este dato y en todos sus diferentes aspectos (es decir, que el hecho se encontrase desplegado o reproducido en más de una columna de la base de datos, como podría tratarse del caso de tablas agregadas de apoyo, y además con diferente criterio de almacenamiento). Esta redundancia de orígenes para un mismo hecho es una realidad deseada en tecnología de DataWarehouse, dado que uno de los principales pilares en los que se basa es en poder ofrecer unos cálculos on-line muy veloces gracias a la tarea (más lenta pero invisible) realizada en tiempo batch a la hora de obtener tablas agregadas, que almacenarán esos hechos precalculados a determinados niveles de dimensionalidad o agregación (ya visto anteriormente).

Haciendo un pequeño inciso, en caso de haber existido más de una expresión de hecho, cada una de ellas con sus correspondientes tablas de orígenes (como sería el caso de un hecho que utilizase tanto la tabla de hechos ‘CHARTEVENTS’, procedente de Physionet y con un elevado volumen, como la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’, calculada en tiempo batch para simplificar cálculos en tiempo de explotación), el aspecto de este formulario variaría, al necesitar relacionar diferentes tratamientos de columnas con unas y otras tablas.

El aspecto del formulario para la expresión del hecho que emplea la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’ será el mostrado en la Figura 4.22.

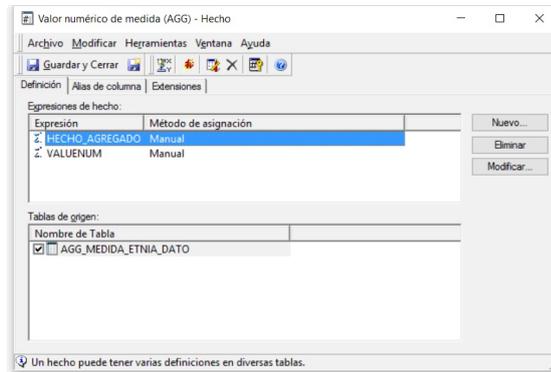


Figura 4.22: Expresión del hecho sobre la tabla ‘AGG_MEDIDA_ETNIA_DATO’.

Por su parte, la presencia de este formulario para la expresión del hecho que emplea la tabla de hechos original ‘CHARTEVENTS’ será la dispuesta en la Figura 4.23.

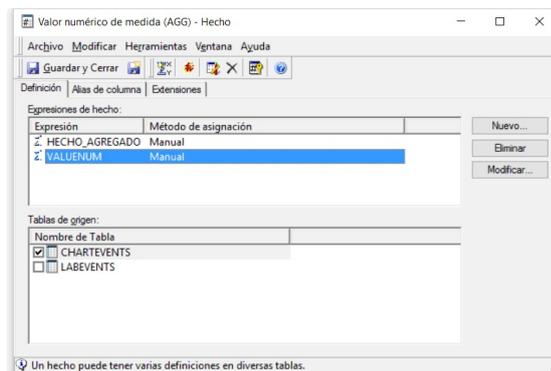


Figura 4.23: Expresión del hecho sobre la tabla ‘CHARTEVENTS’.

Siguiendo con el hecho de ejemplo inicial y más sencillo, esta información no se obtendrá mediante un acceso directo a una columna de la tabla ‘PATIENTS’ (que, por otra parte, puede verse que será la única tabla que, en esta ocasión, permita recuperar esta información a través del almacén, tal y como muestra la sección ‘Tablas de origen’), sino a través de una transformación de una columna que contiene parte de la información necesaria para el hecho deseado. Puesto que se desea obtener un hecho que indique la edad del paciente en el momento de su fallecimiento, y en la tabla de ‘PATIENTS’ sólo se dispone de la fecha de

nacimiento ‘DOB’ y de la fecha de fallecimiento ‘DOD’ en caso de encontrarse ya fallecido el paciente, será necesario realizar una operación sobre ambas columnas (con el fin de no necesitar incorporar una columna adicional al almacén, que sería otra alternativa). Esta operación consistirá en contabilizar los meses transcurridos entre la fecha de nacimiento y la de fallecimiento y dividir el resultado entre 12 meses que presenta cada año.

El aspecto que presentará la sentencia SQL proporcionada por la herramienta a la hora de elaborar un informe que contenga un indicador que utilice este hecho (sobre el que calcule el promedio mediante la función ‘AVG’) se refleja en la Figura 4.24.

```
select TO_NUMBER (SUBSTR (a12.DOB_YYYYMM, 1, 4)) DOB_YYYY,
a11.GENDER GENDER,
a11.GENDER || '-' || DECODE (a11.GENDER, 'M', 'HOMBRE', 'F', 'MUJER') GENDER_TEXT,
avg(MONTHS_BETWEEN (a11.DOD, a11.DOB) / 12) WJXBFST
from PATIENTS a11
join (SELECT DISTINCT TO_CHAR (DOB, 'YYYY-MM') DOB_YYYYMM, TO_CHAR (DOB, 'MONTH YYYY') DOB_MONTHYYYY FROM PATIENTS) a12
on (TO_CHAR (a11.DOB, 'YYYY-MM') = a12.DOB_YYYYMM)
where (a11.DOD is not null
and TO_NUMBER (SUBSTR (a12.DOB_YYYYMM, 1, 4)) between 2100 and 2150)
group by TO_NUMBER (SUBSTR (a12.DOB_YYYYMM, 1, 4)),
a11.GENDER,
a11.GENDER || '-' || DECODE (a11.GENDER, 'M', 'HOMBRE', 'F', 'MUJER)
```

Figura 4.24: Consulta SQL de un informe que utiliza el hecho.

En ella queda constancia de esa transformación realizada a partir de fecha de nacimiento y de fallecimiento para lograr determinar la edad del paciente medida en meses.

Por lo tanto, vemos como es posible no sólo definir un hecho como una columna de la cual recabar información, sino incluso construir un hecho ‘redundante’ (y no aumentar el volumen del datawarehouse) a partir de otros ya existentes, trasladando a la herramienta de explotación todo lo que se separe del almacenaje de datos (es decir, no parece recomendable generar una columna adicional en una tabla del almacén si el valor de dicha columna puede obtenerse, en tiempo de análisis, a partir de otras columnas ya existentes).

Posteriormente, se verá cómo este hecho puede ser empleado por un indicador para realizar cálculos dado un determinado nivel de dimensionalidad del informe (es decir, siendo dicho dato del indicador calculado frente a una serie de atributos que conformen el grado de detalle al que el informe desea calcular la métrica).

4.1.4 Jerarquías

En el momento de analizar los atributos, se ha visto que éstos no tenían por qué ser ‘ortogonales’, es decir, no tenían por qué estar desligados los unos de los otros en su totalidad. De hecho, es muy frecuente que los atributos se encuentren relacionados entre sí y dispongan de relaciones de padre a hijo o de hijo a padre, en función de la cardinalidad de las relaciones de sus distintos elementos (uno a uno, uno a muchos...).

Es por ello que existe la posibilidad ya no sólo de reflejar esas relaciones padres-hijos en la definición de atributos (algo que facilitará al motor analítico de MicroStrategy el determinar los pesos lógicos de las tablas, las claves únicas identificativas, gestionar la composición de sentencias SQL dinámicas...), sino también de reflejar estas relaciones entre atributos en un objeto llamado jerarquía.

Por tanto, una jerarquía representará las relaciones existentes entre atributos de una misma tipología, de una misma dimensión de estudio, así como configurará a la herramienta a la hora de poder explorar los elementos de los atributos (estableciendo qué niveles de la jerarquía servirán o no como punto de entrada en la exploración a través del Explorador de Datos, cuántos elementos mostrar, o cuántos elementos cachear con el fin de mejorar el rendimiento de la exploración de atributos cuyas tablas de look-up sean muy voluminosas o, como se ha visto anteriormente, sean tablas lógicas calculadas al vuelo sobre una tabla de hechos).

Continuando con el ejemplo de los pacientes, se define la jerarquía mostrada en la Figura 4.25.

Se trata de una jerarquía bastante pura, con relaciones 1 a N entre padre e hijos (respectivamente), en la que se ha establecido como punto de entrada a la exploración (check verde) todo aquel atributo que sea padre de al menos otro atributo de la jerarquía (quedando excluido por tanto, únicamente, el atributo Paciente, pilar fundamental de esta jerarquía, único atributo sin hijos de esta jerarquía). No obstante, en caso de desear explorar de manera directa el atributo Paciente, bastaría con establecerlo también como punto de entrada.

Pero este mismo esquema de atributos inter-relacionados con un atributo

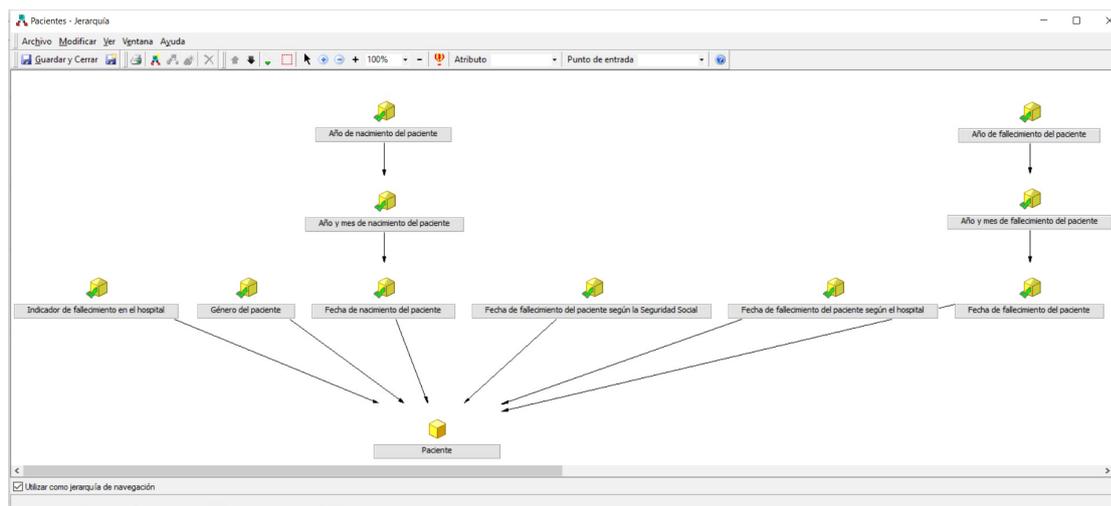


Figura 4.25: Definición de la jerarquía ‘Pacientes’.

común (en este caso ‘Paciente’) se repetirá en más de una ocasión. Será el caso del atributo ‘Admisión’ y todos sus atributos padres (que aportarán datos acerca de las admisiones para poder filtrar o catalogar éstas), será el caso de los ‘Sanitarios’, será el caso de los ‘Servicios’ prestados, será el caso de las ‘Estancias en la UCI’, será el caso de las distintas ‘Medidas de signos vitales’, o será el caso de las distintas ‘Pruebas de laboratorio’ a las que se vaya sometiendo el paciente. Es decir, al final toda la información existente en el datawarehouse se podrá segmentar en áreas o jerarquías de atributos perfectamente relacionados entre ellos, con un eje común. Y, además, todas estas jerarquías estarán relacionadas a su vez: de nada sirve la jerarquía de sanitarios si no se puede establecer una relación entre sanitarios y pacientes, o de nada sirve conocer los servicios prestados a un paciente si se desconoce en qué admisión concreta se le han prestado y a qué paciente corresponde dicha admisión.

De tal manera que, a pesar de que, como objetos de MicroStrategy, se definan una serie de jerarquías con cierta compartimentación lógica (para luego poder verse así reflejados, agrupados, en las navegaciones, por ejemplo), existirá una jerarquía global que englobe todos los atributos del sistema (mediante la unión de las jerarquías previamente creadas). El aspecto, muy general, de esta jerarquía dependerá de lo amplio funcionalmente del entorno concreto y será parecido al mostrado en la Figura 4.26.

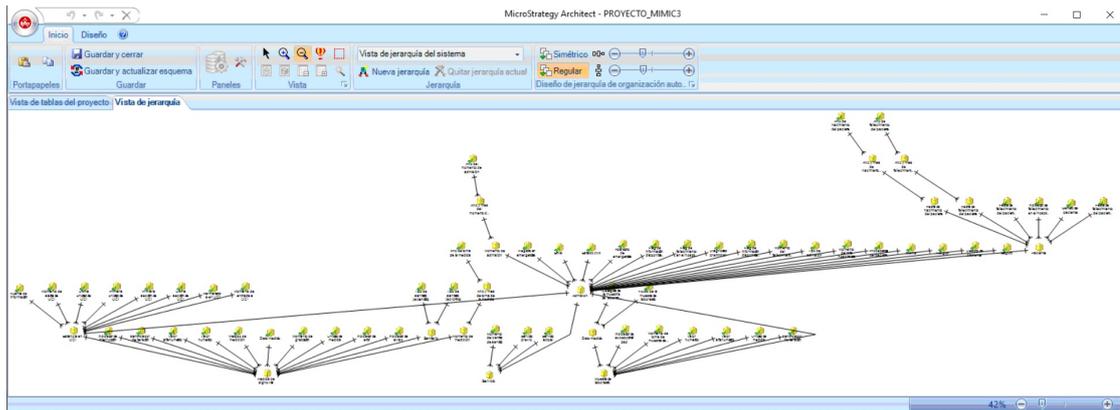


Figura 4.26: Vista global de las jerarquías del sistema.

4.2 Creación de objetos públicos

Una vez definidos los objetos de esquema, que servirán de capa de abstracción entre la herramienta de explotación y el modelo físico de la base de datos o Warehouse, llega el momento en el cual se puede comenzar con la creación de objetos públicos. Éstos se apoyarán en los objetos de esquema recién creados, aunque también será habitual la utilización de unos objetos públicos en otros objetos públicos.

Se pasa a detallar los principales objetos públicos existentes en MicroStrategy, su definición sobre el proyecto Mimic, así como algunos ejemplos de obtención final de resultados.

4.2.1 Indicadores

Una vez explicado el significado y la importancia de los dos principales objetos de esquema (atributos y hechos), es posible proceder a avanzar en el procesamiento de esta información, es decir, en el cálculo de métricas sobre dichos hechos en base a una determinada dimensionalidad o combinación de atributos.

Una métrica o indicador consistirá, básicamente y en la mayoría de los casos, en la aplicación de una fórmula a un hecho, junto a un posible filtro que aplicar a los datos de partida. Como anteriormente se ha explicado, no tiene por qué ser un hecho, sino que también puede ser un atributo, el dato base sobre el que se aplique la fórmula. Y, de manera recursiva, un indicador puede definirse directamente

como una operación (una división, por ejemplo) de otros indicadores previamente definidos en el proyecto (sin necesidad de hacer referencia, de nuevo, a los hechos empleados por esos dos indicadores que desean dividirse).

Para su definición, bastará con acudir al formulario de mantenimiento de indicadores y, para proseguir con el ejemplo anterior, avanzaremos en la definición de un indicador que emplee el hecho recién definido como ‘Edad en el momento del fallecimiento del paciente’, según muestra la Figura 4.27.

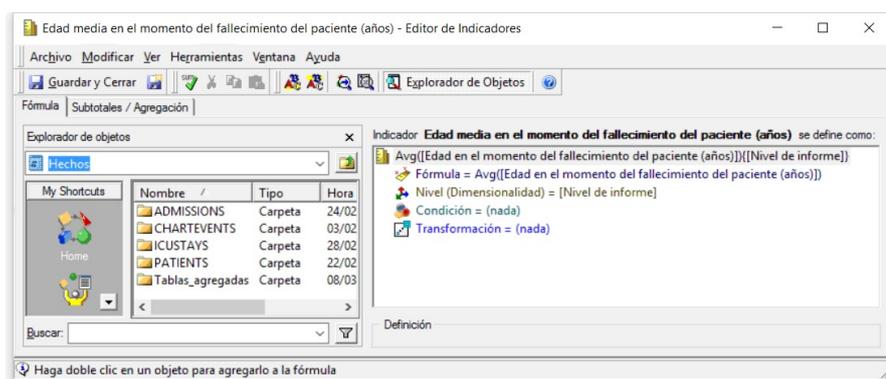


Figura 4.27: Definición de un indicador como promedio de un hecho.

Bastará con indicar el hecho de partida que se tomará, y qué función desea serle aplicada, con el fin de disponer de la fórmula final. Se podrá utilizar un filtro en caso de así desearse (que califique sobre algún atributo relacionado con el hecho en cuestión, o un filtro que califique sobre otro indicador...), se podrá especificar una dimensionalidad concreta de resolución del indicador (para que la herramienta no calcule, por defecto, el indicador al nivel dimensional dado por la definición de atributos en cada momento), o aplicarse transformaciones (con el fin de poder calcular ese mismo indicador como tasa de variación entre el período deseado y el inmediatamente anterior, por ejemplo).

Puesto que el interés en este caso es obtener la edad promedio que presentan los pacientes en el momento de fallecer (y que coincidirá con la del propio ‘Paciente’ concreto en caso de definirse a ese nivel de detalle el informe correspondiente), la función a emplear consistirá en ‘AVG’ y el hecho sobre el que calcularla será el creado en el paso previo. No se aplicará filtro ninguno al no desearse condicionar por nada (género del paciente, por ejemplo).

4.2.2 Filtros

Otro objeto público muy utilizado es el filtro. Consiste en un objeto que permitirá restringir resultados bajo una determinada condición. Esta condición podrá ser establecida sobre los resultados completos de un informe, o sólo sobre un determinado indicador, o incluso en un atributo a la hora de explorar sus elementos. Ello dependerá de en qué objeto sea empleado el filtro en cuestión.

Los métodos más habituales de filtrar son sobre los diferentes valores presentes en un determinado atributo (es decir, en su tabla maestra o de look-up) o restringiendo los posibles valores que pueda tomar un indicador.

Es posible realizar tanto filtros estáticos (es decir, aquellos que contienen toda la parametrización necesaria para saber cómo actuar) como dinámicos (en cuyo caso, en tiempo de ejecución el usuario sería consultado sobre algún punto de flexibilidad así incluido en el filtro).

Siguiendo con el caso de los pacientes y su edad en el momento de fallecer, parece interesante definir un filtro que sólo permita obtener resultados para pacientes ya fallecidos, de manera que los pacientes que aún conservan la vida no distorsionen las cifras de la estadística deseada.

Para definir un filtro que condicione únicamente por pacientes fallecidos bastará con utilizar el formulario de definición de filtros y cualificar el atributo ‘Fecha de fallecimiento del paciente’ como indica la Figura 4.28.

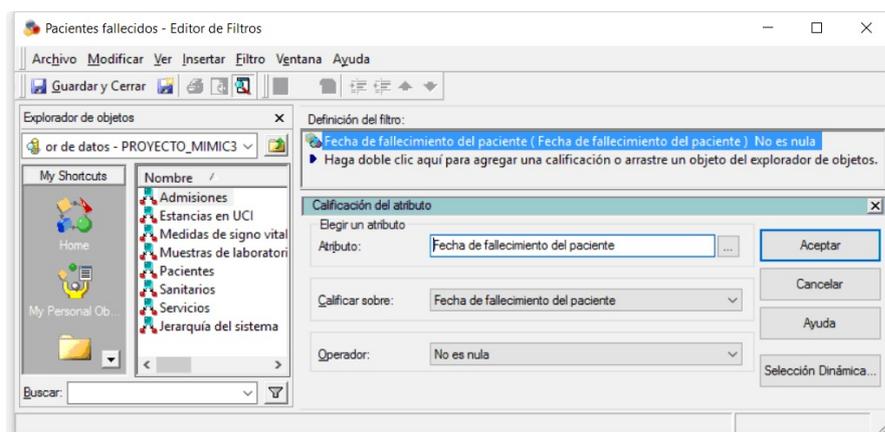


Figura 4.28: Definición del filtro ‘Pacientes fallecidos’.

Bastará con que éste se encuentre cumplimentado para poder garantizar que

el paciente ya no vive.

En caso de quererse un filtro más variable, dinámico, cuya definición general exista pero cuyos valores concretos de criba sean elegidos en el momento de ejecución, se empleará este mismo mantenimiento de filtros pero se utilizará una selección dinámica en su contenido.

Por ejemplo, deseando hacer un filtro que pregunte en tiempo de ejecución para qué años de nacimiento de los pacientes desea obtenerse la información resultante, se diseñará el filtro de la Figura 4.29.

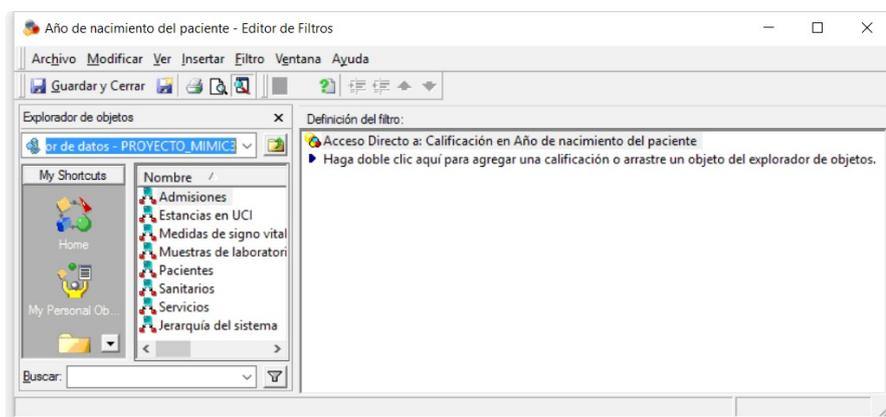


Figura 4.29: Definición de un filtro dinámico sobre el año de nacimiento.

En su interior sólo existe un acceso directo a una selección dinámica, objeto público que pasa a describirse a continuación.

4.2.3 Selecciones dinámicas

Una selección dinámica consiste en un objeto público que permite flexibilizar un único objeto (un filtro, un indicador, un informe...) en más de un comportamiento, siendo así reducida la cifra final de objetos resultantes en la herramienta.

Continuando con el ejemplo anterior, si se desea definir una selección dinámica que solicite, en tiempo de ejecución, el año de nacimiento de los pacientes sobre los que se desea obtener la estadística, bastará con calificar el atributo 'Año de nacimiento del paciente' como indica la Figura 4.30.

Será posible definir un filtro previo bajo el cual mostrar únicamente los datos resultantes como selección posible de esta segunda selección dinámica, definir

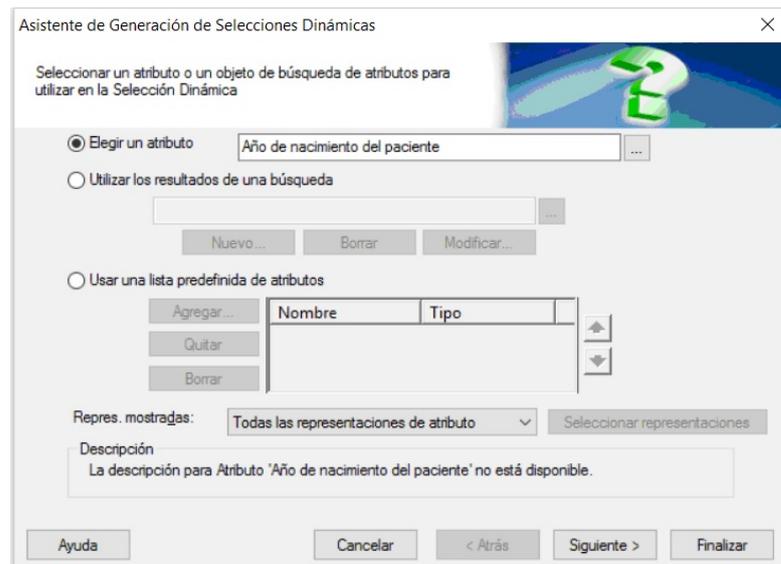


Figura 4.30: Selección dinámica para solicitar el año de nacimiento.

el menor o mayor número de elementos tolerado que podrá elegirse en esta selección dinámica, o incluso preseleccionar un patrón de elementos concretos que, posteriormente, el usuario pueda conservar o modificar, como se refleja en la Figura 4.31.

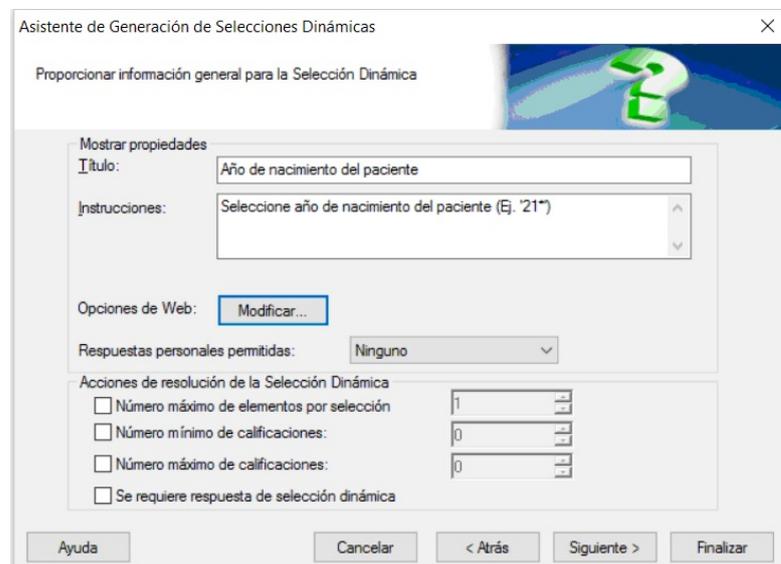


Figura 4.31: Información general a definir en la selección dinámica.

En este caso, no se utilizará ninguna de estas opciones adicionales, dado que no es necesario hacer uso de ellas para alcanzar el objetivo deseado.

4.2.4 Informes

La definición de un informe consistirá en la unión de los componentes que se han ido viendo hasta el momento:

- Atributos: permitirán determinar una dimensionalidad en base a cuya disposición o layout sean ubicados los resultados.
- Indicadores: serán calculados acordes a la definición de la plantilla sobre la que serán mostrados.
- Filtros: permitirán, estática o dinámicamente, condicionar o limitar los resultados devueltos.

Por tanto, la definición de un informe se realizará accediendo al formulario de mantenimiento correspondiente y seleccionando cada uno de los elementos recién comentados (todos o parte de ellos).

Por mantener el mismo ejemplo seguido hasta el momento, se define el informe que permitirá consultar la edad media de los pacientes en el momento de fallecer, visualizados en serie histórica, siendo separadas las series históricas por género del paciente, para aquellos pacientes que ya han fallecido y que, además, nacieron entre el año 2100 y el año 2199, mostrándose en la Figura 4.32.

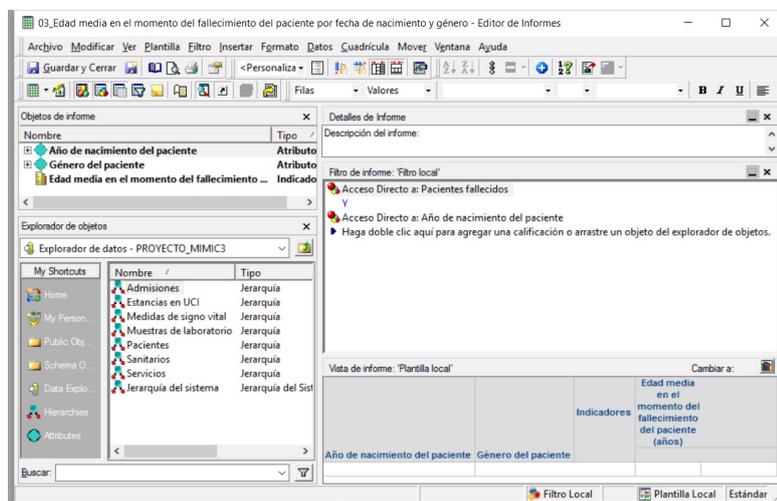


Figura 4.32: Definición de un informe de edad media de fallecimiento.

Se puede observar la disposición de los dos atributos que servirán de desglose de la información en la plantilla (se trataría del año de nacimiento del paciente y del género del paciente), así como del indicador que se calculará, al igual que

puede apreciarse el acceso directo a los dos filtros anteriormente mencionados, que han sido unidos entre sí mediante el operador lógico ‘Y’.

Durante la ejecución del informe, se solicitará especificar el único grado de libertad que se ha otorgado al informe, consistente en el rango de años de nacimiento deseado de los pacientes. La Figura 4.33 muestra el caso en que se elige el rango comprendido entre el año 2100 al año 2199.

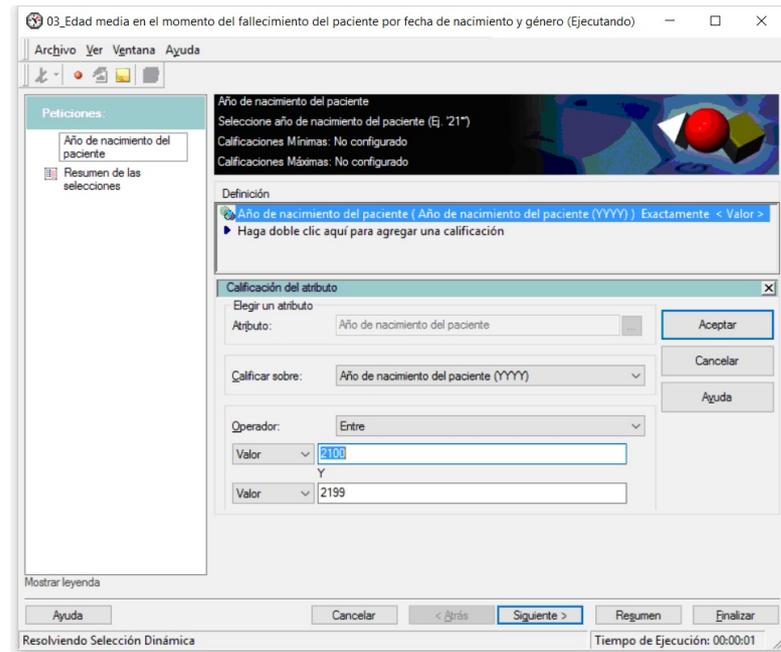


Figura 4.33: Solicitud de años de nacimiento de los pacientes a consultar.

En ese caso, el aspecto del informe ejecutado será el reflejado en la Figura 4.34.

En él, el analista de los datos podrá apreciar la evolución de las cifras, bien separadamente por géneros o bien de manera conjunta, de edad media de los diferentes fallecidos en función de en qué año nacieron (tratando de establecer la relación entre la longevidad y la fecha de nacimiento).

No obstante, un determinado informe es posible ejecutarlo de manera tabular, es decir, en forma de cuadrícula (como acaba de verse en esta ocasión), en forma gráfica (siendo muchos los tipos de gráficos existentes) o de ambas formas a la vez (combinando en una misma visualización datos tabulados frente a su aspecto evolutivo en una gráfica).

En el caso del informe concreto recién analizado, su representación gráfica

Año de nacimiento del paciente	Género del paciente	Edad media en el momento del fallecimiento del paciente (años)
2100	M - HOMBRE	70,5
2100	F - MUJER	71,7
2100	Promedio	71,1
2101	M - HOMBRE	68,5
2101	F - MUJER	73,0
2101	Promedio	71,2
2102	M - HOMBRE	71,1
2102	F - MUJER	71,0
2102	Promedio	71,1
2103	M - HOMBRE	67,0
2103	F - MUJER	73,1
2103	Promedio	70,0
2104	M - HOMBRE	67,5
2104	F - MUJER	69,1
2104	Promedio	68,3
2105	M - HOMBRE	70,3
2105	F - MUJER	71,7
2105	Promedio	71,0
2106	M - HOMBRE	69,1
2106	F - MUJER	73,3
2106	Promedio	71,2

Figura 4.34: Aspecto de la resolución del informe en modo cuadrícula.

podría ser la mostrada en la Figura 4.35.

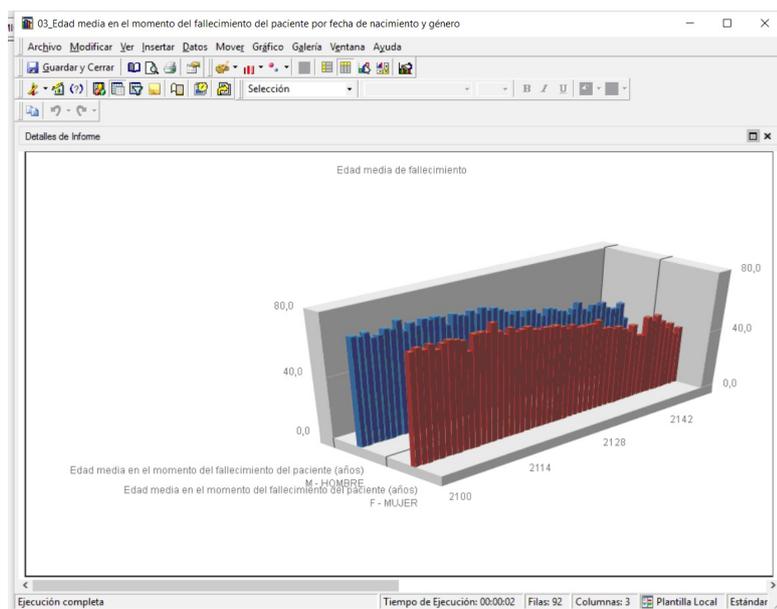


Figura 4.35: Aspecto de la resolución del informe en modo gráfico.

Indistintamente de la presentación visual de los resultados (cuadrícula o gráfica), la sintaxis SQL empleada por la herramienta de explotación MicroStrategy con el fin de poder calcular las métricas al nivel de dimensionalidad indicado por los atributos de la plantilla, con los filtros estáticos o dinámicos empleados, etc. quedaría resumida a una serie de pasos o etapas SQL mostradas en la Figura 4.36 encargadas de, entre todas ellas, lograr el objetivo: un resultado

integral que dé respuesta a la solicitud realizada.

```

Paso0 - Hora de inicio de pases de consulta:      30/04/2016 18:58:20
        Hora de finalización de pases de consulta: 30/04/2016 18:58:20
        Ejecución de la consulta:                0:00:00.00
        Recuperación y procesamiento de datos:   0:00:00.06
        Transferencia de datos desde origen u orígenes de datos: 0:00:00.06
        Otro procesamiento: 0:00:00.04
        Rows selected: 163
select  TO_NUMBER(SUBSTR (a12.DOB_YYYYMM, 1, 4)) DOB_YYYY,
        a11.GENDER GENDER,
        a11.GENDER || '-' || DECODE (a11.GENDER, 'M', 'HOMBRE', 'F', 'MUJER') GENDER_TEXT,
        avg(MONTHS_BETWEEN (a11.DOD, a11.DOB) / 12) WJXBFS1
from    PATIENTS      a11
        join          (SELECT DISTINCT TO_CHAR (DOB, 'YYYY-MM') DOB_YYYYMM, TO_CHAR (DOB, 'MONTH YYYY') DOB_MONTHYYYY FROM PATIENTS) a12
        on           (TO_CHAR (a11.DOB, 'YYYY-MM') = a12.DOB_YYYYMM)
where   (a11.DOD is not null
and TO_NUMBER (SUBSTR (a12.DOB_YYYYMM, 1, 4)) between 2100 and 2199)
group by TO_NUMBER (SUBSTR (a12.DOB_YYYYMM, 1, 4)),
         a11.GENDER,
         a11.GENDER || '-' || DECODE (a11.GENDER, 'M', 'HOMBRE', 'F', 'MUJER')

Paso1 - Hora de inicio de pases de consulta:      30/04/2016 18:58:20
        Hora de finalización de pases de consulta: 30/04/2016 18:58:20
        Ejecución de la consulta:                0:00:00.00
        Recuperación y procesamiento de datos:   0:00:00.00
        Transferencia de datos desde origen u orígenes de datos: 0:00:00.00
        Otro procesamiento: 0:00:00.00
[Populate Report Data]

[Pasos de cálculo en el motor analítico:
 1. Calcular subtotal: <Promedio>
 2. Realizar tabulación cruzada
]
    
```

Figura 4.36: Sintaxis de la sentencia SQL de resolución del informe.

Si bien el objetivo del proyecto es proporcionar el entorno tecnológico físico (gestor de base de datos, herramientas) y la infraestructura lógica básica (diseño, modelización, objetos de esquema) para explotar la información proporcionada por Physionet, también se han creado una serie de informes representativos, tanto por áreas funcionales (pacientes, admisiones, servicios, estancias en la UCI...) como por las diferentes topologías de informes (principalmente gráficas). Así, se han creado una serie de informes que pueden ser mostrados de manera rápida.

En primer lugar, y tratando la información de pacientes, un informe que, de manera gráfica (circular) y sin realizar consulta alguna al usuario que lo solicita, muestra el porcentaje de pacientes tratados y que han fallecido frente al porcentaje de pacientes tratados que no han fallecido, diferenciados por género. Queda reflejado en la Figura 4.37.

Combinando ya información de pacientes y de estancias en UCI, se elabora un informe gráfico (también estático, sin necesidad de parametrización por la persona que lo ejecute) que muestre la evolución temporal de las cifras de fallecimientos de pacientes como de estancias en UCI de éstos, diferenciados por género, como refleja la Figura 4.38.

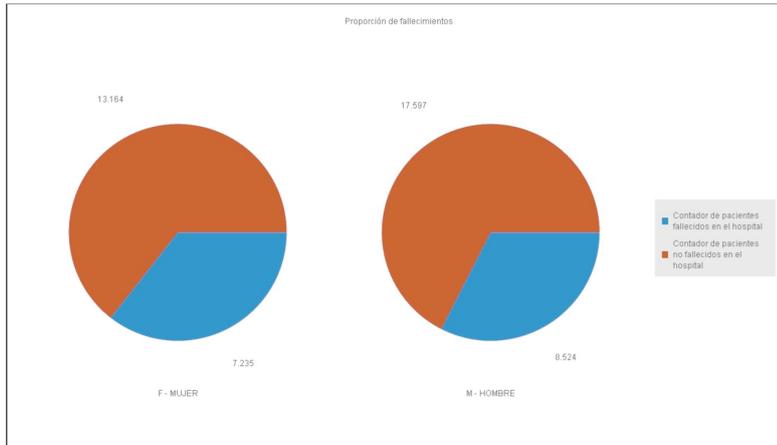


Figura 4.37: Informe gráfico del porcentaje de pacientes fallecidos por género.

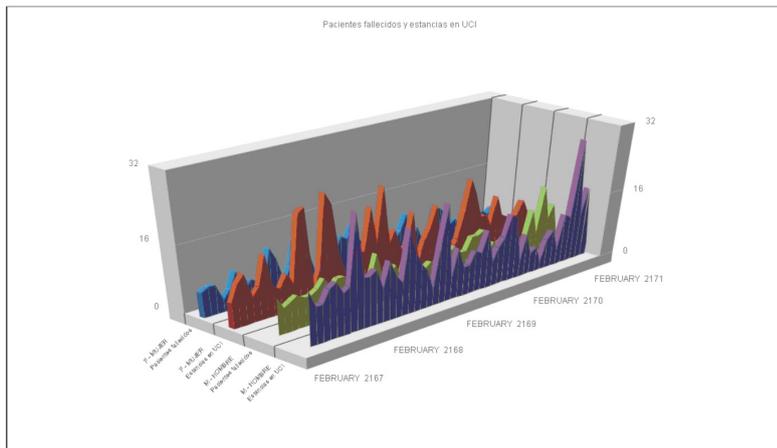


Figura 4.38: Evolución temporal de pacientes y estancias en UCI de pacientes fallecidos.

Otro informe mostrado en la Figura 4.39 calculará los contadores, paginados por tipo de seguro asociado al paciente, de admisiones practicadas, segmentándose la información por estado civil y etnia, permitiéndose la selección dinámica de qué seguros médicos contemplar.

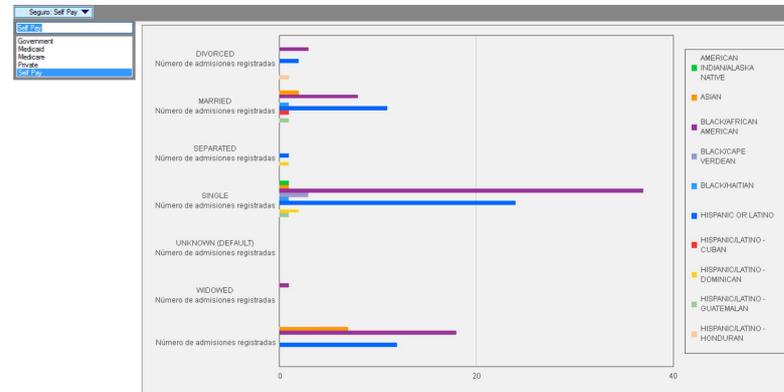


Figura 4.39: Admisiones por seguro médico, estado civil y etnia del paciente.

Prescindiendo de la etnia como dimensión y variando la tipología gráfica, la información puede ser mostrada ahora en términos porcentuales como en la Figura 4.40.

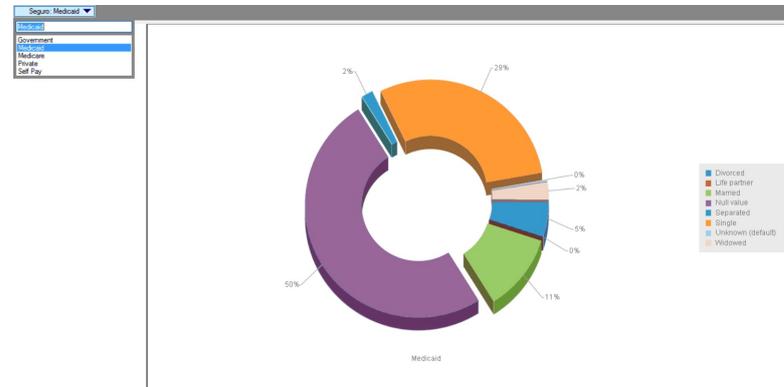


Figura 4.40: Admisiones por etnia y seguro médico del paciente.

Otro informe, visible en la Figura 4.41, permitirá al usuario especificar qué primeras secciones en UCI fueron visitadas por el paciente para considerar éstas o no dentro del ámbito del informe, y mostrará el conteo de muestras de laboratorio practicadas por tipo de muestra y segmentadas por género.

El siguiente informe es estático (por lo que el usuario no debe especificar ninguna parametrización en tiempo de ejecución) y muestra, en serie histórica,

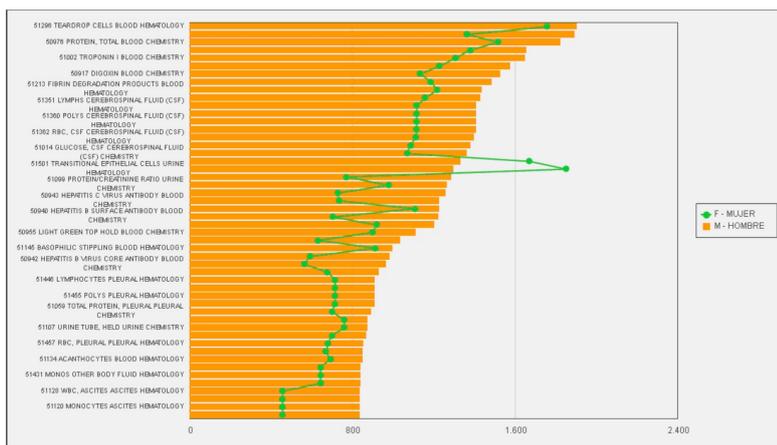


Figura 4.41: Muestras de laboratorio por género y tipo de muestra.

la evolución de la cifra de medidas de signo vital tomadas a pacientes fallecidos, desglosadas por la última unidad que visitó el paciente durante su estancia en UCI, tal y como se muestra en la Figura 4.42.

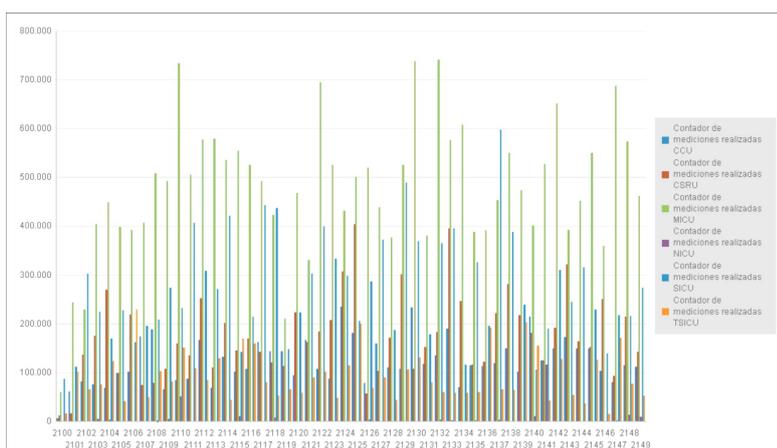


Figura 4.42: Medidas tomadas a pacientes fallecidos por última unidad de UCI.

Otra muestra de informe elaborado, ilustrado en la Figura 4.43, consiste en aquél que, siendo estático, muestra el tiempo de permanencia en UCI de los pacientes, en función de qué sección de UCI fue la de entrada y cuál la de salida.

4.2.5 Navegaciones

Una de las principales potencias analíticas de este tipo de plataformas es la capacidad que poseen de seguir indagando, de manera dinámica, acerca de unos datos previamente obtenidos a un nivel de menor detalle, a un nivel más agregado.

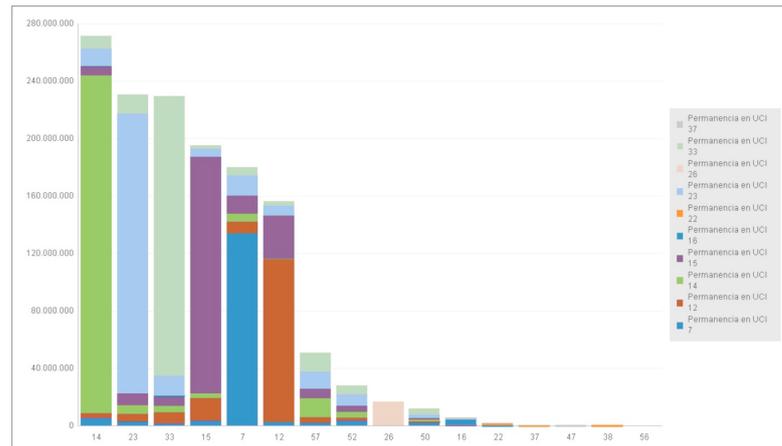


Figura 4.43: Tiempo de permanencia en UCI según el flujo entrada-salida.

Aunque no se trate de un objeto más a poder definir en la herramienta, pero sí estar estrechamente relacionado con los informes recién explicados, se explica a continuación en qué consiste.

Como se ha visto, un informe será resuelto al nivel de detalle que indique la disposición de atributos en la plantilla (bien sea en filas, en columnas, en páginas...). Será el modo de disponer el indicador frente a dichos atributos lo que indique a la herramienta el grado de cardinalidad al que calcular dicho indicador.

Pero, partiendo de unos resultados a nivel agregado por una serie de atributos, puede resultar de interés seguir indagando más acerca de una cifra concreta (porque parezca ser llamativa, porque represente un elevado porcentaje de la cifra global y merezca la pena un mayor estudio...).

Para ello, se dispone de la facilidad de navegar, consistente en seguir profundizando (si se hace descendentemente) o en perder detalle (si se hace ascendentemente), o en obtener información ‘transversal’ (si se realiza en dimensiones más bien ‘ortogonales’) acerca de un dato obtenido inicialmente a un nivel de detalle diferente del que se obtendrá tras la navegación.

Disponer de unas u otras opciones de navegación dependerá de qué relaciones hijos-padres hayan sido definidas a la hora de establecer la caracterización de los atributos, de qué jerarquías hayan sido definidas en la aplicación, de qué puntos de entrada hayan sido establecidos en dichas jerarquías, o incluso de si se ha establecido un ‘Mapa de navegación’ concreto asociado a un determinado informe

o a un determinado atributo (algo que establecería, por encima de las jerarquías definidas de forma global en la herramienta, qué posibilidades de navegación tiene ese informe o ese atributo).

Como primer ejemplo, se parte de un informe que muestra el número de admisiones identificadas, desglosado por raza del paciente y estado civil del paciente, pudiendo además ser filtrado por el tipo de seguro médico asociado al paciente (que, a su vez, servirá de eje de paginación de los resultados). La Figura 4.44 muestra el aspecto del asistente de ejecución de este informe en el momento en el que se permite el condicionamiento por tipo de seguro.

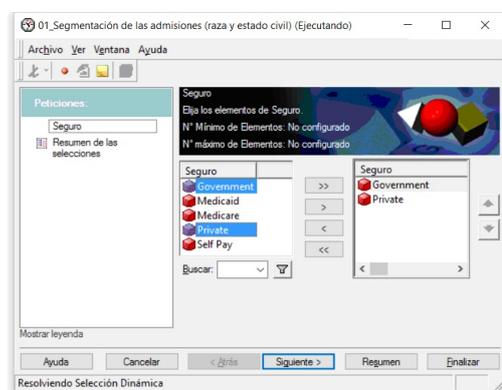


Figura 4.44: Filtro dinámico por tipo de seguro médico en el informe.

La Figura 4.45 muestra el aspecto tras la obtención de resultados.

Indicadores	Estado civil	Etnia											
		AMERICAN INDIAN/ALASKA NATIVE	ASIAN	ASIAN - ASIAN	ASIAN - CHINESE	ASIAN - FILIPINO	ASIAN - OTHER	ASIAN - THAI	ASIAN - VIETNAMESE	BLACK/AFRICAN	BLACK/CA AMERICAN	BLACK/CA AMERICAN - HISPANIC	
Número de admisiones registradas	DIVORCED		1						2	1		3	9
	MARRIED		32	5	12								42
	SEPARATED		1										7
	SINGLE		14		4	4	1	2		1		1	186
	UNKNOWN (DEFAULT)												
	WIDOWED		4		1								9
	Total	1	25		1			4	2		1	136	309
		1	27	5	17	4	1	4	2		5		

Figura 4.45: Aspecto del informe una vez resuelto.

Si, por ejemplo, se pretende profundizar algo más en el detalle de la cifra correspondiente al tipo de seguro 'Government', raza 'Asian' y estado civil 'Married', bastará con posicionarse sobre la cifra y, dada la parametrización que se ha ido haciendo de jerarquías, atributos, relaciones..., se obtendrá un árbol de

posibles destinos a los que navegar con el fin de disponer de una visión diferente del detalle del informe, tal y como refleja la Figura 4.46.

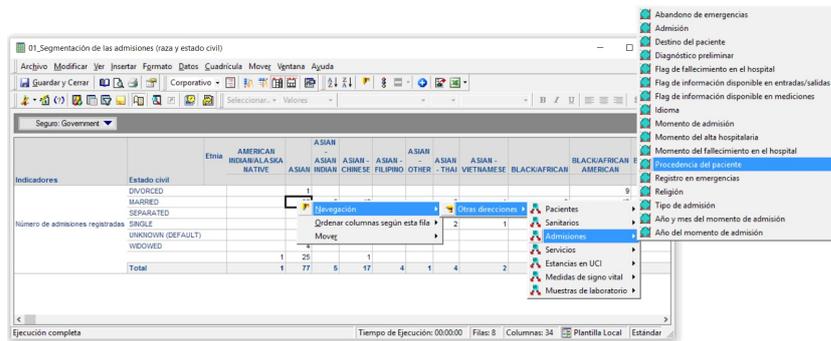


Figura 4.46: Alternativas de navegación en el informe.

Si, tal y como se muestra en dicha imagen, se hace uso del atributo ‘Procedencia del paciente’, el informe que se ejecutará a continuación tendrá la presencia de la Figura 4.47.

Seguro: Government		Etnia		ASIAN	Total
Indicadores	Estado civil	Procedencia del paciente			
Número de admisiones registradas	MARRIED	CLINIC REFERRAL/PREMATURE		6	6
		EMERGENCY ROOM ADMIT		14	14
		PHYS REFERRAL/NORMAL DELI		8	8
		TRANSFER FROM HOSPEXTRAM		3	3
		TRANSFER FROM SKILLED NUR		1	1
		Total	Total		32
Total			32	32	

Figura 4.47: Informe tras navegar a ‘Procedencia del paciente’.

Si, a continuación, se quiere seguir profundizando en detalle y acudir a conocer qué Pacientes concretos conforman la cifra correspondiente a ‘Clinic referral/premature’, bastará con navegar de nuevo desde dicha cifra al atributo ‘Paciente’ como indica la Figura 4.48.

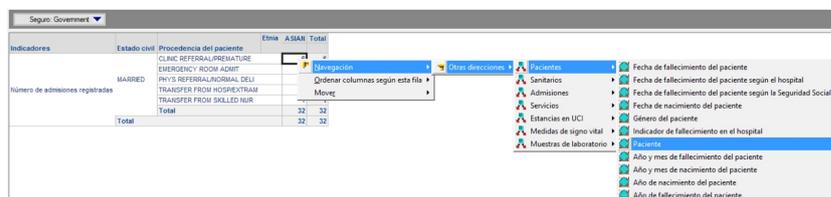


Figura 4.48: Navegación para llegar al nivel de detalle de ‘Paciente’.

Así se llegará, como refleja la Figura 4.49, al detalle de Pacientes concretos que han integrado, de manera agregada, esa cifra de partida.

Indicadores	Estado civil	Procedencia del paciente	Paciente	Etnia	ASIAN	Total
			10505 10/02/2043	M - HOMBRE	0	1 1
			15747 04/06/2071 30/11/2142 30/11/2142	M - HOMBRE	1	1 1
		CLINIC REFERRAL-PREATURE	19410 10/09/2089 22/08/2153 22/08/2153	M - HOMBRE	1	1 1
Número de admisiones registradas	MARRIED		48453 15/06/2143 11/01/2201 11/01/2201	M - HOMBRE	1	2 2
			74945 12/07/2129 14/04/2189	F - MUJER	1	1 1
		Total				6 6
		Total				6 6
		Total				6 6

Figura 4.49: Resultado del informe ya navegado a ‘Paciente’.

Se trata de una funcionalidad de elevada demanda, dados los grandes resultados que proporciona. Permite prescindir de infinidad de informes prediseñados, así como poder indagar de forma natural acerca de resultados de partida más o menos interesantes (es decir, otorga a la herramienta de análisis de una gran capacidad de trazabilidad de un dato desde su comienzo hasta su final durante las etapas de vida del dato a través de todo el workflow del negocio). De cara a la minería de datos (o data mining), permite obtener conclusiones acerca de comportamientos o determinar relaciones entre atributos, o entre indicadores y atributos que, a primera vista, podrían pasar desapercibidas.

Además, es posible navegar a una plantilla predeterminada de informe (por ejemplo), incluso sin heredar todos los filtros que se han ido aplicando con anterioridad a los informes de partida, es decir, condicionando la navegación a a plantilla únicamente con el identificador del atributo desde el que se navegue. Esto es útil, por ejemplo, de cara a poder navegar, desde el atributo Paciente, a una plantilla predeterminada a la que parezca de interés acudir siempre que se haya llegado ya al nivel de detalle de paciente tras varias navegaciones. Para ello, bastará con modificar, como indica la Figura 4.50, el Mapa de Navegación por defecto del atributo Paciente y establecer una plantilla a la que poder navegar.

Al otorgársele prioridad alta, se le asignará de manera automática un hipervínculo (accediendo por MicroStrategy Web, no así accediendo por MicroStrategy Developer) en las diferentes apariciones del atributo en los informes de manera que, de forma predeterminada, pinchar en el hipervínculo signifique navegar a esa plantilla, heredando tan sólo el identificador de Paciente de partida.

Por ejemplo, se parte de un informe (consultado en MicroStrategy Web, de ahí el cambio de apariencia) que muestre el ranking (generalmente conocidos como informes ‘Top-N’) de los 5 Pacientes con un mayor número de Admisiones

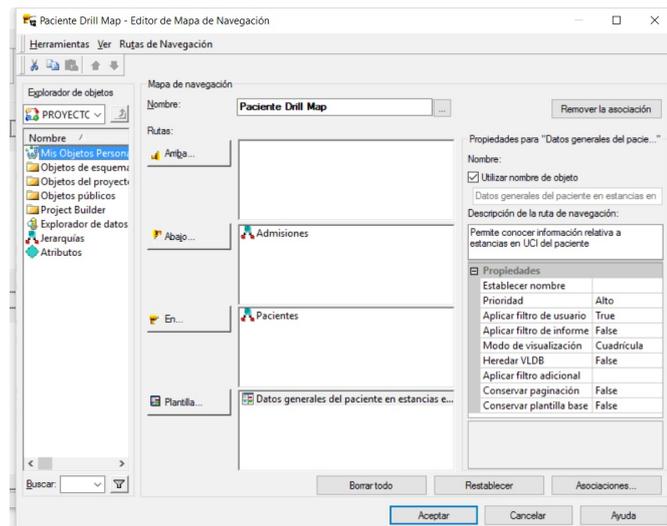


Figura 4.50: Modificación del mapa de navegación del atributo ‘Paciente’.

registradas, tal y como muestra la Figura 4.51.



Figura 4.51: Informe de los cinco pacientes con más admisiones en web.

Se observará que cada representación del atributo ‘Paciente’ aparecerá subrayada como síntoma de que lleva emparejada un hipervínculo, cuyo destino (la plantilla de interés) será informada mediante un pop-up con sólo pasar el cursor del ratón sobre el mismo. Con hacer click en dicho hipervínculo, se navegará a una plantilla diseñada con una serie de indicadores generales de interés asociados a un determinado Paciente como muestra la Figura 4.52.



Figura 4.52: Resultado de navegación por defecto a plantilla desde web.

Un ejemplo curioso es aquel que ya se comentaba con anterioridad de disponer

de un hecho que, entre sus expresiones de atributo diferenciadas, es capaz de obtener la información tanto de una tabla agregada como de las propias tablas originales de hechos proporcionadas por Physionet. En este caso, un informe elaborado con MicroStrategy y que se amolde, tanto en dimensionalidad como en hechos utilizados, a la tabla agregada prediseñada, empleará dicha tabla agregada de manera prioritaria, al percibir que esa consulta tardará menos empleándola que tomando la otra alternativa de la tabla de hechos original.

Para determinar qué tabla será menos costosa de emplear, se basará en el peso lógico que haya asociado a cada tabla (de ahí la importancia de actualizar esquema tras cada cambio de diseño de modelo realizado). La Figura 4.53 refleja que la tabla original ‘CHARTEVENTS’ tiene un peso lógico elevado.

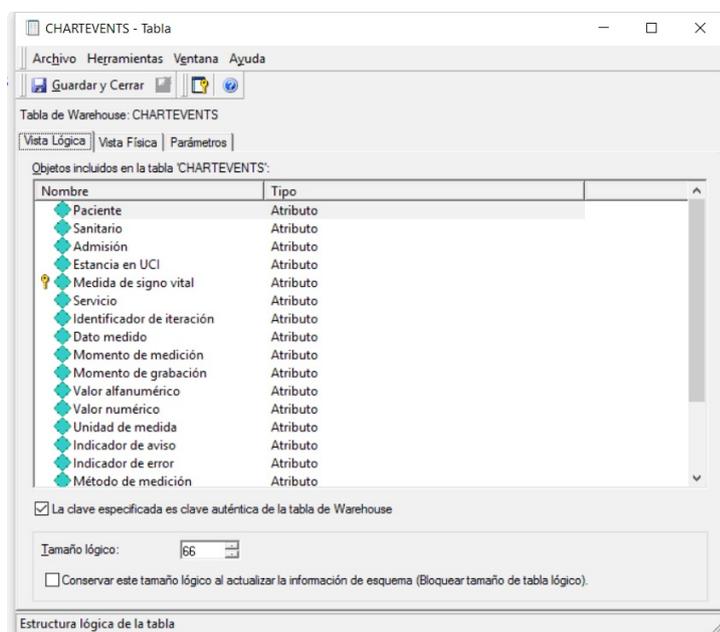


Figura 4.53: Elevado peso lógico de la tabla ‘CHARTEVENTS’.

Sin embargo, la Figura 4.54 refleja que la tabla agregada ‘AGG_MEDIDA_ETNIA_DATO’ presupone un coste menor de consulta dado su menor peso lógico.

En la Figura 4.55 se muestra el informe básico (antes de navegar).

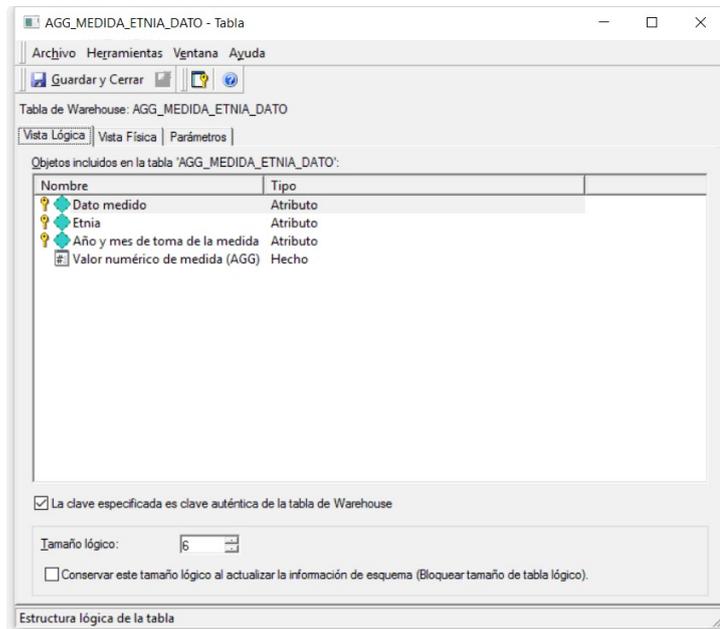


Figura 4.54: Menor peso lógico de la tabla ‘AGG_MEDIDA_ETNIA_DATO’.

Informe_contra_agregada

Archivo Modificar Ver Insertar Formato Datos Cuadrícula Mover Ventana Ayuda

Guardar y Cerrar Personaliza Filas Valores

Detalles de Informe

Indicadores	Etnia	AMERICAN INDIAN/ALASKA NATIVE	ASIAN	ASIAN - CHINESE	BLACK/AFRICAN AMERICAN
		Suma del valor numérico de medida (AGG)			
Dato medido					
220277	O2 saturation pulseoxymetry			6.590	23.159
220292	Minute Volume Alarm - Low				
220293	Minute Volume Alarm - High				
220339	PEEP set				
220545	Hematocrit (serum)		4.989	1.132	3.654
220546	WBC		2.087	222	1.224
220580	Ammonia				
220581	Amylase				
220587	AST		73		364
220602	Chloride (serum)		1.300	318	1.759
220603	Cholesterol		219		194
220615	Creatinine		125	51	254
220621	Glucose (serum)		1.746	342	2.098
220624	HDL		59		66
220632	LDH		831		1.488
220635	Magnesium		225	45	396
220644	ALT		53		309
220645	Sodium (serum)		1.615	424	2.511
220650	Total Protein				

Figura 4.55: Informe resuelto contra la tabla agregada.

Éste acudiría a la tabla agregada a obtener sus resultados, como puede verse en la consulta SQL realizada por MicroStrategy contra la base de datos y capturada en la Figura 4.56.

```
select  a11.ITEMID ITEMID,
        a12.LABEL LABEL,
        a12.ABBREVIATION ABBREVIATION,
        a11.ETHNICITY ETHNICITY,
        sum(a11.HECHO_AGREGADO) WJXBFS1
from    AGG_MEDIDA_ETNIA_DATO a11
join    D_ITEMS a12
on      (a11.ITEMID = a12.ITEMID)
where   a11.CHARTTIME_YYYYMM in (210008, 210009, 210010)
group by
        a11.ITEMID,
        a12.LABEL,
        a12.ABBREVIATION,
        a11.ETHNICITY
```

Figura 4.56: Consulta SQL empleada contra la tabla agregada.

Supongamos la navegación realizada sobre este informe original mostrada en la Figura 4.57.

	AMERICAN INDIAN/ALASKA NATIVE	ASIAN	ASIAN - CHINESE	BLACK/AFRICAN AMERICAN	HISPANIC OR LATINO	NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER	URGE TO OBTAIN
Indicadores	Suma del valor numérico de medida (AGG)	Suma del valor numérico de medida (AGG)					
Dato medido							
220277			6.590				2
220292							
220293							
220339							
220545		4.989	1.132				
220546		2.087	222				
220580							
220581							
220607			73				
220602			1.300	318			
220603							
220615							
220621			1				
220624							
220632							
220635			225	45			
220644			53				
220645			1.615	424			
220650							

Figura 4.57: Selección de un destino de navegación desde el informe.

Ésta proporcionará un nivel de dimensionalidad de plantilla para el que el hecho precalculado sobre la tabla agregada no está capacitado, como refleja la Figura 4.58.

En ese caso, como refleja la Figura 4.59, el informe será calculado (gracias a las diferentes expresiones de hecho definidas) sobre la propia tabla de hechos 'CHARTEVENTS' y demás apoyos necesarios (teniendo, necesariamente, que reducirse el rendimiento).

Es una prueba más de la potencia de esta herramienta que, dinámicamente, obtiene la información del origen de datos que mejor rendimiento vaya a

Dato medido	Sanitario	Etnia	ASIAN
220603	Cholesterol Cholesterol	20889	IMD IMD 219

Figura 4.58: Dimensionalidad de informe disconforme con la tabla agregada.

```

select
  a11.ITEMID ITEMID,
  a13.LABEL LABEL,
  a13.ABBREVIATION ABBREVIATION,
  a11.CGID CGID,
  a14.LABEL LABEL0,
  a14.DESCRPTION DESCRIPTION,
  a12.ETHNICITY ETHNICITY,
  sum(a11.VALUENUM) WJXBFS1
from
  CHARTEVENTS a11
  join ADMISSIONS a12
    on (a11.HADM_ID = a12.HADM_ID)
  join D_ITEMS a13
    on (a11.ITEMID = a13.ITEMID)
  join CAREGIVERS a14
    on (a11.CGID = a14.CGID)
where
  (a11.ITEMID in (220603)
  and a12.ETHNICITY in ('ASIAN')
  and a11.CHARTTIME_YYYYMM in (210008, 210009, 210010))
group by
  a11.ITEMID,
  a13.LABEL,
  a13.ABBREVIATION,
  a11.CGID,
  a14.LABEL,
  a14.DESCRPTION,
  a12.ETHNICITY

```

Figura 4.59: Consulta SQL empleada contra la tabla de hechos original.

proporcionar pero que, en caso de obstáculos, restablece ese origen de la información a aquél que sea capaz de proporcionársela aun con un peor rendimiento.

4.2.6 Tableros de control

Cada día es más frecuente utilizar este tipo de herramientas de explotación, no para la simple ejecución de un determinado informe, sino para tener una rápida visión global de la totalidad de los datos, sobre todo de los más recientes. Este tipo de uso es más demandado por perfiles ejecutivos o de alta dirección que por perfiles analistas o más técnicos.

Un tablero de control es una disposición física de información procedente de uno o más informes, que puede ser distribuida sobre la superficie disponible de una

manera diferente a la utilizada en los informes de origen, de manera que, de un primer vistazo, pueda obtenerse cierta información de interés relativa a uno o más ámbitos (en nuestro caso de los pacientes y sus datos de admisiones).

Se trata de un objeto que puede ser consultado desde el entorno web de MicroStrategy y, por poner un ejemplo, se muestra la Figura 4.60.

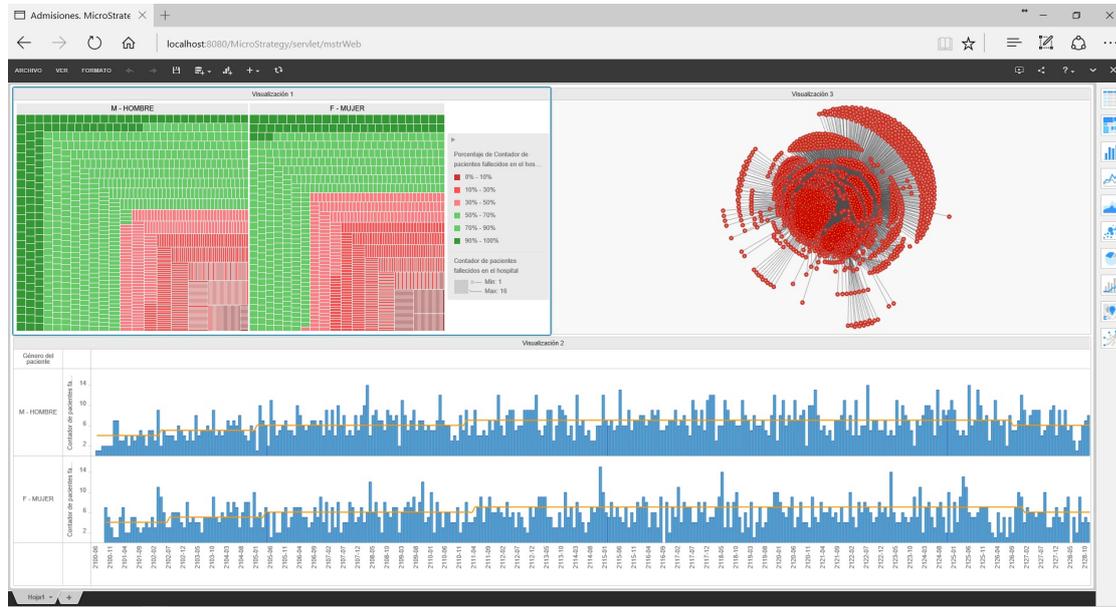


Figura 4.60: Composición de visualizaciones en un tablero de control.

Es fácil ver que, esta composición, permite obtener una visión general de más de un informe o, incluso (como se da en este caso), con tres aspectos o visualizaciones diferentes (gráficos principalmente) aun tratándose de la información de sólo dos informes de partida (previamente definidos en MicroStrategy Developer).

Se puede seguir disfrutando de la misma funcionalidad de navegación, por ejemplo, tras la composición de informes, de manera que los informes que conforman el tablero de control son totalmente dinámicos de cara a usuario.

4.3 Creación de objetos del proyecto

4.3.1 Filtros de seguridad

Como se ha visto anteriormente, un tipo de objeto público son los filtros, que permiten condicionar los resultados devueltos por un atributo al ser explorado, por

un informe al ser resuelto, por un indicador al ser calculado, etc. Pero, además, este tipo de objeto público puede ser utilizado como una garantía de seguridad de manera que condicione las opciones de consulta en la herramienta.

De esta manera, si un determinado filtro previamente diseñado en Developer es asociado a uno o más usuarios de acceso, éstos se verán, irremediablemente, condicionados por dicho filtro a la hora de utilizar el proyecto.

Hasta el momento, en todas las ocasiones se ha estado empleando un usuario ‘Administrator’, de manera que la función desempeñada era la de un arquitecto que está diseñando la estructura básica del proyecto para que éste pueda ser utilizado posteriormente por usuarios con menor autorización (usuarios de sólo consulta, o usuarios que puedan definir objetos públicos pero jamás objetos de esquema, por ejemplo).

Sin embargo, supongamos que se define un nuevo tipo de usuario que, como decimos, dispondrá de funcionalidades de más alto nivel, menos ligadas al diseño del proyecto y más ligadas a su explotación. Imaginemos que dicho usuario es un sanitario concreto.

La creación de dicho usuario se realizará desde el módulo de administración localizado en el origen de proyecto ‘MIMIC3_ORIGEN_PROYECTO_3LEVELS’ mostrado en la Figura 4.61.

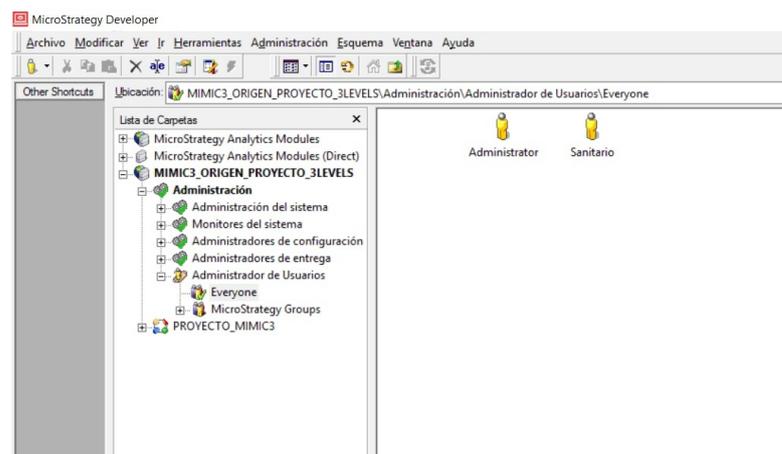


Figura 4.61: Administración de usuarios del proyecto en MicroStrategy.

Es posible configurar un filtro que, necesariamente, vaya asociado a dicho usuario de la plataforma de manera que, bien sin su conocimiento o bien habiendo

sido dado a conocer, el usuario lleve siempre asociado. Para ello, se accede a la sección de Filtro de seguridad de dicho usuario como indica la Figura 4.62.

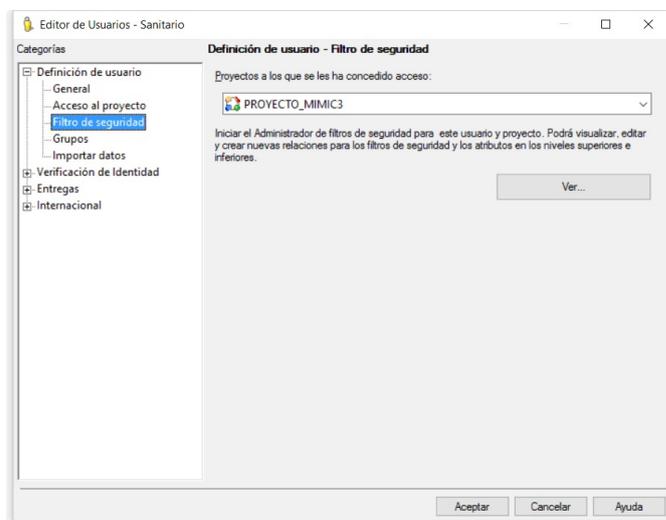


Figura 4.62: Mantenimiento del filtro de seguridad de un usuario concreto.

Como muestra la Figura 4.63, se asociará un determinado filtro que se haya elaborado con anterioridad.

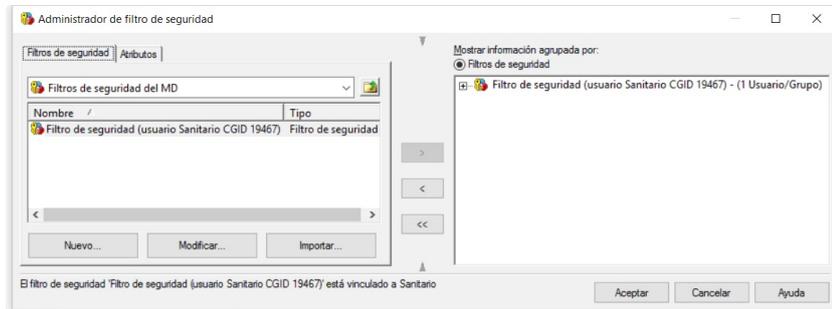


Figura 4.63: Selección y asociación de un filtro de seguridad concreto.

Puede que, por ejemplo, restrinja la información a consultar a los datos asociados al sanitario identificado mediante $CGID = 19467$.

Se trata de una garantía de que, dicho usuario, no podrá consultar nada ajeno a su ámbito de actuación (por una parte) pero, además, simplifica la labor de consulta de dicho sanitario (que podrá prescindir de recordar, en todo momento, condicionar todas sus consultas por los datos que únicamente le corresponden a él. Por tanto, un filtro de seguridad es muy práctico.

Los efectos de la actuación de este filtro de seguridad pueden hacerse constar, por ejemplo, si se contrasta la información devuelta por un determinado informe ejecutado con el usuario ‘Administrator’ (que, por lo tanto, no conllevará filtrado implícito ninguno, consultando la totalidad del warehouse). Por ejemplo, en caso de consultar (con usuario ‘Administrator’, es decir, sin restricciones sobre datos) las mediciones de signo vital realizadas a lo largo del año 2149 desglosadas por su finalización (errónea o no), se obtiene la Figura 4.64.

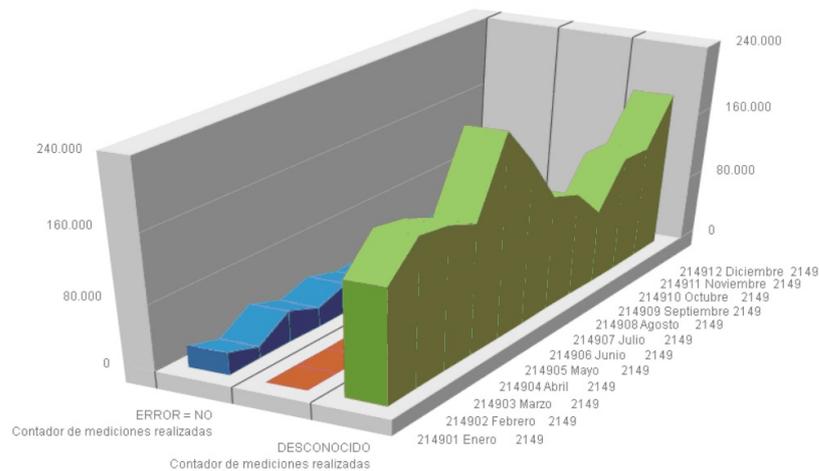


Figura 4.64: Informe resuelto sin restricciones con usuario ‘Administrator’.

La sentencia SQL asociada se muestra en l Figura 4.65.

```

select  a11.CHARTTIME_YYYYMM CHARTTIME_YYYYMM,
        a12.CHARTTIME_MONTHYYYY CHARTTIME_MONTHYYYY,
        a11.ERROR ERROR,
        DECODE(a11.ERROR, 0, 'ERROR = NO', 1, 'ERROR = SI', NULL, 'DESCONOCIDO') ERROR_TEXT,
        count(a11.ITEMID) WJXBFS1
from    CHARTEVENTS a11
join    LU_CHARTIME_YYYYMM a12
on      (a11.CHARTTIME_YYYYMM = a12.CHARTTIME_YYYYMM)
where   TO_NUMBER(SUBSTR(TO_CHAR(a12.CHARTTIME_YYYYMM), 1, 4)) = 2149
group  by a11.CHARTTIME_YYYYMM,
          a12.CHARTTIME_MONTHYYYY,
          a11.ERROR,
          DECODE(a11.ERROR, 0, 'ERROR = NO', 1, 'ERROR = SI', NULL, 'DESCONOCIDO')
    
```

Figura 4.65: Consulta SQL del informe resuelto sin filtro de seguridad.

Si ese mismo informe (a igualdad de parametrización) es ejecutado por el usuario ‘Sanitario’, los datos obtenidos son los mostrados en la Figura 4.66.

En la Figura 4.67 se puede observar el filtro soportado por el informe de manera implícita en la visión SQL de los resultados.

Puede demostrarse, mediante la navegación hacia el atributo ‘Sanitario’ desde el informe resuelto por el usuario ‘administrator’, que el valor devuelto por el informe

Año y mes de toma de la medida	Indicadores	Indicador de error	ERROR = NO
		Contador de mediciones realizadas	Contador de mediciones realizadas
214909	Septiembre 2149		4

Figura 4.66: Informe resuelto con filtro de seguridad con usuario ‘Sanitario’.

```

select  a11.CHARTTIME_YYYYMM CHARTTIME_YYYYMM,
        a12.CHARTTIME_MONTHYYYY CHARTTIME_MONTHYYYY,
        a11.ERROR_ERROR,
        DECODE(a11.ERROR, 0, 'ERROR = NO', 1, 'ERROR = SI', NULL, 'DESCONOCIDO') ERROR_TEXT,
        count(a11.ITEMID) WJXBFS1
from    CHARTEVENTS a11
join    LU_CHARTTIME_YYYYMM a12
      on (a11.CHARTTIME_YYYYMM = a12.CHARTTIME_YYYYMM)
where   (TO_NUMBER(SUBSTR(TO_CHAR(a12.CHARTTIME_YYYYMM), 1, 4)) = 2149
and a11.CGID = 19467)
group by a11.CHARTTIME_YYYYMM,
        a12.CHARTTIME_MONTHYYYY,
        a11.ERROR_ERROR,
        DECODE(a11.ERROR, 0, 'ERROR = NO', 1, 'ERROR = SI', NULL, 'DESCONOCIDO')

```

Figura 4.67: Condicionamiento implícito debido al filtro de seguridad.

condicionado por el filtro de seguridad del usuario ‘sanitario’ (correspondiente al identificador de Sanitario 19467) es correcto. La Figura 4.68 lo demuestra.

Año y mes de toma de la medida	Sanitario	Indicador de error	ERROR = NO	ERROR = Sí	DESCONOCIDO
			Contador de mediciones realizadas	Contador de mediciones realizadas	Contador de mediciones realizadas
19412	RRT Read Only				1
19451	RN				214
19467	MD Attending		4		
19523	PCT PCT/NA		8		
19525	RN RN				398
19562	RN RN		62		
19596	RN RN		364	3	
19599	RNC Read Only				684
19610	RN RN		8		60
19611	RN RN		161		
19612	RN				1.181
19637	RRT Respiratory				122
19650	RN RN		117		
19653	RRT Respiratory				41
19667	CoWker PCT/NA				6
19681	RN RN		53		
19689	RN RN		102		
19747	RRT Respiratory		18		37
19755	RT Respiratory		284		
19761	UCO UCO		9		
19766	Rehab Rehabilitation		18		

Figura 4.68: Comprobación de datos mediante el uso de navegación.

4.4 Cálculo predictivo

4.4.1 Métricas de formación

El conocimiento de los datos reales es fundamental en la toma de decisiones, siendo el principal uso que se espera de este tipo de explotaciones de datos. Si, además, se les dota de dinamismo mediante la facilidad de construcción de informes o la posibilidad de indagación mediante la utilidad de la navegación, comienza a

proporcionársele la capacidad de análisis de datos o DataMining. Si, por último, la propia herramienta es capaz de establecer esas relaciones, nexos o predicciones, en lugar de proporcionar la información base para que las previsiones las haga el usuario, se proporciona al usuario la capacidad de únicamente tomar las decisiones, sin apenas necesidad de análisis de la información.

Para ello, MicroStrategy posibilita la definición de una serie de indicadores o métricas diferentes, en el sentido de que no operarán y calcularán datos sobre los datos ya existentes, sino que proyectarán los datos ya existentes a futuro, siendo capaces de establecer tendencias próximas pero de las que aún no se dispone de información.

Para poner un ejemplo, se tratará de establecer un comportamiento futuro de una de las métricas ya analizadas en presente hasta el momento y que puede ser de mayor interés para valorar el correcto funcionamiento de una unidad de cuidados intensivos: el número de pacientes fallecidos.

Para ello, se comenzará con la definición de una métrica de formación a partir del asistente de este tipo de métricas, diferenciadas de las habituales, como refleja la Figura 4.69.



Figura 4.69: Comienzo del asistente de indicador de formación.

Será necesario, en primer lugar, qué clase de indicador de formación va a crearse

de entre los disponibles en la aplicación, siendo la elección en esta ocasión una regresión exponencial, como se muestra en la Figura 4.70.

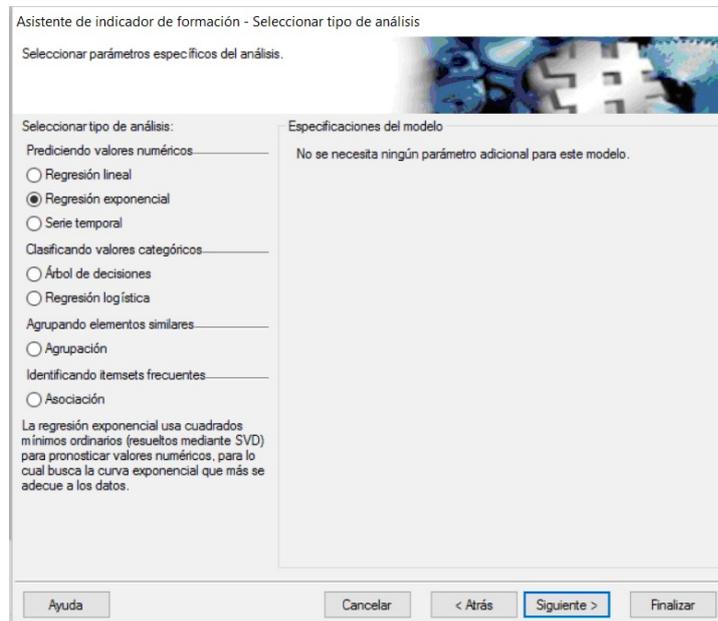


Figura 4.70: Selección del tipo de análisis a realizar.

A continuación, se detallarán los indicadores independientes (en este caso se trataría del ‘Contador de pacientes’, variable a partir de la cual se supondrá que la métrica dependiente tiene una relación basada en una función) y los indicadores dependientes (en este caso de tratará de ‘Contador de pacientes fallecidos en el hospital’, métrica que se cree dependiente de la anterior y cuyo valor futuro parece poder ser pronosticable), como se indica en la Figura 4.71.

En último lugar, como muestra la Figura 4.72, se especificará qué indicador o indicadores se desea obtener como resultado de la elaboración de este estudio predictivo.

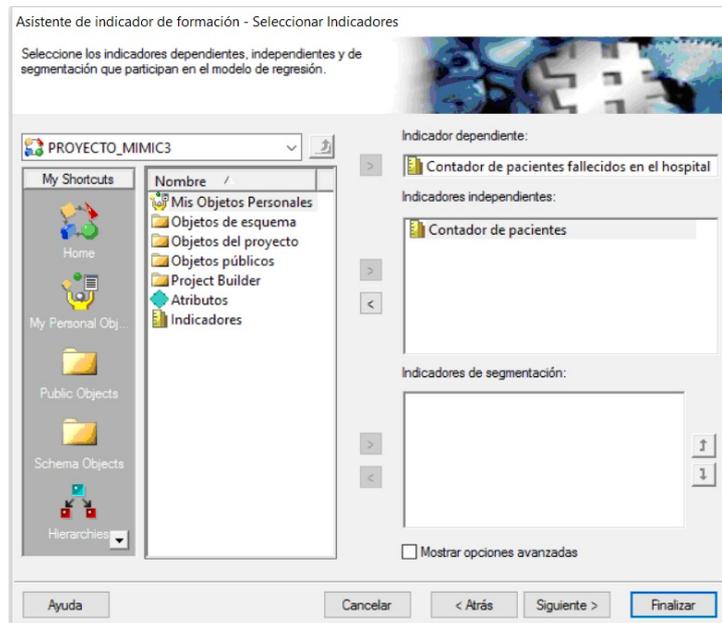


Figura 4.71: Selección de variables dependientes e independientes para el análisis.

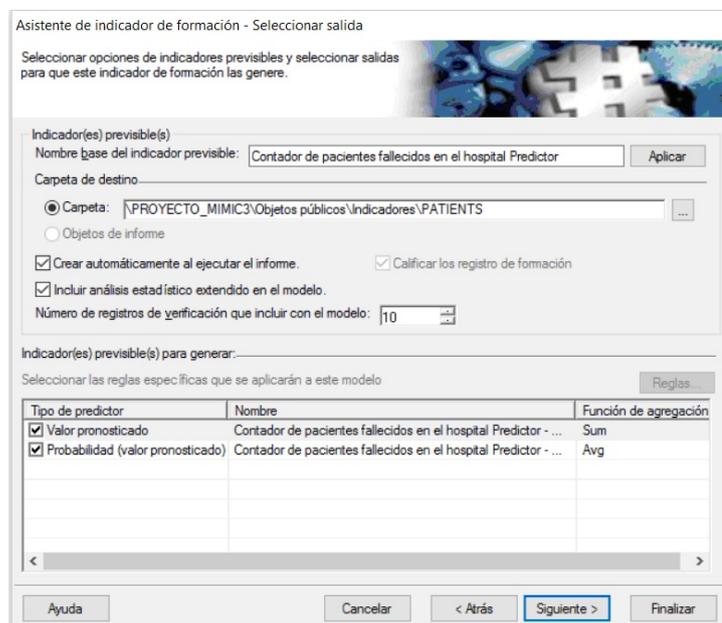


Figura 4.72: Especificación de objetos resultantes del indicador de formación.

Esto irá seguido, como indica la Figura 4.73, de un resumen de la parametrización de esta definición.

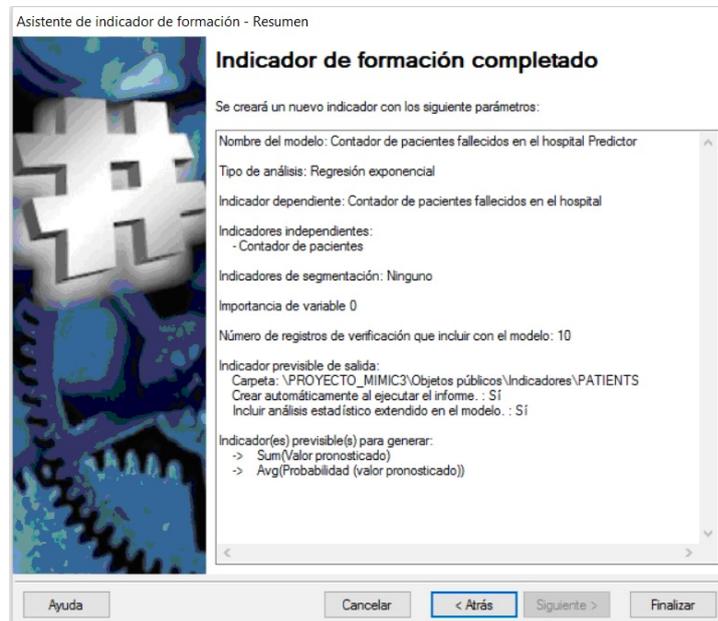


Figura 4.73: Resumen de la definición del indicador de formación.

Con esta parametrización, se creará un indicador llamado ‘Probabilidad (valor pronosticado)’ que dispondrá de la información reflejada en la Figura 4.74.

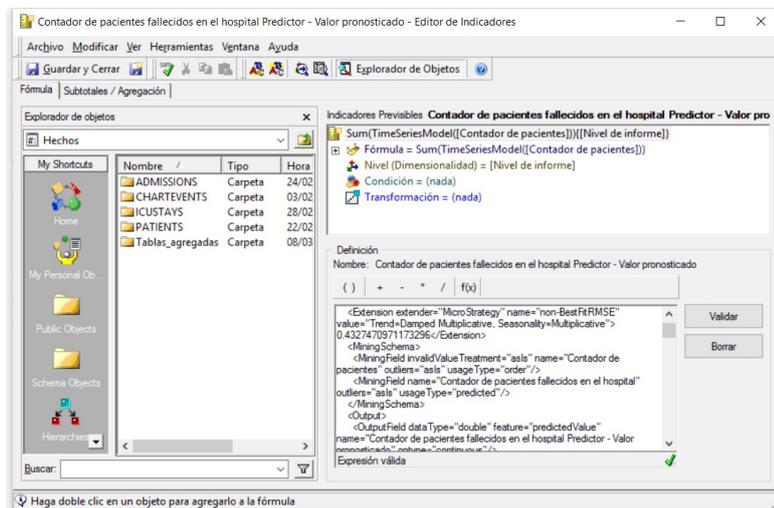


Figura 4.74: Contenido interno del indicador probabilístico automáticamente creado.

Realmente, consistirá en un cubo de coeficientes, valores calculados en base a la regresión solicitada. Se podrá extraer información acerca de este indicador de formación a través de la funcionalidad de ‘Ver modelo previsible’, como ver el modelo en sí tal y como refleja la Figura 4.75.

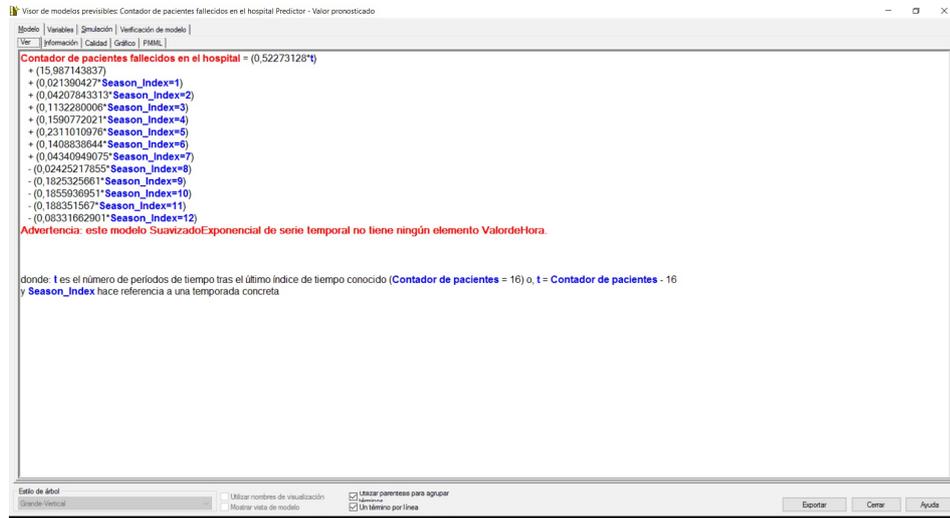


Figura 4.75: Cubo de coeficientes resultantes de la regresión.

También se podrá consultar su supuesta calidad como indica la Figura 4.76.

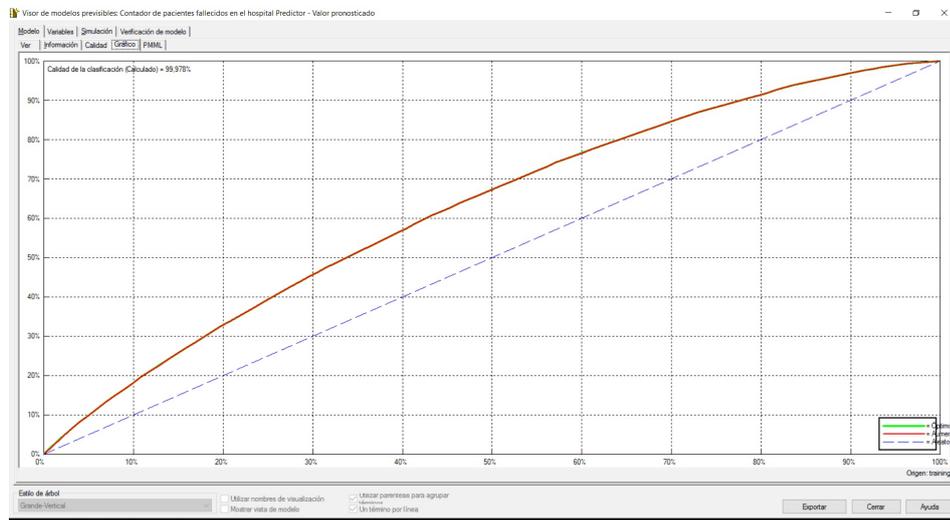


Figura 4.76: Visor de la calidad de los coeficientes obtenidos.

Tal vez lo más interesante sea poder exportar a formato PMML (estándar de intercambio de modelos predictivos) este modelo para su uso por otras herramientas de análisis estadístico más especializadas, como se muestra en la

y pronosticados, lo que aporta garantía al modelo predictivo. En caso de querer ser más exhaustivos y no ser suficiente con una visualización gráfica, bastaría con transformar la apariencia de esta ejecución de informe de Gráfica a Cuadrícula, aportando incluso un indicador nuevo al informe (calculado en el interior de este mismo informe mediante la utilidad de ‘Insertar cálculo’, no siendo necesario definir un nuevo objeto) que calcule el porcentaje de distancia entre la métrica real y la pronosticada y, adicionalmente, estableciendo un umbral sobre dicha métrica porcentual que provoque que las celdas con un valor que despunte (superior al 25 %) sean coloreadas de amarillo, como se ve en la Figura 4.79.

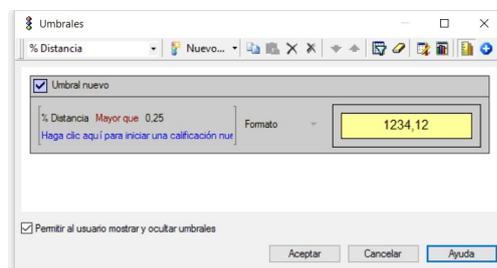


Figura 4.79: Definición de un umbral sobre el porcentaje de desviación.

El resultado del informe en aspecto de cuadrícula con este indicador ‘umbral’ de aviso ya definido e incorporado sería el indicado en la Figura 4.80.

Género del paciente: F - MUJER		Indicadores	Contador de pacientes fallecidos en el hospital	Predicador del contador de pacientes fallecidos	% Distancia
F - MUJER					
F - MUJER					
M - HOMBRE					
Total					
Año y mes de fallecimiento del paciente					
SEPTEMBER	2100		7	6,38	8,86%
OCTOBER	2100		5	4,41	11,75%
NOVEMBER	2100		3	3,05	1,72%
DECEMBER	2100		1	2,11	111,04%
JANUARY	2101		1	2,11	111,04%
FEBRUARY	2101		5	4,41	11,75%
MARCH	2101		7	6,38	8,86%
APRIL	2101		2	2,54	26,89%
MAY	2101		2	2,54	26,89%
JUNE	2101		5	4,41	11,75%
JULY	2101		5	4,41	11,75%
AUGUST	2101		3	3,05	1,72%
SEPTEMBER	2101		2	2,54	26,89%
OCTOBER	2101		3	3,05	1,72%
NOVEMBER	2101		4	3,67	8,27%
DECEMBER	2101		3	3,05	1,72%
JANUARY	2102		5	4,41	11,75%
FEBRUARY	2102		3	3,05	1,72%
MARCH	2102		11	13,34	21,26%
APRIL	2102		9	9,22	2,50%
MAY	2102		6	5,31	11,57%
JUNE	2102		3	3,05	1,72%
JULY	2102		3	3,05	1,72%
AUGUST	2102		6	5,31	11,57%
SEPTEMBER	2102		6	5,31	11,57%
OCTOBER	2102		6	5,31	11,57%
NOVEMBER	2102		4	3,67	8,27%
DECEMBER	2102		3	3,05	1,72%

Figura 4.80: Informe resuelto contemplando el umbral de aviso por desviación.

4.5 Ejemplo de aplicación

4.5.1 Caso práctico

Con el fin de acreditar la utilidad de este tipo de sistemas, dada la facilidad de obtención de datos que proporciona, se va a proceder a representar un ejemplo práctico de directa aplicación.

Supongamos que el gerente de un hospital desea conocer qué porcentaje de los pacientes tratados por cada determinado asistente sanitario ha terminado falleciendo. Está claro que no será una relación de causa-efecto, dado que a un paciente le tratan varios asistentes y que un asistente trata a varios pacientes (es decir, la relación entre ambos atributos es de muchos a muchos), pero puede llegar a orientar sobre la calidad de los trabajos realizados. A continuación se relatan los pasos a seguir en el sistema (en este caso empleando MicroStrategy Developer como canal) para llegar a obtener dicha información.

En primer lugar, se tratará de crear un indicador que calcule el número de diferentes Pacientes tratados, indistintamente de si éstos han fallecido o no. Para ello, se empleará como base del indicador el atributo Paciente, se empleará la función ‘Count’ con la opción ‘Distinct’, y se guardará dicho indicador, tal y como muestra la Figura 4.81.

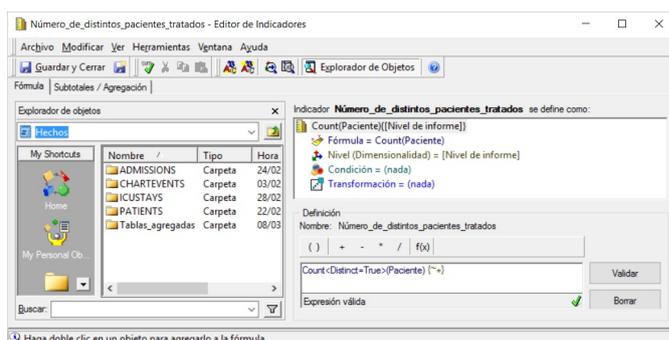


Figura 4.81: Indicador de distintos pacientes tratados.

En segundo lugar, se optará por replicar (mediante copia, por ejemplo) este primer indicador pero, en esta ocasión, se le aplicará un filtro del que ya se disponía con anterioridad, que filtraría únicamente por Pacientes que ya han fallecido. La definición de este segundo indicador queda reflejada en la Figura 4.82.

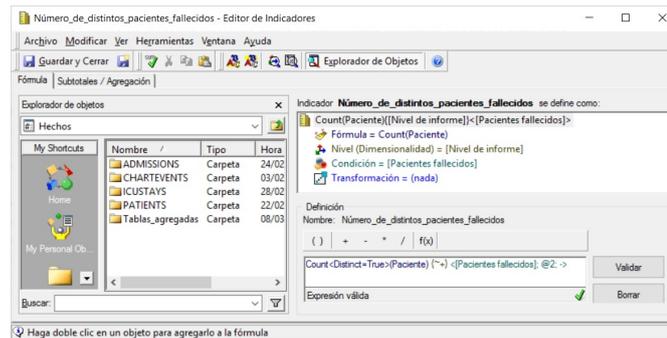


Figura 4.82: Indicador de distintos pacientes fallecidos.

En tercer lugar, se definirá un nuevo indicador que calcule el cociente de los dos primeros. En la Figura 4.83 se puede observar la definición de este tercer indicador.

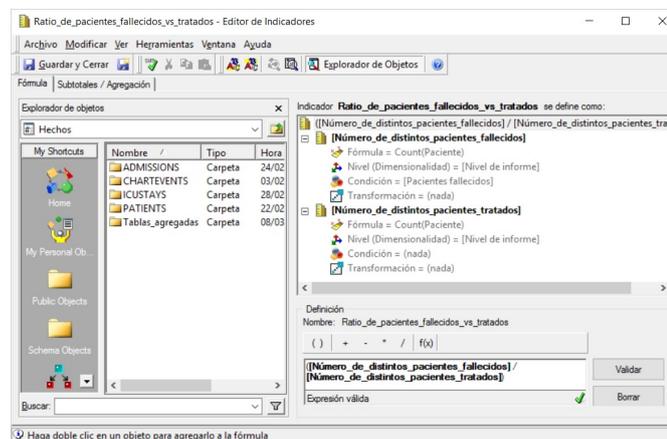


Figura 4.83: Indicador de tasa de Pacientes fallecidos vs. tratados.

A continuación, bastará con definir un informe que, en su plantilla, muestre este tercer indicador enfrentado al atributo Sanitario (para que el cálculo se realice con el criterio de agrupación de cada Sanitario independientemente de los demás, sin agrupación). Además, se optará por alterar la visualización de los datos, indicando que la tasa se visualice en porcentaje con dos decimales, y que las filas del informe sean proporcionadas en orden ascendente de resultados del indicador. La Figura 4.84 muestra cómo sería la definición de este informe (que se mostrará en formato de cuadrícula).

La ejecución de este informe no solicitará en absoluto ningún condicionante por parte del gerente del hospital, sino que atacará toda la información sin filtrar

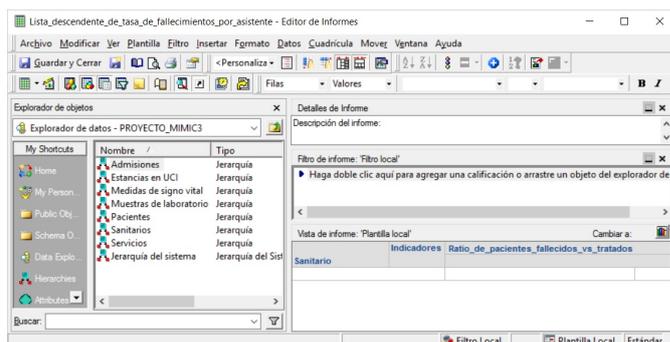


Figura 4.84: Definición del informe de interés.

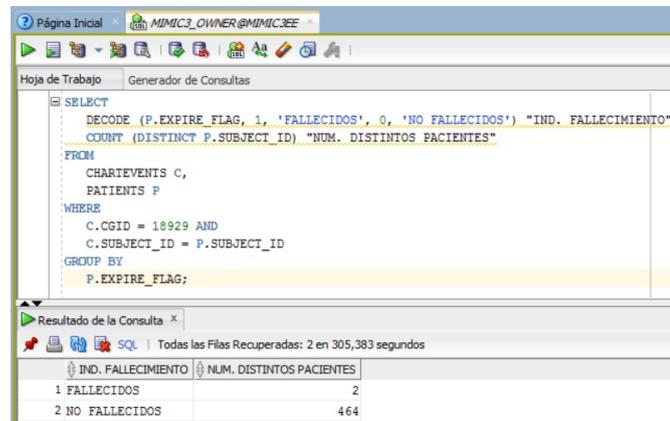
por nada. Los resultados de esta ejecución pueden observarse en la Figura 4.85.

Sanitario	Indicadores	Ratio_de_pacientes_fallecidos_vs_tratados
19371 RN		0,18%
20680 RN	Read Only	0,18%
15393 RN		0,20%
17206 RN		0,20%
19137 RN		0,23%
18789 RN		0,23%
16864 rn		0,25%
17671 RNC		0,26%
15986 RN		0,26%
14828 RN	Read Only	0,27%
15931 RN		0,29%
14927 RN		0,31%
18433 RN		0,32%
17125 RN		0,36%
19836 RN	Read Only	0,37%
19271 RN	Read Only	0,39%
20296 RN		0,40%
16833 RN		0,40%
17919 RN		0,41%
16914 RN		0,42%
18929 RN		0,43%
17447 RN	Read Only	0,43%

Figura 4.85: Resultados de ejecución del informe.

Para comprobar que los datos visualizados en el informe son correctos, se tomará como muestra el Paciente mostrado en último lugar en la anterior figura y, con la herramienta SQL Developer, se atacará directamente la base de datos Oracle del Warehouse y, mediante una consulta SQL libre (atacando la tabla 'CHARTEVENTS' de medidas de signos vitales, la tabla 'PATIENTS' de Pacientes, filtrando por el flag de expiración 'EXPIRE_FLAG' que identifica los Pacientes ya fallecidos, etc.), se obtendrá la cifra de Pacientes que han sido

atendidos por el sanitario en cuestión, así como cuántos de ellos han fallecido. La Figura 4.86 muestra los resultados que demuestran que el cálculo es correcto.



The screenshot shows the Oracle SQL Developer interface. The top pane displays the following SQL query:

```
SELECT  
  DECODE (P.EXPIRE_FLAG, 1, 'FALLECIDOS', 0, 'NO FALLECIDOS') "IND. FALLECIMIENTO",  
  COUNT (DISTINCT P.SUBJECT_ID) "NUM. DISTINTOS PACIENTES"  
FROM  
  CHARTEVENTS C,  
  PATIENTS P  
WHERE  
  C.CGID = 18929 AND  
  C.SUBJECT_ID = P.SUBJECT_ID  
GROUP BY  
  P.EXPIRE_FLAG;
```

The bottom pane, titled "Resultado de la Consulta", shows the results of the query. It indicates that 2 rows were recovered in 305,383 seconds. The results are as follows:

IND. FALLECIMIENTO	NUM. DISTINTOS PACIENTES
1 FALLECIDOS	2
2 NO FALLECIDOS	464

Figura 4.86: Comprobación de resultados en Oracle.

Por tanto, queda acreditada la sencillez con la que, alguien ajeno al conocimiento técnico (pero sí conocedor del negocio), puede obtener datos con unos escasos pasos.

Capítulo 5

Conclusiones

Una vez desarrollado este proyecto basado en la aplicación de las tecnologías de la información en el ámbito de la salud, concretamente estableciéndose un elaborado estudio de Business Intelligence sobre una base de datos de pacientes que han ido practicando admisiones en diferentes hospitales, es posible destacar un ejemplo de presupuesto para una configuración más o menos genérica, obtener una serie de conclusiones, así como de futuras líneas de mayor profundización.

Por la parte de un presupuesto general que permitiese la disposición de un sistema similar al desarrollado, se deben diferenciar costes de desarrollo, costes de infraestructura hardware y coste de software, básicamente. Mientras que no estudiaremos el coste de desarrollo (personal técnico cualificado para poder llevar a cabo las tareas de análisis, desarrollo o implantación, muy variable en base a la dificultad técnica o funcional del sistema concreto) ni de infraestructura (igualmente, variará mucho de si se dispone de un servidor básico a si se dispone de las últimas máquinas Exadata existentes en el mercado, especialmente dirigidas a estos fines de análisis de información), sí que es posible identificar unos gastos más o menos constantes en las licencias de software.

Por la parte de Oracle, el coste de la licencia de una versión Enterprise Edition como la empleada para este proyecto con uso indefinido (dado que es posible contratarla con fecha de vencimiento) será del entorno de los 50.000 euros (siempre partiendo de una configuración básica de usuarios, de procesadores del

servidor, de contrato de mantenimiento...). Conviene contratar, adicionalmente, un servicio de soporte y actualizaciones valorado en aproximadamente 11.000 euros, así como algún componente básico para Datawarehouse como será la capacidad de particionamiento (con un coste aproximado de 12.000 euros adicionales). Queda perfectamente reflejado en el siguiente link (precios en dólares, negociables en España, etc.):

<http://www.oracle.com/us/corporate/pricing/technology-price-list-070617.pdf>

Por la parte de MicroStrategy, suele licenciarse en conjuntos de componentes que, habitualmente, serán utilizados en la gran generalidad de desarrollos. Un paquete compuesto por un software adecuado para el desarrollo en desktop (totalmente necesario para la parte de arquitecto del sistema, pero también muy empleado por usuarios avanzados que no ven cubiertas sus necesidades con las funcionalidades web) rondaría los 6.000 euros, mientras que un paquete compuesto por cinco licencias de explotación (más destinadas a usuarios de consulta, que explotan la información y sólo puntualmente diseñan, a lo más algún informe vía web) rondaría los 10.000 euros. Estos precios se ven reflejados en la tabla de precios presente en la Figura 5.1. Estos precios son altamente revisables en base a qué proveedor realice el desarrollo y su relación con dicho fabricante.

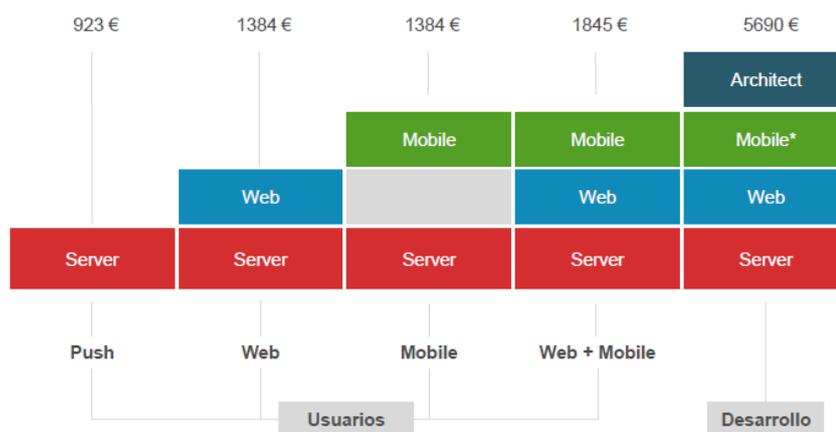


Figura 5.1: Costes de licencias de MicroStrategy.

En lo relativo a las conclusiones obtenidas de este trabajo, hay que destacar la

importancia que presenta en proyectos de Business Intelligence la correcta toma de conocimientos funcionales acerca del negocio que va a tratar de explotarse. Un incorrecto conocimiento acerca del funcionamiento de dicho negocio conllevará la incorrecta toma de decisiones en la etapa de modelización y, dándose esta coyuntura, el proyecto se verá abocado al fracaso, dado que una incorrecta modelización implica una incorrecta explotación. Es, por tanto, fundamental aplicar el tiempo necesario al entendimiento de la manera de relacionarse las distintas secciones del negocio. Sin duda, en este proyecto concreto ésta ha sido una de las etapas en las que se ha invertido un mayor esfuerzo: comprender cómo se relacionaban pacientes, admisiones, servicios, estancias en la UCI o sanitarios, así como la manera en la que la información relativa a cada uno de estos actores era almacenada en los ficheros de partida.

Por otra parte, este proyecto ha sido una línea de investigación acerca de las posibilidades de explotación que una herramienta como MicroStrategy puede presentar sobre unos datos de esta categoría. Aunque los datos hubieran sido manipulados en origen con el fin de frustrar su posible seguimiento o la elaboración de estadísticas con sentido, dando lugar a series temporales ilógicas, o aunque no se haya partido de una colección concreta de informes requeridos a proporcionar y simplemente se hayan elaborado ejemplos de lo que puede llegarse a alcanzar, queda patente que es posible establecer una plataforma de análisis que, gracias al tiempo dedicado por el arquitecto de la aplicación, simplifique el posterior diseño de informes por parte del usuario final y agilice la velocidad de respuesta de éstos. Queda patente, consultando el espacio de foros relacionados con Mimic, la continua dedicación por parte de los analistas a redactar programas o sentencias SQL con el fin de obtener un estudio concreto, algo que una plataforma como ésta ha quedado patente que puede realizar mediante una rápida definición y sin conocimientos de programación. Se trata de herramientas que posibilitan la abstracción del nivel más bajo de la arquitectura.

Es muy importante destacar el amplio alcance que el Business Intelligence presenta en la actualidad en nuestra sociedad. Se trata de una tecnología continuamente demandada, donde es necesaria cada vez más capacidad humana,

y que no para de crecer. Es un área laboral que siempre requiere de gente preparada y cuyos especialistas deben mantenerse continuamente actualizados, dada la velocidad de vértigo a la que evolucionan tanto las herramientas de análisis como las funcionalidades que se espera de ellas por parte de los clientes.

A futuro, es evidente que el internet de las cosas será un escenario de gran aplicación de tecnologías de explotación como ésta. El presente auge del Big Data conllevará una elevada demanda de personal cualificado (de hecho, ya lo hace), siendo por tanto muy importante el aprendizaje en estas materias. Animo personalmente a la formación en el ámbito de Business Analytics, dado que se trata de un área en el que existe una gran expectativa laboral y que, además, es muy gratificante a nivel personal dados los palpables resultados que el profesional percibe.

Capítulo 6

Planificación del proyecto

La elaboración del proyecto se ha desarrollado por una única persona a lo largo de un período de tiempo natural de aproximadamente cinco meses.

Puesto que el proyecto ha consistido en una serie de etapas todas ellas secuenciales, dependientes cada una de la anterior, y con ciertos períodos de descanso (debidos a períodos vacacionales, enfermedad...), el diagrama que refleja la progresión de cada una de estas etapas a lo largo del calendario sería el mostrado en la Figura 6.1.

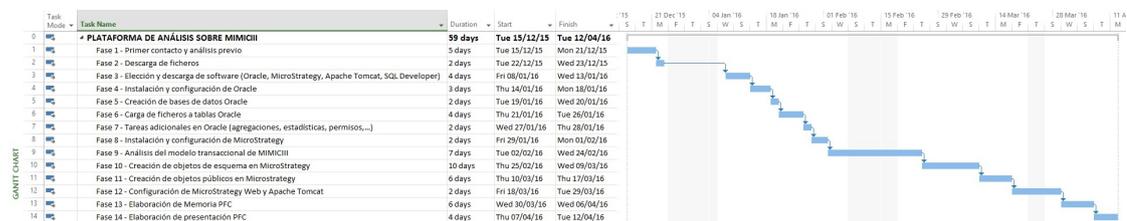


Figura 6.1: Diagrama de Gantt del proyecto.

Bibliografía

- [1] MicroStrategy (2015). *MicroStrategy 10.1 Advanced Reporting Guide*. MicroStrategy Incorporated.
<http://www2.microstrategy.com/producthelp/10.2/manuals/en/AdvancedReporting.pdf>
- [2] — (Agosto de 2015). *MicroStrategy 10.1 Project Design Guide*. MicroStrategy Incorporated.
<http://www2.microstrategy.com/producthelp/10.2/manuals/en/ProjectDesign.pdf>
- [3] — (Diciembre de 2015). *MicroStrategy 10.2 Installation and Configuration Guide*. MicroStrategy Incorporated.
<http://www2.microstrategy.com/producthelp/10.2/manuals/en/InstallationConfig.pdf>
- [4] Oracle (Enero de 2015). *Oracle Database Quick Installation Guide, 11g Release 2 (11.2) for Microsoft Windows x64 (64-Bit)*. Oracle.
https://docs.oracle.com/cd/E11882_01/install.112/e49703.pdf
- [5] — (Junio de 2015). *Oracle Database SQL Developer Installation Guide (Release 4.1)*. Oracle.
https://docs.oracle.com/cd/E55747_01/doc.41/e57986.pdf

Anexo A

Descarga e instalación de Oracle

A.1 Descarga de Oracle

El primer paso a realizar consistirá en darse de alta en el sitio web general de Oracle, de manera que sea posible identificarse posteriormente frente a este proveedor mediante un determinado usuario (que consistirá en la dirección de correo electrónico que se proporcione durante el proceso de alta o registro) y su correspondiente contraseña.

Para ello, bastará con acudir a la página web asociada a la siguiente URL:

<https://profile.oracle.com/myprofile/account/create-account.jsp>

y rellenar la información requerida. Ésta se muestra en la Figura A.1.

The image shows a screenshot of a web browser displaying the Oracle account creation page. The browser's address bar shows the URL 'profile.oracle.com/myprofile/account/create-account.jsp'. The page has a red header with the Oracle logo. Below the header, the text reads 'Crear una cuenta en Oracle' and 'La cuenta Oracle le permitirá acceder a varios servicios, aplicaciones y comunidades en línea. Cuando cree su cuenta, puede elegir si desea recibir comunicaciones de Oracle y registrarse a suscripciones.' There is a section for 'Información necesaria' with several input fields: 'Dirección de correo electrónico*', 'Contraseña*', 'Volver a escribir contraseña*', 'País*' (with a dropdown menu), 'Nombre*' (with sub-fields for 'Nombre de pila' and 'Apellido'), 'Cargo*', 'Teléfono de trabajo*', 'Número de empresa*', 'Dirección*', 'Ciudad*', and 'Código postal*'. At the bottom of the form, there is a red button labeled 'Crear una cuenta'.

Figura A.1: Formulario de alta en la web de Oracle.

Aunque no conlleva ningún compromiso, este alta o registro será necesario para poder ser identificados (igualmente, sin ningún compromiso) a la hora de realizar descargas de productos desde este proveedor de software.

Una vez que se dispone de un usuario de acceso a las funcionalidades web que Oracle proporciona, para la instalación del producto ‘Oracle Database 11g Release 2’ basta con acudir a la página general de descargas de software de Oracle a través de la siguiente URL:

<http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index-092322.html>

El aspecto de esta página una vez cargada es el mostrado en la Figura A.2.

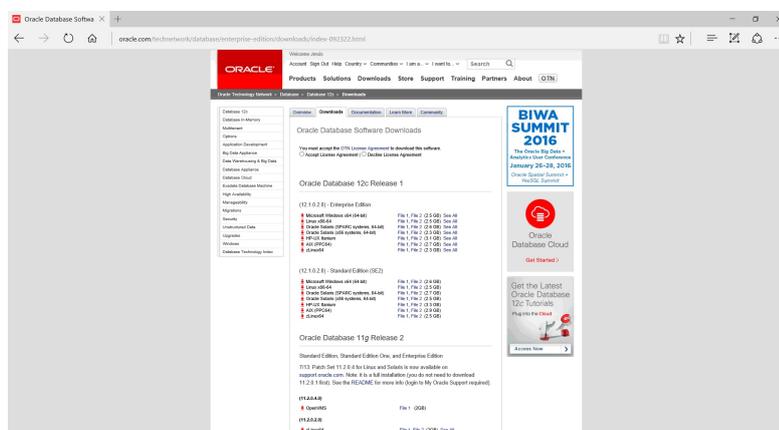


Figura A.2: Sitio web de descargas de Oracle.

Dado que se desea instalar la modalidad ‘Enterprise Edition’ correspondiente a la versión ‘Oracle Database 11g Release 2’ de Oracle, se centrará la atención en la sección de la página correspondiente a dicha versión, como refleja la Figura A.3.

En este caso, se va a realizar la instalación de dicho producto en un puesto local (ordenador portátil), dotado de un procesador de 64 bits (concretamente Intel(r) Core(tm) i7-6700HQ) y de un sistema operativo también de 64 bits (concretamente Windows 10 Home), como se ve en la Figura A.4.

Por lo tanto, interesa seleccionar la entrada existente en la lista para la versión 11.2.0.1.0 y para Microsoft Windows (x64).

Se trata, en realidad, de dos archivos (comprimidos), que bastará con descargar a alguna ubicación del puesto local. Seleccionando la opción de ‘See all’, se mostrará

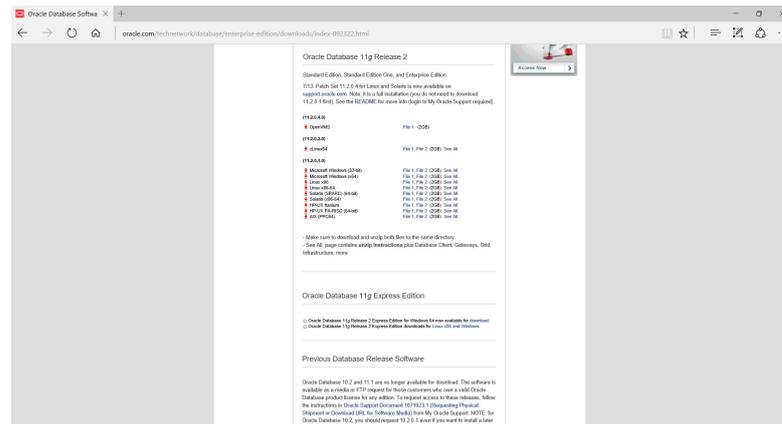


Figura A.3: Descargas disponibles de Oracle Database 11g Release 2.

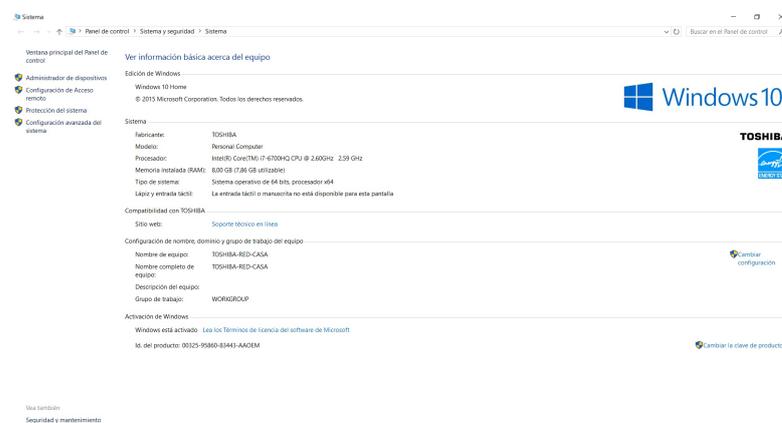


Figura A.4: Especificaciones del puesto de trabajo.

la página de descarga correspondiente a esta versión de producto concreta, como refleja la Figura A.5.

Aceptadas las condiciones de licencia, se seleccionará separadamente la descarga de los dos archivos correspondientes a ‘Oracle Database 11g Release 2 (11.2.0.1.0) for Microsoft Windows (x64)’, iniciándose la descarga del archivo. Este punto se ilustra en la Figura A.6.

Repitiendo este mismo proceso para ambos archivos, se dispondrá de dos archivos comprimidos de un elevado volumen de información depositados en el lugar correspondiente a las descargas de Windows y que, tras su traslado a la carpeta deseada definitiva, presentarán el aspecto de la Figura A.7.

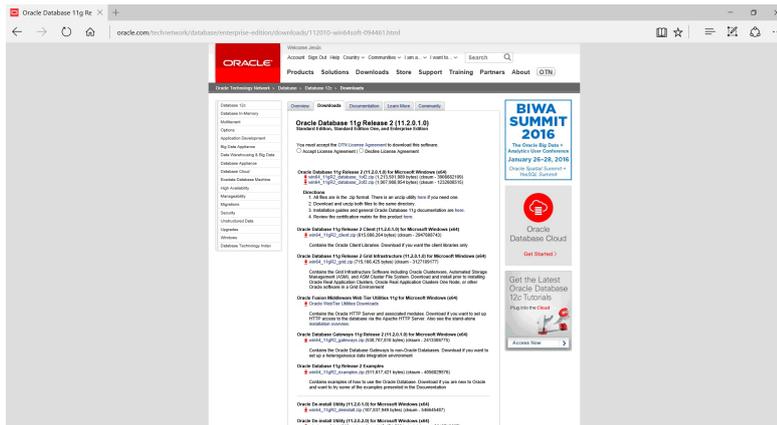


Figura A.5: Accesos directos a las descargas de archivos.

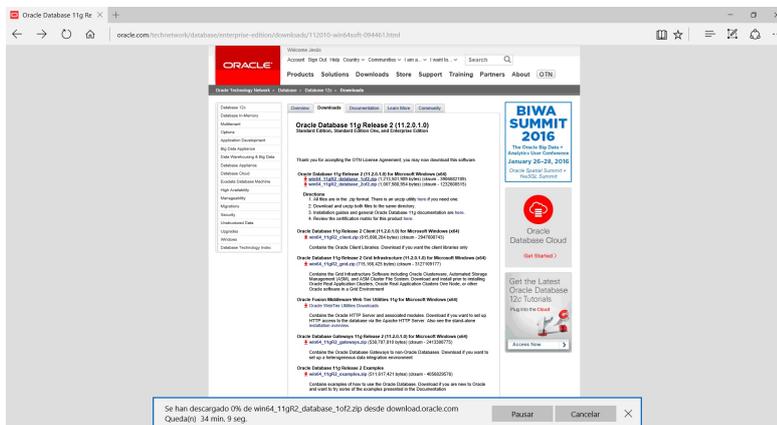


Figura A.6: Aspecto durante la descarga de archivos.

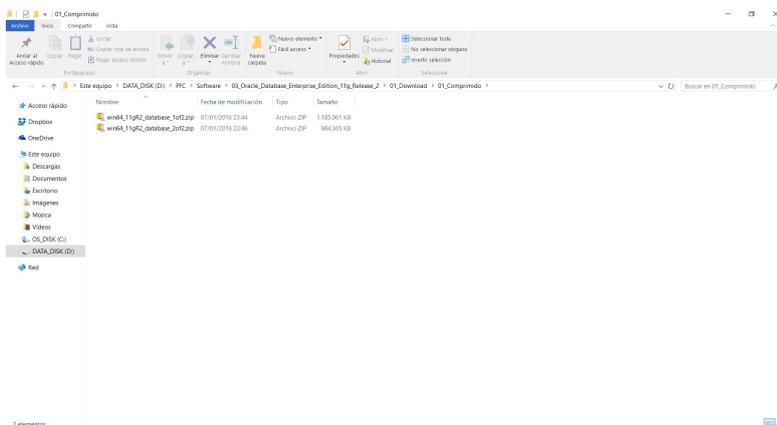


Figura A.7: Archivos de instalación de Oracle descargados.

Se requiere, a continuación, hacer uso de alguna herramienta de descompresión de estos archivos comprimidos con el fin de poder extraer los archivos originales (no comprimidos) de su interior.

En este caso, se empleará una versión gratuita (de evaluación) del conocido software de compresión y descompresión Winzip, como refleja la Figura A.8.



Figura A.8: Versión de evaluación de Winzip.

Se seleccionarán los archivos que se desea descomprimir, así como la ubicación elegida para la descarga de dicho archivo original, punto mostrado en la Figura A.9.

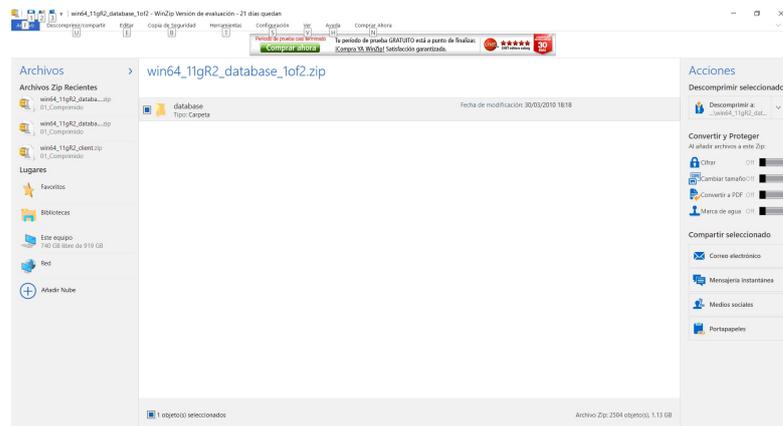


Figura A.9: Configuración de la descompresión de los archivos.

El resultado consistirá en una serie de carpetas y archivos que, por indicaciones de Oracle, deberán ser ubicadas en la misma ruta. Con esta indicación, a lo que Oracle se refiere es a que el primer archivo contiene una única carpeta llamada

‘database’ de la cual cuelga, por ejemplo, una subcarpeta llamada ‘stage’ (entre otras subcarpetas), como refleja la Figura A.10.

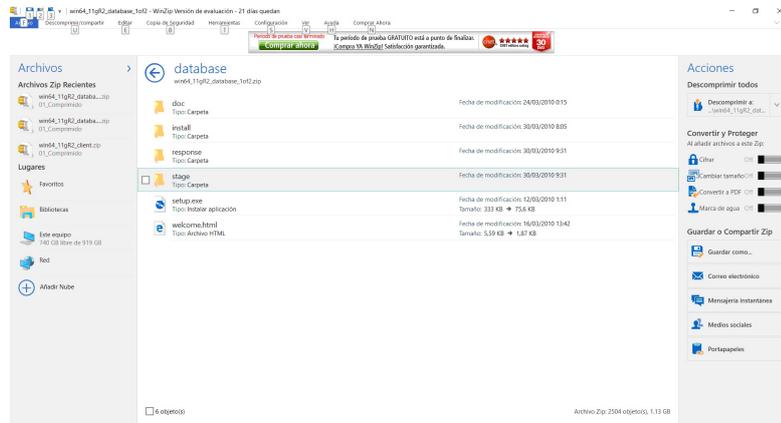


Figura A.10: Resultado de la descompresión del primer archivo.

Por su parte, el segundo archivo también contiene esa misma carpeta ‘database’ y esa misma subcarpeta ‘stage’, pero en este caso sola y únicamente la subcarpeta ‘stage’, como se ve en la Figura A.11.



Figura A.11: Resultado de la descompresión del segundo archivo.

Hay que ser cuidadoso y, por tanto, lograr que la descompresión de uno de los archivos comprimidos (el segundo que se descomprima) no sobrescriba lo ya descomprimido del otro archivo (el primero que ya fuera descomprimido con anterioridad), sino que, finalmente, se logre el árbol deseado de carpetas y archivos esperado en el que, la subcarpeta ‘stage’, estará compuesta por la unión (y esto es lo importante, lograr la fusión de componentes de los dos archivos) de contenidos de dicha subcarpeta ‘stage’ de ambos archivos.

La situación final del árbol de carpetas y ficheros, una vez terminada la descompresión de ambos archivos, quedará como muestra la Figura A.12.

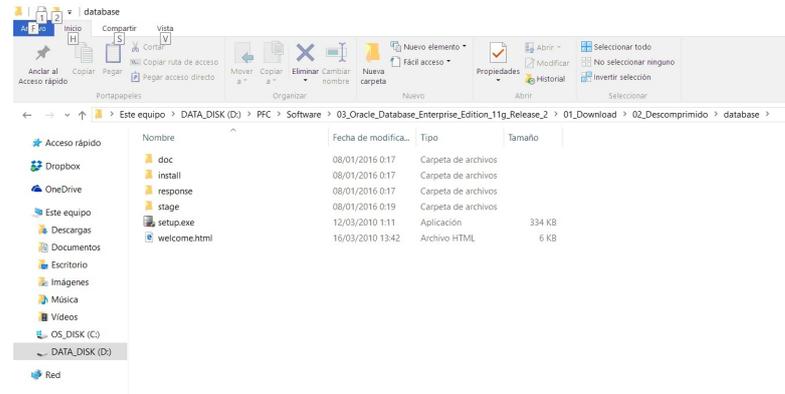


Figura A.12: Situación final tras la correcta descompresión.

A.2 Instalación de Oracle

Una vez disponibles en el puesto local los archivos descargados y descomprimidos correspondientes a la instalación de ‘Oracle Database 11g Release 2’, se trata de instalar el producto deseado de este gestor de base de datos.

Para ello, se ejecutará (con un usuario con capacidades de administración en el puesto local) el archivo ‘setup.exe’ como indica la Figura A.13.

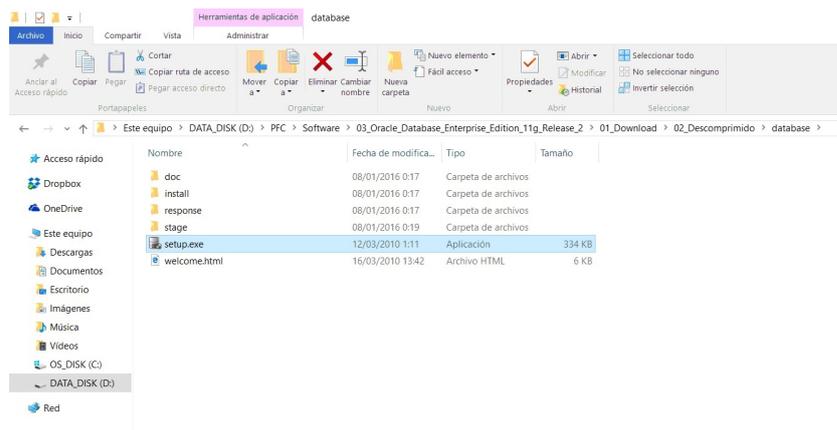


Figura A.13: Localización del archivo de instalación.

Windows advertirá de que se realizarán cambios en el PC debido a dicha instalación, algo que debe tolerarse para poder continuar. Una vez aceptada esta

advertencia del sistema operativo, la instalación arrancará de la manera que indica la Figura A.14.

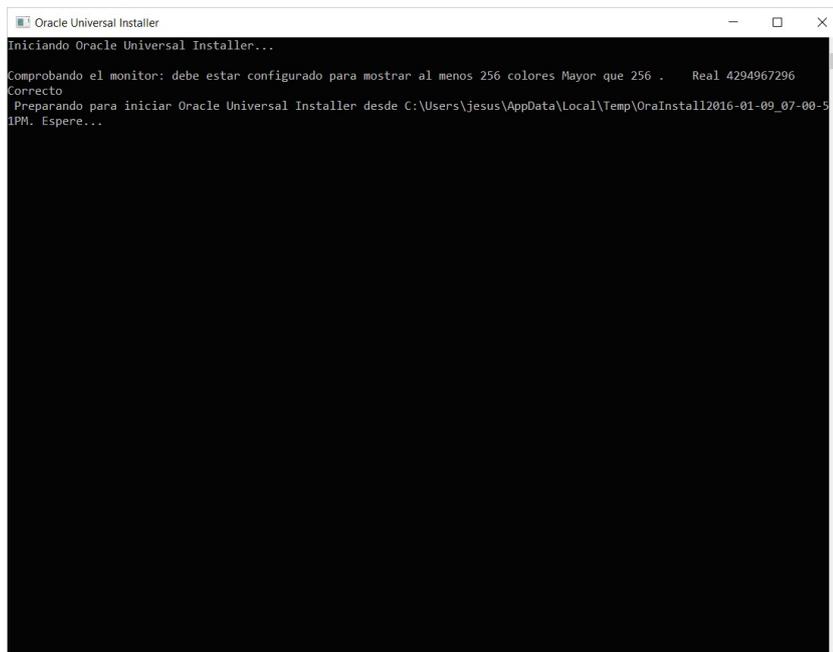


Figura A.14: Comienzo del proceso de instalación.

Puesto que la versión de sistema operativo es Windows 10, una versión aún no certificada para la instalación de 'Oracle Database 11g Release 2' (lo que no significa que no sea compatible), se dispondrá de un mensaje de aviso (ver Figura A.15) acerca de este inconveniente (de cara a posibles incompatibilidades futuras).

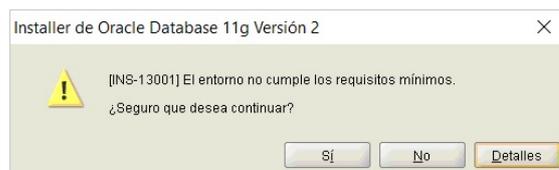


Figura A.15: Advertencia de incompatibilidad del sistema.

Tras pulsar el botón de 'Sí' aceptando continuar con el proceso de instalación, dicha instalación prosigue su camino.

El siguiente paso del asistente de instalación consistirá en proporcionar un correo electrónico de contacto. Se recomienda proporcionar aquel que ya sirvió para proceder al acceso generalizado a funcionalidades como las descargas de Oracle. De la misma manera, se solicitará la incorporación a un archivo de notificaciones

periódicas de seguridad, de manera que se pueda seguir informado de cualquier corrección o mejora relativa a la seguridad del producto. Esto queda representado en la Figura A.16.

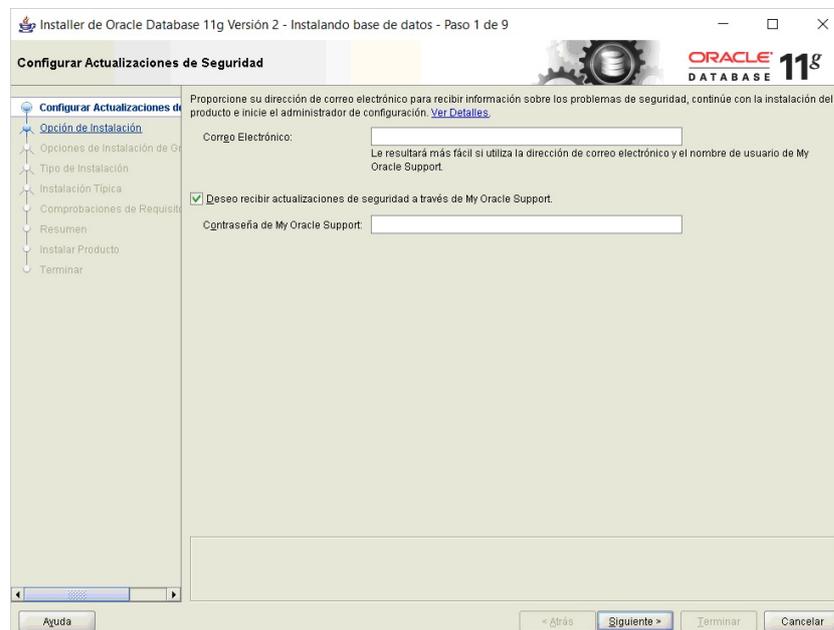


Figura A.16: Configuración de actualizaciones de seguridad.

A continuación, comenzará lo que ya en sí es la instalación del software de gestor de base de datos Oracle, así como, si se desea, la creación de una base de datos concreta. Puesto que el interés de este proyecto es tanto la instalación del producto como la creación de una base de datos, en esta ocasión se optará por esta primera opción de la Figura A.17.

El siguiente paso del asistente consiste en determinar el alcance de la instalación del software dependiendo tanto de la complejidad del sistema deseado como de su infraestructura técnica o hardware involucrado.

Dado que la instalación va a ser realizada en un puesto local (ordenador portátil), sin necesidad de utilizar muchas de las variadas posibilidades técnicas que Oracle ofrece, se opta por la primera de las dos opciones, más acotada que la segunda de ellas, como muestra la Figura A.18.



Figura A.17: Selección de instalación y creación de una base de datos.



Figura A.18: Selección de la versión de escritorio para la instalación.

A continuación, se debe proporcionar la información correspondiente a dicha instalación. Se trata de indicar los directorios de Windows en los que se ubicará la instalación del software, los propios archivos que compondrán la base de datos, la versión concreta de base de datos Oracle 11g release 2 que se desee instalar en base a los requisitos de sus posteriores uso (en este caso, se optará por la Enterprise Edition), el juego de caracteres con que será creada esta instalación (escogiéndose el set de caracteres más ampliado de UTF8 para lograr una mayor compatibilidad con los datos de origen), el nombre de la base de datos (siendo muy importante este aspecto, llamándose ‘MIMIC3EE.MIMIC’ en este caso concreto) o la contraseña de los principales usuarios (SYS, SYSTEM...) administradores de ésta.

Toda esta parametrización puede verse resumida en la Figura A.19 capturada de este fundamental paso del asistente de instalación.

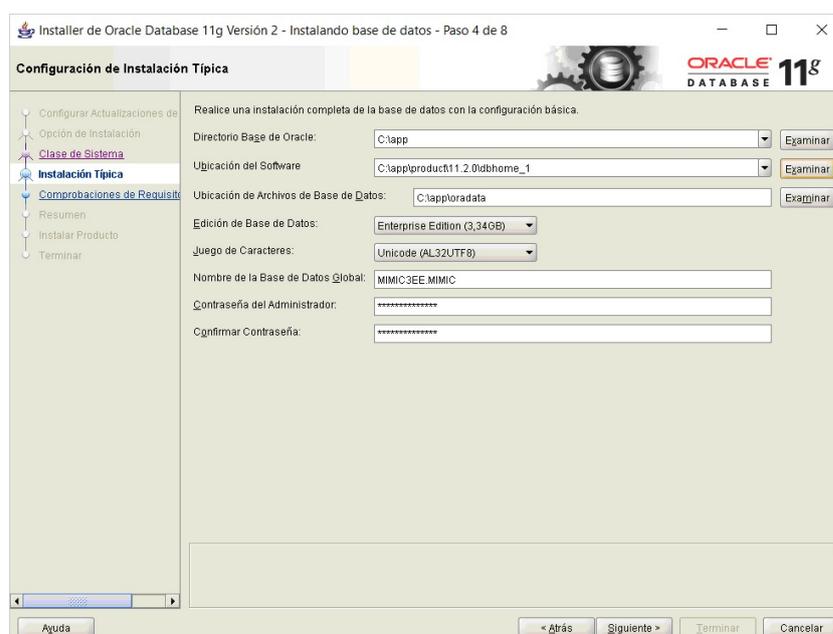


Figura A.19: Configuración de parámetros esenciales de la base de datos.

Rellenada esta información, puede crearse un archivo con estos datos, con el fin de poder consultarlos más adelante si fuera necesario. Basta con pulsar el botón ‘Guardar Archivo de Respuesta...’ del siguiente paso del asistente de instalación, como indica la Figura A.20.

En la Figura A.21 se especifica la ubicación deseada para dicho archivo.

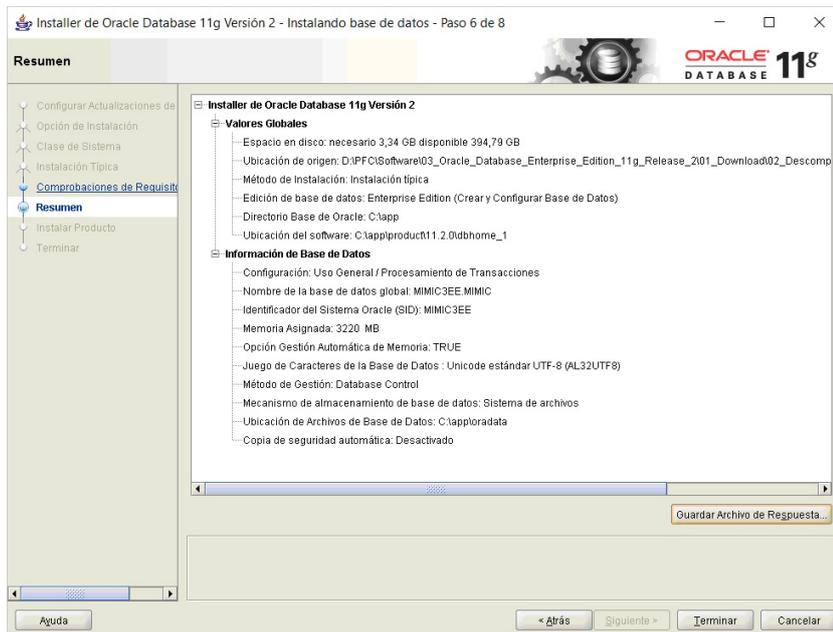


Figura A.20: Resumen final de la instalación de Oracle.



Figura A.21: Respaldo del archivo de resumen de la instalación.

Tras pulsar el botón ‘Terminar’, arrancará en primer lugar el proceso de la instalación tal y como muestra la Figura A.22.

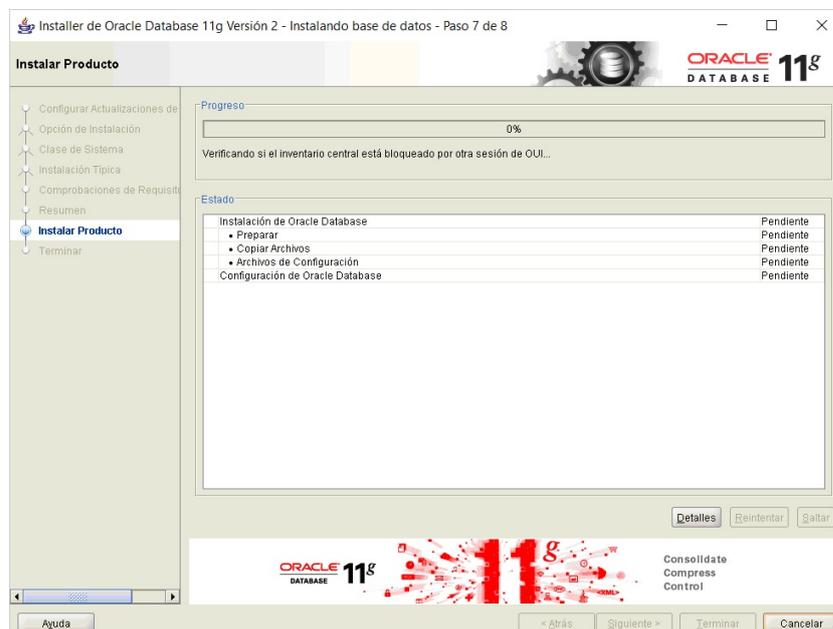


Figura A.22: Progreso de la instalación de Oracle.

En segundo lugar arrancará el proceso de la configuración, como se ilustra en la Figura A.23.

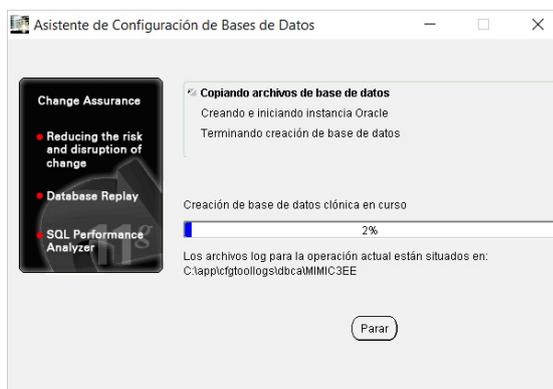


Figura A.23: Progreso de la creación de la base de datos.

Una vez finalizadas la instalación y configuración, se obtiene un mensaje como el de la Figura A.24, en el que aún sería posible gestionar (es decir, bloquear, desbloquear o modificar) las contraseñas de los distintos usuarios existentes en la nueva base de datos.



Figura A.24: Finalización de la creación de la base de datos.

En el caso concreto del proyecto que se está estudiando, no se hace uso de dicha opción de gestión de las contraseñas.

Como culminación del proceso, se obtendrá un último mensaje mostrado en la Figura A.25, dándose por finalizada la creación del repositorio de base de datos en el que se almacenará la diferente información de datos, metadatos o estadísticas del proyecto.

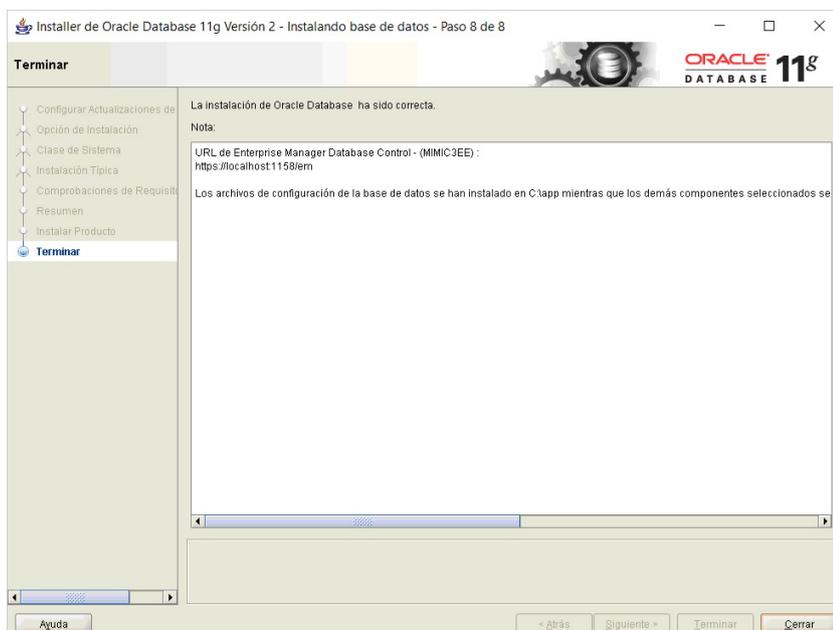


Figura A.25: Mensaje de terminación y URL de Enterprise Manager.

Anexo B

Descarga e instalación de SQL Developer

B.1 Descarga de SQL Developer

Con la finalidad de disponer de un entorno más o menos gráfico desde el que poder interactuar con la base de datos Oracle recién instalada y configurada, evitando así tener que utilizar una consola del estilo de SqlPlus y línea de comandos, se procede a descargar la herramienta propia de Oracle para dicho propósito, SQL Developer, desde la siguiente URL:

<http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads/index.html>

la cual, una vez cargada, muestra un aspecto como el de la Figura B.1.

Puesto que lo deseable es descargar la versión correspondiente a un sistema operativo de Windows de 64 bits, se optará por dicha opción de descarga (bien la primera o bien la segunda opción, dependiendo de que se desee obtener también el Java Development Kit o JDK, que es necesario para el funcionamiento de SQL Developer). Bastará con pulsar el botón de ‘Download’ junto a la descarga deseada, y la descarga comenzará a procesarse (de nuevo, es requerido autenticarse con una cuenta previamente creada en el sitio de Oracle). Esto queda palpable en la Figura B.2.

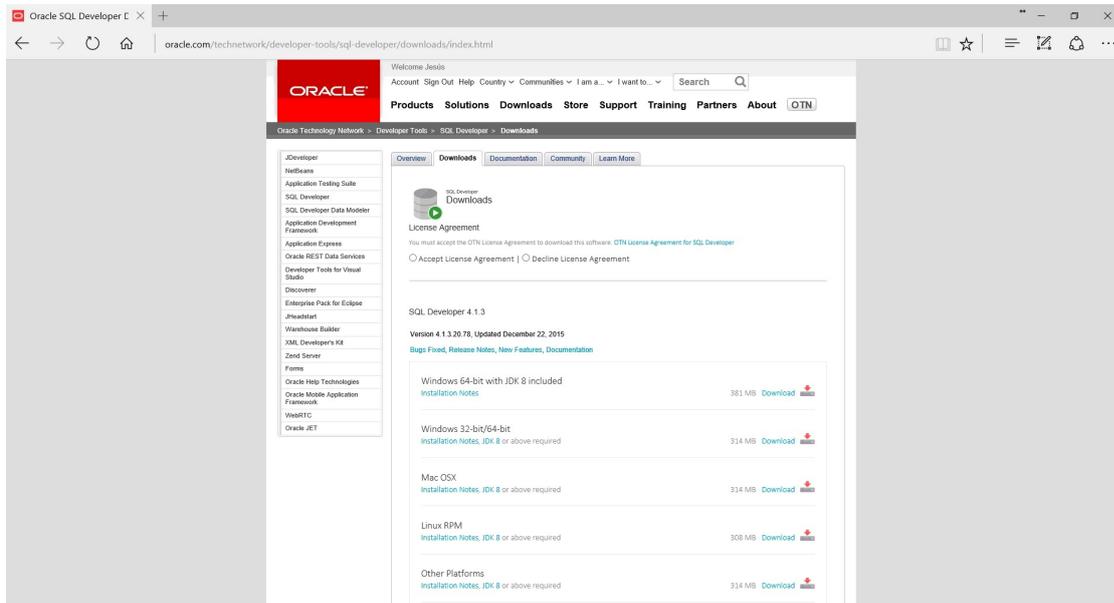


Figura B.1: Sitio de descarga de SQL Developer.

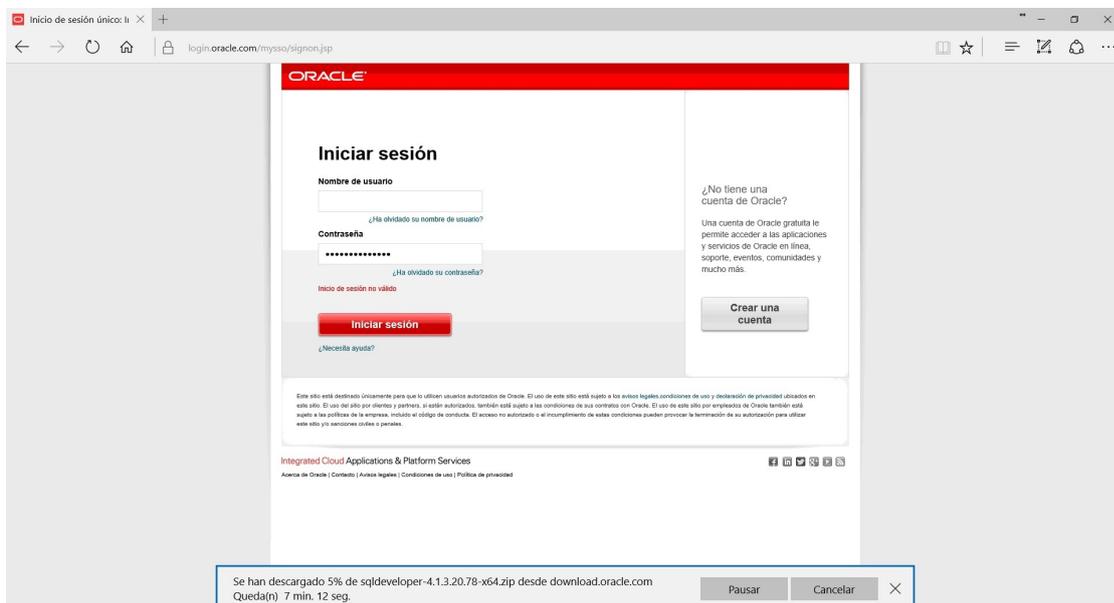


Figura B.2: Progreso de la descarga de SQL Developer.

El resultado será un archivo instalable comprimido disponible en la carpeta correspondiente a ‘Descargas’ del sistema operativo, lo cual se muestra en la Figura B.3.

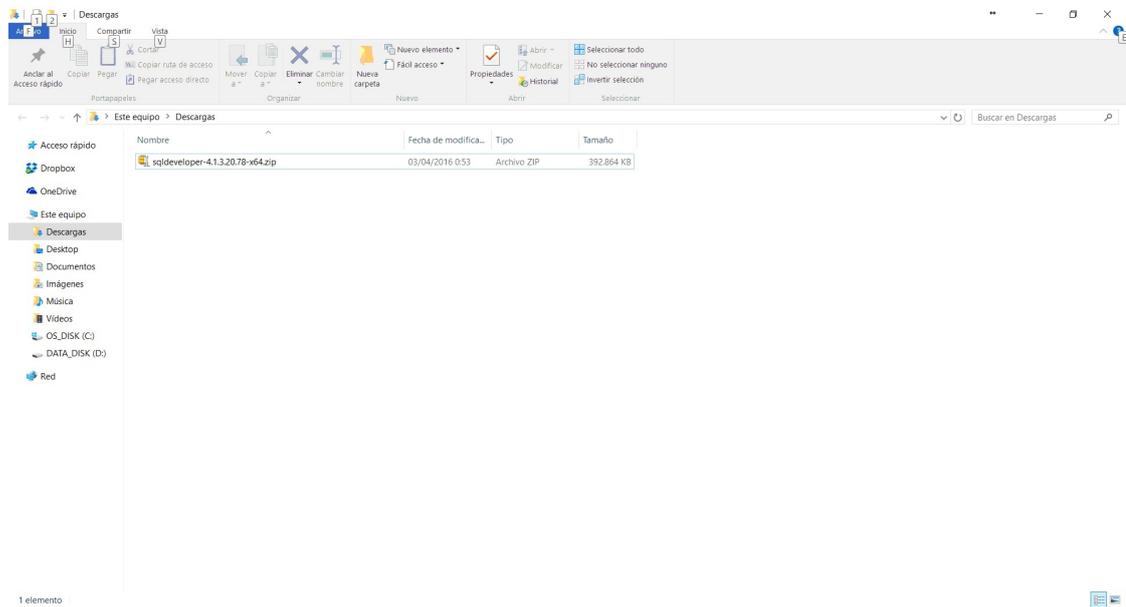


Figura B.3: Archivo instalador comprimido de SQL Developer.

B.2 Instalación de SQL Developer

Partiendo del archivo instalable recién descargado, comienza la instalación del software.

Para ello, basta con descomprimir, en primer lugar, el archivo descargado. En esta ocasión, en lugar de utilizar el software Winzip, se empleará la propia herramienta de compresión y descompresión integrada en el sistema operativo Windows, algo palpable en la Figura B.4.

El resultado será una carpeta que, con la sencillez de cortarla y pegarla en el directorio destino deseado, como si de una versión ‘portable’ se tratase, permitirá (sin necesidad de instalación adicional alguna) que la herramienta SQL Developer funcione correctamente.

El aspecto de esta carpeta resultado de la descompresión del archivo descargado queda reflejado en la Figura B.5 obtenida en el puesto de trabajo del proyecto.

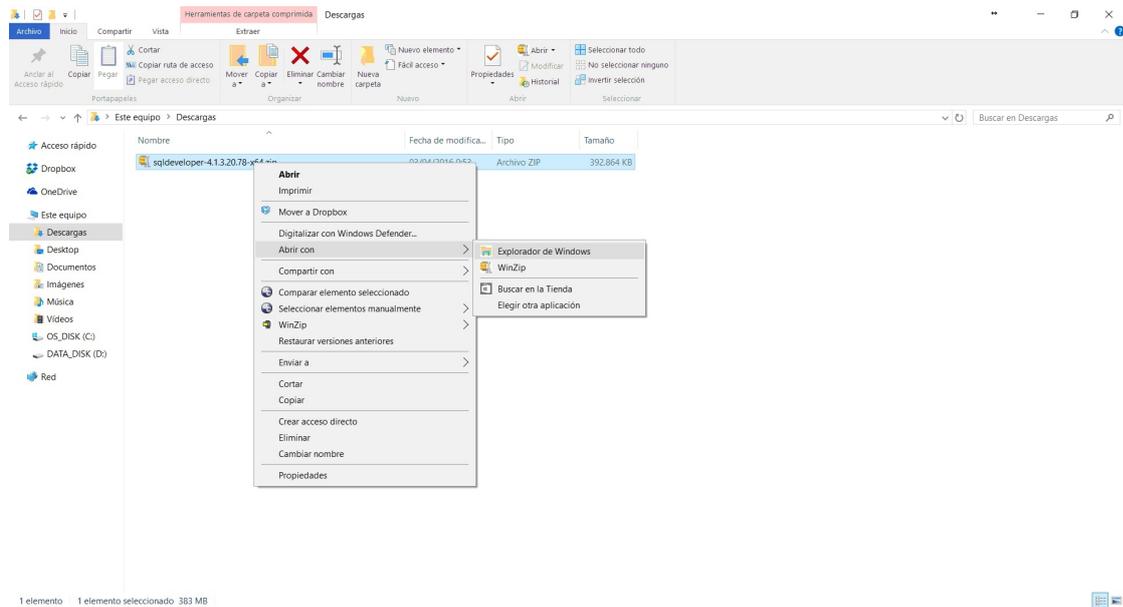


Figura B.4: Descompresión del instalador de SQL Developer.

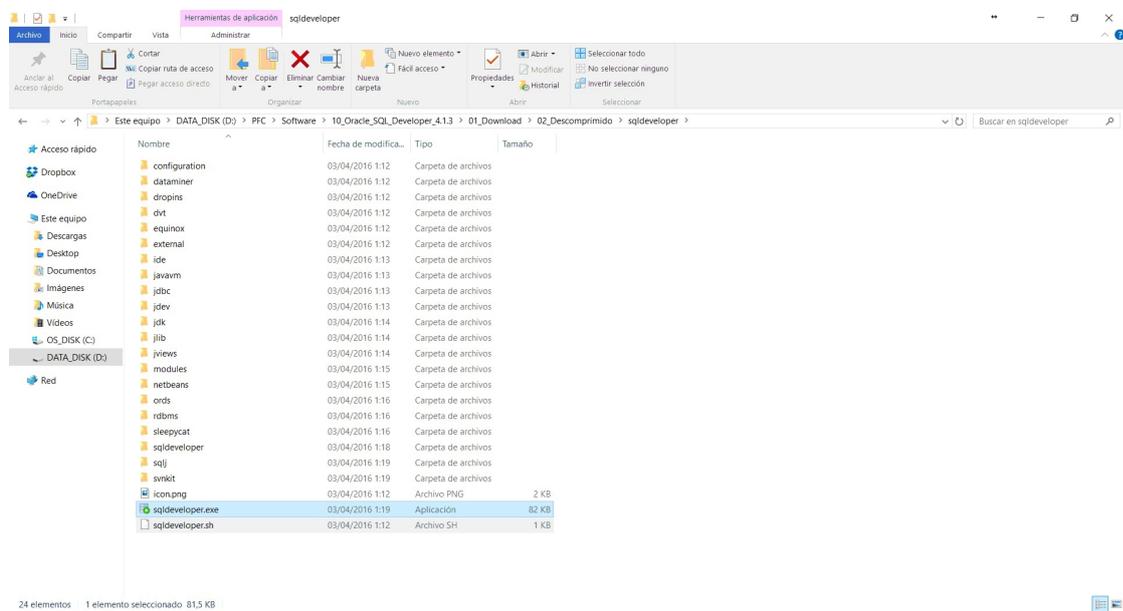


Figura B.5: Aspecto portable de SQL Developer en el puesto de trabajo.

Sin necesidad de realizar ninguna operación adicional, este software queda plenamente utilizable a falta de establecer la conexión o las conexiones deseadas con la base de datos anteriormente instalada y configurada.

Este proceso quedará reflejado en el siguiente anexo con mucho más detalle, dado que de momento tan sólo se dispone de una instancia de base de datos Oracle con sus usuarios administradores principales y de una herramienta de administración de la misma, de modo que llega el momento de establecer tantos esquemas como el proyecto requiera.

Anexo C

Creación de la base de datos

C.1 Acceso mediante usuario administrador

Una vez finalizada la instalación del producto ‘Oracle Database 11g Release 2’, y definida una instancia de base de datos ‘MIMIC3EE.MIMIC’ sobre la que poder trabajar, llega el momento de definir un entorno de desarrollo. Para ello, se dispondrá de la aplicación ya descargada e instalada ‘SQL Developer’.

A través de ella estableceremos una conexión que, mediante el usuario administrador ‘SYSTEM’, permita acceder a la base de datos ‘MIMIC3EE’ para seguir con la creación de los usuarios necesarios así como de todos los objetos de éstos.

Por lo tanto, se establecerá en esta herramienta una nueva conexión que permita conectarse de forma cliente-servidor (aunque, en esta ocasión, ambos extremos de la conexión compartan un mismo host local u ordenador portátil), como refleja la Figura C.1.

De este formulario de parametrización de conexión, se conoce el usuario (‘SYSTEM’, usuario administrador de la base de datos), su contraseña (la establecida en uno de los primeros pasos del asistente de instalación del producto base de Oracle), el ‘SID’ o nombre de la base de datos (acrónimo de System ID, definido como ‘MIMIC3EE’, también parametrizado en otro de los pasos del asistente de instalación y configuración del gestor de base de datos), el puesto

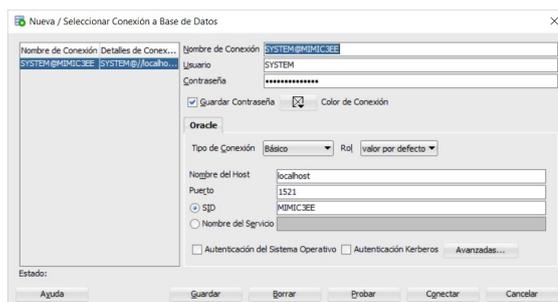


Figura C.1: Conexión para el usuario administrador ‘SYSTEM’.

servidor (será ‘localhost’ dado que reside en la misma máquina que el cliente desde donde se trata de conectar).

Pero no es conocido el puerto de conexión. Aunque siempre acostumbra a ser el puerto ‘1521’, posibles duplicidades de distintas instalaciones de Oracle (de distintas versiones, o en distintas ubicaciones...) en un mismo puesto podrían provocar que no se tratase del esperado. Para salir de dudas, basta con consultar esto a través de una invocación de ‘tnsping MIMIC3EE’ desde la línea de comandos de Windows, lo que se ve en la Figura C.2.

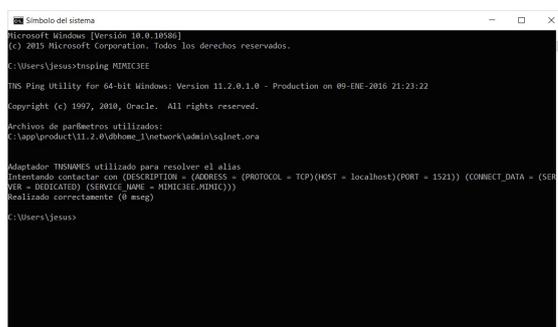


Figura C.2: Ping a la instancia ‘MIMIC3EE’.

Puede observarse que, en esta ocasión, el puerto de conexión será el esperado puerto ‘1521’.

De esta manera, conocidos todos los parámetros con los que poder definir la conexión entre el puesto cliente y el puesto servidor en la herramienta SQL Developer, se define dicha conexión y se establece la misma, de manera que pueda disponerse de un entorno de administración (más visual que abrir una consola SqlPlus desde el propio Terminal de comandos de Windows) desde el que continuar con la gestión.

C.2 Creación del usuario propietario del esquema

Como la intención es crear un usuario llamado ‘MIMIC3_OWNER’, que sea propietario de las tablas, índices... (es decir, propietario del esquema del almacén de datos), interesa proporcionarle (y, por tanto, disponer de él antes de crear dicho usuario) el tablespace (espacio de ubicación de objetos en Oracle) que, por defecto dispondrá dicho usuario (es decir, aquél en el que, si no se indica lo contrario, se ubicará cada objeto que dicho usuario cree). Con el fin de crear dicho tablespace (que, en sí, es un objeto lógico propio de Oracle, no visible desde Windows), será necesario especificar qué ficheros (es decir, archivos de nuestro sistema operativo Windows) o datafiles darán cuerpo a dicho tablespace.

Existen una serie de relaciones entre qué tamaño de bloque dispone la base de datos Oracle y qué tamaño máximo podrá tener un datafile, como refleja la Figura C.3.

<code>db_block_size</code>	<code>Datafile upperlimit</code>
-----	-----
2kb	8GB
4kb	16GB
8kb	32GB
16kb	64GB
32kb	128GB

Figura C.3: Relación de tamaños de bloque y de datafile.

Dado que se pretende dotar al tablespace por defecto del usuario ‘MIMIC3_OWNER’ de unos 100GB de capacidad, es necesario consultar el tamaño de bloque de la instalación mediante la sentencia de la Figura C.4.

Puesto que el tamaño de bloque corresponde a 8Kb, el tamaño máximo de un datafile soportado será de 32GB. Por tanto, con el objetivo de crear un tablespace de 100GB, será conveniente crear cuatro datafiles de 25GB cada uno (logrando, mediante su unificación, obtener un tablespace de 100GB, el objetivo deseado). Para ello, desde la herramienta se lanzará la sentencia mostrada en la Figura C.5.

La Figura C.6 muestra la ubicación y tamaños de los cuatro datafiles en el correspondiente directorio de Windows.

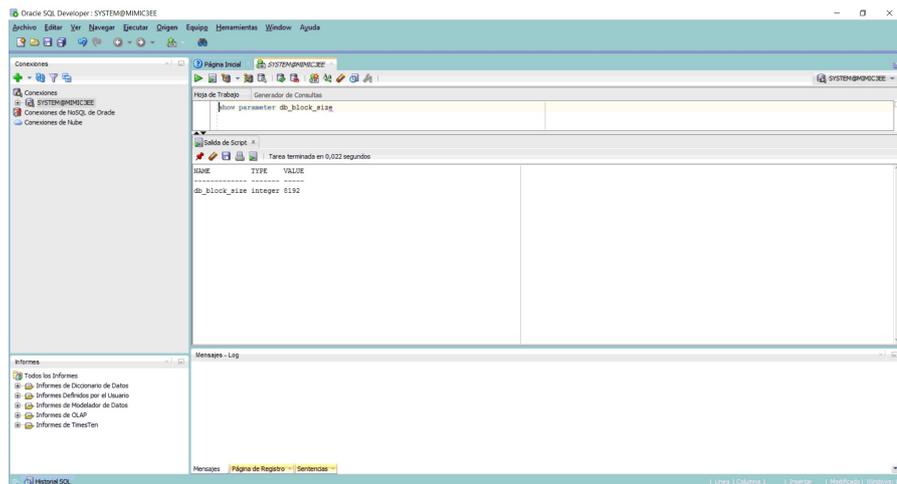


Figura C.4: Consulta SQL para conocer el tamaño de bloque.

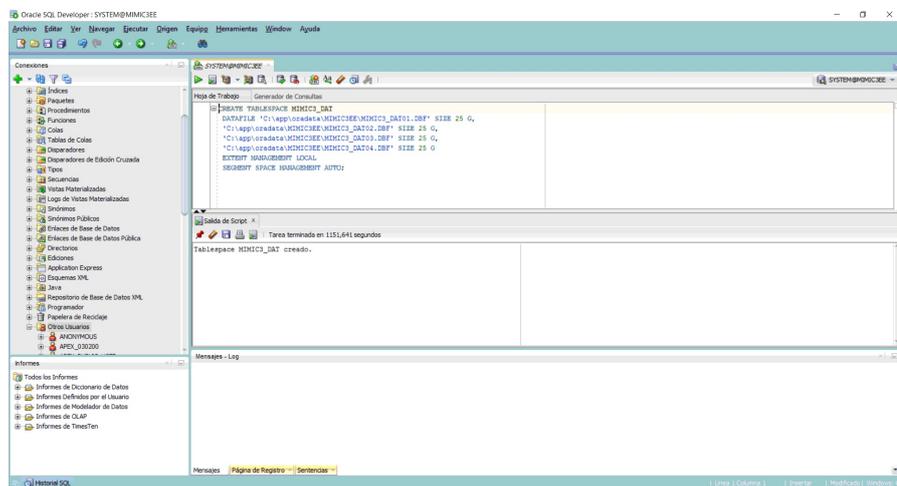


Figura C.5: Creación de un tablespace mediante cuatro datafiles.

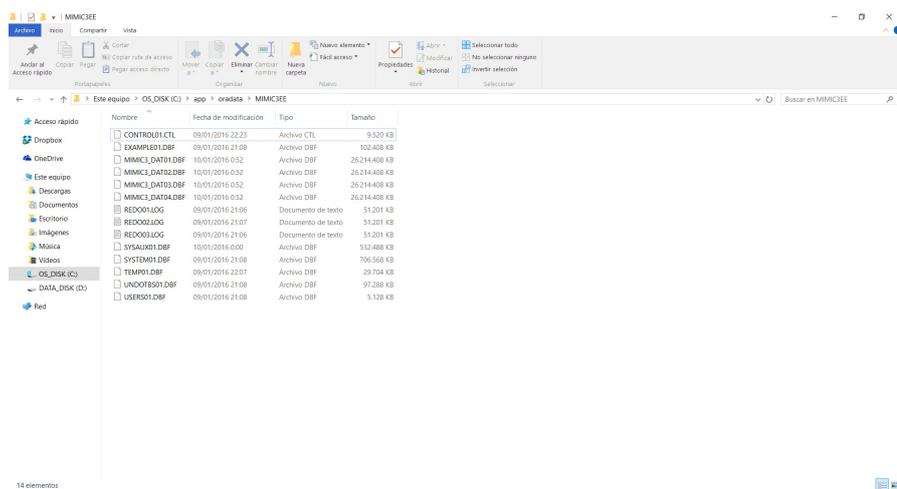


Figura C.6: Aspecto de los cuatro datafiles en el puesto de trabajo.

Ya es posible crear un nuevo usuario ‘MIMIC3_OWNER’ con este tablespace por defecto asociado, ejecutando la sentencia presente en la Figura C.7.

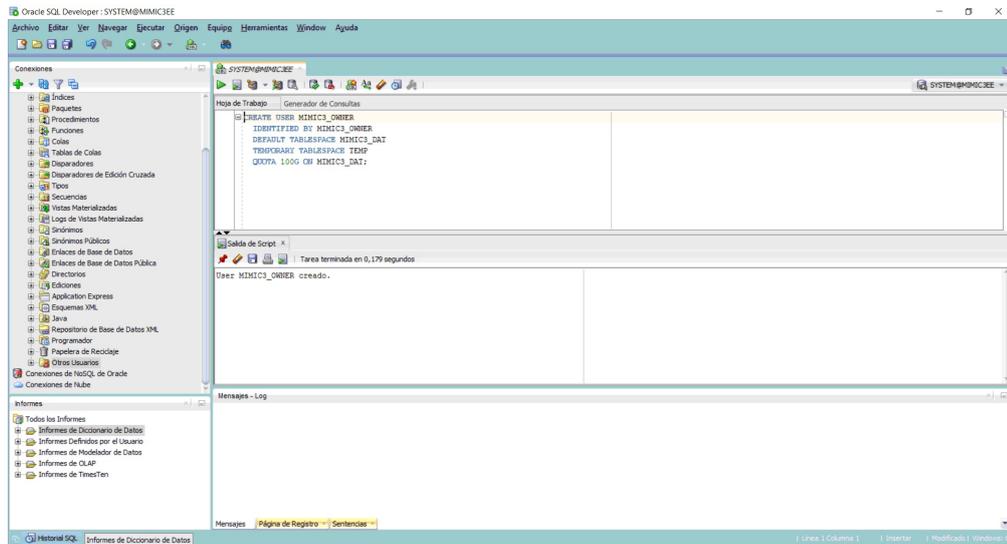


Figura C.7: Creación del usuario propietario del esquema ‘MIMIC3_OWNER’.

Además, con la finalidad de que este nuevo usuario disponga de la capacidad necesaria para poder realizar tareas básicas para su correcto funcionamiento, es necesario otorgarle una serie de privilegios, lo que se realiza mediante la ejecución de las sentencias mostradas en la Figura C.8.

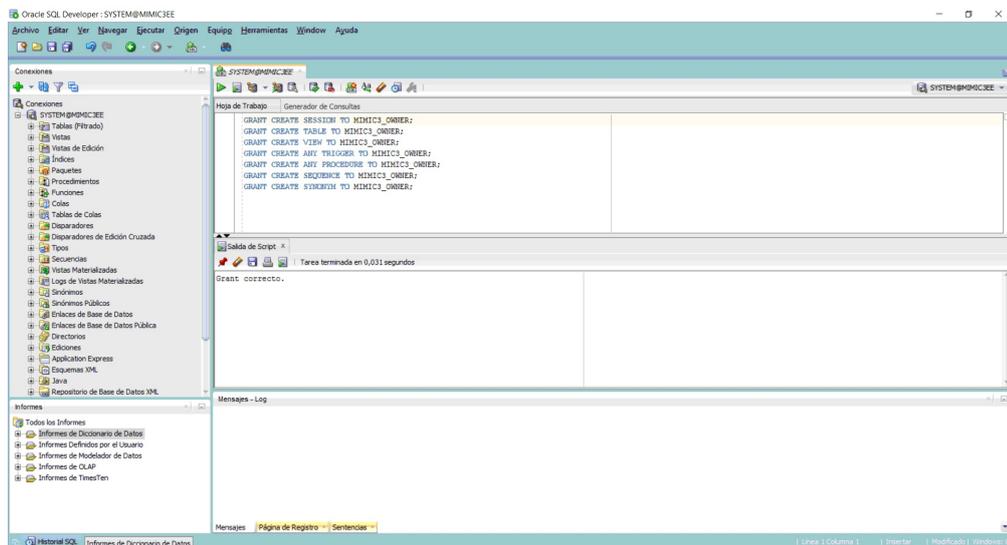


Figura C.8: Concesión de privilegios al usuario ‘MIMIC3_OWNER’.

Con el fin de poder comenzar a trabajar con el usuario propietario del

esquema ‘MIMIC3_OWNER’, se creará una nueva conexión en la herramienta SQL Developer, con la información asociada que refleja la Figura C.9.

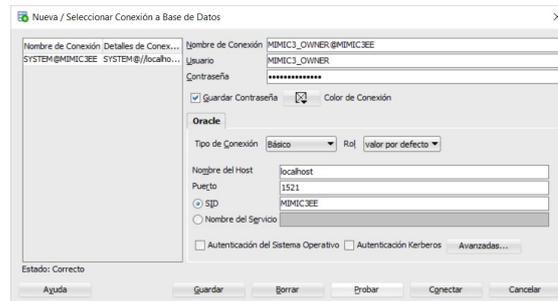


Figura C.9: Conexión para el usuario propietario ‘MIMIC3_OWNER’.

De esta manera, el usuario ‘MIMIC3_OWNER’ dispone de la capacidad de conectarse a la instancia Oracle ‘MIMIC3EE’, poder crear sus propios objetos, etc.

C.3 Creación de la estructura de tablas

El siguiente paso, por lo tanto, consistirá en crear el esquema de tablas e índices necesario para poder almacenar la información procedente de los ficheros MIMIC. Para ello se construyen una serie de sentencias DDL que permitirán crear una estructura de tablas e índices en Oracle capaces de almacenar la equivalente organización de la información observada en los ficheros de MIMIC.

Para ello, se crea un archivo DDL de construcción de tablas partiendo de una plantilla inicial recuperada de la propia organización MIMIC, pero que se encontraba obsoleto (supuestamente correspondería a alguna anterior versión de estructura de tablas y ficheros de MIMIC). De esta manera, ese script fue adaptado a la nueva lista y estructura de ficheros proporcionados por MIMIC, con el fin de poder disponer de un tabla por cada uno de los ficheros primarios proporcionados por la organización, y respetando todas y cada una de las columnas a cargar, con su formato correcto, etc.

A continuación, bastaría con identificarse en la herramienta SQL Developer con el usuario y contraseña recién creados, y proceder a ejecutar las sentencias DDL definidas, creándose así la estructura de tablas necesaria.

C.4 Carga de ficheros a tablas

El siguiente objetivo es rellenar dichas tablas con la información existente en los ficheros proporcionados por MIMIC. Para ello, se realizarán las correspondientes cargas también aprovechando la funcionalidad disponible en la herramienta SQL Developer.

Se seleccionará, para cada tabla, la opción ‘Importar Datos’ disponible en el correspondiente menú contextual (ilustrado en la Figura C.10).

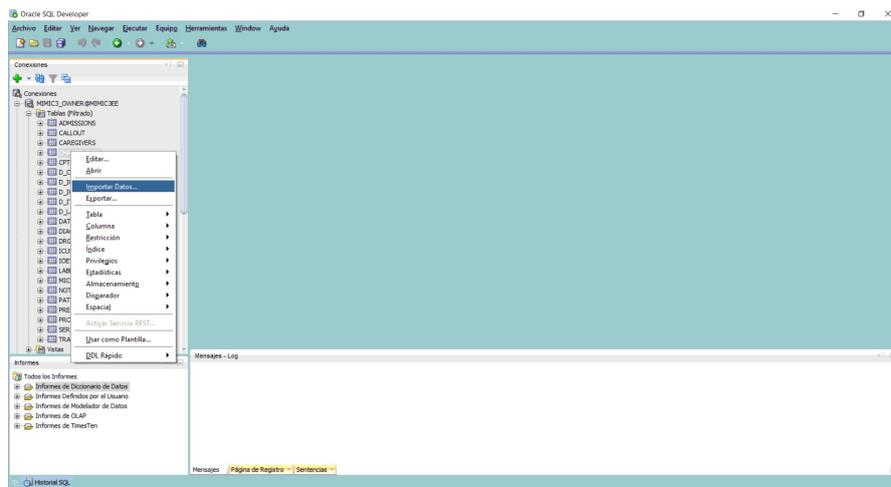


Figura C.10: Opción ‘Importar Datos’ de SQL Developer.

También se seleccionará el archivo origen con el que cargar dicha tabla, como muestra la Figura C.11.

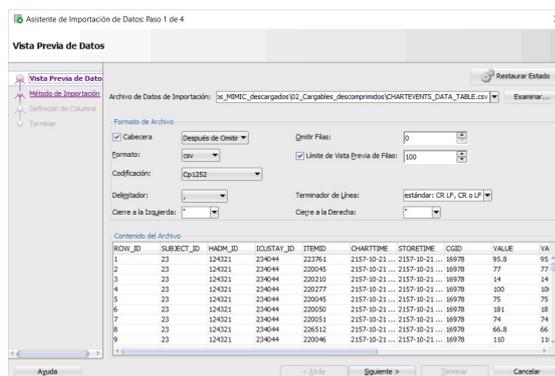


Figura C.11: Selección del archivo de datos a importar.

En este formulario del asistente de carga será necesario, además, especificar más concretamente el formato que caracteriza al fichero que se trata de cargar. De

esta manera, será necesario indicar si el fichero contiene cabeceras en su primera fila (cuyo contenido, por tanto, debe descartarse como información a cargar en la tabla), si existe algún tipo de delimitador entre las diferentes columnas, si los valores de cada columna pueden venir entrecomillados, etc.

A continuación, se podrá especificar el método de carga a emplear: bien la propia utilidad de carga de SQL Developer, bien realizar la carga mediante la utilidad de Oracle llamada SqlLoader de manera que se generen los correspondientes archivos lanzador .bat y de control .ctl, etc. En este caso se optará por la primera opción, algo palpable en la Figura C.12.

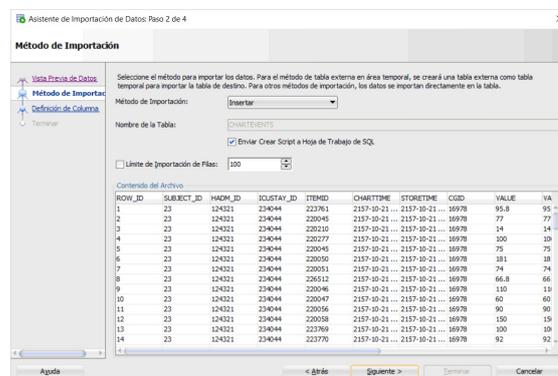


Figura C.12: Selección del método de carga de datos a utilizar.

Además, podrán elegirse las columnas concretas del fichero a atender y las que se deben obviar, en caso de quererse realizar una carga de información parcial desde fichero. Esto puede apreciarse en la Figura C.13.

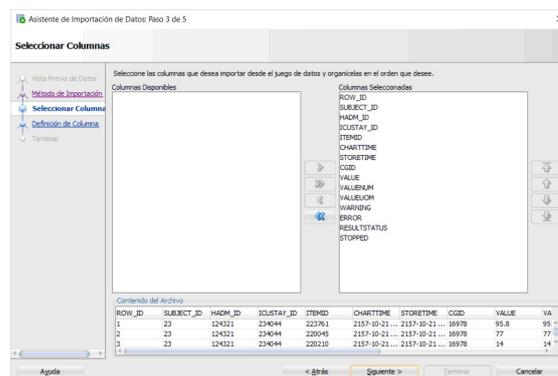


Figura C.13: Elección de columnas concretas a importar.

En el siguiente paso del asistente de carga será necesario especificar el formato de máscara con el que, determinadas columnas de tipo de dato distinto del

alfanumérico, se encuentran codificadas en el fichero de texto. Es muy importante de cara a especificar correctamente la máscara de columnas de tipo fecha, por ejemplo (que, en el caso de todos los ficheros proporcionados por MIMIC, se trata siempre de la máscara ‘YYYY-MM-DD HH24:MI:SS’), lo que se refleja en la Figura C.14.

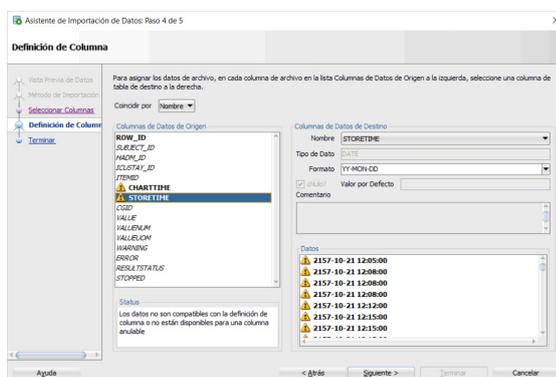


Figura C.14: Corrección de las máscaras por defecto para tipos ‘DATE’.

Una vez terminada la parametrización de la carga, basta con presionar el botón de importar para que la carga se realice apropiadamente.

Esta carga inicial, realizada en esta ocasión con motivo de la existencia de unos determinados ficheros proporcionados por Physionet, correspondería con la carga inicial de un sistema de producción, es decir, a la primera toma de contacto entre los sistemas transaccional y datawarehouse entre los que, una vez superada esta primera carga, se entablaría una comunicación periódica de traslado de cambios incrementales. Es decir: una vez realizada esta primera carga, la alimentación del datawarehouse se realizaría mediante algún otro procedimiento, no repitiéndose de nuevo este proceso. Realmente, el sistema transaccional residente en cada uno de los hospitales colaboradores informaría apropiadamente su sistema informático, el cual comunicaría de una u otra manera los cambios (ya no iniciales, sino incrementales) al sistema del datawarehouse.

En función de las diferentes tecnologías empleadas por el sistema transaccional y el sistema de datawarehouse, esta interfaz de comunicación (practicada con cierta periodicidad, posiblemente diaria y realizada en franja nocturna para actualizaciones batch) podrá consistir en la entrega de ficheros con un aspecto

parecido a los empleados para la carga inicial (siendo dichos ficheros un repositorio de cambios a realizar sobre el punto de partida de la carga inicial, que podrían ser cargados mediante la utilidad SqlLoader ya planteada anteriormente), o en la comunicación mediante accesos directos entre bases de datos (bien a través de un database link si ambos extremos fueran bases de datos Oracle compatibles, o bien a través de un Gateway en caso de estar operado el datawarehouse con Oracle pero estar gestionado el transaccional con DB2, por ejemplo).

Son muchas las modalidades de carga incremental que pueden utilizarse en los distintos casos reales de un entorno de producción. La elección de dicha modalidad dependerá, de una u otra manera, de las diferentes tecnologías empleadas por ambos extremos de la comunicación, por el volumen de datos relativos a cambios a gestionar, de la existencia o no de un determinado producto que sirva de herramienta de extracción - transformación - carga de los datos en el datawarehouse (comúnmente denominados ETC o ETL), etc.

Además, lo habitual es que el sistema de datawarehouse aplique estos cambios incrementales en su primera sección de base de datos, casi réplica local del transaccional, aquella cuyo interés es disponer de un reflejo sincronizado del transaccional pero que, en principio, no será la sección de base de datos explotable. Será necesario disponer, tras la actualización incremental de esa primera sección, de unos procesos de actualización de las propias tablas del datawarehouse (lo que sería esa segunda sección explotable) que, con una estructura más apropiada para la explotación, dispondrán de esta misma información pero con una orientación agregada, indexada, con modelos lógicos más sencillos de explotar como el de estrella o el de copo de nieve, con la información más depurada y normalizada, etc.

Todos estos procesos de comunicación transaccional - datawarehouse y de actualización de tablas agregadas del datawarehouse conforman el proceso batch que, necesariamente, debe efectuarse con cierta periodicidad con el fin de mantener sincronizados ambos sistemas.

C.5 Ajustes adicionales tras las cargas

Una vez realizada la carga, se crea un paquete almacenado en el esquema del usuario propietario 'MIMIC3_OWNER' con el fin de que contenga algunas utilidades de uso general.

En concreto, conviene pasar estadísticas a las tablas que acaban de ser cargadas tras su reciente creación. Este proceso consiste en proporcionar al gestor de base de datos Oracle información acerca de los datos: cuántas filas contiene cada tabla, con qué longitud media, con cuántos distintos valores, con qué tipo de distribución entre los diferentes valores de cada una de las columnas sobre las que se apoya algún índice...

Es fundamental realizar esta tarea con el fin de que, las consultas posteriormente lanzadas contra estas tablas, sean gestionadas de manera correcta por Oracle a través del gestor de costes que éste posee y en el que se basa para emplear uno u otro algoritmo de recuperación de la información.

Por tanto, en primer lugar se desarrolla y compila un paquete almacenado denominado 'PACK_GENERICO_MIMIC3', que de momento sólo contendrá un procedimiento almacenado llamado 'PROC_ESTADISTICAS_MIMIC3' con el que proporcionar información estadística a la tabla que reciba por parámetro en su invocación.

Una vez disponible este paquete PL/SQL almacenado en el esquema propietario, es posible invocarlo tantas veces como tablas se deseen analizar estadísticamente. Para ello, se desarrolla un pequeño procedimiento PL/SQL no almacenado que recorrerá todas las tablas del esquema propietario e invocará a su toma de estadísticas, mostrado en la Figura C.15.

C.6 Creación de usuarios de metadata y warehouse

El siguiente paso será la instalación de una herramienta de explotación de esta información residente en el esquema Oracle. Esta herramienta necesitará al menos

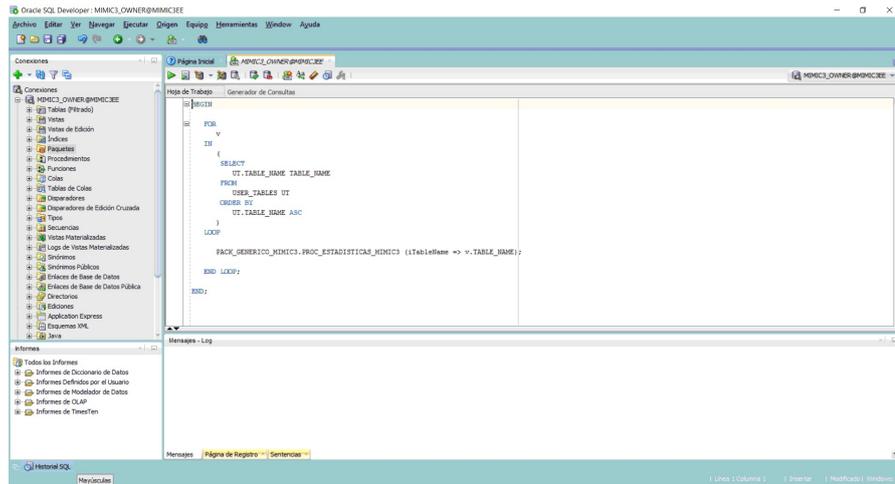


Figura C.15: Cálculo de estadísticas a todo el esquema Oracle.

dos usuarios en la misma base de datos en la que reside el esquema a explotar:

- usuario propietario de los metadatos: se trata de un usuario que también creará su propio esquema, pero que en este caso dispondrá no de tablas con la información de interés, sino de tablas con metainformación sobre ésta, esto es, dispondrá de un modelo de datos en el que almacenará toda la información acerca de la modelización lógica y física del almacén de datos a explotar. Igualmente, se encargará de almacenar todos los objetos que se vayan diseñando en dicha herramienta (indicadores, filtros, informes...). Se denominará 'MIMIC3.METADATA'.

- usuario de acceso a los datos: se trata de otro usuario adicional que será encargado de acceder a la base de datos que desea explotarse para proporcionar conectividad a la herramienta de explotación, consultando las tablas indicadas en cada momento por la metainformación (según los datos que quieran obtenerse en el informe que se esté ejecutando), creando tablas temporales con las que realizar las operativas que la dificultad del informe requiera, etc. Se denominará 'MIMIC3.WAREHOUSE'.

Por lo tanto, se procede a crear sendos usuarios que permitan almacenar los metadatos de la herramienta de explotación (primer usuario, mostrado en la Figura C.16) y acceder desde la herramienta de explotación al almacén de datos a analizar (segundo usuario, mostrado en la Figura C.17).

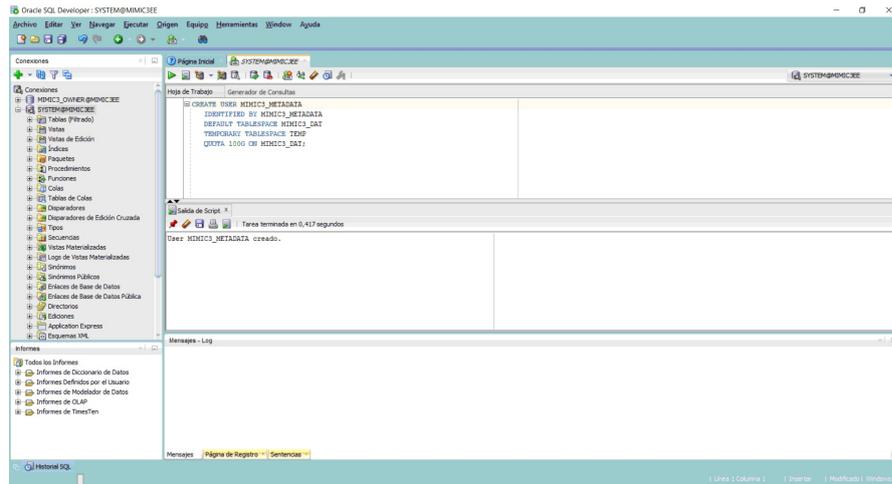


Figura C.16: Creación del usuario de Metadatos ‘MIMIC3_METADATA’.

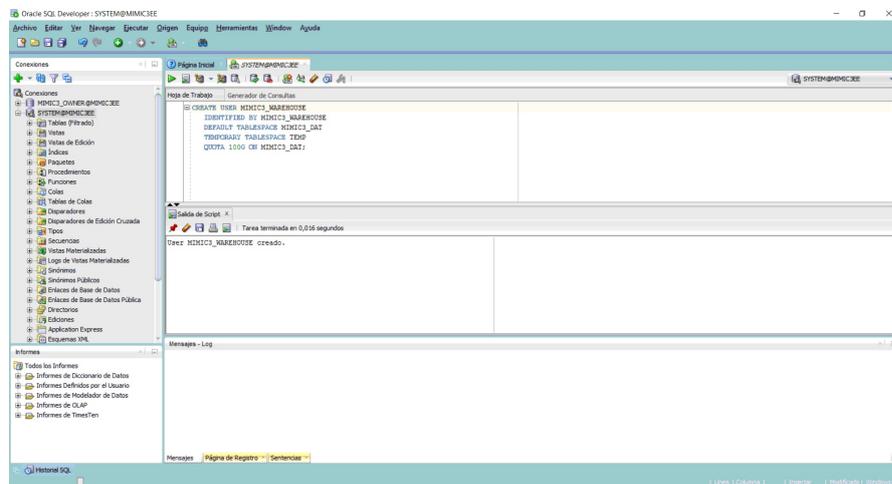


Figura C.17: Creación del usuario de conexión ‘MIMIC3_WAREHOUSE’.

Para estos dos usuarios, en esta ocasión, se seguirá manejando el único tablespace creado hasta el momento. En ocasiones, se puede desear un tablespace diferente para cada uno de los usuarios (dependiendo de la disposición de file systems del sistema operativo), o se puede compartir el mismo tablespace pero estableciendo cuotas complementarias que, entre todas, lleguen a sumar el 10% del espacio disponible en el tablespace.

Dado que, en esta ocasión, se está trabajando en un puesto local, con un único disco duro como almacenamiento, y puesto que no se van a realizar cargas incrementales en el almacén de datos a explotar que lleguen a poner en riesgo la disponibilidad de espacio en el único tablespace creado hasta el momento, se

compartirá dicho tablespace entre todos los usuarios necesarios para este proyecto, y además con una cuota completa sobre dicho tablespace.

Adicionalmente, y a igualdad de lo hecho en su momento con el usuario propietario del esquema ‘MIMIC3_OWNER’, será necesario conceder una serie de privilegios a estos dos usuarios con el fin de garantizar su correcto comportamiento posterior (que puedan acceder a crear una sesión, que puedan crear tablas temporales durante la ejecución de informes, etc.).

Por ello, se conceden permisos al primer usuario (o usuario de Metadata), tal y como se muestra en la Figura C.18.

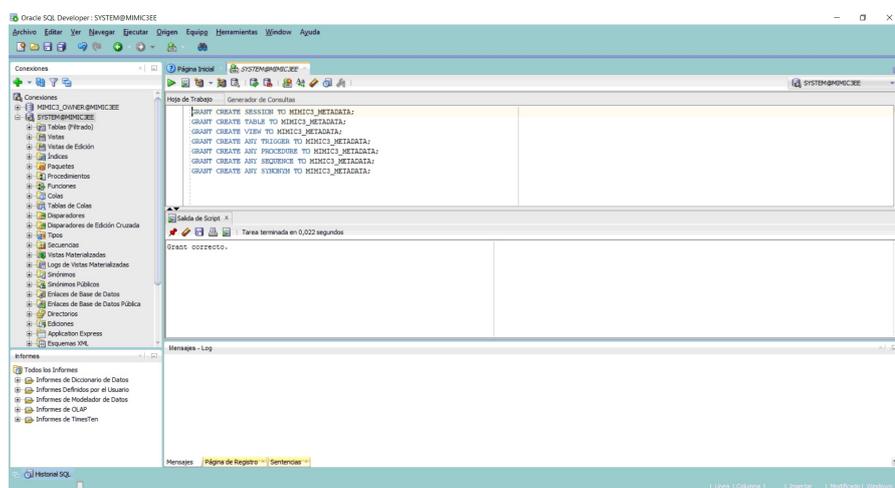


Figura C.18: Concesión de privilegios al usuario ‘MIMIC3_METADATA’.

Y, a continuación, se hace lo mismo con el segundo usuario (o usuario de Warehouse), como refleja la Figura C.19.

Por último, se establecen dos cadenas nuevas de conexión en la herramienta cliente SQL Developer con el fin de poder gestionar, desde dicha herramienta, las necesidades que, relativas a dichos usuarios, puedan surgir durante el desarrollo del proyecto.

En primer lugar, la Figura C.20 muestra la cadena de conexión definida para el usuario de Metadata ‘MIMIC3_METADATA’.

En segundo lugar, la Figura C.21 muestra la cadena de conexión definida para el usuario de conexión ‘MIMIC3_WAREHOUSE’.

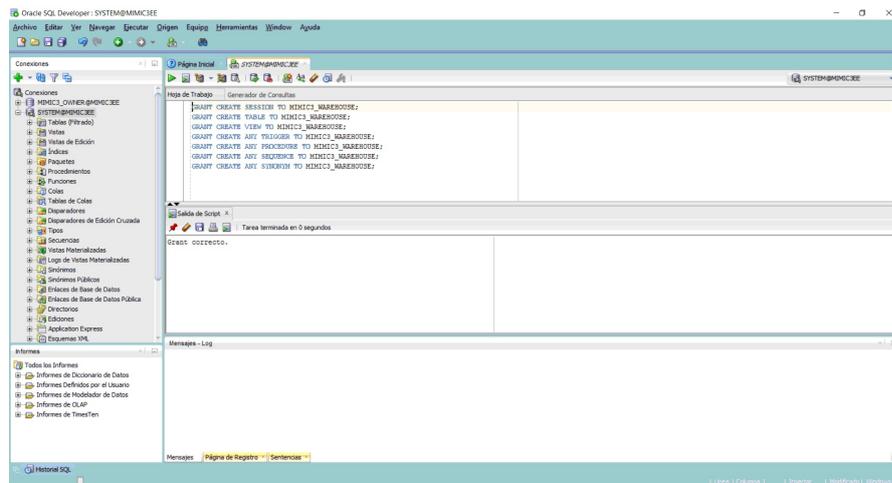


Figura C.19: Concesión de privilegios al usuario 'MIMIC3_WAREHOUSE'.

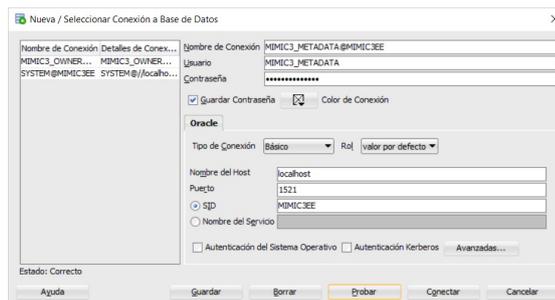


Figura C.20: Conexión para el usuario de Metadatos 'MIMIC3_METADATA'.

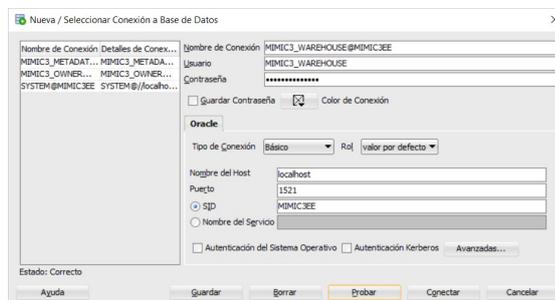


Figura C.21: Conexión para el usuario de conexión 'MIMIC3_WAREHOUSE'.

Finaliza, con estos últimos pasos, la tarea de ajuste de la base de datos para que, a continuación, pueda procederse a instalar y configurar la herramienta de explotación de la información MicroStrategy.

C.7 Visión general del modelo de datos físico

El modelo de datos primario de MIMICIHI, es decir, aquel proporcionado por la propia organización Physionet (el procedente del sistema operacional, sin añadidos para la explotación en el Datawarehouse), viene dado por una larga serie de tablas, fundamentalmente todas ellas de hechos, con apenas dimensiones, todas ellas conformadas por un número muy elevado de columnas, con una importante volumetría, con muchísima lógica funcional... Es necesaria una labor de análisis, previa al despliegue del esquema, para poder vislumbrar las necesidades de cada tabla o sus relaciones con las demás. Se trata de aproximadamente 26 tablas que, adicionalmente, deberán ser posteriormente apoyadas por tablas de look-up, por tablas agregadas o por tablas de relaciones jerárquicas. Fue necesario, por tanto, examinar profundamente la documentación existente en la página principal de la organización, así como consultar numerosas veces el propio contenido de las tablas, para poder conocer el funcionamiento del sistema operacional que se encarga de informar dichas tablas.

Con el fin de que se pueda conocer de manera general la topología del modelo de datos físico recién creado en el usuario propietario del Warehouse en Oracle, simplemente a grandes rasgos (sin conocer exactamente las columnas que participan de cada tabla, cuáles conforman la clave primaria, cuáles la clave única, etc.), así como las relaciones entre ellas (sin tampoco llegar a conocer qué columnas definen exactamente dichas relaciones), en la Figura C.22 se muestra una visión general del modelo de datos.

De estas tablas, es importante destacar la información almacenada en la tabla de pacientes 'PATIENTS', de la que dependerá prácticamente el resto del modelo de datos. Esta tabla informará del género del paciente, de su fecha de nacimiento y, en caso de fallecimiento, de su fecha concreta.

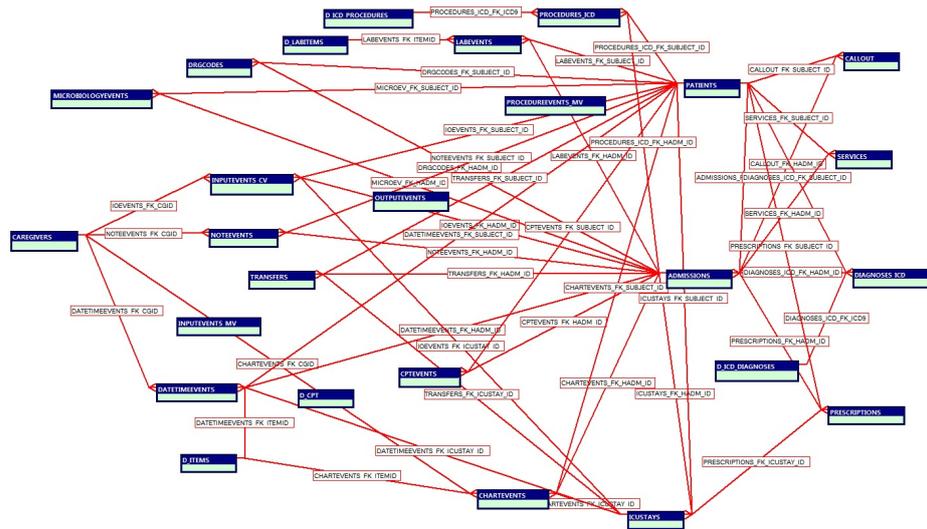


Figura C.22: Modelo de datos físico de MIMICIII.

Otra tabla fundamental será la tabla de admisiones ‘ADMISSIONS’, tal vez de igual importancia que la anterior (sin pacientes o sin admisiones, el resto de la base de datos desaparecería). Esta tabla contendrá tanto datos de la admisión en sí (fecha de admisión, tipo de la admisión, por dónde se ha realizado) como más relativos al paciente en el momento de la admisión (religión, estado civil, idioma) o al propio motivo de la admisión (como es el caso de un diagnóstico precoz).

La tabla de estancias en UCI llamada ‘ICUSTAYS’ es también crucial, dado que indicará las diferentes estancias en UCI sufridas por el paciente durante su admisión, con información como la de qué unidad fue la que le dio entrada y cual la que le dio salida.

La tabla ‘CHARTEVENTS’ es también primordial. Lista los distintos valores que han ido tomándose de los signos vitales del paciente en cada una de las estancias en la UCI. Se trata, con diferencia, de la tabla más voluminosa del sistema.

Por último, podrían también destacarse las tablas ‘SERVICES’ (que refleja los distintos servicios que el paciente ha ido recorriendo a lo largo de su estancia) o la tabla ‘LABEVENTS’ (en la que se almacenan los resultados de cada prueba de laboratorio practicada al paciente durante su admisión).

Anexo D

Descarga e instalación de MicroStrategy

D.1 Descarga de MicroStrategy

El siguiente paso será la instalación de la herramienta de explotación de la información. La herramienta escogida para la elaboración de este proyecto será MicroStrategy, sin duda el número uno mundial como tecnología de explotación en Business Analytics.

Se comienza accediendo e identificándose en la Web de descargas de componentes de MicroStrategy, dada por la siguiente URL:

<https://software.microstrategy.com/Download>

Su aspecto antes de identificarse es el presente en la captura de pantalla correspondiente a la Figura D.1.

Una vez identificados, el aspecto de esta pantalla pasa a ser el existente en la Figura D.2, de nuevo procedente de una captura de pantalla obtenida.

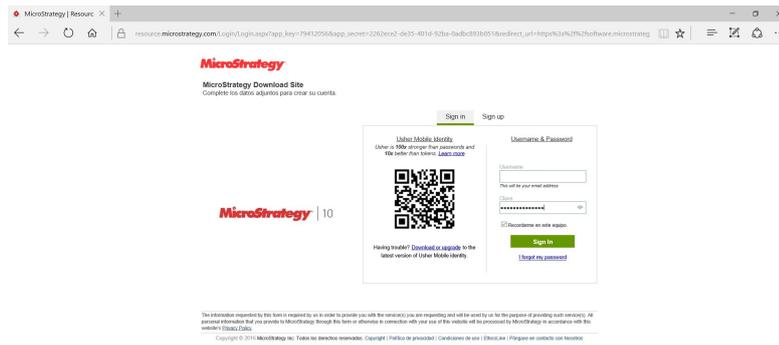


Figura D.1: Sitio web de descargas de MicroStrategy.

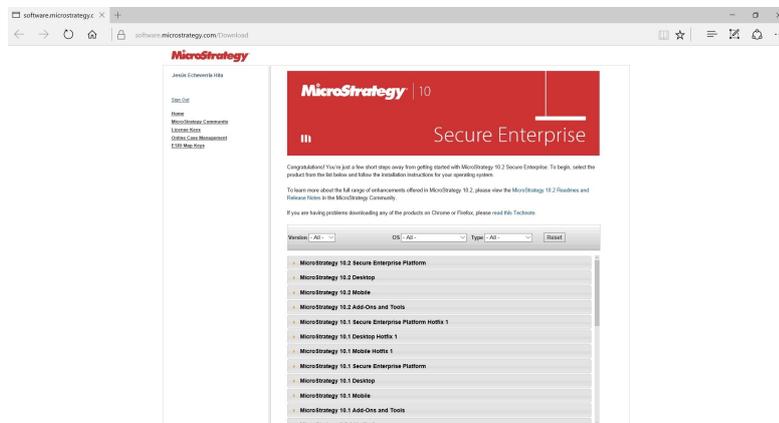


Figura D.2: Descargas disponibles tras la identificación.

Se filtra y selecciona la descarga correspondiente a la versión 10.2.0 para sistema operativo Windows de la suite (conjunto de distintas aplicaciones que conforman un entorno de desarrollo, mantenimiento y explotación) denominada Secure Enterprise Platform, como refleja la Figura D.3.

La descarga consiste en un único archivo comprimido con el aspecto mostrado en la Figura D.4.

Éste, una vez descomprimido (de nuevo, empleando herramientas de compresión y descompresión como WinZip o la herramienta ya integrada en el propio sistema operativo Windows), presenta la apariencia de la Figura D.5.

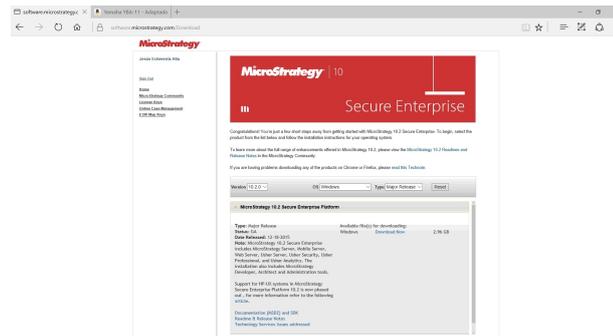


Figura D.3: Versión 10.2.0 de MicroStrategy para Windows.

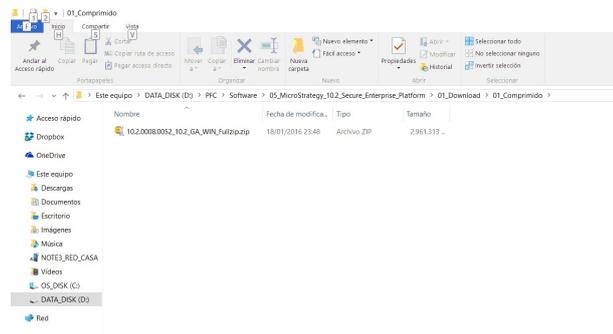


Figura D.4: Archivo instalador comprimido tras la descarga.

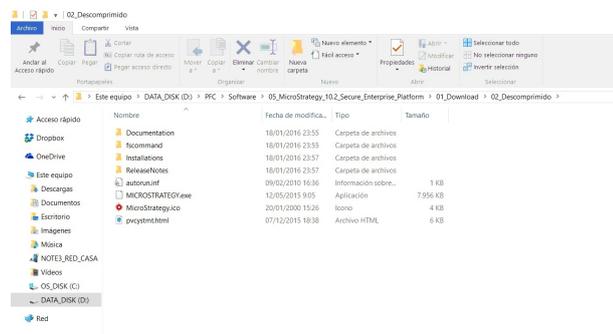


Figura D.5: Archivos de instalación ya descomprimidos.

D.2 Instalación de MicroStrategy

Para comenzar con la instalación, basta con ejecutar el archivo denominado MICROSTRATEGY.exe, de manera que se accederá a un primer formulario de acceso como el mostrado en la Figura D.6.

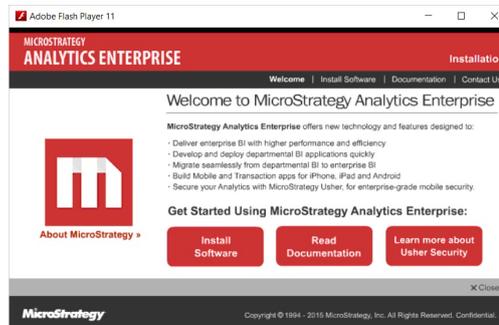


Figura D.6: Pantalla de bienvenida del proceso de instalación.

Seleccionando la opción de 'Install Software', el asistente conduce al siguiente paso de instalación, consistente en una advertencia sobre la necesidad de disponer de una clave de acceso a la instalación, que se ve en la Figura D.7.

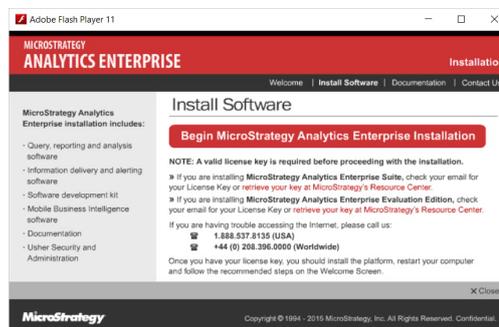


Figura D.7: Aviso de la necesidad de una clave de activación.

Basta con presionar la zona roja donde se indica 'Begin MicroStrategy Analytics Enterprise Installation' para superar este paso del asistente y pasar a la selección del idioma deseado, como se indica en la Figura D.8.

Tras elegir el idioma deseado, en la Figura D.9 se muestra una barra de progreso relativa a la preparación del resto de instalación.

Una vez alcanzado el final de la barra de progreso, se muestra un primer formulario de inicio de la instalación visible en la Figura D.10.

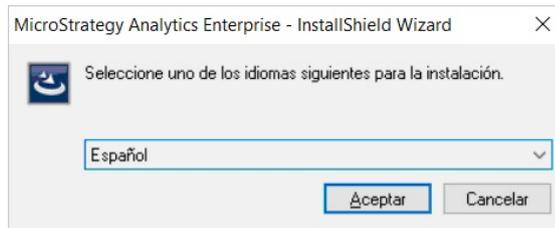


Figura D.8: Selección de idioma de MicroStrategy.

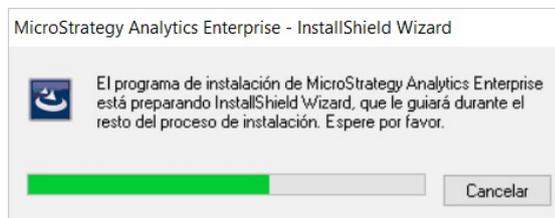


Figura D.9: Progreso de la preparación de la instalación.

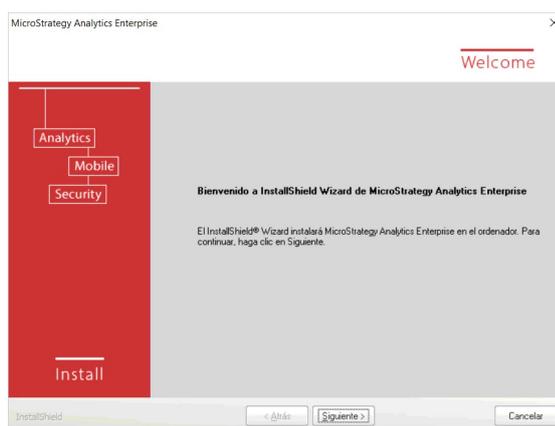


Figura D.10: Arranque del recorrido de instalación.

Una nueva advertencia avisa sobre a necesidad de detener ciertos servicios Windows durante la instalación del software, tal y como indica la Figura D.11.

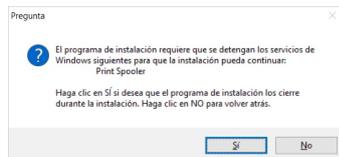


Figura D.11: Aviso de detención necesaria de servicios Windows.

Tras pulsar el botón de ‘Sí’, aparece un nuevo formulario de aceptación de condiciones de licencia, mostrado en la Figura D.12.

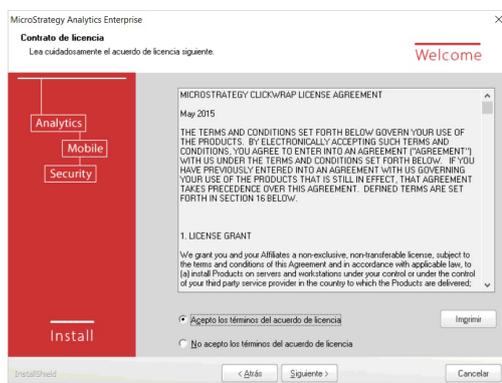


Figura D.12: Aceptación de las condiciones de la licencia.

Tras aceptar, la instalación prosigue, solicitando la cumplimentación de datos relativos a la persona instaladora, su compañía y la clave del producto, lo que se aprecia en la Figura D.13.

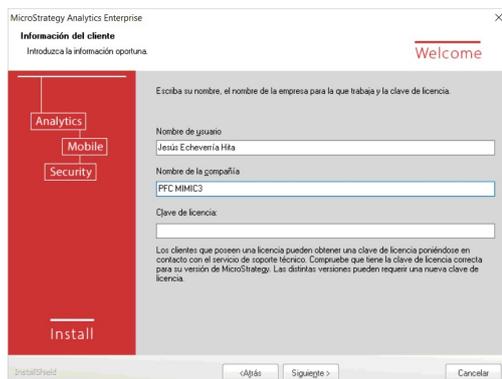


Figura D.13: Aportación de datos personales y clave de licencia.

Tras cumplimentar este formulario, el siguiente paso permitirá optar por

modificar las rutas de instalación, como se ve en la Figura D.14.

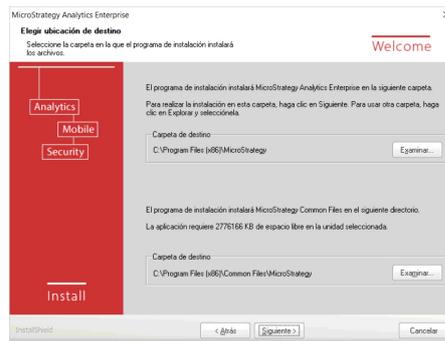


Figura D.14: Selección de rutas de instalación del software.

En la Figura D.15 se listan todos los componentes que integran la suite.

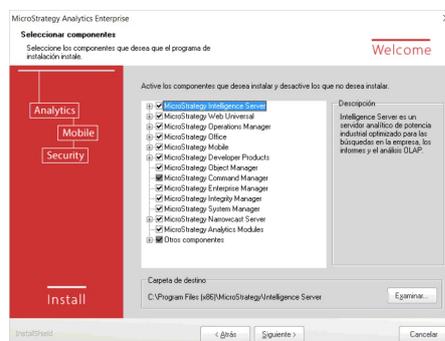


Figura D.15: Selección de componentes integrantes de la instalación.

Se opta por la lista de componentes de las Figura D.16 y Figura D.17.

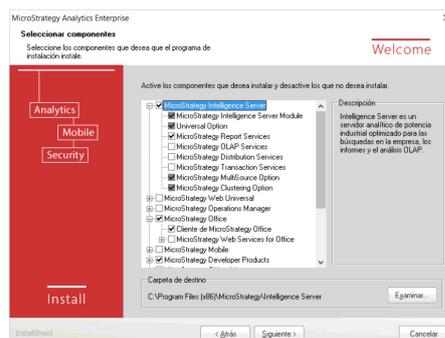


Figura D.16: Primera sección de componentes seleccionados.

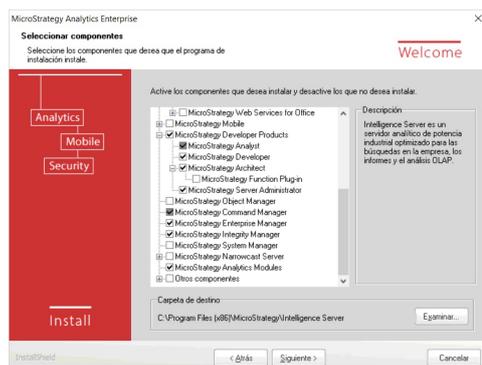


Figura D.17: Segunda sección de componentes seleccionados.

En caso de proseguir con el asistente de instalación y no disponer en el puesto de instalación de alguno de los componentes ajenos a MicroStrategy pero requeridos por éste, se obtiene un mensaje de aviso como el mostrado en la Figura D.18.



Figura D.18: Advertencia de necesidad de software adicional.

Adicionalmente, dado que el puesto local corresponde a una versión de sistema operativo Windows 10, la Figura D.19 advierte igualmente de las limitaciones de uso (restringidos a pruebas, demostraciones...) de la herramienta en dicho entorno (siendo necesario un sistema operativo de servidor real para su empleo en entornos de producción).



Figura D.19: Limitaciones de uso de la versión de sistema operativo.

Se solicita una contraseña con la que poder acceder a la herramienta de supervisión del correcto estado de la plataforma, como refleja la Figura D.20.

Figura D.20: Código de acceso al módulo Health Center.

Se mostrará un aviso como el de la Figura D.21 de la necesidad de un código de activación o tan sólo se dispondrá de hasta 7 días naturales de utilización gratuita.

Figura D.21: Recordatorio de uso de License Manager.

Las Figuras D.22 y Figura D.23 corresponden a la cumplimentación de los datos necesarios para proceder a la solicitud vía Internet de un código de activación.

Figura D.22: Primera sección de la activación del servidor.

Figura D.23: Segunda sección de la activación del servidor.

Asimismo, en la Figura D.24 se hace la solicitud definitiva de dicha clave de activación.

Figura D.24: Solicitud de un código de activación del servidor.

Se solicitará información sobre las credenciales con las que arrancar el servicio Windows que es requerido para el correcto funcionamiento de Intelligence Server, como indica la Figura D.25.

Figura D.25: Credenciales de arranque de Intelligence Server.

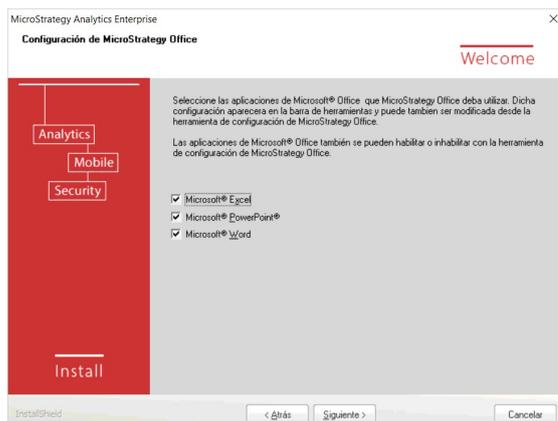


Figura D.28: Elección de asociaciones entre MicroStrategy y Microsoft Office.

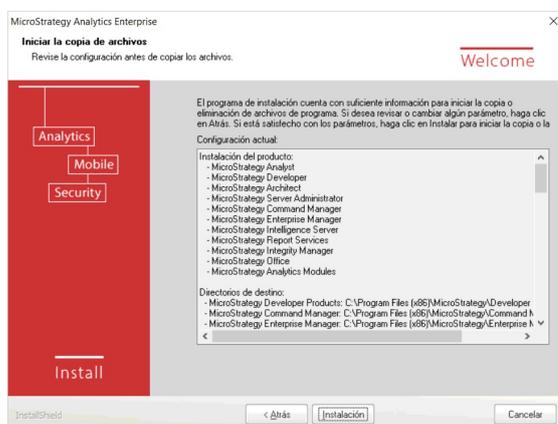


Figura D.29: Resumen de la parametrización de copia de archivos.



Figura D.30: Comienzo de la instalación de MicroStrategy.

Una vez finalizada la instalación, se pregunta si desea consultarse el archivo ‘Léame ahora’, como refleja la Figura D.31.



Figura D.31: Finalización de la instalación.

D.3 Configuración de MicroStrategy

Tras la instalación de MicroStrategy, llega el momento de su configuración, lo que implica un reinicio del sistema operativo, seguido de un acceso a Menú Inicio – Todas las aplicaciones – MicroStrategy Tools – Configuration Wizard.

Se comienza por establecer las conexiones con la base de datos con el fin de proporcionar soporte a la Metadata, el Warehouse y las estadísticas que el propio producto va tomando acerca del uso de sus componentes, mostrado en la Figura D.32.

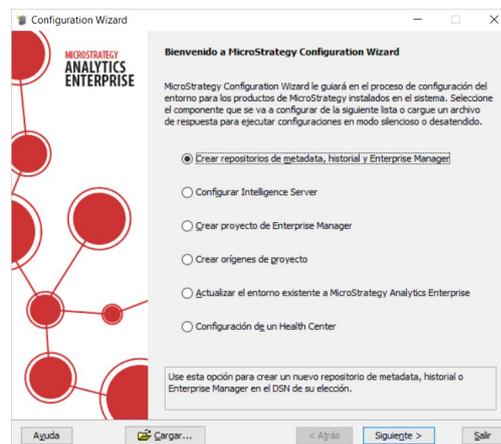


Figura D.32: Selección de creación de repositorios.

Una vez indicado que se desea configurar este primer punto, el siguiente avance en el asistente de configuración (mostrado en la Figura D.33) permite precisar qué repositorios concretamente quieren ser conectados.

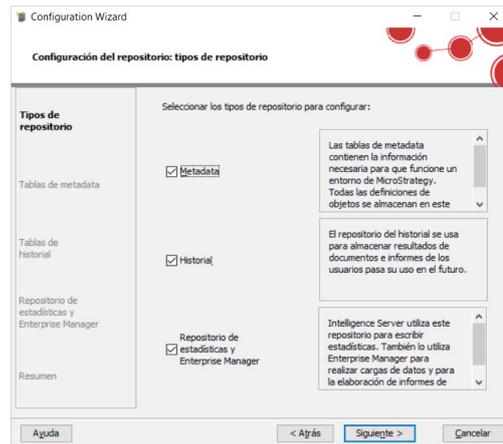


Figura D.33: Definición de repositorios concretos a crear.

Es totalmente necesario indicar el primero de los puntos, relativo a la Metadata, dado que será necesario un repositorio de base de datos en el que ir almacenando el avance del desarrollo (definiciones de tablas del warehouse, de atributos, de hechos, de métricas, de plantillas, de filtros, de informes, etc.).

De igual manera será interesante establecer un historial en el que poder ir almacenando, a modo de caché, informes previamente ejecutados (para no tener que resolverlos de nuevo contra la propia base de datos del warehouse).

Y, de igual manera, conviene activar el guardado de estadísticas para poder disponer de información de tiempos de resolución, sentencias SQL que resolvieron un determinado informe a una determinada fecha, etc.

Marcadas las tres opciones, se avanza al siguiente paso del asistente, reflejado en la Figura D.34.

En este paso será necesario establecer un origen de datos que indique a MicroStrategy la instancia, usuario y contraseña de la base de datos que servirá, desde ese momento, para que se cree la estructura necesaria para ir almacenando los metadatos que en la herramienta vayan creándose.

Se presionando en el botón de 'Nuevo', como muestra la Figura D.35.

Entonces se accede a un asistente de conectividad proporcionado por MicroStrategy, que permitirá establecer cuantos DSNs se necesiten (Figura D.36).

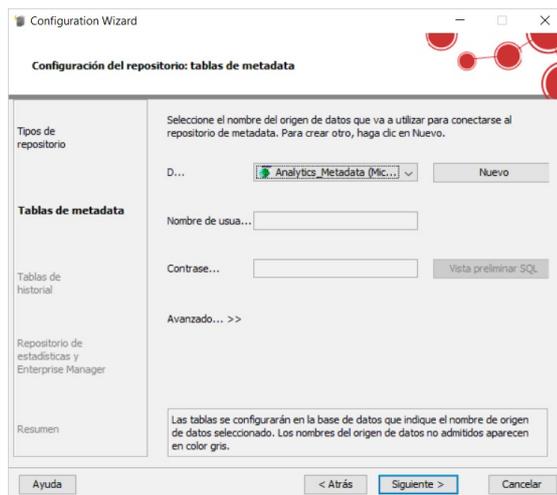


Figura D.34: Especificación del DSN de Metadata.

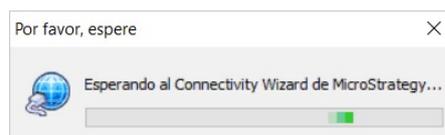


Figura D.35: Acceso al asistente de conectividad de MicroStrategy.

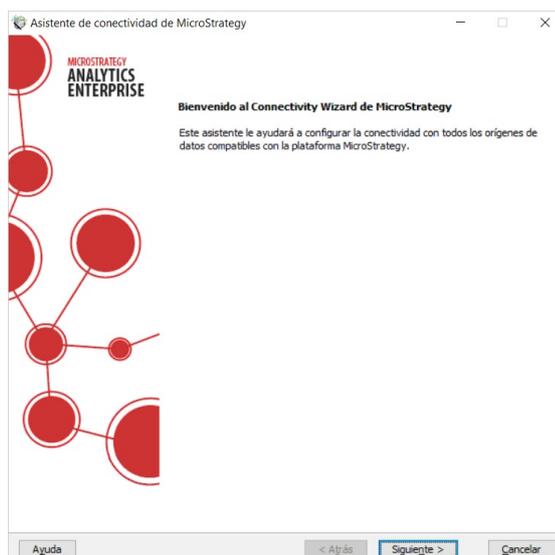


Figura D.36: Bienvenida al asistente de conectividad.

Tras pulsar el botón de ‘Siguiente’ se accede al siguiente paso del asistente, en el que se elige el tipo de base de datos que servirá de repositorio. En este caso, se trata de Oracle y, además, se marca la opción que permitirá, al finalizar, crear un DSN adicional (para conectar con el warehouse), como puede verse en la Figura D.37.



Figura D.37: Tipo de base de datos que contendrá el repositorio.

En el siguiente paso del asistente, como refleja la Figura D.38, se establecerá la parametrización de este primer DSN que sirva de conexión para que MicroStrategy acceda a su propia metadata.



Figura D.38: Ruta de conexión a emplear en este repositorio.

Una vez aportada la información correspondiente a la identificación del host en

el que reside el repositorio (el puesto local), el puerto de conexión (en este caso es el 1521) y el SID de la base de datos (MIMIC3EE en esta ocasión), en la Figura D.39 se ve que puede hacerse uso del botón de ‘Prueba...’ (recomendable) en el que se solicita introducir un usuario y contraseña que sirva de tentativa de conexión (de nuevo, recomendable probar con el usuario y contraseñas que realmente se emplearán para la conexión de la Metadata).

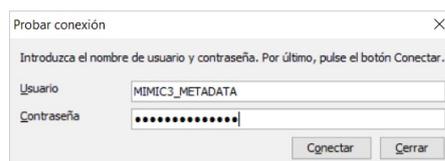


Figura D.39: Prueba de conectividad mediante usuario y contraseña.

El botón de ‘Conectar’ indicará, como en la Figura D.40, si la prueba de conexión es o no satisfactoria.

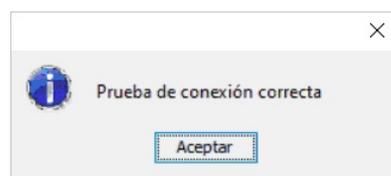


Figura D.40: Resultado satisfactorio de la prueba de conectividad.

Bastará, como indica la Figura D.41, con presionar el botón de ‘Aceptar’ de este último formulario del asistente, así como también el botón de ‘Finalizar’ del asistente que ha permitido configurar este DSN.



Figura D.41: Mensaje de DSN creado correctamente.

Puesto que se había solicitado configurar DSNs adicionales a continuación de la anterior etapa, se regresa al punto inicial de configuración de DSN.

En este caso, mostrado en la Figura D.42, la selección será de nuevo una base de datos Oracle, pero no se optará por solicitar posteriores creaciones de DSNs.

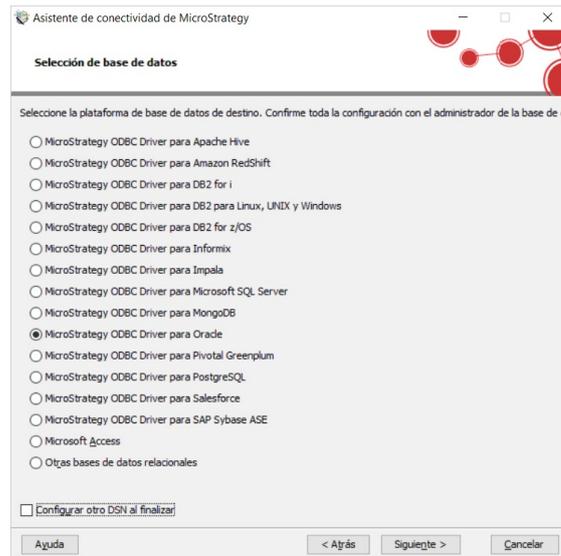


Figura D.42: Tipo de base de datos del segundo repositorio.

El nombre del DSN reflejará su posterior empleo para el acceso al Warehouse y no a la Metadata, como muestra la Figura D.43.

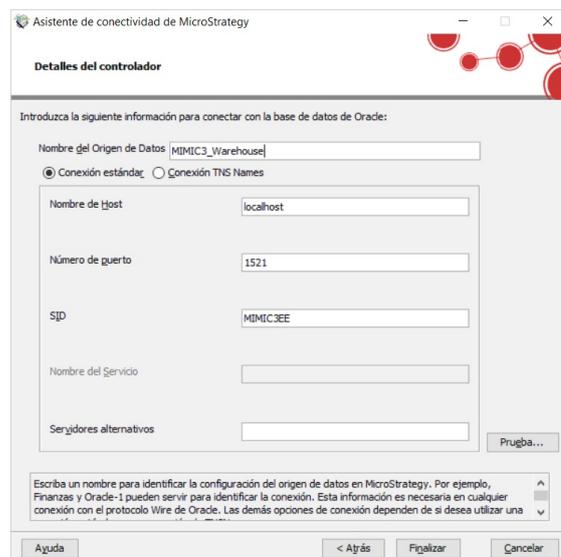


Figura D.43: Ruta de conexión al segundo repositorio.

Se realiza una prueba de conectividad con el usuario de acceso al Warehouse con el fin de garantizar el correcto acceso al almacén de datos, como refleja la Figura D.44.

Como es de esperar, la prueba de conectividad vuelve a resultar satisfactoria y así queda reflejado a través de un mensaje en la Figura D.45.

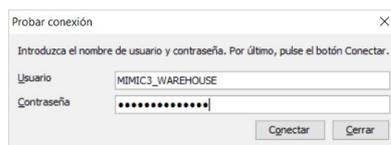


Figura D.44: Prueba de conectividad del segundo repositorio.



Figura D.45: Resultado correcto de la segunda prueba de conectividad.

Tras pulsar los botones de ‘Aceptar’ y ‘Finalizar’, el nuevo DSN queda creado de manera definitiva para su posterior uso, algo reflejado en la Figura D.46.



Figura D.46: Mensaje de segundo DSN creado.

Puesto que no se había indicado la necesidad de seguir creando nuevos DSNs, el asistente retorna de nuevo al punto de partida antes de haberse accedido a este asistente de creación de DSNs. La diferencia es que, ahora, será ya posible establecer, como origen de datos de la Metadata, tal y como indica la Figura D.47, el DSN recién creado.

Haciendo uso del botón ‘Vista preliminar SQL’, se puede observar (ver Figura D.48) el script que MicroStrategy empleará a la hora de crear, en esa conexión, el esquema de tablas necesarias para almacenar toda la Metadata:

Se indicará este mismo DSN para los otros dos repositorios que MicroStrategy necesita mantener tras las opciones definidas. En la Figura D.49, el Historial.

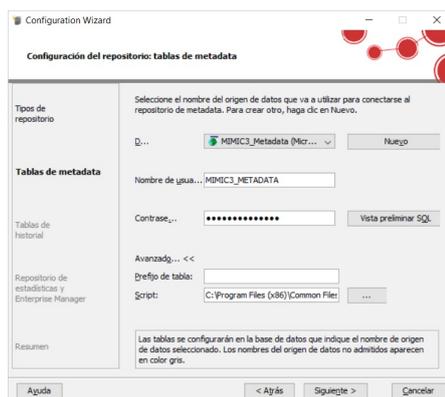


Figura D.47: Selección del DSN de Metadata recién definido.

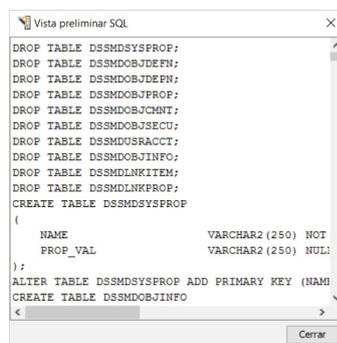


Figura D.48: Vista preliminar del SQL de creación de la Metadata.

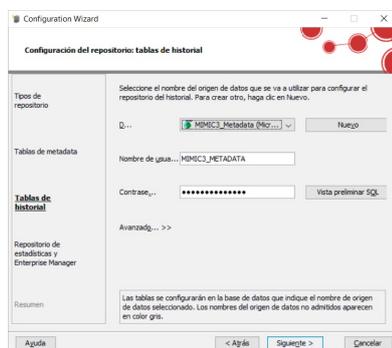


Figura D.49: Selección del DSN de Historial.

En la Figura D.50, las Estadísticas.

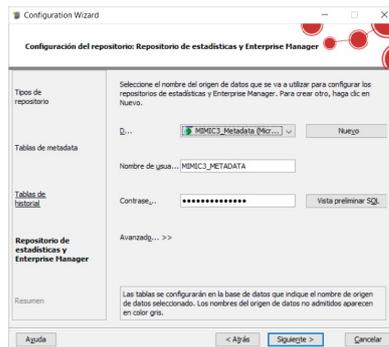


Figura D.50: Selección del DSN de Estadísticas.

Se mostrará un resumen de las conexiones a establecer, como refleja la Figura D.51.

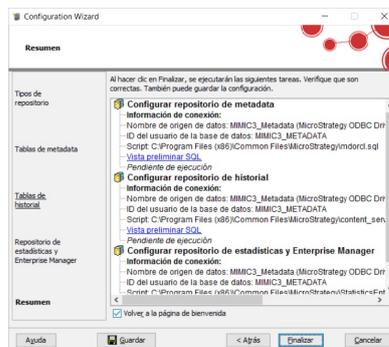


Figura D.51: Resumen de repositorios recién declarados.

Pulsando el botón de 'Finalizar', se crearán dichos esquemas, como muestra la Figura D.52.

Una vez ejecutados estos scripts, en la Figura D.53 se muestra un resumen de estado.

El botón de 'Devolver' permite regresar al Configuration Wizard, como muestra la Figura D.54.

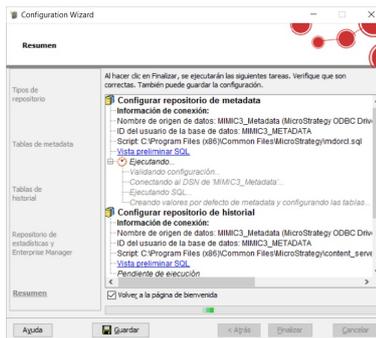


Figura D.52: Arranque de la creación de los distintos repositorios.

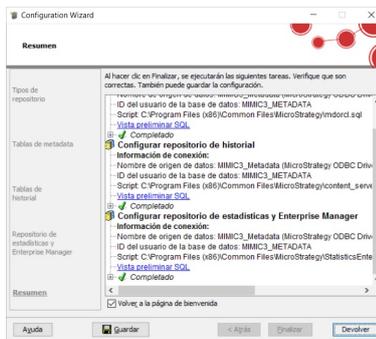


Figura D.53: Resumen de la creación de los repositorios.

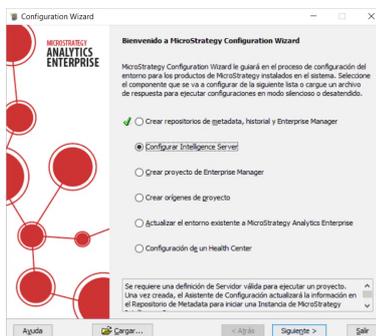


Figura D.54: Regreso al menú principal de configuración de la instalación.

A continuación, la Figura D.55 muestra cómo se establecerá una configuración válida para Intelligence Server.

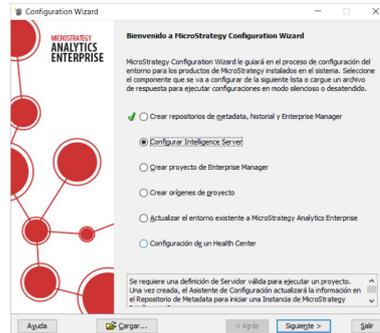


Figura D.55: Avance hacia la configuración de Intelligence Server.

Será necesario indicar la conexión de base de datos a utilizar para acceder a la Metadata, como se ve en la Figura D.56.

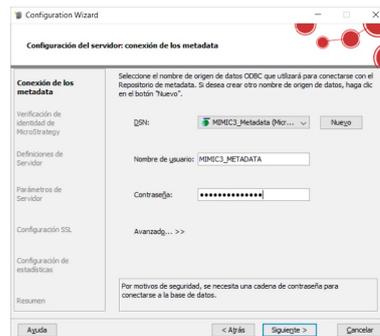


Figura D.56: Selección de la conexión válida para el repositorio de Metadata.

Tras este paso, se debe establecer un usuario y contraseña de conexión al Intelligence Server que, siendo la primera vez que se trata de conectar, debe dejarse sin cambio tal y como advierte el mensaje de la Figura D.57.

Una vez establecidos usuario y contraseña, en la Figura D.58 se entabla la conexión entre Intelligence Server y Metadata.

Es necesario configurar cierta flexibilidad al Firewall en caso de disponer de uno, como puede verse en la Figura D.59.

A continuación, se parametrizará (ver la Figura D.60) la nomenclatura a emplear para la definición de servidor.

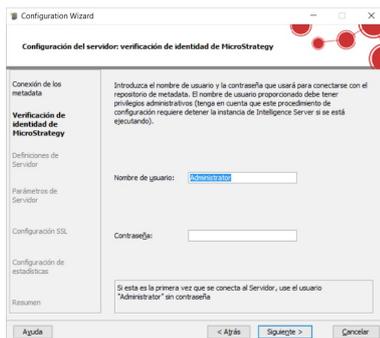


Figura D.57: Alta de usuario y contraseña administrador.

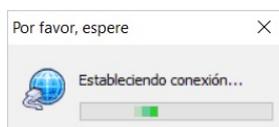


Figura D.58: Intento de conexión con el repositorio de Metadata.



Figura D.59: Desbloqueo de características del Firewall de Windows.

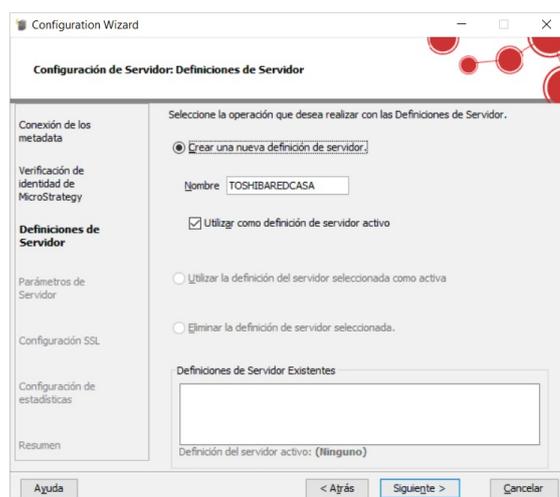


Figura D.60: Definición de servidores de Intelligence Server.

Deberá establecerse un puerto de comunicación y, si se desea, podrá solicitarse un aviso por cada uno de los DSNs que se echen en falta, como se ve en la Figura D.61.

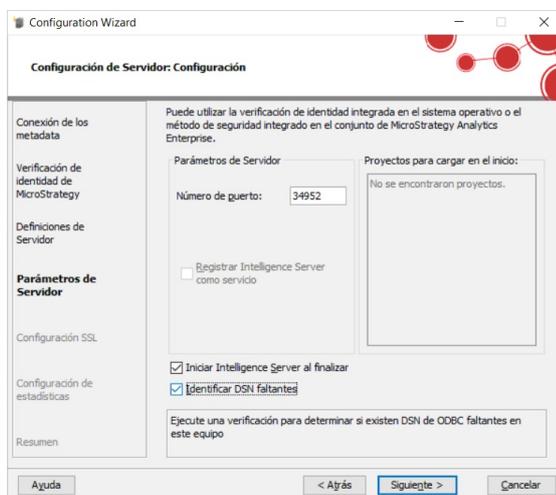


Figura D.61: Puerto de comunicaciones de Intelligence Server.

Un siguiente paso en el asistente permitiría configurar comunicaciones más seguras mediante el empleo de SSL, pero en esta ocasión (Figura D.62) no se parametrizará esta sección dado que ambos extremos de la comunicación residirán en el mismo puesto de trabajo.

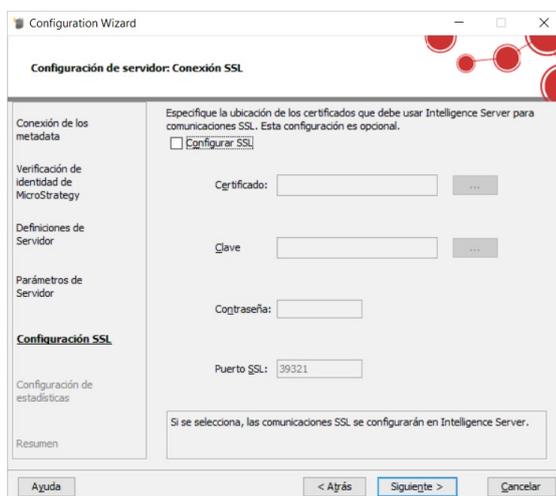


Figura D.62: Posibilidad de empleo de SSL en las comunicaciones.

De nuevo, en la Figura D.63 se solicitará un acceso a una base de datos que permita crear una estructura de Estadísticas común a todo el Intelligence Server

y que, una vez más, volverá a ser la creada con anterioridad para tal propósito.

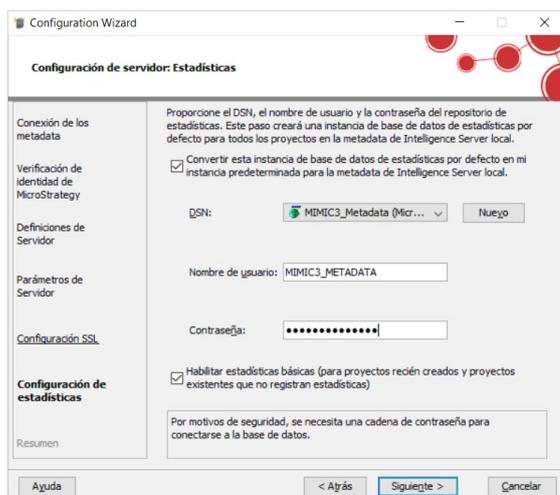


Figura D.63: Selección de la conexión válida para el repositorio de Estadísticas.

Un último paso de este asistente de configuración del Intelligence Server mostrará un resumen de los valores a inicializar, con la opción de guardado a fichero de toda esta información, como se aprecia en la Figura D.64.

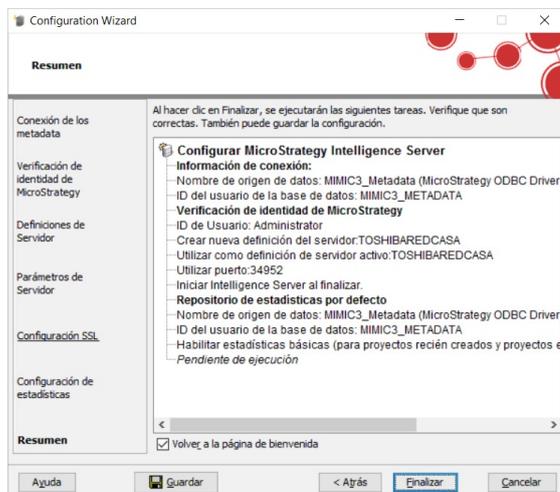


Figura D.64: Resumen de la configuración de Intelligence Server.

Tras pulsar el botón de ‘Finalizar’, se trata de establecer toda esta configuración (ver Figura D.65) y, a continuación, si el resultado es satisfactorio (lo esperado), se muestra un resumen de resultados de la actuación como el de la Figura D.66.

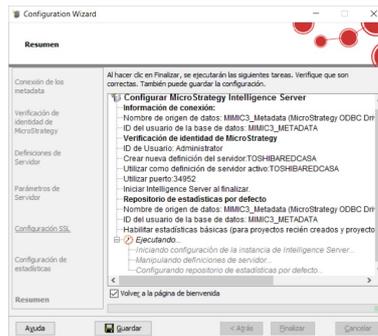


Figura D.65: Ejecución de esta configuración.

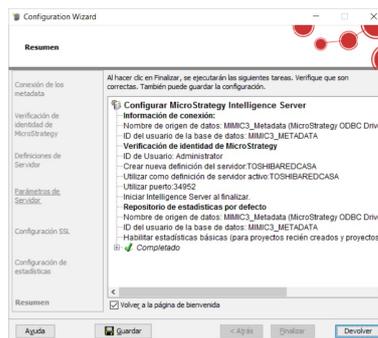


Figura D.66: Resumen de la ejecución de la configuración.

Pulsando el botón de ‘Devolver’, se regresa al Configuration Wizard como muestra la Figura D.67:

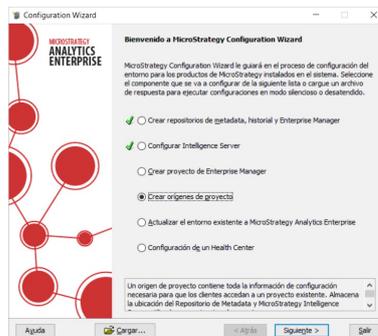
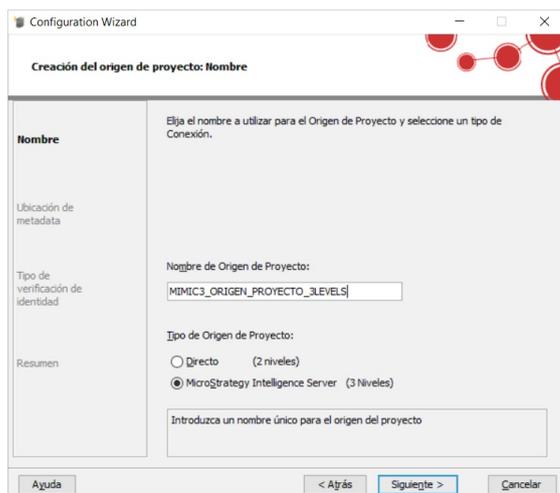


Figura D.67: Regreso al menú principal del asistente de configuración.

Tal y como sugiere este asistente mediante la opción preseleccionada, el siguiente paso a configurar conviene que sea la creación de un origen de proyecto, del cual dependerán tantos proyectos de MicroStrategy como se desee.

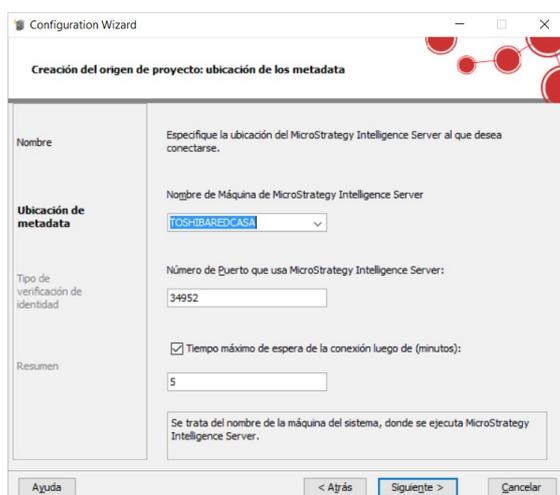
Por lo tanto, en la Figura D.68 se procede a crear un origen de proyecto establecido a tres niveles.



The screenshot shows the 'Creación del origen de proyecto: Nombre' step of the Configuration Wizard. The window title is 'Configuration Wizard'. The main instruction is 'Elija el nombre a utilizar para el Origen de Proyecto y seleccione un tipo de Conexión.' The left sidebar contains 'Nombre', 'Ubicación de metadata', 'Tipo de verificación de identidad', and 'Resumen'. The 'Nombre' section has a text input field containing 'MIMIC3_ORIGEN_PROYECTO_3LEVELS'. The 'Tipo de Origen de Proyecto' section has two radio buttons: 'Directo (2 niveles)' and 'MicroStrategy Intelligence Server (3 Niveles)', with the latter selected. Below this is a text input field with the placeholder 'Introduzca un nombre único para el origen del proyecto'. At the bottom are buttons for 'Ayuda', '< Atrás', 'Siguiente >', and 'Cancelar'.

Figura D.68: Nombre del nuevo origen de proyecto a crear.

El siguiente paso del asistente, mostrado en la Figura D.69, se auto completa con la información acerca del servidor en el que reside la información de la Metadata a emplear.



The screenshot shows the 'Creación del origen de proyecto: ubicación de los metadata' step of the Configuration Wizard. The window title is 'Configuration Wizard'. The main instruction is 'Especifique la ubicación del MicroStrategy Intelligence Server al que desea conectarse.' The left sidebar contains 'Nombre', 'Ubicación de metadata', 'Tipo de verificación de identidad', and 'Resumen'. The 'Ubicación de metadata' section has a dropdown menu showing 'TOSHIBAREDCASA'. The 'Número de Puerto que usa MicroStrategy Intelligence Server:' section has a text input field containing '34952'. There is a checked checkbox for 'Tiempo máximo de espera de la conexión luego de (minutos):' with a text input field containing '5'. Below this is a text box with the note 'Se trata del nombre de la máquina del sistema, donde se ejecuta MicroStrategy Intelligence Server.' At the bottom are buttons for 'Ayuda', '< Atrás', 'Siguiente >', and 'Cancelar'.

Figura D.69: Resumen de la Metadata a emplear por el origen de proyecto.

A continuación, se necesita configurar el mecanismo de seguridad que regirá el modo de conectarse a este nuevo origen de proyecto, mostrado esto en la Figura D.70.

Se muestra un breve resumen (que, de nuevo, puede respaldarse a un fichero y que se aprecia en la Figura D.71) de las actuaciones a acometer.

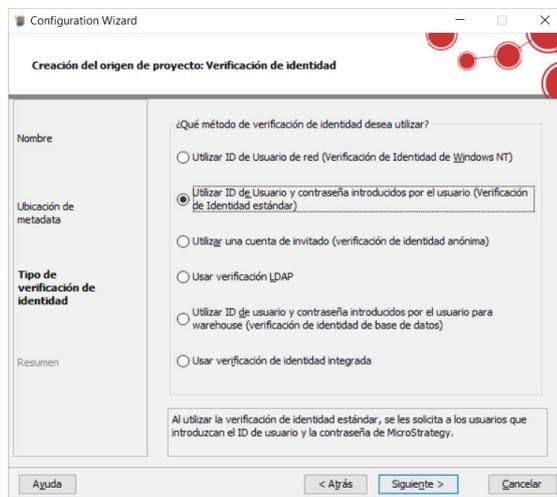


Figura D.70: Definición del tipo de seguridad a emplear.

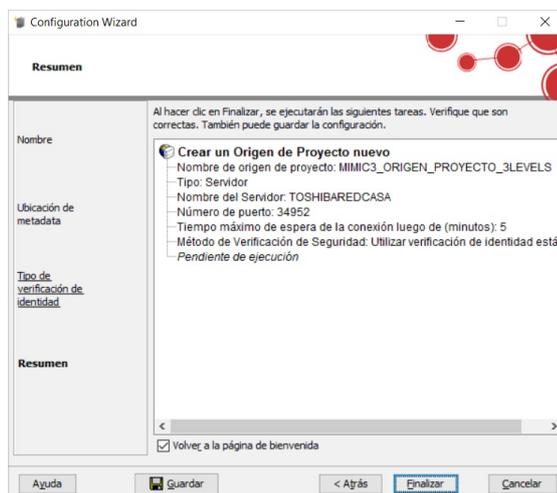


Figura D.71: Resumen previo a la ejecución del origen de proyecto.

El botón de ‘Finalizar’ permite, finalmente, establecer toda esa parametrización.

Esto queda reflejado en la captura de pantalla del correspondiente paso del asistente de la Figura D.72.

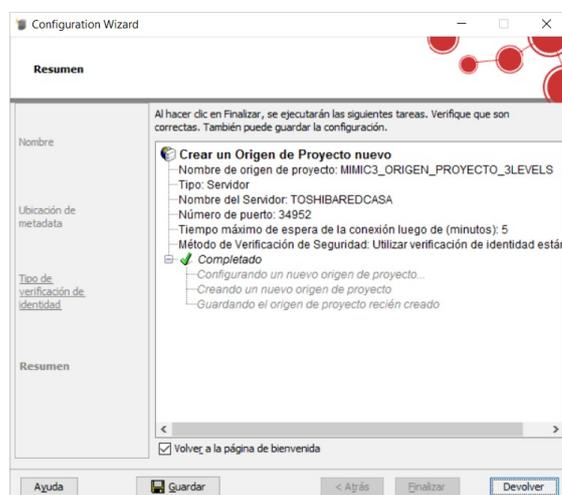


Figura D.72: Ejecución de la creación del origen de proyecto.

Tras pulsar el botón de ‘Devolver’, se regresa en la Figura D.73 de nuevo al formulario principal de Configuration Wizard.

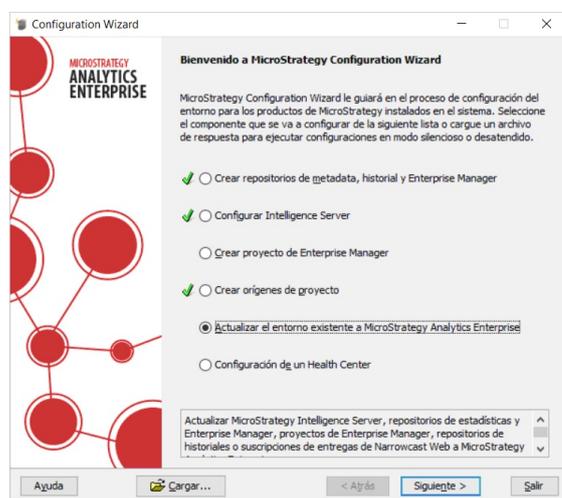


Figura D.73: Regreso al menú principal del asistente de configuración.

Por el momento, no se procede a crear ninguna configuración adicional, dando por parametrizado el producto para los propósitos de este proyecto. De esta forma, basta con pulsar el botón de ‘Salir’ para abandonar Configuration Wizard.

Con el fin de activar la instalación (dado que, en caso contrario, transcurridos 7 días desde la instalación ésta deja de mantener todo el desarrollo realizado), basta con acceder a la utilidad de License Manager (ver Figura D.74) proporcionada por la suite de MicroStrategy y, dentro de ésta, acceder a la solapa correspondiente a Administración de licencias.

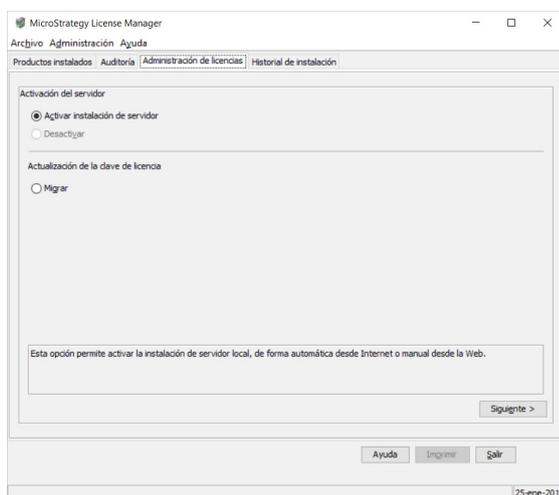


Figura D.74: Acceso a License Manager.

Se introducirá la clave de activación que MicroStrategy envió a la cuenta de correo electrónico indicada en el momento de realizar la instalación, como refleja la Figura D.75.

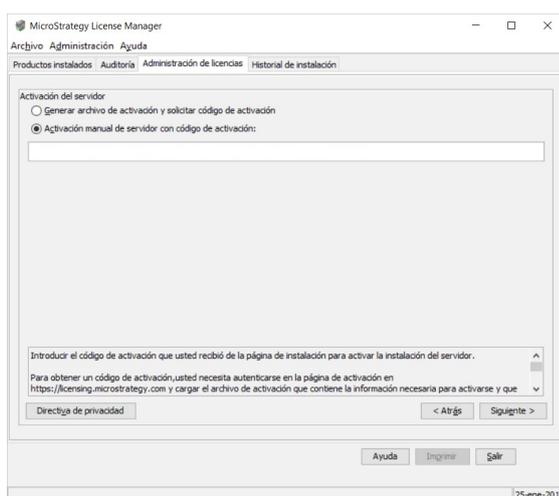


Figura D.75: Introducción de la clave de activación del producto.

Tras pulsar el botón de 'Siguiente', se obtendrá un mensaje como el de la Figura

D.76 corroborando que la instalación ha quedado correctamente activada y que, por lo tanto, queda olvidada la limitación temporal de 7 días de uso.

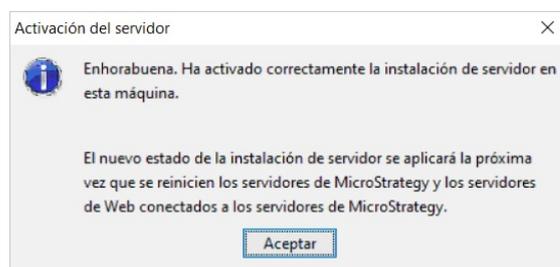


Figura D.76: Mensaje satisfactorio de la activación del producto.

Para finalizar, basta con pulsar el botón de ‘Terminado’ como muestra la Figura D.77.

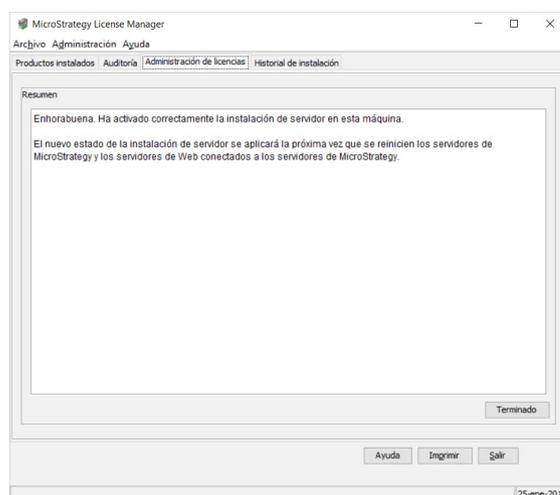


Figura D.77: Estado de la licencia en License Manager.

D.4 Creación de un proyecto

Una vez que se dispone de la instalación correcta de la herramienta de explotación, así como de la activación de la misma, cabe recordar que, hasta el momento, tan solo se ha definido un origen de proyecto, llamado ‘MIMIC3_ORIGEN_PROYECTO_3LEVELS’, que vendría a ser el nodo origen del que dependerían una serie de posibles proyectos.

Dada la necesidad de este proyecto, se procede a definir un nuevo proyecto que dependa de dicho origen de proyecto. Se denominará 'PROYECTO_MIMIC3'. Para ello, desde la herramienta MicroStrategy Developer incluida en la suite de MicroStrategy, se accede a la opción de 'Crear Proyecto Nuevo' como indica la Figura D.78.

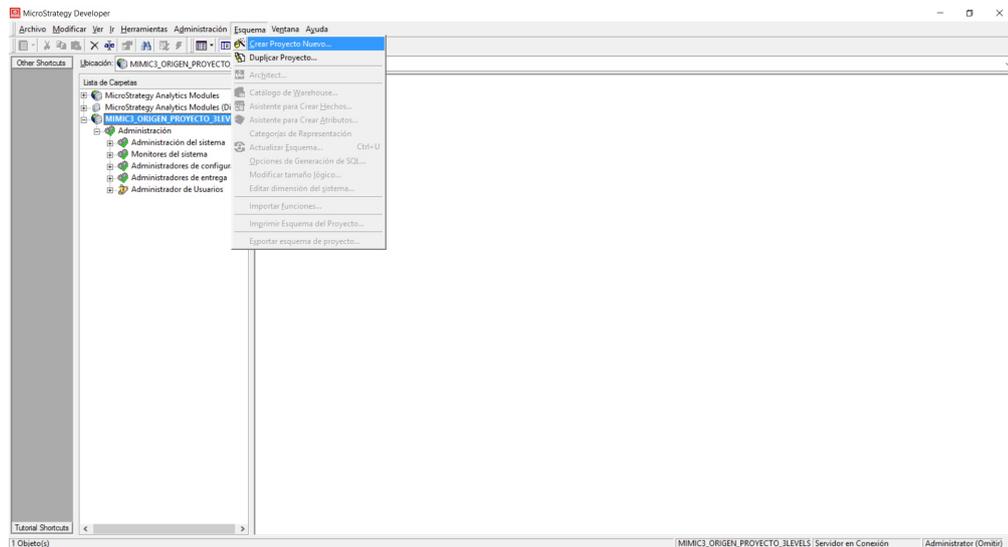


Figura D.78: Acceso a la creación de un nuevo proyecto.

De esta manera, en la Figura D.79 se accede a un nuevo asistente de definición de un nuevo proyecto de MicroStrategy Developer.

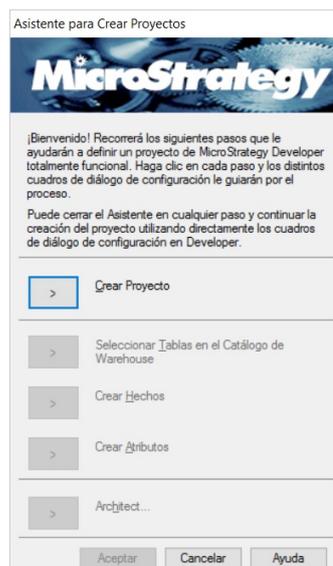


Figura D.79: Asistente de creación de proyecto.

Tras pulsar el botón de ‘Crear Proyecto’, se deben rellenar una serie de datos identificativos básicos relativos al nuevo proyecto, así como establecer una serie de parámetros mínimos como son la ruta en la que depositar los documentos que MicroStrategy genere o la disponibilidad en el proyecto de una cuenta de usuario invitado.

Esta parametrización queda reflejada en la captura de pantalla de la Figura D.80.

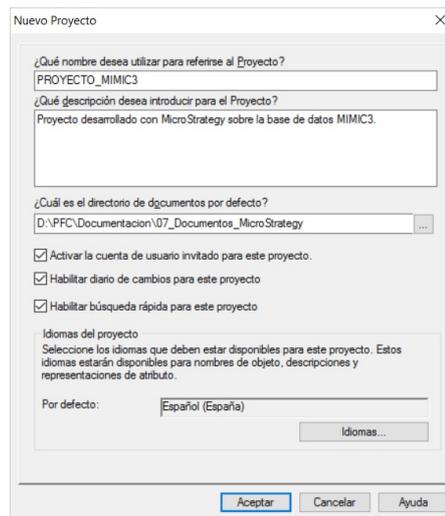
La imagen muestra una ventana de diálogo titulada 'Nuevo Proyecto'. El contenido incluye: un campo de texto con el nombre 'PROYECTO_MIMIC3'; un campo de descripción con el texto 'Proyecto desarrollado con MicroStrategy sobre la base de datos MIMIC3.'; un campo de directorio de documentos con la ruta 'D:\PFC\Documentacion\07_Documentos_MicroStrategy'; tres casillas de verificación activadas: 'Activar la cuenta de usuario invitado para este proyecto.', 'Habilitar diario de cambios para este proyecto' y 'Habilitar búsqueda rápida para este proyecto.'; una sección de idiomas con el texto 'Idiomas del proyecto' y 'Seleccione los idiomas que deben estar disponibles para este proyecto. Estos idiomas estarán disponibles para nombres de objeto, descripciones y representaciones de atributo.', donde el idioma seleccionado es 'Español (España)'. En la parte inferior hay botones para 'Aceptar', 'Cancelar' y 'Ayuda'.

Figura D.80: Definición de parámetros generales del proyecto.

Tras pulsar el botón de ‘Aceptar’, en la Figura D.81 se procede a la creación del nuevo proyecto.

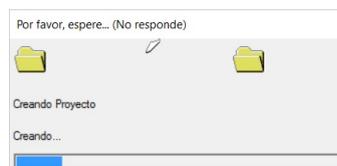


Figura D.81: Creación del nuevo proyecto.

Tras esto se solicita la especificación de los roles asociados al usuario actual (ver la Figura D.82).

Una vez creado el proyecto, el siguiente paso consistirá en especificar la lista de tablas pertenecientes al warehouse cuya información desea ser explotada, lo cual queda reflejado en la Figura D.83.

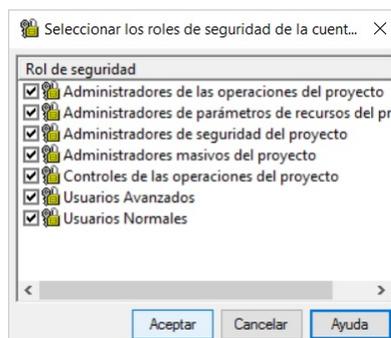


Figura D.82: Asignación de roles al usuario actual en el proyecto.

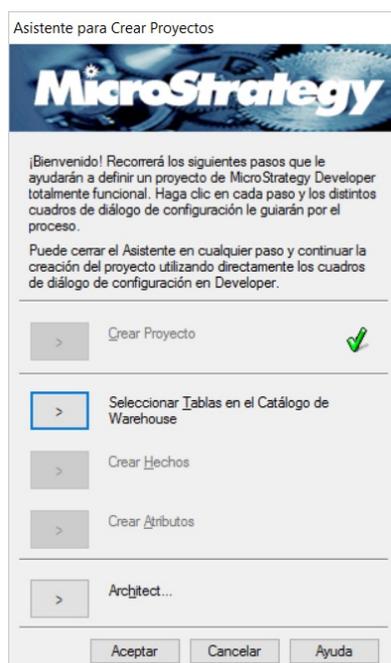


Figura D.83: Acceso al catálogo de Warehouse del proyecto.

Tras acceder a dicho menú de selección de tablas integrantes del catálogo de warehouse del proyecto, se muestra el formulario reflejado en la Figura D.84.

La lista de posibles orígenes de datos proporcionados por el combo no llega en ningún momento a referenciar el warehouse de interés, tal y como puede apreciarse en la Figura D.85.

Así que se accederá al botón de 'Nuevo' como muestra la Figura D.86.

A continuación se definirá el tipo de warehouse que se empleará para este nuevo proyecto (ver Figura D.87).

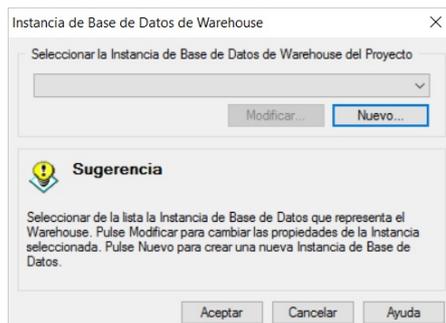


Figura D.84: Selección de instancia asociada al Warehouse.

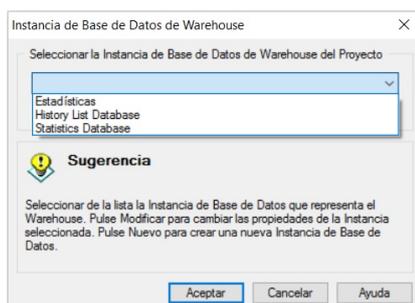


Figura D.85: Instancia de Warehouse no disponible en la selección.



Figura D.86: Asistente de creación de la instancia de Warehouse.

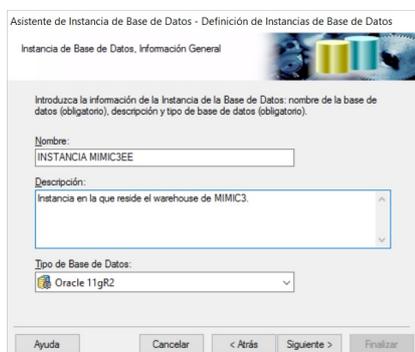


Figura D.87: Tipo de base de datos de la instancia de Warehouse.

En la Figura D.88 se especificará el ODBC a emplear (creado con anterioridad) para establecer la conexión con dicho warehouse.

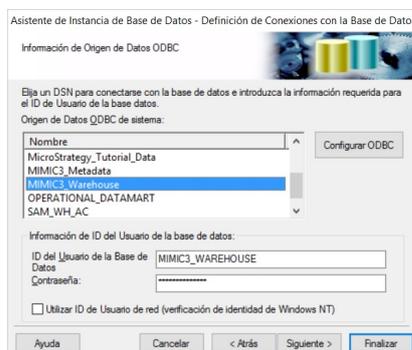


Figura D.88: Selección del ODBC correspondiente al Warehouse.

Después, se mostrará en la Figura D.89 un breve resumen final de los parámetros introducidos.

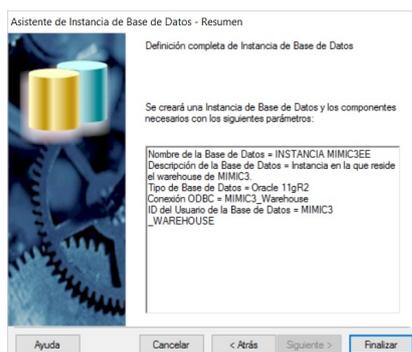


Figura D.89: Resumen de parámetros de la instancia de Warehouse.

Y el nuevo catálogo de warehouse queda parametrizado, como se ve en la Figura D.90.

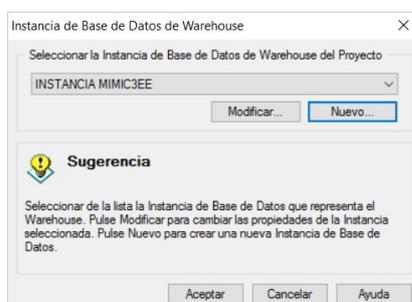


Figura D.90: Selección de la instancia de Warehouse recién creada.

En caso de no localizarse objetos (tablas, vistas, sinónimos...) en el esquema definido como catálogo de warehouse, se obtiene una advertencia como la de la Figura D.91.

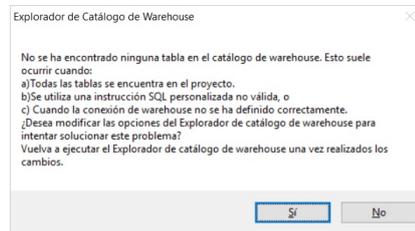


Figura D.91: Advertencia de catálogo de warehouse vacío en el proyecto.

Por tanto, antes de continuar, conviene conceder permisos de consulta sobre las tablas del esquema 'MIMIC3_OWNER' al esquema 'MIMIC3_WAREHOUSE', así como crear sinónimos desde 'MIMIC3_WAREHOUSE' hacia los objetos de 'MIMIC3_OWNER' sobre los que se acaban de conceder permisos de consulta.

Para ello, mediante la herramienta SQL Developer, se ejecutarán (ver Figura D.92) las concesiones de permisos de consulta sobre las tablas ya creadas y rellenas (desde el usuario 'MIMIC3_OWNER', mediante un programa PL/SQL no almacenado).

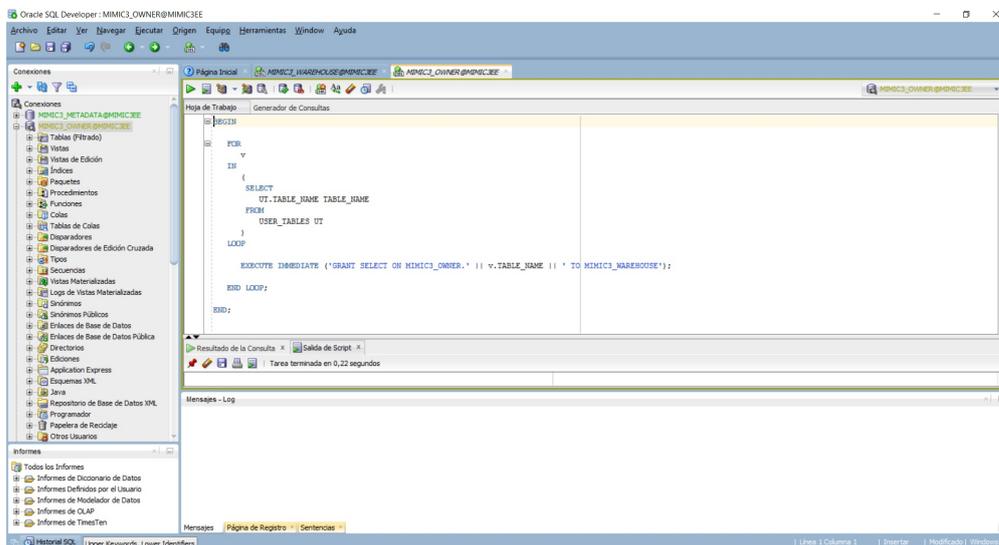


Figura D.92: Concesión de permisos al usuario de conexión de MicroStrategy.

A continuación, se procede a crear los respectivos sinónimos (desde el usuario 'MIMIC3_WAREHOUSE') como se ve en la Figura D.93.

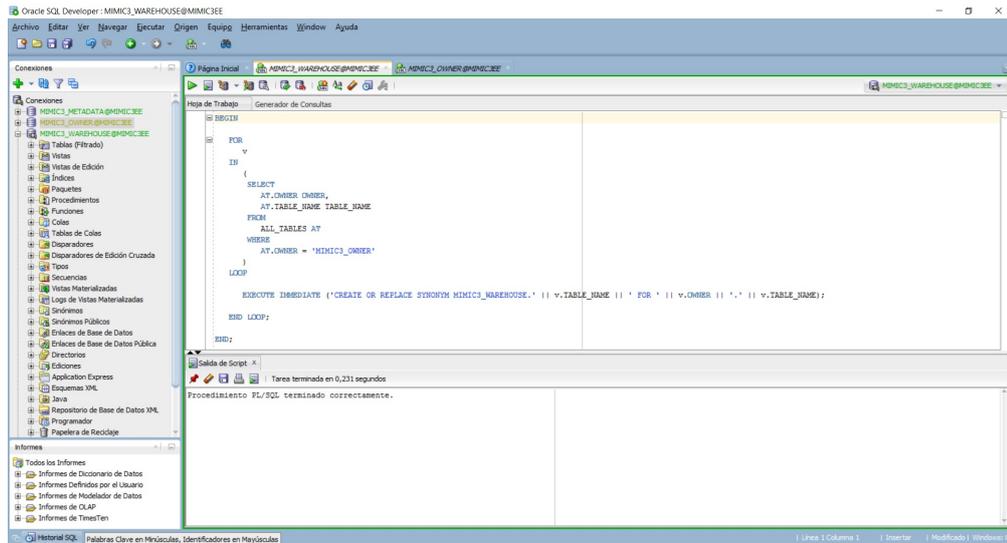


Figura D.93: Creación de sinónimos hacia objetos del usuario propietario.

De esta manera, se puede repetir el intento de creación del catálogo de warehouse pulsando el botón de ‘Sí’, pero se detecta el warehouse vacío en la Figura D.94.

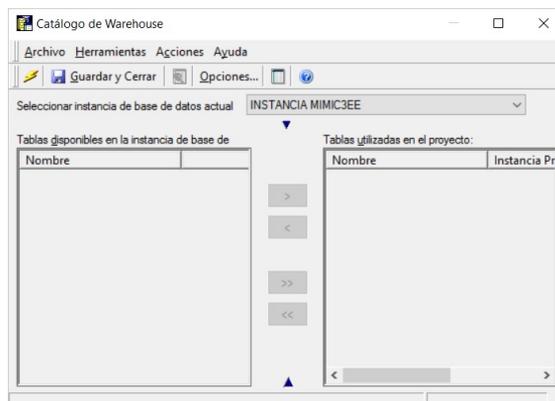


Figura D.94: El catálogo de warehouse continúa vacío.

Para resolver esta situación, será necesario modificar la sentencia SQL con la que MicroStrategy trata de conocer qué tablas residen en el esquema de Warehouse y que, muy posiblemente, acceda a la tabla del catálogo llamada ‘USER_TABLES’/‘ALL_TABLES’, pero no a la denominada ‘USER_SYNONYMS’/‘ALL_SYNONYMS’.

Pulsando el botón de ‘Opciones’ se accede al formulario de la Figura D.95 en el cual, primero se accede a la etiqueta de ‘Configuración de Leído’.

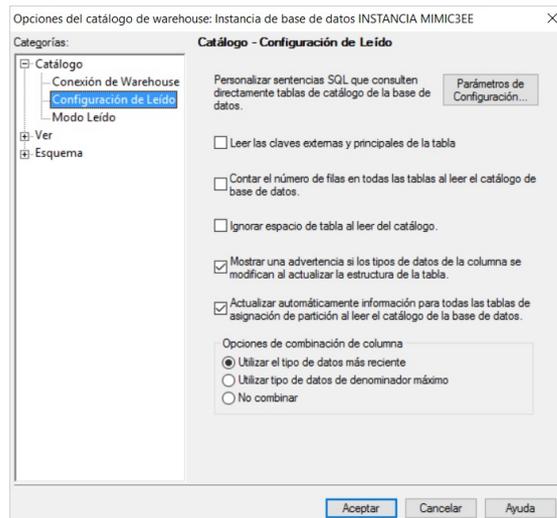


Figura D.95: Formulario de opciones del catálogo de warehouse.

Posteriormente, se accede al botón de ‘Parámetros de Configuración’, donde se pueden modificar estas consultas SQL de recuperación de objetos de Oracle (tablas, vistas, columnas...) disponibles en el warehouse, lo que se ve en la Figura D.96.

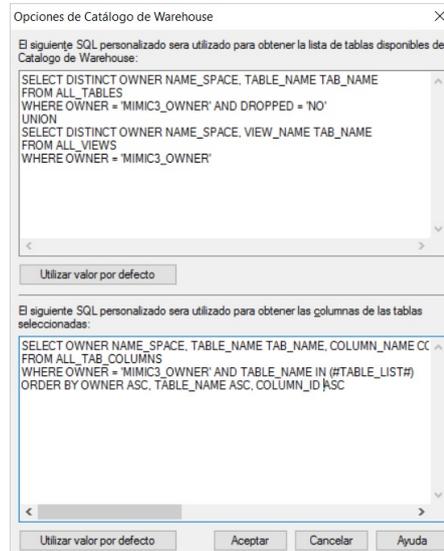


Figura D.96: Configuración de sentencia SQL de carga del catálogo de warehouse.

Basta con recargar el warehouse y aparecen (Figura D.97) todos los objetos deseados, de los cuales se elegirán (Figura D.98) aquéllos que se desee que integren el catálogo de warehouse.

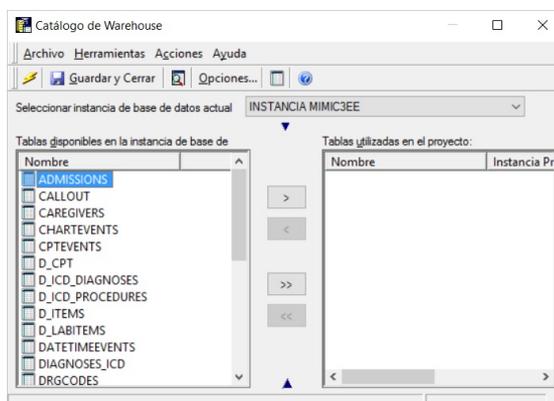


Figura D.97: Catálogo de warehouse correctamente cargado con objetos disponibles.

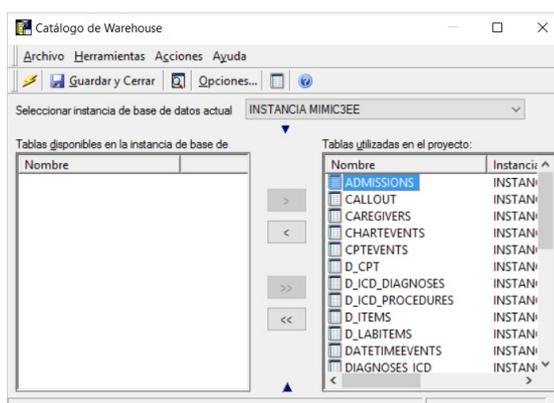


Figura D.98: Selección de objetos utilizables en el proyecto.

Tras pulsar el botón de 'Guardar y Cerrar', es posible establecer un comentario de lo realizado en la Figura D.99.

Y, tras pulsar en 'Aceptar', la información de tablas, columnas, tipos de datos... será respaldada al repositorio de Metadata que la herramienta dispone en Oracle, como se muestra en la Figura D.100.

Puesto que, por el momento, no se va a proceder a definir Hechos, Atributos u otros objetos de esquema del proyecto (como se ve en la Figura D.101), aquí se detiene el proceso de definición del proyecto.

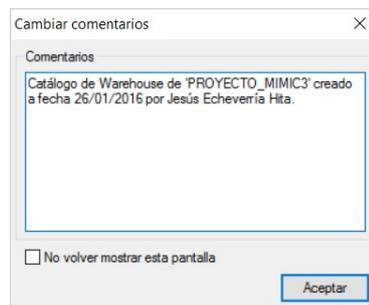


Figura D.99: Comentarios tras los cambios en el catálogo de warehouse.

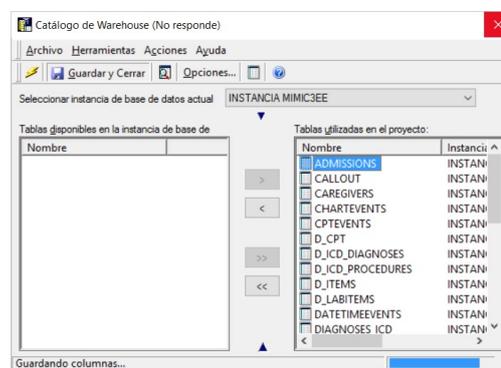


Figura D.100: Respaldo del catálogo de warehouse al repositorio de Metadata.

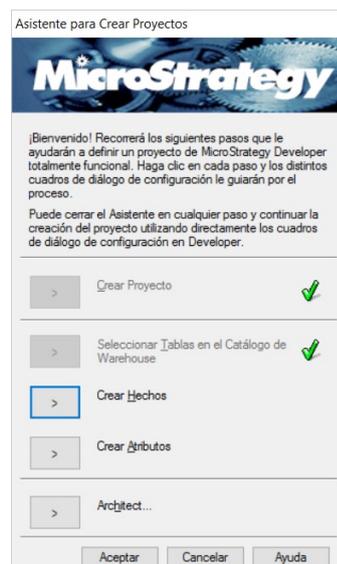


Figura D.101: Posibilidad de creación de objetos de esquema del proyecto.

Tras pulsar el botón de ‘Aceptar’, un mensaje mostrado en la Figura D.102 indica cómo poder proseguir más adelante con el resto de definiciones empleando el propio MicroStrategy Developer incluido en la suite.

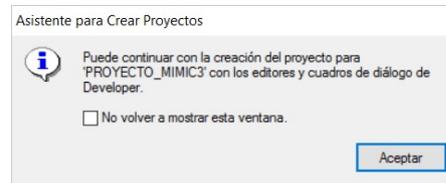


Figura D.102: Finalización del asistente de creación de proyectos.