



UNIVERSIDADE DA CORUÑA

Departamento de Computación

**EVOLUCIÓN TECNOLÓGICA E  
INGENIERÍA DE SOFTWARE**  
(UNA VISIÓN DESDE LA PERSPECTIVA DE LA EMPRESA)

**TESIS DOCTORAL**

**Doctorando:**

**Sebastián Bamonde Rodríguez**

**Directora:**

**Nieves Rodríguez Brisaboa**

**A Coruña, 2013**



**Ph. D. Thesis supervised by**  
**Tesis doctoral dirigida por**  
**Tese doutoral dirixida por**

**Nieves Rodríguez Brisaboa**  
Departamento de Computación  
Facultade de Informática  
Universidade da Coruña  
15071 A Coruña (España)  
Tel: +34 981 167000 ext. 1243  
Fax: +34 981 167160  
brisaboa@udc.es



*a mi familia,  
muy especialmente a Soco*



# Agradecimientos

Este trabajo de tesis lo debo al apoyo y empuje de la Doctora Nieves Rodríguez Brisaboa, a quien he tenido la satisfacción de tener como alumna y ver después como elegía para su desarrollo como docente el ámbito de las Bases de Datos. Sin su dirección, asesoramiento y ayuda no me habría sido posible desarrollarlo y sobre todo darle sentido como tesis doctoral.

Quiero expresar también mi agradecimiento al apoyo que siempre he tenido de los compañeros del Laboratorio de Base de Datos de la Facultad de Informática da Coruña, especialmente a Luis González Ares con quien he compartido asignaturas durante varios años. El camino que él recorrió, en el desarrollo de su tesis, ha sido de gran ayuda para mí y sus consejos siempre han resultado acertados hasta al punto de poder considerarlo codirector de esta tesis. Gracias Luis por el esfuerzo de haberla leído con tanto detalle y por todas las sugerencias que me has hecho.

También quiero mencionar a Enrique Seoane y Alida Alvaredo quienes "heredaron" el sistema que constituye la segunda parte de este trabajo y vivieron en primera persona su adaptación a los cambios tecnológicos y funcionales que el tiempo ha ido demandando. Su ayuda ha sido fundamental para recordar ese camino de evolución.

Agradezco la colaboración prestada por otras personas con las que pude compartir mi experiencia profesional y especialmente a aquellas que me han ayudado a recordar hechos y fechas a los que la memoria no alcanzaba, citarlos a todos sería muy largo y siempre correría el riesgo de olvidarme de alguno.

Por último, y para mí lo más importante, quiero expresar el agradecimiento a mi familia, mi esposa Soco en primer lugar por haberle robado el tiempo dedicado a este trabajo tardío, a mis hijos Sebastián, su esposa Lola, Santiago y Jesús, a mis nietos, que con el tiempo sabrán valorarlo, y a mis padres que siempre han sido el ejemplo a seguir.





# Resumen

Después de más de 35 años de dedicación simultánea a la docencia y a la profesión informática, ésta en el ámbito de la empresa privada, al cerrar esta última actividad se me presenta la oportunidad de terminar mi tesis doctoral, antes negada por la dificultad de encontrar continuidad en el tiempo necesario para poder hacerlo. El largo camino recorrido en la empresa ha hecho posible que participase en una gran cantidad de proyectos, ejerciendo distintos roles: programador, analista o jefe de proyecto, administrador de base de datos y teleproceso, director de un departamento importante de desarrollo de proyectos y director de arquitectura de aplicaciones y metodología. Estos proyectos y actividades me han permitido disfrutar de la experiencia de usar múltiples tecnologías y plataformas, tanto de ejecución como de desarrollo, realizando, orientando o dirigiendo el desarrollo de aplicaciones y la selección de tecnologías, soluciones y plataformas en distintos ámbitos y en diferentes momentos del estado del arte. Las tecnologías empleadas han tenido mayor o menor difusión y distinto grado de permanencia en el mercado. La naturaleza de los proyectos ha sido muy variada, en la misma medida que lo han sido los contextos para los que se planteaban las aplicaciones. Las tecnologías y plataformas fueron evolucionando y con ello obligaron a actualizar y adaptar los sistemas, al mismo tiempo que se desarrollaban otros nuevos.

Este trabajo de tesis doctoral tiene como objetivo analizar la evolución de la tecnología informática en la empresa. Se apoya en la historia profesional vivida, extrayendo de esa experiencia conclusiones que puedan servir como aportaciones aplicables a entornos de trabajo similares al que aquí se plantea. Un objetivo fundamental es refrendar el papel fundamental de algunas prácticas de la Ingeniería del Software y de algunas soluciones de arquitectura que han resultado cruciales en esta experiencia, prácticas y soluciones a los que el tiempo ha puesto en valor. Así mismo se considera fundamental el análisis que se realiza de las decisiones que fue necesario tomar en la evolución de los sistemas para introducir cambios tecnológicos.



# Resumo

Despois de máis de 35 anos de dedicación simultánea á docencia e á profesión informática, ésta no ámbito da empresa privada, ao pechar esta última actividade preséntaseme a oportunidade de terminar a miña tese doutoral, antes negada pola dificultade de encontrar continuidade no tempo necesario para poder abordala. O longo camiño percorrido na empresa fixo posible que participase nunha gran cantidade de proxectos, exercendo distintos roles: programador, analista ou xefe de proxecto, administrador de base de datos e teleproceso, director dun departamento importante de desenvolvemento de proxectos e director de arquitectura de aplicacións e metodoloxía. Estes proxectos e actividades permitíronme gozar da experiencia de usar múltiples tecnoloxías e plataformas, tanto de execución coma de desenvolvemento, realizando, orientando ou dirixindo o desenvolvemento de aplicacións e a selección de tecnoloxías, solucións e plataformas en distintos ámbitos e en diferentes momentos do estado da arte. As tecnoloxías empregadas tiveron maior ou menor difusión e distinto grao de permanencia no mercado. A natureza dos proxectos foi moi variada, na mesma medida que o foron os contextos para os que se formulaban as aplicacións. As tecnoloxías e plataformas foron evolucionando e con iso obrigaron a actualizar e adaptar os sistemas, ao mesmo tempo que se desenvolvían outros novos.

Este traballo de tese doutoral ten como obxectivo analizar a evolución da tecnoloxía informática na empresa. Apóiase na historia profesional vivida, extraendo desa experiencia conclusións que poidan servir como achegas aplicables a ámbitos de traballo similares ao que aquí se formula. Un obxectivo fundamental é referendar o papel fundamental dalgunhas prácticas da Enxeñaría do Software e dalgunhas solucións de arquitectura que resultaron cruciais nesta experiencia, prácticas e solucións aos que o tempo puxo en valor. Así mesmo considérase fundamental a análise que se realiza das decisións que foi necesario tomar na evolución dos sistemas para introducir cambios tecnolóxicos.



# Abstract

After over 35 years of dedication to teaching and simultaneously to the computing profession, this in the private enterprise, closing this last activity I got the opportunity to finish my PhD, previously denied by the difficulty of finding continuity in the time needed to address it. The long journey in the private enterprise brought me the opportunity to participate in a lot of projects in different roles: programmer, analyst or project manager, database and teleprocessing administrator, director of a major development department and architecture and methodology director . These projects and activities have allowed me to enjoy the experience of using multiple technologies and platforms, both in execution and development, performing, directing or managing application development and selection of technologies, solutions and platforms in different areas and at different times state of the art. The technologies had varying diffusion and different degree of permanence in the market. The nature of the projects has been varied, to the same extent that they have been the contexts for which applications were raised. The technologies and platforms have evolved and thus forced to update and adapt the systems, while developing new ones.

This PhD work aims to analyze the evolution of information technology in the enterprise. It is based on the professional history lived, drawing conclusions from that experience that can serve as inputs applicable to similar work environments to that proposed here. A key objective is to endorse the role of some practices of software engineering and some architecture solutions that have been crucial in this experience, practices and solutions that the time has given value. It is also considered essential the analysis performed on the decisions that have been taken in the development of systems in order to introduce technological changes.



# El tiempo

Al ser el foco de esta tesis doctoral la evolución informática en la empresa, si tuviese un personaje lo sería "el tiempo". El discurrir de la profesión informática enfrentada a continuos cambios obliga a tenerlo siempre presente. El pasado es el legado que debemos respetar y aprender de él, lo que exige su conocimiento y análisis, evitando aquello que ha demostrado ser contraproducente para, finalmente, construir sobre él. Los sistemas que diseñamos difícilmente serán completamente nuevos, necesitan heredar, convivir y construirse sobre los que les precedieron. Si el pasado es importante el presente no lo es menos, es el espacio para la toma de decisiones donde se definen y construyen los sistemas de ese futuro que está llegando a cada instante, convirtiendo el presente en pasado.

La percepción del tiempo anterior no está lejos de lo que expresaba el poeta D. Jorge Manrique (1440-1479) en sus coplas a la muerte de su padre:

.....  
*Pues si vemos lo presente  
cómo en un punto ses ido  
e acabado,  
si juzgamos sabiamente,  
daremos lo non venido  
por pasado.  
Non se engañe nadi, no,  
pensando que ha de durar  
lo que espera  
más que duró lo que vio,  
pues que todo ha de passar,  
por tal manera.*  
.....





# La historia

Esta tesis doctoral está sustentada sobre la actividad profesional realizada en una entidad financiera de ámbito nacional durante los últimos 25 años. Corresponde a la última etapa de mi trayectoria profesional, la más larga y la de mayor riqueza, tanto por su duración como por la amplitud del campo de utilización de la tecnología y los medios disponibles.

La magnitud de los proyectos abordados y los cambios tecnológicos vividos suponen una experiencia irrepetible que permite tener una visión suficientemente rica de la joven historia de esta rama del conocimiento, que genéricamente denominamos "Informática", y de su aplicación en la empresa. La velocidad con la que evoluciona la Informática, hace desaparecer con rapidez el conocimiento adquirido en las situaciones pasadas y con ello la explicación de cómo y por qué se ha llegado al punto en el que nos encontramos, haciendo de ese modo más difícil el entendimiento de la situación actual y la evolución de los sistemas. La importancia de un trabajo como éste reside precisamente ahí, en dejar constancia de lo que ocurrió y por qué sucedió de la forma que lo hizo, analizando motivos y consecuencias. Para entender el presente y poder proyectar el futuro necesitamos conocer el pasado. Si lo que aquí se trata no se escribe y analiza ahora, esa parte de la historia se perderá para siempre, las personas que lo conocen irán desapareciendo, las empresas y su documentación también, pues van siendo absorbidas por otras que pueden tener su propia historia pero que harán desaparecer el conocimiento de experiencias como la que aquí se plantea, imposibilitando en el futuro el análisis y la extracción de las lecciones y conclusiones que de ello se puedan derivar.

Las tecnologías juegan un papel primordial en la modificación de la función de producción, posibilitando el desarrollo de nuevos productos o permitiendo nuevas formas de producirlos o comercializarlos. La importancia del impacto de la tecnología en el crecimiento económico la pone de manifiesto Robert Solow<sup>1</sup> [67] cuando cita que en los Estados Unidos, durante el período 1929-1982, la tercera parte

---

<sup>1</sup>Robert Solow (1924) fue Premio Nobel de Economía en 1978 por sus trabajos sobre el crecimiento económico.

del crecimiento económico fue debida a la evolución tecnológica. Si trasladásemos ese estudio al período 1986-2011 el impacto más importante lo encontraríamos, sin ningún género de duda, en el uso de las tecnologías de la información.

La historia en definición de E. Carr [12] es la ciencia que tiene como objeto de estudio el pasado de la humanidad y como método el propio de las ciencias sociales. Trata por tanto, de modo genérico, del estudio del comportamiento de hombre en sociedad, pero a diferencia de otras ciencias le interesa su comportamiento en el pasado. No se trata de una narración en sentido estricto de los hechos acontecidos, sino de analizarlos para entender el presente y poder anticiparse así al futuro [30]. El objetivo de esta tesis es analizar una parte de esta historia reciente en un contexto concreto, el de una empresa financiera, y con un alcance determinado, el de los Sistemas de Información. En las sociedades modernas "la información" es un factor de producción más, de la misma importancia que la tierra, el trabajo o el capital. El conocimiento de como evolucionó su uso y el de las tecnologías aplicadas, tiene el mismo interés que el estudio de la utilización de cualquiera de los otros factores de producción y por lo tanto es importante realizarlo.

En toda interpretación de tipo histórico hay siempre elementos subjetivos. En este trabajo los cambios que se describen no son interpretaciones, son hechos que sucedieron, vividos en primera persona y en algunos casos hasta también realizados en persona, como es el caso del proyecto que se describe en la Parte II de esta tesis. Sin embargo, los análisis y valoraciones que se hacen de los hechos si son subjetivos y obedecen a la interpretación del autor.

La idea del cambio en el tiempo expresada en el preámbulo de este documento la aplica también E. Carr [30] en la entrevista que aquí transcribo:

*El pasado no desaparece súbita y completamente en el presente. El dicho francés "le saisit le vif" (aferrarse a la vida) expresa la influencia que, en ocasiones, parece ejercer el pasado en el presente. Por otra parte, incluso cuando permanecen intactas las formas externas del pasado, su contenido está sujeto a un proceso de cambio continuo, que en ocasiones pasa desapercibido. La historia es producto de la interrelación de los principios de continuidad y cambio.*

Indudablemente esta cita se refiere a la "Historia" con mayúsculas, mientras que en este trabajo la idea está limitada al ámbito de los sistemas de información en la empresa y las tecnologías utilizadas.

# Trayectoria profesional y docente

Teniendo en cuenta la componente histórica que tiene este trabajo, dedicaré este capítulo del preámbulo a hacer una breve reseña de la trayectoria profesional y académica seguida, que en cierto modo también es historia, ya que la trayectoria que se describe ha sido similar a la de muchos profesionales que se incorporaron a la empresa en los años 70 y principios de los 80, antes de que las Facultades y Escuelas de Informática tuviesen la presencia en los estudios universitarios que hoy tienen. En el caso particular de Galicia los estudios de Informática no se implantaron hasta el año 1986 en el que nació la Escuela Universitaria de Informática de La Coruña, prácticamente coincidiendo con el inicio del período que vamos a analizar.

Me licencié en Ciencias (Sección de Matemáticas) en la Universidad de Santiago en el año 1974. En aquel momento los estudios de Matemáticas no ofrecían la posibilidad de una especialización más allá de lo que puede representar la elección de alguna asignatura optativa. El Plan de Estudios era de naturaleza muy teórica, orientado fundamentalmente a la formación de docentes en las enseñanzas medias o superiores. En este contexto, al realizar el tercer curso, tuve la posibilidad de asistir a unas clases optativas del lenguaje Fortran IV donde vi abrirse una horizonte diferente al docente anteriormente reseñado. En 1972, coincidiendo con el cuarto curso de los estudios de Matemáticas, se planteó una oferta de becas de la Fundación Barrié de la Maza para trabajar en el Centro de Cálculo de la Universidad de Santiago, financiado por la citada Fundación, y que en aquél momento estaba ubicado en la propia Facultad de Ciencias y dirigido por el profesor José M<sup>a</sup> Busta<sup>2</sup>. Conseguí la beca y simultáneamente, de nuevo pensando en la apertura del campo profesional, inicié estudios de Económicas que finalmente no pude concluir por incorporarme, en 1975, como profesor de Matemáticas y Estadística en la propia Facultad de Económicas<sup>3</sup>.

---

<sup>2</sup>José M<sup>a</sup> Busta (Lugo 1946-2008). Fue director del Centro de Cálculo de la Universidad de Santiago desde el año 1970.

<sup>3</sup>Colegio Universitario de La Coruña dependiente de la Universidad de Santiago de Compostela.

La experiencia profesional como becario se limitó al mundo del Fortran IV, desarrollando pequeñas soluciones para distintos departamentos de la universidad en el campo de la estadística o del análisis numérico. La plataforma objetivo fue inicialmente un IBM 1130 con 16K de memoria, capacidad muy pequeña para los tamaños de memoria a los que hoy estamos acostumbrados. En 1973 el Ministerio de Educación y Ciencia dotó al centro de cálculo con un terminal DCT-2000, conectado al ordenador Univac 1100 alojado en el propio Ministerio, que amplió sustancialmente la capacidad de proceso pero no la orientación del trabajo que realizaba.

En el año 1974, el Centro de Cálculo decidió abrirse a la empresa, iniciando una colaboración con los astilleros ASTANO<sup>4</sup> en un proyecto de diseño de tuberías, denominado SADAT<sup>5</sup>, al que fui asignado. La singularidad de este proyecto y la tecnología utilizada me ha llevado a incorporar una reseña del mismo al final de esta tesis (apéndice A. pág. 157) por considerar que también esta experiencia tiene un importante interés histórico. El inicio del trabajo se realizó desde el Centro de Cálculo de la Universidad, simultaneando el mismo con la finalización de los estudios de Matemáticas. Al concluir ese mismo año la Licenciatura se prorrogó la beca, trabajando a partir de entonces en el propio astillero. Finalmente, un año después, me incorporé a la plantilla de ASTANO en el Departamento Técnico, dentro del denominado Grupo de Programas Técnicos responsable en el astillero del desarrollo de la informática técnica. El ámbito de trabajo continuó siendo el mismo, muy cercano al análisis numérico y ahora también a la geometría, pero lejos de los sistemas de gestión empresarial que constituyen el foco de análisis de esta tesis. En esta misma etapa es cuando inicio mi carrera docente en la Universidad de Santiago (1975) como profesor, primero contratado y posteriormente asociado, en el citado Colegio Universitario de La Coruña.

En 1980 se produjo un giro profesional importante pasando a trabajar en el Sector Agropecuario, centrado alrededor de la fabricación de piensos compuestos, medicamentos para ganado, correctores de pienso, complementos alimenticios, selección, reproducción y cría de ganado porcino y producción avícola en granjas propias o en colaboración con granjeros. Este salto profesional hizo que abandonara el ámbito técnico para pasar de lleno al mundo de la gestión empresarial: Producción, Almacén, Ventas, Clientes, Contabilidad, Control Presupuestario, Nóminas etc. pasaron a ser a partir de entonces el ámbito de las aplicaciones que desarrollábamos quedando algún resquicio para la aplicación de los conocimientos adquiridos en la licenciatura de Matemáticas en aplicaciones como: Formulación de Piensos (programación lineal) y Selección Genética de Porcino (estadística)<sup>6</sup>. El pequeño

---

<sup>4</sup>Astilleros y Talleres del Noroeste. Actualmente Navantia.

<sup>5</sup>Sistema Astano de Diseño Automático de Tuberías.

<sup>6</sup>El programa de selección genética se realizaba bajo la dirección de la Universidad de Glasgow

tamaño de la instalación informática obligaba a jugar con frecuencia más de un papel en la misma, como fueron los de analista, técnico de sistemas y administrador de base de datos. Fue entonces cuando tuve la oportunidad de entrar en contacto con dos plataformas que marcaron profundamente mi futuro profesional y docente, concretamente las bases de datos y los entonces denominados monitores de teleproceso o gestores transaccionales.

Coincidiendo con el final de esta etapa (1985) participé activamente en el nacimiento de la Escuela Universitaria de Informática [28], invitado por el profesor José Luis Freire Nistal<sup>7</sup>, del que había sido alumno en la Facultad de Matemáticas, colaborando en la definición del primer Plan de Estudios y convirtiéndome desde su inicio en profesor de la misma, impartiendo al principio asignaturas de Programación y Arquitectura de la Información, para pasar posteriormente a la definición y puesta en marcha de las asignaturas de Base de Datos en los distintos cursos en la medida en la que la carrera iba creciendo.

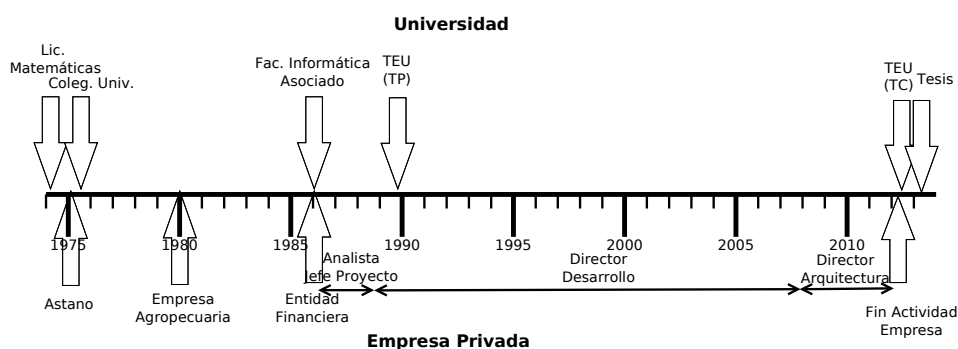
Al final del año 1985 se produjo un nuevo cambio profesional cuando se me ofreció la posibilidad de incorporación a una entidad financiera de ámbito nacional, a la que en lo sucesivo me referiré como la Entidad Financiera, que en aquel momento había iniciado una profunda transformación de sus sistemas, abandonando la plataforma anterior, basada en ordenadores Bull, en favor de sistemas IBM. Este es el entorno profesional en el que se va a centrar esta tesis y se explica en mayor detalle en la primera parte de esta tesis. En los 27 años de trabajo en esta empresa hubo una primera etapa en la que la función principal fue la de Analista/Jefe de Proyecto. Fue la etapa del desarrollo del sistema que constituye la segunda parte de esta tesis, en la que además tuve la responsabilidad de formar, en bases de datos relacionales y en el gestor transaccional, a los distintos equipos de trabajo que realizaban la migración de la plataforma, actuando también como soporte técnico para esos equipos y como analista en aplicaciones como Préstamos o Tarjetas de Débito. En una segunda fase, que comenzó en 1989, me hice cargo de la dirección del departamento de desarrollo informático que integraba en aquél momento la plataforma central, la plataforma de oficinas y el incipiente desarrollo departamental. El desarrollo departamental pasó más adelante a otro departamento pero se incorporó a éste el desarrollo en Internet. En esta etapa que duró hasta el año 2007 el trabajo fundamental fue el de dirección de equipos de proyecto, colaborando con otras unidades en la selección de tecnologías y soluciones de negocio. El número de personas de los equipos superaba el centenar entre personal de la propia entidad y personal externo. A partir del año 2007, ya hasta el final de mi actividad en la empresa, pasé a dirigir una unidad más reducida, alrededor de 12 personas, cuya misión era la de selección de tecnologías, realización de prototipos, desarrollo de proyectos con nuevas tecnologías y soporte técnico y metodológico al departamento de desarrollo.

---

<sup>7</sup>J.L. Freire fue Director comisionado para la puesta en marcha de la Escuela y primer Decano de la Facultad de Informática (1991).

En diciembre de 1989 accedí al cuerpo de Titulares de Escuela Universitaria en el Área de Lenguajes y Sistemas con dedicación a tiempo parcial por continuar con el trabajo en la empresa privada.

Mi actividad en la Entidad Financiera terminó en Marzo de 2012 solicitando en ese momento mi incorporación a tiempo completo a la Facultad de Informática que se concretó en Setiembre de ese mismo año. Con el cambio de dedicación se me encarga la coordinación del itinerario de Sistemas de Información, correspondiente al Grado en Ingeniería Informática, y de dos asignaturas que nacen con el nuevo Master en Ingeniería Informática: Planificación Estratégica de Sistemas de Información y Dirección de Proyectos, ambas en directa relación con el contenido de este trabajo.



**Figura 1:** Trayectoria profesional y docente

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Motivación . . . . .	1
1.2. Objetivos . . . . .	1
1.3. Estructura del documento . . . . .	3
<b>I Evolución de la tecnología informática en la empresa</b>	<b>7</b>
<b>2. Introducción</b>	<b>9</b>
<b>3. Situación tecnológica en 1985</b>	<b>15</b>
3.1. Objetivos . . . . .	15
3.2. Plataforma hardware y software inicial . . . . .	15
3.3. Arquitectura de aplicaciones . . . . .	16
3.4. Metodología de desarrollo . . . . .	18
3.4.1. Administración de Datos . . . . .	19
3.4.2. Gestión de proyectos . . . . .	20
3.4.3. Gestión de la calidad . . . . .	21
<b>4. El proyecto de migración de plataforma</b>	<b>23</b>
4.1. Objetivos . . . . .	23
4.2. Condicionantes técnicos . . . . .	23
4.3. Fases de la migración . . . . .	25
4.4. Planificación de los procesos "batch" . . . . .	26
4.5. Conclusiones . . . . .	27

<b>5. Migración a bases de datos relacionales</b>	<b>29</b>
5.1. Objetivos . . . . .	29
5.2. Situación general . . . . .	29
5.3. Cronología . . . . .	30
5.4. Problemática de este tipo de migración . . . . .	31
5.5. Conclusiones . . . . .	32
<b>6. Evolución del Terminal Financiero</b>	<b>33</b>
6.1. Objetivos . . . . .	33
6.2. Historia inicial . . . . .	34
6.3. Evolución en la Entidad Financiera . . . . .	36
6.3.1. El terminal OS/2 . . . . .	36
6.3.2. El terminal Windows . . . . .	40
6.3.3. El terminal Eclipse-RCP . . . . .	40
6.4. Conclusiones . . . . .	44
<b>7. Incorporación a Internet: Webs y soluciones de Banca Electrónica</b>	<b>45</b>
7.1. Objetivos . . . . .	45
7.2. Contexto general . . . . .	45
7.3. Planteamiento . . . . .	46
7.4. La Banca Electrónica . . . . .	47
7.5. Conclusiones . . . . .	51
<b>8. Gestión documental</b>	<b>53</b>
8.1. Objetivos . . . . .	53
8.2. Las información documental en las entidades financieras . . . . .	53
8.3. Planteamiento y evolución . . . . .	54
8.4. Proyecto de base de datos documental . . . . .	57
8.5. Conclusiones . . . . .	58
<b>9. Arquitectura de integración de aplicaciones y BPM</b>	<b>59</b>
9.1. Objetivos . . . . .	59
9.2. Planteamiento general . . . . .	59
9.3. Planteamiento en la Entidad Financiera . . . . .	61
9.4. Conclusiones . . . . .	62



---

<b>10. Sistemas Informacionales y "Data warehouse"</b>	<b>65</b>
10.1. Objetivos . . . . .	65
10.2. Situación general . . . . .	65
10.3. Planteamiento en la Entidad Financiera . . . . .	66
10.4. Conclusiones . . . . .	69
<b>11. Lecciones aprendidas</b>	<b>71</b>
<b>II La persistencia de las soluciones</b>	<b>75</b>
<b>12. Introducción</b>	<b>77</b>
<b>13. Antecedentes</b>	<b>81</b>
13.1. Contexto de desarrollo del proyecto . . . . .	81
13.2. Plataforma tecnológica . . . . .	83
<b>14. El proyecto concreto</b>	<b>87</b>
14.1. El FAP . . . . .	87
14.2. Mensajes de control . . . . .	91
14.3. Estados del FAP . . . . .	93
14.4. Operaciones . . . . .	94
14.5. Integración con aplicaciones . . . . .	95
<b>15. El diseño planteado</b>	<b>97</b>
15.1. Subsistema de Entrada de Mensajes . . . . .	98
15.1.1. Control del FAP . . . . .	98
15.1.2. Transformación de formato . . . . .	100
15.2. Integración con el Servidor Financiero . . . . .	102
15.3. Subsistema de Emisión de Mensajes . . . . .	103
15.4. Bases de datos . . . . .	103
15.4.1. Parámetros Aplicación/Sesión . . . . .	104
15.4.2. Control de Aplicación/Sesión . . . . .	105
15.4.3. Control de Totales . . . . .	105
15.4.4. Control de Canales . . . . .	105
15.4.5. Registro de mensajes ("log") . . . . .	105
15.5. Terminal de Control y Consola de Operación . . . . .	106
15.6. Integración con la Planificación de Procesos . . . . .	106

<b>16. Arranque y recuperación</b>	<b>109</b>
16.1. Juego de estados . . . . .	109
16.2. Arranque del sistema . . . . .	111
16.3. Recuperación del sistema . . . . .	112
<b>17. Operación y monitorización</b>	<b>115</b>
17.1. Consola de operación . . . . .	115
<b>18. Evolución del sistema</b>	<b>117</b>
<b>19. Algunos datos de volúmenes</b>	<b>119</b>
<b>20. Lecciones aprendidas</b>	<b>123</b>
20.1. Visión del diseño . . . . .	123
20.2. Visión de los requisitos no funcionales . . . . .	126
20.3. Visión de la integración empresarial . . . . .	127
<b>III La Administración de Datos</b>	<b>129</b>
<b>21. Introducción</b>	<b>131</b>
<b>22. Administración de Datos</b>	<b>133</b>
22.1. Principios de la Administración de Datos . . . . .	133
22.2. El valor de la información . . . . .	134
22.3. Resultados obtenidos . . . . .	136
<b>23. Diccionario de Datos</b>	<b>139</b>
23.1. Concepto . . . . .	139
23.2. El entorno . . . . .	140
23.3. Modelo de información . . . . .	142
23.4. Funciones soportadas . . . . .	143
23.5. Datos cuantitativos . . . . .	143
<b>24. Lecciones aprendidas</b>	<b>147</b>
<b>IV Resumen final</b>	<b>149</b>
<b>25. Conclusiones</b>	<b>151</b>

---

<b>A. SADAT: Sistema Astano de Diseño Automático de Tuberías</b>	<b>157</b>
A.1. El proyecto . . . . .	157
A.2. El sistema gráfico Adage . . . . .	159
<b>B. Proyecto Cajeros: Estructura de los mensajes</b>	<b>161</b>
B.1. Cabeceras . . . . .	161
B.1.1. Cabecera de control . . . . .	161
B.1.2. Cabecera de datos . . . . .	162
B.2. Cuerpo . . . . .	163
<b>C. Diccionario de Datos: Jerarquías</b>	<b>165</b>
<b>D. Diccionario de Datos: Modelo de información</b>	<b>169</b>
<b>E. Siglas y acrónimos</b>	<b>177</b>
<b>Bibliografía</b>	<b>183</b>



# Índice de figuras

1. Trayectoria profesional y docente . . . . .	XXII
2.1. Hitos más relevantes . . . . .	13
2.2. Potencia instalada . . . . .	14
3.1. Servidor y Terminal Financiero . . . . .	16
4.1. Proyecto de migración de plataforma . . . . .	25
5.1. Evolución a BD relacionales . . . . .	31
6.1. Evolución del Terminal Financiero . . . . .	37
7.1. Arquitectura de 4 niveles . . . . .	48
7.2. Evolución web y Banca Electrónica . . . . .	50
8.1. Evolución temporal de los sistemas documentales . . . . .	57
10.1. Arquitectura del Data Warehouse . . . . .	67
13.1. Topología de la red . . . . .	82
13.2. Arquitectura de tres niveles . . . . .	83
13.3. Esquema de comunicación . . . . .	84
13.4. Comunicación técnica . . . . .	85
14.1. Estructura de los mensajes . . . . .	88
14.2. Estados del FAP . . . . .	89
14.3. Duración de sesiones . . . . .	90
14.4. Estados FAP en la Entidad Financiera . . . . .	93
14.5. Integración funcional . . . . .	95

---

15.1. Pila de comunicaciones OSI . . . . .	98
15.2. Subsistema de entrada de mensajes . . . . .	99
15.3. Proceso cabecera FAP y OSI . . . . .	100
15.4. Mapas, estructuras y transformaciones . . . . .	101
15.5. Subsistema de integración . . . . .	102
15.6. Modelo conceptual global . . . . .	104
16.1. Estados de la aplicación . . . . .	110
16.2. Estados de los totales . . . . .	111
16.3. Tablas en memoria . . . . .	112
18.1. Evolución del sistema . . . . .	118
19.1. Mensajes entrantes por día . . . . .	120
19.2. Mensajes entrantes cada cuarto de hora . . . . .	121
22.1. Administración de datos y diseño de base de datos . . . . .	135
22.2. Evolución del volumen de datos . . . . .	136
23.1. Diccionario y entornos . . . . .	141
23.2. Modelo de información del Diccionario . . . . .	142
23.3. Evolución - Número de aplicaciones . . . . .	144
23.4. Evolución - Número de tablas . . . . .	144
23.5. Evolución - Número de mensajes . . . . .	145
23.6. Evolución - Número de programas . . . . .	145
23.7. Evolución - Número de entidades . . . . .	146
B.1. Cabecera del FAP . . . . .	161
B.2. Cabecera de datos . . . . .	162
C.1. Jerarquía EDPS (Extended Data Processing Structure) . . . . .	166
C.2. Jerarquía SAS (Structured Analysis) . . . . .	166
C.3. Jerarquía SDS (Structured Development Structure) . . . . .	167
D.1. Documentación de procesos . . . . .	170
D.2. Documentación transacciones . . . . .	171
D.3. Documentación programas . . . . .	172
D.4. Documentación ficheros . . . . .	173
D.5. Documentación datasets . . . . .	174

D.6. Documentación registros . . . . .	175
D.7. Documentación mensajes . . . . .	176





# Capítulo 1

## Introducción

### 1.1. Motivación

Treinta y cinco años de experiencia profesional es tiempo suficiente para haber sido testigo directo de muchas situaciones de cambio en la utilización de la informática. El ejercicio de la profesión informática en diferentes tipos de empresas me ha permitido vivir en primera persona cómo se produjeron esos cambios y las dificultades que presenta la incorporación de nuevas tecnologías en las organizaciones.

Ahora, con el tiempo suficiente para hacerlo, se me presenta la oportunidad de reflexionar sobre esa experiencia y extraer de la misma conclusiones que puedan ser útiles en situaciones similares a las que en este trabajo se analizan, realizando de esta forma una aportación a la historia de la informática en la empresa en el final del siglo XX y principio del siglo XXI.

### 1.2. Objetivos

El objetivo de esta tesis doctoral es analizar en el tiempo la evolución de la tecnología informática en una empresa. Se trata de conocer cómo fue esa evolución utilizando dos perspectivas: la del cambio y la de la continuidad, representadas, respectivamente, por la incorporación de nuevas tecnologías y por la persistencia de las soluciones. La primera de las dos perspectivas explica y analiza cómo se han producido los cambios tecnológicos que tuvieron un mayor impacto para la empresa, bien por su magnitud o bien por su relevancia; la segunda es dual de la primera pues, utilizando un proyecto concreto, se centra en cuáles han sido los criterios de diseño que permitieron su vida útil durante más de 25 años. Entre ambas perspectivas hay

un nexo de unión en el papel fundamental que tuvo la Administración de Datos en esta evolución, que se expone como tercera parte de este trabajo.

Esta Tesis no se ajusta al formato clásico de un trabajo experimental en Ciencias o Ingeniería [15]; no existe una hipótesis que se vaya a validar y por lo tanto tampoco el diseño de un experimento de laboratorio para hacerlo. En ese sentido su metodología podría estar más cercana a la de los trabajos en Ciencias Sociales, especialmente en su primera parte, pero sin olvidar que el tema está en el marco de la evolución tecnológica y no en el de la evolución social. Su parte experimental está sustituida por una observación directa de la realidad durante un período de tiempo significativo, 25 años. Los resultados se han obtenido de la observación, el conocimiento y el análisis de los cambios tecnológicos que hubo en una empresa, de cómo estos cambios se fueron incorporando y de cuáles fueron los criterios para hacerlo, señalando especialmente todas aquellas situaciones que puedan ser útiles en casos similares. La validez de las conclusiones que se obtienen es fruto del trabajo llevado a cabo en una entidad financiera durante un período de tiempo dilatado y sería difícil realizar un contraste adicional por tratarse de hechos reales que no puede repetirse en un laboratorio, solamente podrán compararse en un futuro con situaciones reales similares.

El trabajo realizado es sustancial, por la importancia del tema, por el período de tiempo abarcado y por la variedad de tecnologías y situaciones tratadas, y es original por obedecer a las observaciones y análisis del propio autor.

La aportación de este trabajo es doble:

1. *Escribir una historia de la evolución de la informática, en los últimos 25 años, desde la perspectiva de una entidad financiera importante.*
2. *Analizar cómo se produjo esa evolución, qué soluciones se arbitraron y extraer conclusiones de la forma en la que se hizo, para su aplicación en futuras situaciones similares.*

### 1.3. Estructura del documento

El documento se compone de cuatro partes y cinco apéndices. Cada una de las partes comienza con una introducción, continúa con el análisis de cada uno de los casos planteados, estableciendo para cada uno de ellos conclusiones, y termina con una síntesis de las lecciones aprendidas, considerando el conjunto de las conclusiones y las lecciones la aportación de esta tesis.

- Parte I. Evolución tecnológica en la empresa.

En esta parte se describe y analiza un período de algo más de 25 años que va desde el año 1986 hasta finales del año 2011. Para efectuar este análisis se han elegido siete escenarios de evolución tecnológica que comienzan con un cambio total de la plataforma central de proceso. Cada uno de los escenarios se centra en un cambio tecnológico del que se describe su motivación, las decisiones para su incorporación y la forma en la que se acometió, terminando con las conclusiones que se pueden extraer de la experiencia relatada.

- Parte II. La persistencia de las soluciones.

Esta parte analiza un desarrollo concreto: el de la integración de una entidad financiera en una red de cajeros. Se trata de un sistema con características de criticidad muy significativas, que habiendo sido diseñado en el año 1986 y puesto en producción en el año 1987, continúa prestando servicio en el año 2013. El foco en esta parte es la importancia del diseño y cómo un buen diseño permite la fácil adaptación en un entorno sometido a una evolución tecnológica como la que se describe en la primera parte de este trabajo.

- Parte III. La Administración de Datos.

La tercera parte de la tesis se ha dedicado a una práctica de la Ingeniería del Software, la Administración de Datos, cuya importancia ya se pone de manifiesto en las dos partes anteriores. Se trata de una práctica muy poco desarrollada en la mayor parte de las organizaciones y que la experiencia en la que nos apoyamos demuestra que es de la máxima importancia. En esta parte se explican sus fundamentos y una forma de abordarla en una instalación informática compleja que resultó exitosa.

- Parte IV. Resumen final.

Se hace un resumen del trabajo realizado explicando en que forma se han conseguido los objetivos planteados. También se tratan en esta parte aspectos transversales que fueron tratados de forma particular en las conclusiones de cada capítulo.

- Apéndices.

- A. SADAT: Sistema Astano de Diseño Automático de Tuberías.

Se describe un proyecto singular que utilizó una tecnología de diseño gráfico muy avanzada para la época en la que se desarrolló (1974-1979). Tiene interés histórico pues no es fácil imaginarse un entorno como el sistema gráfico interactivo utilizado, en un momento del tiempo en el que en el ordenador principal trabajaba con fichas perforadas y los programadores codificaban sus programas en papel.

- B. Proyecto Cajeros: Estructura de los mensajes.

Se describen con mayor detalle la estructura de los mensajes que fluyen en el sistema de Cajeros. La descripción se ha hecho a un nivel suficiente para que se pueda entender el diseño, especialmente la transformación de mensajes, sin llegar al detalle de estructura concreta, formatos y longitudes por tratarse de un protocolo privado y no disponer en este momento de esa información.

- C. Diccionario de Datos: Jerarquías.

Se muestran los esquemas de tres estructuras definidas de forma estándar en el Diccionario de Datos.

- D. Diccionario de Datos: Modelo de información.

Se detallan gráficamente los distintos tipos de objetos que maneja el Diccionario y sus relaciones. Los esquemas presentados son equivalentes a "vistas" del Diccionario en el sentido relacional del término.

#### E. Siglas y acrónimos.

Se resumen las siglas y acrónimos empleados en el texto.

A lo largo del documento aparecen frecuentes acrónimos, normalmente referidos a tecnologías o fabricantes, cuyo significado se detalla en la primera aparición de los mismos y que finalmente se recogen al final del documento (apéndice E. pág. 177).



## Parte I

# Evolución de la tecnología informática en la empresa





## Capítulo 2

# Introducción

El objetivo de esta primera parte de la tesis es analizar la evolución de la tecnología informática desde dentro de la empresa, utilizando como referencia una entidad financiera de ámbito nacional. También se presentan aquellos aspectos de la Ingeniería del Software que presentan un rasgo diferenciador en esta experiencia concreta, demostrando su importancia en cualquier tipo de instalación pero en particular en aquellas de características similares a la que nos ocupa, que son:

- Parque importante de aplicaciones con diferente antigüedad, en continuo crecimiento y evolución.
- Volumen muy elevado de datos heterogéneos con relevancia especial de la información histórica.
- Variedad de plataformas y tecnologías empleadas.
- Equipos de desarrollo numerosos con presencia importante de recursos externos y subcontratación.

El período de tiempo que se va a considerar es el que abarca desde principio del año 1986 hasta el final del año 2011. El trabajo se centra en los cambios tecnológicos más importantes que sufrieron los sistemas de una entidad financiera en este período, haciéndolo siempre desde la perspectiva del software. Para ello se han seleccionado los hechos más significativos, denominándolos "hitos en la evolución" a pesar de tratarse cada uno de ellos de uno o más proyectos y de que en alguno de los casos tuvieron una duración importante.

Los hechos que se van a tratar en esta tesis se han estructurado en los capítulos siguientes:

- Migración de la plataforma Bull a IBM.

La informática de la empresa estaba soportada por ordenadores Honeywell-Bull que en el año 1984 se había decidido sustituir por ordenadores de la marca IBM. El proyecto de migración estableció el entorno sobre el que se incorporaron las tecnologías que son objeto de análisis en esta tesis.

La migración se realizó en dos fases: "Migración Teleproceso" y "Migración Batch". Fue realizada durante los siete primeros años del período de tiempo que tratamos.

- Migración a bases de datos relacionales.

Las bases de datos relacionales se empezaron a utilizar en las empresas de forma importante en la década de los 80. La migración desde bases de datos diseñadas con otras tecnologías supuso un hito de evolución informática muy importante. En este capítulo se analiza cómo sucedió en la instalación que tratamos, las dificultades encontradas y las ventajas que este cambio suponía.

No se trató un único proyecto sino que se hizo por aplicaciones, al ritmo al que éstas sufrían algún tipo de rediseño. Su duración fue larga, 15 años aproximadamente.

- Evolución del Terminal Financiero.

El Terminal Financiero constituye un entorno fundamental en las entidades financieras, en otro tipo de empresas hablaríamos de la plataforma cliente. Su evolución y sobre todo los criterios que la sustentaron supone una parte muy importante de la historia de la informática en este tipo de entornos.

El Terminal Financiero pasó por tres grandes etapas: Terminal en modo Carácter (1985-1992), Terminal Gráfico (1992-2007), Terminal Gráfico Integrado (2007-2013).

- Proyecto Año 2000.

El proyecto del Año 2000 fue común a todas las empresas. Supuso una revisión profunda de todos los formatos de fechas, su tratamiento y su presentación. Fue un proyecto que se pudo anticipar en el tiempo pero el trabajo más importante tuvo lugar en los dos años anteriores. Tiene características propias por tratarse de un problema horizontal que afectó a todas las aplicaciones.

- Proyecto Euro.

Se trata del proyecto para la sustitución de la peseta por el euro. El cambio al euro se produjo el 1 de Enero del año 2002, dos años después del efecto 2000. Los proyectos del Euro y del Año 2000 tienen características similares: involucrar a prácticamente todas las aplicaciones y tener una fecha de puesta en producción determinada e inamovible. Éste tiene sin embargo algunas connotaciones diferentes: la necesidad de mantener la doble moneda, tanto internamente como a nivel de presentación, el corto espacio de tiempo para hacerlo y la necesidad de los decimales. La experiencia del proyecto Año 2000 fue muy importante para acometerlo.

- Incorporación a Internet: Webs y soluciones de Banca Electrónica.

La incorporación de las entidades financieras a Internet supuso un hito muy importante en la informática de estas empresas. En todo el proceso de incorporación a Internet tiene especial relevancia la Banca Electrónica por reutilizar operativa bancaria existente en otros canales. Los dos hitos más significativos de esta evolución lo representan la Banca Electrónica de Empresas (BE Empresas) y la Banca Electrónica de Particulares (BE-Particulares).

- Gestión documental.

En este capítulo se analiza la gestión documental en sentido amplio, los temas que se tratan son: distribución y almacenamiento de listados, distribución física de correspondencia a clientes, distribución virtual y bases de datos documentales (BD. G. Documental). En el gráfico resumen (fig. 2.1), se representa solamente esta última.

- Arquitecturas de integración de aplicaciones y BPM<sup>1</sup>.

Se analizan los orígenes de la integración y su evolución hasta la situación actual con el escenario que plantean los EAI<sup>2</sup> y las plataformas BPM. La integración de aplicaciones se presenta en dos situaciones claramente diferenciadas: integración interna e integración externa, en este capítulo se analiza la evolución de las necesidades de integración y cuáles fueron los condicionantes que llevaron a la estandarización.

- Sistemas Informacionales - "Data Warehouse".

Los sistemas informacionales actuales tienen sus raíces en los primitivos Centros de Información y en los sistemas EIS<sup>3</sup>. El planteamiento en este capítulo es eminentemente práctico, se centra en los aspectos de gestión y en los de integración con los sistemas operacionales, haciendo resaltar la importancia de la Administración de Datos en este contexto.

Los aspectos relacionados con lo que supuso el Año 2000 y el cambio al Euro se tratarán en la Parte III de esta tesis por el papel central que en ellos jugó la función de Administración de Datos.

Es preciso resaltar que se han seleccionado aquellos cambios que presentan una mayor singularidad tecnológica, dejando fuera del alcance de este trabajo otros muchos también importantes como: nuevas aplicaciones, ampliaciones de las existentes y los cambios inducidos por la propia evolución de la plataforma del ordenador central, fundamentalmente en lo que concierne a versiones del sistema operativo, gestores de base de datos, gestores transaccionales o software de comunicaciones.

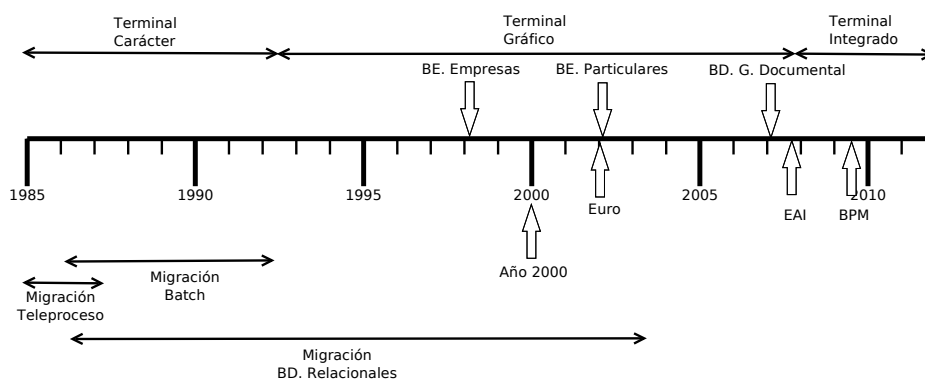
También se analizan en esta parte de la tesis los condicionantes que presenta la evolución de una instalación con las características anteriores. Las necesidades funcionales están en continuo cambio, ya que las empresas buscan continuamente la mejora de sus sistemas que cada vez influyen más en la marcha del negocio. La tecnología nos va ofreciendo nuevas posibilidades para satisfacer las necesidades pero nunca se puede olvidar el pasado, hay que construir siempre teniéndolo en cuenta, ya que rara vez existe la posibilidad de hacer tabla rasa y partir de cero. El tiempo es el hilo conductor de esta parte de la tesis en la que se analizan los cambios tecnológicos

---

<sup>1</sup>BPM: Business Process Modeling.

<sup>2</sup>EAI: Enterprise Application Integration

<sup>3</sup>EIS: Executive Information Systems

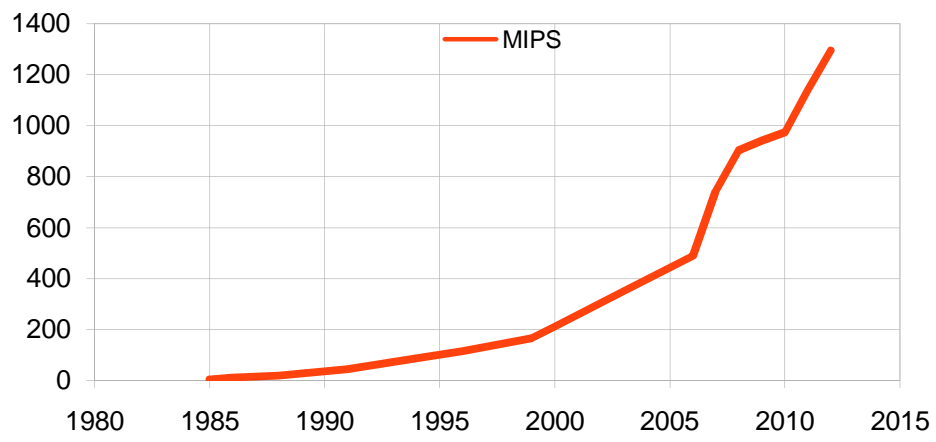


**Figura 2.1:** Hitos más relevantes

más significativos, desde una situación de partida que se describe en cada caso, y se explica el porqué de las decisiones tomadas y en que modo se acompañaron.

En 25 años la capacidad de las plataformas utilizadas creció de forma muy significativa. En la figura 2.2 se refleja la evolución de la capacidad de proceso disponible en la instalación que va a servir como referencia. Es importante tener en cuenta que éste ha sido uno de los factores que más influyó en cada momento para poder optar por una u otra alternativa, decisiones que pueden ser acertadas para una determinada capacidad de proceso pueden ser incorrectas en otras situaciones. Es significativo hacer notar que la potencia instalada se multiplicó por 130, pasando de 10 Mips<sup>4</sup> en el comienzo de la etapa analizada, año 1986, a 1298 Mips al final de la misma en el año 2011.

<sup>4</sup>MIPS: Millones de instrucciones por segundo (medida de potencia de un ordenador).



**Figura 2.2:** Potencia instalada

## Capítulo 3

# Situación tecnológica en 1985

### 3.1. Objetivos

Se describe en este capítulo la situación de partida para la evolución tecnológica que se va a tratar, tanto desde el punto de vista de las plataformas hardware y software, como desde el de la arquitectura de aplicaciones existente en la entidad financiera en 1985, inicio del período de tiempo de estudio. También se explicará la metodología de desarrollo utilizada y tres prácticas de la ingeniería del software que resultaron cruciales en la experiencia que se va a analizar.

### 3.2. Plataforma hardware y software inicial

El instante cero de este análisis se sitúa en el año 1985, momento en el que la Entidad Financiera había decidido acometer una profunda renovación tecnológica, migrando su plataforma principal de ordenadores, en ese momento con tecnología Honeywell-Bull DPS-8, sistema operativo GCOS-8, monitor transaccional DM-IV/TP y gestor de base datos DMIV [11] [17], a plataforma IBM, con sistema operativo MVS <sup>1</sup> [38], gestor transaccional CICS<sup>2</sup> y base de datos jerárquica IMS<sup>3</sup> [39].

En aquel momento, año 1985, la informática en el entorno de las oficinas estaba resuelta mediante la utilización de terminales Olivetti y Nixdorf que se comunicaban

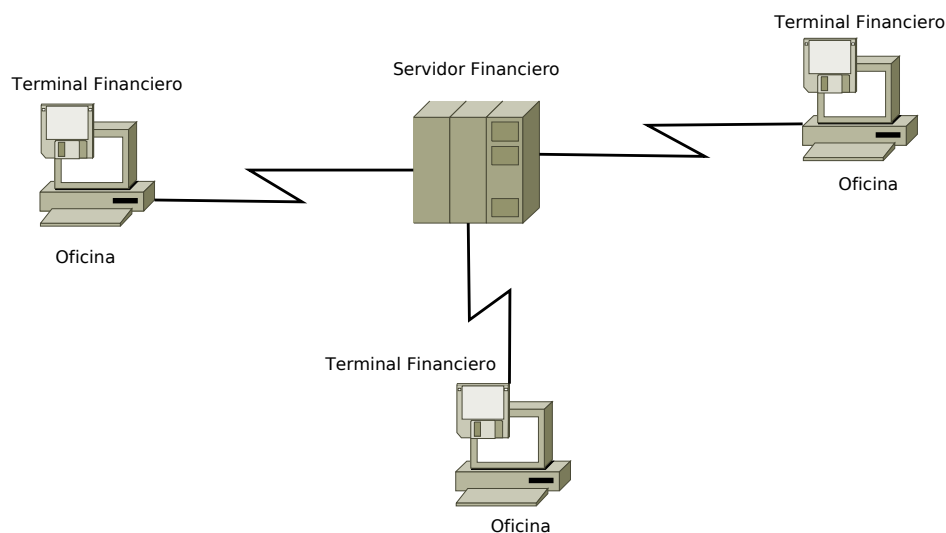
---

<sup>1</sup>MVS: Multiple Virtual Storage. Sistema operativo de IBM (1974).

<sup>2</sup>CICS: siglas de Customer Information Control System. Realmente un gestor transaccional de propósito general.

<sup>3</sup>IMS: Information Management System. Base de Datos Jerárquica de IBM.

con el ordenador central mediante el envío de mensajes. Se trataba de una arquitectura cliente/servidor en la que la presentación y la lógica de presentación residían en el terminal de la oficina, que denominaremos en adelante Terminal Financiero de acuerdo con el nombre con el que se conocen habitualmente este tipo de entornos. Las pantallas que se utilizaban eran las llamadas "de fósforo verde", trabajaban en modo carácter y los equipos que las soportaban se programaban en ensamblador y en Cobol.



**Figura 3.1:** Servidor y Terminal Financiero

### 3.3. Arquitectura de aplicaciones

La Entidad Financiera tenía entonces un parque de aplicaciones típico de una entidad de sus características, destacando la relevancia de aplicaciones como Cuentas Corrientes, Cuentas de Ahorro, Cuentas de Crédito, Cuentas a Plazo Fijo, Valores, Préstamos, Transferencias, Tarjetas y Cartera. Se trataba de aplicaciones, soportadas por tecnología de base de datos con modelo lógico en red, utilizables en línea desde las oficinas en modo transaccional. Estas aplicaciones, invocadas como servicios desde el Terminal Financiero de las oficinas, constituían lo que generalmente se denomina el Servidor Financiero (fig. 3.1). Una característica importante es que estas aplicaciones no compartían el concepto de titular de contrato, teniendo cada una de ellas su propio fichero de titulares, situación común



en aquella época en la que el concepto de sistema integrado estaba muy poco desarrollado, limitándose a una integración contable. Para paliar los problemas de redundancia e inconsistencia de información, se planteó en el año 1985, todavía en el entorno Bull, un diseño muy ambicioso, pionero en el sector, que consistía en una base de datos central de personas y contratos relacionados (Fichero Central de Clientes) que integrase a todas las aplicaciones que utilizaban algún concepto de persona o contrato.

El diseño fue muy laborioso por la necesidad de poner de acuerdo a un número muy elevado de personas: los usuarios de las distintas aplicaciones. Las aplicaciones tuvieron que adaptarse para integrarse con la nueva base de datos, sustituyendo y eliminando paulatinamente sus ficheros de titulares e incorporando éstos a la nueva base de datos compartida entre todas ellas. Para su implantación fue necesario realizar un trabajo muy minucioso, de gran envergadura, realmente un proyecto de depuración, puesta en común y carga de datos de personas, de sus domicilios y de sus relaciones con contratos, para el que fue necesario constituir un equipo centralizado, en contacto permanente con las oficinas, pues las inconsistencias entre los ficheros de titulares de las distintas aplicaciones eran muy frecuentes siendo necesario aclarar cada una de las situaciones de conflicto que se planteaban con los nombres o con los domicilios. El arranque de esta macro base de datos se hizo en el entorno Bull en el año 1987 incorporándose después dentro del proyecto de migración a IBM donde arrancó un año después, ya sobre tecnología relacional, siendo éste el primer desarrollo importante de la entidad con este tipo de tecnología.

Aunque el primer diseño del Fichero Central de Clientes se hizo sobre base de datos en red, una característica importante fue que en esta base de datos se optó por la identificación de las personas mediante un subrogado, en la línea de lo propuesto por Codd para su modelo RM/T [14] [21], aunque limitado a la propia empresa, evitando el problema de identificación mediante documentos oficiales, como el DNI, de diferentes formatos en los diferentes países o el caso singular de los menores que podían no tener documento.

Un aspecto también singular del diseño de esta base de datos fue la forma en la que se resolvió el diseño de la parte correspondiente a los domicilios de las personas, estableciendo un concepto de "Callejero". En él se identificaba cada una de las calles de una población mediante un subrogado y se establecía para cada rango de numeración de una calle el distrito postal de pertenencia, de forma que una persona -que podía tener más de un domicilio- se ubicaba en un número de portal, de un número de calle, de un número de población, de una provincia. El diseño fue muy ambicioso, con un coste de carga y mantenimiento muy importante, permitiendo una gran flexibilidad en los procesos comerciales pues este tipo de planteamiento permitía una aproximación de tipo geográfica al cliente. El talón de Aquiles de este

diseño fue el crecimiento de las poblaciones y la expansión de las oficinas bancarias, que convirtió en muy costoso el mantenimiento de este callejero por la necesidad de actualizaciones constantes, que provocaban demora en los procesos de alta de clientes al efectuarse el mantenimiento del callejero de modo centralizado.

### 3.4. Metodología de desarrollo

Las metodologías en uso en 1985 eran fundamentalmente de ciclo de vida en cascada. En el caso que planteamos se había decidido la implantación de una metodología de tipo diferente.

En concreto la metodología seleccionada había sido PDM-80 [18]. Esta metodología se caracterizaba por hacer fuerte énfasis en el modelado de datos, utilizando una técnica denominada ELKA<sup>4</sup> [61], de la familia de los modelos ERA<sup>5</sup>, caracterizada por su foco relacional. En la definición de procesos la técnica utilizada era SADT<sup>6</sup>, que utilizaba diagramas SA<sup>7</sup> [64] tanto a nivel proceso de negocio como a nivel de flujo de interacción.

La metodología contemplaba las siguientes fases para la definición de la base de datos:

1. Auditoria operacional.  
Su objetivo era el análisis del ámbito de aplicación, sus procesos y sus necesidades de información.
2. Diseño conceptual.  
Se corresponde con el modelado conceptual de datos.
3. Diseño de base de datos.  
Se trata del diseño lógico de la base de datos.
4. Análisis heurístico.  
Valida la base de datos diseñada con ayuda de lenguajes de interrogación de bases de datos. Define las funciones de acceso a los datos y finalmente los módulos de acceso a las mismas.
5. Test de entorno.  
Su cometido fundamental era resolver los problemas de integración desarrollando las funciones definidas en la fase anterior.

---

<sup>4</sup>ELKA: Element Link Key Attribute.

<sup>5</sup>ERA: Entity-Relationship-Attribute.

<sup>6</sup>SADT: Structured Analysis Design Technique.

<sup>7</sup>SA: Structured Analysis.

## 6. Control de rendimiento y calidad.

Cabe resaltar la importancia temprana del diseño conceptual, el énfasis en la administración de datos y la importancia del prototipado, así como los aspectos relacionados con la calidad del software desarrollado. El punto débil de esta metodología era la consideración de los procesos "batch", pues su fuerte era el diseño de la base de datos y la validación del mismo. Dado que en una entidad financiera los procesos "batch" son de la mayor importancia, esta laguna se salvó al plantear el proyecto de migración que se trata en el próximo capítulo.

Hay algunas características de esta metodología que resultaron determinantes ya que sentaron las bases que facilitaron la evolución de los sistemas, sustentados sobre algunos principios de la Ingeniería de Software que han demostrado su validez y adecuación al entorno. En concreto:

- Administración de Datos
- Gestión de Proyectos
- Gestión de la Calidad

### 3.4.1. Administración de Datos

Frente al planteamiento más usual en aquel momento de diseñar estructuras de datos con una óptica de aplicación aislada, en la que cada aplicación era propietaria de sus datos, la instauración de la función de Administración de Datos [26] supuso un cambio radical en la forma en la que se plantearon y definieron las aplicaciones desde ese momento. El proyecto del Fichero Central de Clientes, mencionado anteriormente, supuso el punto de partida para esta evolución.

Las estructuras de datos y los propios datos se convirtieron desde ese momento en un activo más de la organización que era necesario planificar, coordinar y gestionar, con énfasis en la integración global.

La importancia de este hecho ha justificado que se le dedique la tercera parte de esta tesis para explicar con mayor detalle el significado e impacto de la Administración de Datos y de su herramienta principal de soporte: el Diccionario de Datos. Ambos conceptos han sido centrales en la historia que se describe teniendo un papel primordial en los proyectos de Año 2000 y cambio al Euro.

### 3.4.2. Gestión de proyectos

El desarrollo de proyectos en una instalación compleja exige una cuidadosa gestión de los mismos. Esta gestión comienza con la decisión de cuáles son los proyectos que se van a acometer y con qué prioridades. Fue muy importante que se acometiera un Plan de Sistemas, alineado con los planes y objetivos corporativos globales, en el que se encuadró un Plan de Proyectos, que en este caso era de naturaleza anual y revisión mensual.

El entorno de desarrollo de proyectos de una entidad financiera es muy dinámico, los años a los que hacemos referencia fueron años de expansión y fuerte competencia. Todos los meses aparecían solicitudes nuevas de desarrollo de productos, cada uno más urgente que el anterior, que era necesario encajar en la planificación establecida en ese momento. La variabilidad de un entorno como el mencionado se veía incrementada por los aspectos normativos o legales, ya que las entidades tenían también que adaptarse a todos los cambios y sistemas que se imponían desde el Banco de España (ej. Sistema Nacional de Compensación Electrónica, Circulares Contables del Banco de España), la Comisión Nacional del Mercado de Valores, el Consejo Superior Bancario, Organismos Públicos (AEAT, Diputaciones, Ayuntamientos, Administraciones Autonómicas etc.), sin olvidar la integración con la informática de los clientes, supeditada siempre a los medios de los que éstos disponían.

Este escenario hizo necesaria la existencia de una herramienta de planificación de proyectos, a disposición de los Jefes de Proyecto, que permitiera hacer la asignación de recursos de forma dinámica y poder tener un seguimiento continuo de los proyectos en curso. La herramienta fue instalada sobre el ordenador Bull para poder hacer la propia planificación del proyecto de migración, y en 1986 fue migrada al entorno IBM. Tanto la planificación como la captura de las horas realizadas se hacía sobre formularios en papel que era necesario grabar para ser incorporados en la aplicación en modo "batch". Ya en el entorno IBM la plataforma evolucionó cambiando a una que permitía a los jefes de proyecto hacer la planificación y obtener informes en línea, sin necesidad de esperar a la carga y proceso "batch". La herramienta de captura de horas también evolucionó en la misma dirección utilizando para su desarrollo el entorno colaborativo existente.

En el Plan de Proyectos del año convivían los proyectos en curso, heredados del año anterior, con los nuevos proyectos aprobados para el plan del siguiente año con origen en las solicitudes que planteaban las distintas unidades de negocio. La complejidad que suponía un plan de proyectos en continua evolución, con un peso importante de peticiones sobrevenidas, hizo que tempranamente se constituyesen dos unidades relacionadas con su gestión:

- La Red de Servicio al Usuario

Fue la encargada de valorar, aceptar o rechazar, y priorizar las peticiones de las unidades usuarias. En el caso de ser aceptadas se trataban como proyectos a encajar en el plan o simplemente como peticiones de mantenimiento, sujetas a un ciclo de vida más simplificado, dependiendo de su tamaño.

- La Oficina del Plan de Proyectos

Su función era la de gestionar el Plan de Proyectos, tanto desde el punto de vista del alta de los nuevos proyectos como del de obtener la información de seguimiento para las unidades peticionarias.

El Plan de Proyectos, administrado con flexibilidad, demostró ser una herramienta fundamental para gestionar la actividad en un centro de las características del descrito.

### **3.4.3. Gestión de la calidad**

La calidad de los desarrollos efectuados se cuidó desde el primer momento. Dentro de este marco se definió un conjunto de estándares para la definición de los datos y de los modelos de datos, y unas normas rígidas de programación orientadas al Cobol [10], enfatizando la programación estructurada en línea con los planteamientos de Dijkstra [24][25]. Todos los desarrollos eran revisados por los equipos de calidad correspondientes, tanto en su vertiente dato como en su vertiente proceso. Desde el punto de vista funcional se efectuaban pruebas independientes y control de calidad a distintos niveles.

Otro aspecto relacionado con la calidad que resultó trascendental fue la normativa estricta que se aplicó en el diseño de la interfaz de usuario, inicialmente se hizo para la interfaz en modo carácter y posteriormente en la interfaz gráfica orientada a objetos.

La importancia de la interfaz de usuario fue de primer nivel, tanto por la trascendencia que tiene en el diseño de las aplicaciones, pues en definitiva es la cara de los sistemas, como en lo que se refiere a la necesidad de su estandarización para facilitar el uso y la incorporación de nuevas aplicaciones a la plataforma.

Metodologías más modernas han refrendado este posicionamiento a la vista del papel fundamental que hoy se le da a los "Casos de Uso" [46] en metodologías como el Proceso Unificado de Desarrollo de Software [47]. La importancia del diseño temprano de la interfaz de usuario, realizada por los analistas con la cooperación de los propios usuarios, fue uno de los elementos más importantes para poder garantizar el tiempo de desarrollo de una aplicación. Una forma de conseguir cerrar el diseño de esta interfaz era el tratar de conseguir lo antes posible el diseño del manual del usuario.

Se construyeron entornos diferenciados para desarrollo, en el que trabajaban los equipos de proyecto y hacían sus primeras pruebas, aceptación, orientado al teleproceso buscando garantizar su estabilidad, integración, para poder realizar ciclos completos de aplicaciones antes de su puesta en producción, y producción. La separación entre los entornos de desarrollo, aceptación, integración y producción, fue fundamental para garantizar la estabilidad de los sistemas. Para los entornos de desarrollo, aceptación e integración fue necesario construir versiones reducidas de pruebas, orientadas bien al proceso por lotes o al proceso en línea. La dificultad de obtener entornos reducidos coherentes, sobre todo cuando se trata de entornos muy integrados, no es un problema menor, como no lo es tampoco el conseguir mantenerlos actualizados cuando más de una de las aplicaciones involucradas puede estar sujeta a cambios.

Dado que no se va tratar de nuevo en la tesis la gestión de la calidad, salvo en lo que concierne a calidad de datos, se debe señalar que posteriormente se implantó una herramienta automática de análisis de la calidad del software con la que se analizaban periódicamente las aplicaciones, de modo especial si sufrían modificaciones importantes, calculando sobre las mismas distintos tipos de métricas. Los resultados se recogían en una base de datos histórica que permitía conocer aspectos de evolución de las aplicaciones así como establecer comparativas entre las mismas en base a las métricas establecidas.

El foco en la calidad es absolutamente necesario en cualquier tipo de instalación pero especialmente en instalaciones complejas. El problema se acrecienta en la medida en que colaboran en el desarrollo empresas externas, convirtiéndose algo que es intrínsecamente bueno en una necesidad de supervivencia.

## Capítulo 4

# El proyecto de migración de plataforma

### 4.1. Objetivos

En este capítulo se explican los condicionantes técnicos de la migración de la plataforma Bull a la plataforma IBM. También se explica la forma en la que se planteó la migración y las características del entorno de explotación que se implantó.

### 4.2. Condicionantes técnicos

El proyecto de migración de la plataforma constituye el punto de partida del análisis que se realiza en esta primera parte de la tesis. En este proyecto se establecieron las bases del entorno sobre el que se estudiará la evolución tecnológica que va a ser objeto de esta parte. Es por lo tanto el primer hito que se analiza.

La decisión de abordar la migración de plataforma no fue de tipo técnico, se trató de una decisión estratégica tomada por la dirección de la entidad financiera. Esta decisión desencadenó las acciones tácticas necesarias para efectuar el cambio que son las que se analizan en este capítulo.

La migración de plataforma arrastraba dos cambios fundamentales que exigieron un rediseño a nivel técnico de las aplicaciones: el del gestor transaccional, que pasaba de estar soportado por el DM-IV/TP de Bull al entorno transaccional CICS de IBM, y sobre todo el cambio de tecnología del gestor de base de datos, que en el entorno

Bull consistía en una base de datos con modelo lógico en red y en el caso de la plataforma IBM era una base de datos de tipo jerárquico soportada por el producto IMS.

Existen muchos modos posibles de plantear una migración de plataforma, sobre todo cuando el número de aplicaciones a migrar es importante como sucedía en este caso. La opción elegida para efectuar la migración se sustentó en un requisito fundamental que establecía que no se podía modificar la funcionalidad de las aplicaciones, es decir sería una migración en sentido estricto. Esta decisión afectaba tanto a la funcionalidad del Terminal Financiero, instalado en las oficinas, como al Servidor Financiero, establecido en el ordenador central. El requisito obligaba a que el nuevo Servidor Financiero respondiese a las peticiones de los terminales de oficinas sin que fuese necesario modificar los formatos de los mensajes existentes más allá de lo que fuese imprescindible para adaptarse a la nueva arquitectura de comunicaciones que pasaba a ser SNA<sup>1</sup>. De esta forma la interfaz de usuario no se veía afectada, manteniendo en ella la lógica de presentación, a diferencia de lo que en aquél momento era habitual en plataformas IBM donde la utilización del protocolo 3270<sup>2</sup> llevaba a incorporar la lógica de presentación en el Servidor Financiero.

El planteamiento anterior independizaba al Servidor Financiero del Terminal Financiero. Esta arquitectura facilitó desde el primer momento los desarrollos multicanal, haciendo posible reutilizar la misma lógica de negocio para los diferentes canales: oficinas y cajeros inicialmente, y más adelante aplicaciones de banca electrónica, banca telefónica y centros de atención a usuarios.

A pesar de no modificar funcionalmente el Servidor Financiero, sí fue necesario hacer un rediseño completo del mismo bajo la nueva plataforma que constituían el sistema operativo MVS, el gestor transaccional CICS y la base de datos jerárquica IMS. Este trabajo se hizo respetando, como se ha dicho anteriormente, los formatos de mensajes de petición y respuesta del Terminal Financiero. La dimensión del trabajo a realizar supuso que se definieran tantos subproyectos como aplicaciones tenía en aquel momento la Entidad Financiera. Estos subproyectos cubrían las aplicaciones clásicas de Clientes, Préstamos, Cuentas Corrientes, Cuentas de Ahorro, Cuentas de Crédito, Cuentas Plazo, Valores, Accionistas, Transferencias, Avales, Tarjetas de Crédito, Tarjetas de Débito etc., en definitiva las aplicaciones necesarias para soportar tanto la operativa del negocio bancario como aquellas aplicaciones destinadas a cubrir los aspectos relacionados con la gestión como eran: Contabilidad, Control Presupuestario, Almacén, Nóminas etc.

---

<sup>1</sup>SNA: System Network Architecture (IBM).

<sup>2</sup>3270: Protocolo de la familia SNA.



### 4.3. Fases de la migración

La forma de abordar la migración fue en dos grandes fases. En una primera fase se diseñaron de nuevo las aplicaciones existentes sobre la nueva plataforma para poder soportar de esta forma el proceso en línea de las oficinas, el denominado "teleproceso". Se dejó sin embargo en la antigua plataforma todo el proceso por lotes o "batch". Esto fue posible porque los tratamientos "batch", que por su naturaleza eran masivos, se realizaban fundamentalmente utilizando ficheros descargados de las bases de datos que habían sido actualizadas durante la sesión del teleproceso. Para poder hacerlo de esta forma fue necesario replicar, en la plataforma IBM, las descargas hacia ficheros en los mismos formatos en los que los esperaba el entorno anterior. Los resultados del "batch", que se necesitaban para actualizar las bases de datos, se resolvían en sentido inverso, trasladándolos al nuevo entorno y efectuando allí las actualizaciones.

La tecnología de base de datos a la que se migró fue IMS, que trabajaba con modelo lógico de tipo jerárquico. Solamente aplicaciones de pequeña entidad, sobre todo las nuevas que se iban incorporando, se desarrollaron sobre base de datos relacional. El motivo de esta decisión no fue otro que el mejor rendimiento de las bases de datos jerárquicas en aquel momento. No se puede olvidar que en 1987 la potencia del ordenador central era de poco más que 10 Mips (fig. 2.2 pág. 14).

El teleproceso sobre la plataforma IBM arrancó en el año 1987. En una segunda fase, con solapamiento sobre la primera, se abordó la sustitución progresiva del "batch" que concluyó en 1992. En el periodo intermedio entre ambas fechas las aplicaciones nuevas ya se desarrollaron sobre la nueva plataforma, basculando algunas aplicaciones a tecnología de base de datos relacional y manteniendo las más críticas en bases de datos jerárquicas.

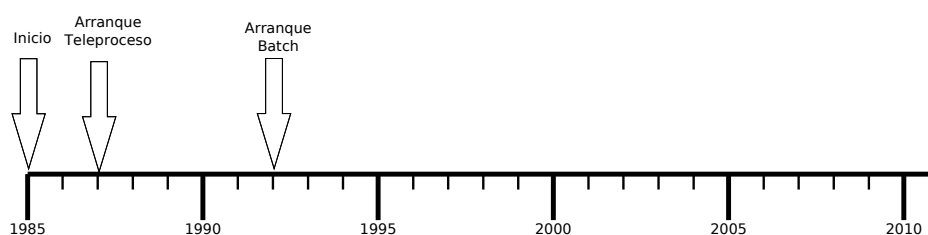


Figura 4.1: Proyecto de migración de plataforma

#### 4.4. Planificación de los procesos "batch"

El proyecto de migración fue también muy importante para sentar las bases de un entorno de explotación robusto evitando, en lo posible, intervenciones de tipo manual en favor de una automatización. En las entidades financieras el volumen de proceso que se realiza fuera de lo que es el entorno en línea es muy elevado. El período de tiempo que transcurre entre el cierre del teleproceso de un día y el arranque del día siguiente se conoce con el nombre de "ventana batch". En este período de ejecutan todos los procesos que generalmente suponen descargas y actualizaciones masivas de bases de datos, así como el tratamiento de un gran número de ficheros. Sin una planificación adecuada de los trabajos a realizar en esta ventana, se producirían solapamientos que podrían provocar inconsistencias en la información y, en caso de haber problemas, llegar a la no disponibilidad del sistema en las oficinas a la hora de su apertura.

Desde el primer momento de la migración el buen diseño de los procesos "batch" fue de la mayor importancia. Se trataba de paliar el menor desarrollo que la metodología de trabajo utilizada tenía en este aspecto. El entorno de explotación se fue mejorando con el tiempo incorporando un "planificador de procesos".

Las herramientas básicas de un planificador son los calendarios y las dependencias. Los calendarios definen que días hay que realizar un proceso, mientras que las dependencias son las que establecen el flujo de ejecución. Sobre los calendarios se montan los conjuntos de procesos, mallas o grupos, en dependencia de otros procesos con los cuales estén relacionados. De esta forma se consigue optimizar el uso de los recursos del procesador, en base al paralelismo en la ejecución de procesos, sin que existan problemas que puedan generar inconsistencias.

Es importante tener en cuenta que el tratamiento "batch" basa sus principios en procesar grandes volúmenes de información, de forma que cada proceso sea una unidad lógica recuperable en caso de fallos. Para ello es conveniente diseñarlo de forma que cada proceso efectúe operaciones simples sobre los datos como pueden ser listados, liquidaciones o cuadros, que puedan ser tratados eficientemente y que en caso de fallo sea fácil encontrar el motivo del error y su consiguiente arreglo para su relanzamiento. Según los requerimientos aportados por los procesos a realizar sobre los datos, se diseñan en modo secuencial, agrupándolos en unidades de trabajo coherentes. Las unidades de trabajo a su vez se relacionan entre sí estableciendo una estructura de malla que habilita el paralelismo, de forma que la finalización de un proceso supone un evento de inicio para otro trabajo. Evidentemente se producen situaciones de espera en las que la continuación de un trabajo puede estar supeditada a la finalización de otros procesos.

## 4.5. Conclusiones

No es fácil determinar que hubiese pasado en el caso de haber planteado una forma diferente de hacer la migración, una opción podría haber sido abordarla por aplicaciones o por grupos de aplicaciones, de forma que cada una de ellas fuese migrando en momentos diferentes del tiempo. Este supuesto alternativo tendría al menos dos tipos de inconvenientes: la conexión de los terminales y la integración "batch".

- La conexión de los terminales hubiera sido un problema complicado por la incompatibilidad de las arquitecturas de red de ambas plataformas, hubiese obligado a mantener en funcionamiento dos terminales diferentes en cada oficina: uno para las aplicaciones ya migradas y otro para las aplicaciones que siguiesen residiendo en la antigua plataforma. Un planteamiento como éste tendría problemas de coste de equipos, de espacio físico y de frustración de los usuarios por tener éstos que trabajar cambiando de un sistema a otro según cual fuese la aplicación a utilizar. No hay que olvidar que la fecha en la que esto sucede es el año 1986, momento en el que ni la capacidad de los equipos, ni la estandarización de los fabricantes era lo que hoy existe.
- La integración a nivel de los procesos "batch", que aquel momento tenían un mayor paso relativo que el que hoy tienen en el conjunto global de la informática de las organizaciones, también se hubiera visto penalizada por la necesidad de trasladar un mayor número de ficheros entre plataformas. Esta necesidad hubiera sido además dinámica pues en la medida en la que se migrasen aplicaciones habría que adaptar estas integraciones a la nueva situación establecida.

De la experiencia en planificación de procesos se han obtenido algunas lecciones que es importante resaltar:

- Prioridad de los procesos.

Una correcta planificación debería ejecutar primero los procesos núcleo (ej. liquidaciones, actualizaciones), después las interfases con otras aplicaciones y por último todo aquello que sea informativo y la alimentación de bases de datos orientadas a las consultas como puede ser el caso del "Data Warehouse".

- Procesos críticos.

También es importante anticipar los procesos críticos; aquellos que generan un mayor número de dependencias o que son más propensos a situaciones de error. En este caso la motivación es la de tener más tiempo para resolverlos en el caso de que se planteen problemas.

- La periodicidad.

Es otro factor a tener en cuenta, ya que no todos los procesos tienen que ejecutarse todos los días, puede haber procesos semanales o mensuales, siempre deben ejecutarse antes los procesos diarios y a continuación los de periodicidad mayor: semanales, mensuales, trimestrales o anuales. De esta forma, en caso de fallo, el tiempo destinado a solucionar los problemas no se convertiría en algo tan crítico.

## Capítulo 5

# Migración a bases de datos relacionales

### 5.1. Objetivos

El capítulo trata de como se produjo la migración entre la tecnología de bases de datos jerárquicas y la de bases de datos relacionales. Recordamos que en el proyecto de migración de plataforma (capítulo 4) las bases de datos que existían previamente eran bases de datos en red, que se migraron a tecnología de bases de datos jerárquicas en el transcurso del mismo.

### 5.2. Situación general

La migración a tecnología relacional de las bases de datos desarrolladas sobre tecnologías existentes anteriormente, fundamentalmente bases de datos en red y bases de datos jerárquicas, fue un problema que se presentó en muchas organizaciones en el período de tiempo que tratamos. El caso que planteamos es uno de ellos, se trata de un proceso de migración desde la tecnología de bases de datos jerárquicas soportadas por el gestor de base de datos IMS de IBM, las establecidas en el proyecto de migración de plataforma, a la tecnología relacional soportada por el gestor de bases de datos DB2 también de IBM [22].

Conviene recordar que a pesar de que en el momento en el que se abordó el proyecto de migración ya estaban en el mercado las bases de datos relacionales, su utilización en los entornos de alto volumen de datos y gran nivel de concurrencia era

todavía muy limitada. Las bases de datos relacionales, en el tipo de entornos como el que se trata en esta tesis, comenzaron su andadura de la mano de los entonces llamados Centros de Información, predecesores de los actuales "Data Warehouse" o Almacenes de Datos, o también de aplicaciones de gestión interna que no tenían la criticidad de las aplicaciones operativas.

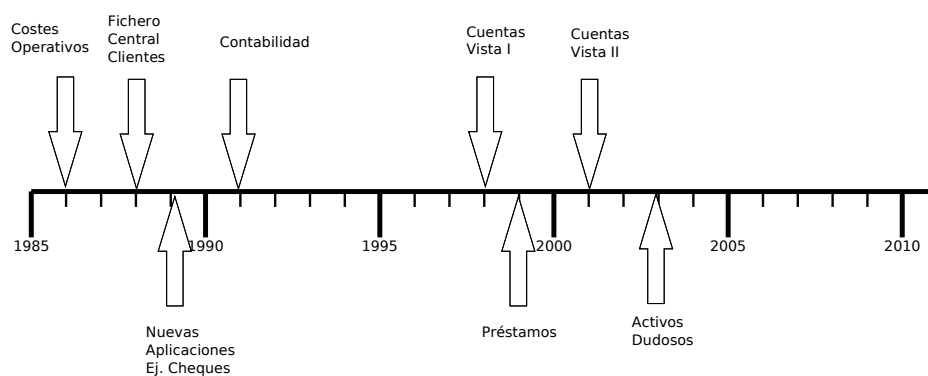
### 5.3. Cronología

La primera implementación sobre tecnología relacional de relevancia en la entidad financiera fue la del ya mencionado Fichero Central de Clientes en el año 1988, dentro del proyecto de migración de plataforma. A pesar de su tamaño y de su nivel de integración con el resto de aplicaciones esta base de datos se rediseñó sobre tecnología relacional, realizando una migración directa de tecnología de base de datos en red a tecnología relacional. La justificación de esta decisión, contraria al criterio general de utilizar bases de datos jerárquicas, fue que prácticamente no interfería en la operativa diaria de los productos salvo en el momento de la contratación y por lo tanto no afectaba de forma significativa al rendimiento. Las aplicaciones utilizaban un concepto de anagrama de cuenta que, almacenado de forma redundante en la base de datos de cada producto, permitía identificar al titular de un contrato en la operativa común de los mismos sin necesidad de acceder al Fichero Central de Clientes. El propósito del anagrama era el de mejorar el rendimiento de las aplicaciones, limitando el número de accesos que tenían que hacer a las bases de datos para identificar al titular de un contrato. En esta decisión también influyó la complejidad de las relaciones existentes, que no era fácil representar en un modelo jerárquico.

Las aplicaciones operativas fueron migrando a lo largo del tiempo a tecnología relacional, bien porque sufrían un rediseño, una modificación importante (ej. Cambio al Euro), o simplemente porque al final resultaban ser un residuo que complicaba la explotación y el mantenimiento por la dispersión de tecnologías de base de datos existentes. Así, las aplicaciones más importantes fueron cambiando a base de datos relacional en diferentes años: Cuentas Vista (1998 y 2001), Préstamos (1999), Activos Dudosos (2003) etc. Un detalle más claro puede verse en la figura 5.1. Entre estas dos fechas (1998-2003) se situarían el resto de las aplicaciones que no se han incorporado al gráfico por entender que no aportan información adicional.

En algún momento del tiempo alguna aplicación, como por ejemplo Cuentas Vista, podía tener parte de su estructura de datos en base de datos relacional y otra parte en base de datos jerárquica, haciendo imprescindible la utilización de un gestor transaccional no solamente en sus funciones de interacción con terminales

o gestión de ejecución de procesos, sino también en lo relativo a su papel en la coordinación de la confirmación o recuperación de las transacciones.



**Figura 5.1:** Evolución a BD relacionales

## 5.4. Problemática de este tipo de migración

La migración de las aplicaciones a bases de datos relacionales ofreció la oportunidad de desacoplar la lógica de acceso a datos de la lógica de negocio. El acceso a las bases de datos jerárquicas era de tipo navegacional, el programador tenía que ser conocedor de la estructura de la base de datos y de las opciones de acceso establecidas como podía ser un índice secundario, que se utilizaba explícitamente, o las relaciones lógicas, que permitían liberar la rigidez de una jerarquía estricta mediante un concepto de padre lógico [19]. Los programas por tanto navegaban la base de datos, estableciendo posicionamiento en las mismas. Al cambiar a bases de datos relacionales el acceso pasó a ser declarativo, sin conceptos de posicionamiento ni estructuras de acceso visibles al programador, ya que en esta tecnología el camino de acceso lo resuelve el optimizador del gestor relacional. El rediseño de las aplicaciones, al cambiar a relacional, permitió construir aplicaciones mejor estructuradas y más sencillas de mantener al liberarse de los condicionantes que imponía el acceso en bases de datos jerárquicas.

## 5.5. Conclusiones

La migración a bases de datos relacionales supuso un cambio tecnológico importante que fue sustentado en las siguientes motivaciones:

- Actualización tecnológica.

Las bases de datos jerárquicas dejaban de ser la apuesta prioritaria de los fabricantes que empezaban a dirigir sus recursos de desarrollo hacia las bases de datos relacionales. También hay que tener en cuenta que el rendimiento de las bases de datos relacionales mejoraba rápidamente, apoyado en el crecimiento de la capacidad de proceso y del tamaño de la memoria.

- Productividad del desarrollo.

Es un aspecto muy importante que afecta tanto a los programadores como a los usuarios finales, ya que ambos grupos empezaban a disponer con esta tecnología de herramientas orientadas a consultas de usuario final que mejoraban su productividad. Desde el punto de vista de los desarrolladores el cambio a tecnología relacional concentraba la dificultad en la formulación de la sentencia SQL correcta, abandonando la dificultad que presentaba la navegación posicional que quedaba sustituida por un recorrido secuencial mediante el concepto de cursor.

- Unificación tecnológica.

Se trataba de simplificar la gestión de la instalación y reducir el coste de las licencias de uso de productos. Los sistemas informacionales iniciales se movían en esta dirección en los llamados Centros de Información, también los sistemas distribuidos utilizaban de forma casi única las bases de datos relacionales. La idea central era unificar la tecnología de base de datos utilizada sobre tecnología relacional, haciendo converger también en esta tecnología las bases de datos de las aplicaciones operativas.

- Formación de equipos.

Los nuevos profesionales que se incorporaban a la empresa llegaban con formación en teoría relacional y SQL, ya que las bases de datos jerárquicas se habían ido eliminando de los programas de estudio.



## Capítulo 6

# Evolución del Terminal Financiero

### 6.1. Objetivos

El Terminal Financiero es uno de los componentes más importantes en la informática de las entidades financieras. Constituye la plataforma cliente más importante dentro de la arquitectura, siendo además la herramienta informática fundamental de la denominada red comercial, cuyo exponente más relevante es la red de oficinas. En las sucursales u oficinas es donde tradicionalmente se ha resuelto la parte más importante de la operativa del negocio, más adelante complementada por las redes de cajeros y las bancas electrónicas. La propia historia de la informática en este tipo de entidades tiene su origen en la resolución de los problemas operativos de las oficinas.

En este capítulo se recorre su historia, analizando cómo evolucionó con los cambios tecnológicos ocurridos en estos 25 años y cómo fue reorientando su concepto, pasando de trabajar como un mero transmisor de operaciones a ser el punto más importante de relación con el cliente.

Por la importancia que el Terminal Financiero tiene en estos entornos se ha considerado interesante hacer un recorrido sobre su evolución, aunque para ello haya que remontarse unos años hacia atrás del punto de partida fijado en esta tesis.

## 6.2. Historia inicial

A mediados de los años 60 aparecieron los ordenadores orientados a procesos de carácter comercial -vulgarmente llamados IBM- aunque existían diversas marcas de renombre en el mercado. La banca apostó decididamente por la incorporación de estos equipos, con la finalidad de aligerar la gran carga administrativa que había en las oficinas de las cada día más extensas redes.

La incorporación de los ordenadores en el trabajo de las oficinas fue lenta, pudiéndose identificar un primer periodo de mecanización, nombre que se le daba en aquel momento a la incorporación de la informática en las entidades, en el que los tratamientos eran exclusivamente en forma de proceso por lotes o "batch".

Se comenzó por la sustitución en oficinas de los pesados libros de hojas intercambiables, que permitían tener las cuentas corrientes, cuentas de crédito, libretas de ahorro y libretas a plazo fijo, ordenadas alfabéticamente para facilitar su localización de forma rápida, por hojas independientes para cada cuenta del cliente. Se actualizaban por medio de maquinas contables, que al mismo tiempo iban generando un diario de operaciones con copia por medio de papel carbón, posteriormente a estos equipos se le fue dotando de perforadoras de bandas de papel o casetes de banda magnética, en sustitución de la copia obtenida con papel carbón, que era una fuente de errores. Una vez cerradas las operaciones, estos diarios grabados en bandas y casetes se enviaban a los Centros de Proceso de Datos, para su grabación o lectura y posterior actualización de las posiciones y movimientos en los ordenadores centrales. Éstos disponían de periferia de entrada, constituida por lectores de tarjetas, bandas y casetes magnéticos. Además de la función de registro, eran fundamentales los procesos para hacer las liquidaciones de dichas cuentas.

Con este primer paso se conseguía no tener que hacer laboriosas liquidaciones en las oficinas, que en aquel momento se hacían semestralmente, en el caso de las cuentas corrientes y las libretas de ahorro, y mensualmente, para las imposiciones a plazo que vencían el mes siguiente. A pesar de estos avances, seguía siendo una tarea pesada, pues las liquidaciones por ordenador obtenían unos listados con los resultados que era necesario enviar a las oficinas, para allí actualizar las posiciones en las cuentas de cada cliente. Otra tarea aún sin resolver era la actualización de libretas pues se hacía manualmente, en maquina de escribir o con pequeños equipos registradores con introducción manual de los movimientos. Estas carencias demandaban a los fabricantes de ordenadores y de máquinas contables continuar con la investigación y desarrollo de nuevos equipos.

La incorporación de ordenadores en las entidades financieras empujó a su vez a la Compañía Telefónica a mejorar sus sistemas de comunicaciones. A mediados

de los años 70, en colaboración con alguno de los bancos en aquel momento más importantes, abordó un proyecto para desarrollar la Red Especial de Transmisión de Datos (RETD) que contaba con una serie de Concentradores de Comunicaciones distribuidos en toda la geografía española y otros centros denominados CCR<sup>1</sup>, que constituían la Red Secundaria de Alto nivel (RSAN) y permitía conectar equipos por medio de sencillos protocolos de comunicaciones, orientados al carácter, similares al utilizado en los teletipos, con una longitud máxima de mensaje de 256 caracteres (incluidos los caracteres de control). Entre estos protocolos se encontraba el NCR270, asíncrono o síncrono, con unas velocidades de transmisión de 200, 1.200 ó 2.400 baudios por segundo.

Se inició un nuevo periodo, demandando a los fabricantes nuevos equipos en oficinas capaces de enviar y recibir mensajes del ordenador central. Al mismo tiempo se deseaba hacer desaparecer las fichas de posición en oficinas para lo que se necesitaban impresoras financieras capaces de llevar el diario de operaciones y actualizar las libretas, recabando directamente los movimientos desde el ordenador central.

Inicialmente fueron apareciendo pequeños equipos, de capacidad reducida, que se programaban por medio de tablas. Con ellos se hacía el control de los datos, la formación de mensajes y su transmisión al ordenador por medio de la red. Uno de los proyectos más interesantes fue el terminal RONDA desarrollado por IBM en colaboración con la Caja de Ahorros de Ronda, pero no terminó de resolver lo que se demandaba por las entidades financieras.

Los distintos fabricantes de equipos de oficina: NCR, OLIVETTI, PHILIPS, NOKIA, etc. comenzaron a desarrollar equipos dotados de más memoria y capacidad de programación, destacando entre ellos el TC808 de Olivetti, programable en lenguaje ensamblador y con casete como soporte de programa y totales contables. A este equipo se le podían conectar impresoras financieras y de impacto, así como lectores de caracteres magnéticos y dispensadores de efectivo. Otro de los equipos con más posibilidades era el Multisistemas 8864 de NIXDORF, soporte de programas y totales en disketes de 8", después de 5"1/4 y por último por transmisión desde los centros de desarrollo, también programable en assembler o en el denominado "parametrico" (pseudo-cobol), posteriormente en cobol y en lenguaje C. Este tipo de terminal podía tener hasta 16 puestos de trabajo locales y también puestos remotos conectados a él por líneas punto a punto de 1.200 baudios, muy útil en localidades con más de una oficina pues estas líneas eran más económicas que las líneas con la red RSAN.

---

<sup>1</sup>CCR:Centros de Comunicación y Retransmisión.

Con estos nuevos equipos las oficinas ya casi tenían una automatización total y se continuaron incorporando aplicaciones a las iniciales de Cuentas, Valores, Cartera, Titulares de Contratos. Un hito importante fue la creación de la Cámara de Compensación Única, centralizada en el Banco de España, desapareciendo las cámaras provinciales que habían sido iniciadas en el Consejo Superior Bancario y una vez desaparecido éste continuadas por la AEB<sup>2</sup>. Los comités técnicos establecidos llevaron a cabo el truncamiento<sup>3</sup> de efectos comerciales y de cheques de cuenta corriente así como intercambios de transferencias, recibos, etc. hasta llegar al nivel de desarrollo que hoy conocemos.

Hubo tradicionalmente dos formas de integrar las oficinas. Unas entidades lo hicieron a través de centros regionales, que generalmente por la noche consolidaban los saldos y los movimientos en el ordenador central, y otras conectaron directamente sus oficinas al ordenador central.

En toda esta época, que duró hasta principios de los 90, los sistemas de comunicación mejoraron de forma exponencial con la aparición de nuevos sistemas operativos y de un nuevo protocolo de comunicaciones orientado a paquetes, el protocolo X.25, que ya permitía el envío de cualquier tipo de información: gráficos, tablas, etc.

Es en ese momento del tiempo cuando las entidades financieras comenzaron a plantearse un cambio tecnológico de su Terminal Financiero. La disponibilidad de los nuevos sistemas operativos con posibilidad de interfaz gráfica, como Windows y OS/2, fue la que abrió la puerta a nuevos planteamientos.

## **6.3. Evolución en la Entidad Financiera**

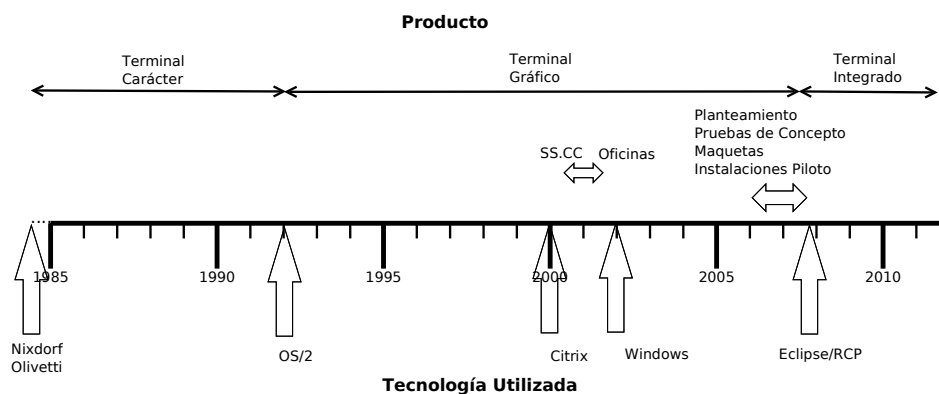
### **6.3.1. El terminal OS/2**

Para el diseño del nuevo terminal se seleccionó el sistema operativo OS/2 por su interfaz gráfica, capacidad multitarea y facilidades de comunicación. Fue en el año 1992 cuando se decidió hacer el cambio, planteando un diseño hacia un nuevo concepto de terminal con características interesantes como las que se describen a continuación:

---

<sup>2</sup>AEB: Asociación Española de Banca.

<sup>3</sup>Envío de datos o imágenes manteniendo los originales de los documentos en donde fueron tomados.



**Figura 6.1:** Evolución del Terminal Financiero

- Interfaz de usuario orientada a objetos

Frente al diseño tradicional de interfaz de texto, orientada a menús o invocación directa de funciones mediante su código técnico o comando, se decidió diseñar una interfaz de usuario con filosofía de orientación a objetos, de forma que las aplicaciones expusieran en la interfaz objetos y acciones sobre los mismos. La mención a orientación a objetos realizada se refiere únicamente a la interfaz en si misma y no al código que la implementa.

En línea con la elección del sistema operativo OS/2 se decidió utilizar para el desarrollo de la nueva interfaz gráfica el modelo propuesto por IBM como "The IBM Common User Access"[41] [40]. Este modelo es el que había sido usado por IBM para la definición de la interfaz del propio sistema operativo OS/2; de esta forma la nueva interfaz del Terminal Financiero presentaba las aplicaciones como objetos y menús contextuales que ofrecían acciones sobre esos objetos. Un ejemplo podría ser el objeto "Préstamo" y una posible acción sería "amortizar anticipadamente", ésta daría lugar a un objeto "Amortización" con sus propias acciones como pueden ser "anular" o "imprimir".

La idea era que los objetos que el usuario necesitase para realizar sus tareas se integrasen y pudieran cooperar con los objetos que proveía el entorno de forma integrada. La interfaz utilizaba tres tipos de objetos: contenedores, objetos de datos y dispositivos. Los contenedores agrupan otros objetos, los objetos de datos son realmente los objetos de negocio, como los explicados anteriormente, y los dispositivos son representaciones de elementos físicos como por ejemplo

una impresora, un lector de barras, un reciclador, un dispensador etc. La acción de imprimir podría realizarse directamente desde el menú contextual del objeto correspondiente o mediante la interfaz de usuario arrastrando el objeto a la impresora.

El resultado de este diseño fue una interfaz de usuario, muy avanzada en aquel momento, que permitía un espacio de trabajo o escritorio compartido por las aplicaciones y una interfaz de usuario uniforme e intuitiva, que hacía sencilla la incorporación de nuevas aplicaciones al entorno por compartir todas ellas el mismo modelo conceptual de la interfaz<sup>4</sup>, forma de representación y modo de interaccionar.

- Arquitectura del terminal

El desarrollo se realizó en lenguaje C, estableciendo un diseño arquitectónico en base a dos capas independientes: el núcleo y las transacciones de negocio, que seguían una filosofía cliente/servidor entre las mismas [60].

1. El núcleo

Aísla a las transacciones de negocio de los aspectos técnicos comunes a todas ellas como pueden ser: manejo de dispositivos (impresión, escaneo, dispensadores de efectivo, recicladores de billetes etc.), envío y recepción de mensajes, servicios de transferencia de ficheros, grabación del diario, además de gestionar todas las clases visuales, rutinas de uso genérico, control de transacciones activas, control de acceso al sistema, acceso a bases de datos locales, fuentes etc.

También es responsabilidad de esta capa el manejo de los hilos de ejecución o "threads" que se establecieron para gestionar los dispositivos. Estos hilos se conectan directamente con los puertos de conexión de los dispositivos en base a una arquitectura de dos niveles: un hilo de ejecución principal y dos adicionales para leer y escribir respectivamente.

---

<sup>4</sup>Modelo mental consistente en el conjunto de relaciones que una persona percibe que existen entre los elementos de una determinada situación, en este caso interfaz.

## 2. Las transacciones de negocio

Constituidas por las librerías dinámicas (DLL<sup>5</sup>) que contenían los programas de las aplicaciones. Estos programas manejaban los paneles de entrada/salida de datos y sus campos, la lógica de validación y la lógica de negocio asociada a la presentación. Las transacciones de negocio invocaban a los servicios técnicos que estaban definidos y expuestos en el núcleo.

### ■ Roles y perfiles

Los usuarios se registraban en el sistema en la propia oficina, donde existía el perfil de administrador del sistema atribuido generalmente al interventor y al director de la oficina. El administrador era quien daba de alta a los usuarios de esa oficina asignándoles un perfil y uno o más roles que podían ser variables. En función del perfil y el rol, el Terminal Financiero permitía el acceso o no a las distintas funcionalidades.

Inicialmente el terminal OS/2 funcionaba distribuyendo semanalmente el código de la aplicación a las oficinas. La distribución de código planteaba inconvenientes importantes, tanto desde el punto de vista de dimensionamiento de equipos como sobre todo de disponibilidad física de la red para transmitir las actualizaciones, sucediendo con frecuencia que alguna oficina quedaba sin actualizar en el proceso de distribución del fin de semana. En estos casos se efectuaban reintentos de tipo manual.

La solución al problema de la distribución de código se consiguió instalando una plataforma de presentación distribuida, instalada en el año 2000 en algunos servicios centrales y en el 2001 en oficinas, concentrando los servidores de aplicaciones en una granja de servidores ubicada en las mismas instalaciones que el ordenador central. La tecnología utilizada fue Citrix, con ella las actualizaciones de las aplicaciones se hacían de forma centralizada en los servidores, permitiendo al mismo tiempo clientes más ligeros en las oficinas, en este caso el cliente ICA<sup>6</sup> de la propia plataforma Citrix.

Conseguir reducir el tamaño del equipo necesario para los terminales de oficinas supuso una reducción de coste muy importante en la explotación del Terminal Financiero si tenemos en cuenta que el número de equipos superaba los 3.000 entre los servicios centrales y las oficinas.

---

<sup>5</sup>DLL: Dynamic Link Library.

<sup>6</sup>ICA: Independent Computing Architecture.

### 6.3.2. El terminal Windows

El abandono por parte de IBM del soporte al sistema OS/2 obligó en el año 2002 a efectuar una migración del terminal. La migración se hizo al sistema operativo Windows, concretamente Windows XP para clientes y Windows 2000 para servidores. La arquitectura del terminal explicada anteriormente simplificó en gran medida esta migración. Por una parte fue necesario actuar sobre el núcleo, que concentraba los servicios técnicos que por su naturaleza tienen gran dependencia del sistema operativo, y por otro lado sobre las aplicaciones, constituidas por las llamadas transacciones de negocio. Para esta segunda parte se desarrollaron convertidores que podían realizar la mayor parte del trabajo de forma automática. Este trabajo se vio facilitado por tratarse del mismo lenguaje C en ambos sistemas.

Otro cambio importante que se realizó en esta conversión fue la migración del protocolo de transporte que pasó de ser X.25 a TCP/IP, para ello se construyó un componente de emulación Telnet 3270E que encapsulaba el tráfico SNA para que la integración con el servidor transaccional del ordenador central no se viese afectada.

### 6.3.3. El terminal Eclipse-RCP

En el año 2006 se decidió diseñar un nuevo Terminal Financiero. En este caso el motor del cambio fue fundamentalmente de tipo funcional, aunque, como se explica a continuación, veremos que llevó consigo un cambio tecnológico de la mayor importancia que abrió nuevas posibilidades de evolución.

El requisito fundamental para el nuevo terminal era pasar de una orientación operativa, es decir el terminal financiero concebido como un ejecutor de operaciones de negocio vinculadas a productos, a una orientación comercial en la que el foco fuese el cliente, la información que se dispone del mismo y las oportunidades comerciales. Todo ello sin abandonar la capacidad operativa, pero buscando facilitar el uso en este nuevo contexto.

Se establecieron una serie de requisitos entre los que se pueden destacar:

- Integración de información.

El terminal tendría que ser capaz de integrarse con otras plataformas como el entorno de colaboración, basado en Notes<sup>7</sup>, el entorno de ofimática, sobre

---

<sup>7</sup>Notes: Entorno colaborativo de IBM.



Microsoft Office, el entorno informacional, con acceso directo a modelos en bases de datos relacionales, o la información de intranet, así como por supuesto a todo el Servidor Financiero ya integrado con la plataforma Windows.

- Orientación a procesos.

Dentro del marco de la usabilidad requerida se planteó la necesidad de procesos guiados y trazabilidad. El concepto de proceso guiado es un paso por encima del concepto convencional de flujo de trabajo, incorporando en la interfaz de usuario una representación de los pasos necesarios para realizar un proceso y su estado, así como facilidades para retroceder a un paso anterior. Este elemento fue denominado Guía Operativa.

- Reutilización de dispositivos.

Se trataba de reutilizar toda la periferia bancaria constituida por: impresoras convencionales, impresoras bancarias, dispensadores, recicladores, escáneres etc.

- Identificación única.

El requisito consistía en tener un único modo de identificación del usuario para todas las aplicaciones que se ejecutasen en el entorno del Terminal Financiero.

En el momento en el que planteó este cambio, la mayor parte de las entidades financieras que en ese momento estaban evolucionando su terminal lo hacían con una filosofía de cliente ligero, utilizando un navegador como cliente de presentación. Sin embargo en este caso la decisión se tomó hacia una tecnología de cliente pesado, basado en Java, por los siguientes motivos:

- Punto de partida

El Terminal Financiero a sustituir era un terminal de tipo gráfico con una experiencia de usuario muy bien valorada. Las aplicaciones basadas en navegador solamente empezaban en ese momento a utilizar tecnología tipo AJAX<sup>8</sup>, haciendo que la apuesta por esta tecnología fuese arriesgada. La

---

<sup>8</sup>AJAX: Asynchronous Javascript and XML.

utilización de un navegador sin más también era un riesgo importante para el éxito del proyecto pues el usuario percibiría una pérdida de usabilidad.

- Integración

Las posibilidades de integrar las plataformas requeridas se resolvían mejor en este tipo de entorno.

- Reutilización

El número de paneles de entrada de datos diferentes que en ese momento tenía el terminal superaba los 4.000. En la prueba de concepto realizada se constató la posibilidad de reutilización en el escritorio de estos paneles, conviviendo el código "C" antiguo con el nuevo Java, compartiendo flujos de trabajo en los que se podían intercalar pantallas definidas con ambas tecnologías. Sin duda éste fue el condicionante de mayor peso.

Una de las principales ventajas de las aplicaciones basadas en un cliente ligero es la no necesidad de distribución de código a la plataforma cliente. El haber optado por un cliente pesado planteaba en principio la dificultad de tener que distribuir el código. Este tipo de distribución, a pesar de utilizar procedimientos automáticos, nunca estaba exenta de incidencias, por tratarse de un entorno en continua evolución en todas las semanas se incorporaban actualizaciones. Este problema estaba sin embargo ya solventado en el Terminal Financiero anterior por la utilización de la tecnología de terminal remoto explicada anteriormente.

Centrados en la plataforma de aplicación de escritorio con tecnología de cliente rico, el mercado se dividía entre dos opciones tecnológicas: .Net o Java. Aquí la decisión fue optar por la plataforma Java utilizando la tecnología RCP<sup>9</sup> de Eclipse, en base a los siguientes criterios:

- Presentación rica con elementos de interfaz propios de la plataforma donde se ejecuta.
- Capacidad de integrar en modo cooperativo aplicaciones desarrolladas sobre diferentes tecnologías: navegador, C, Notes, Java u otras en el futuro.
- Utilización de una plataforma incipiente pero estándar y con un grado de apoyo creciente.

---

<sup>9</sup>RCP: Rich Client Platform

- Capacidad de la plataforma Java.
  
- Distribución limitada.
  
- Capacidad a futuro de optar en cada caso por el tipo de cliente que fuera más conveniente: ligero, pesado.
  
- Existencia en la zona de profesionales cercanos al entorno, tanto por la Facultad de Informática como en la propia entidad al haber sido Java la tecnología utilizada en el desarrollo de la Banca Electrónica.

Para el nuevo terminal se construyó tanto un entorno de desarrollo gráfico, basado en la reutilización de componentes y con el objetivo de código cero, como un entorno de ejecución, que bajo Eclipse/RCP interpretaba el XML producido por el desarrollo. Esta forma de proceder resultó más lenta en ejecución de lo previsto y se decidió, antes de salir a producción, modificar el generador para producir directamente clases Java en lugar de XML [55] y ejecutarlas directamente sin necesidad de interpretar XML. El entorno de desarrollo fue evolucionando para poder generar desde el mismo, además de las aplicaciones de cliente rico, aplicaciones Web desplegadas sobre un servidor J2EE y más adelante procesos de negocio que se tratan en el capítulo 9 correspondiente a integración y BPM.

Por último, vinculado al proyecto del nuevo Terminal Financiero, se decidió abandonar paulatinamente la plataforma inicial de impresión en oficinas, en favor de la tecnología BIRT<sup>10</sup>, desarrollada dentro del marco de los proyectos Eclipse y perfectamente alineada con la plataforma última del terminal. Funcionalmente el comportamiento de BIRT era similar al de la plataforma existente, pero tecnológicamente se integraba con mayor facilidad por estar basado en conceptos Eclipse y Java. Se trataba además de una plataforma "Open Source" que dejaba abierta la posibilidad de migrar, si fuese necesario, a la solución de mercado de la empresa Actuate, origen del proyecto BIRT de Eclipse.

---

<sup>10</sup>BIRT: Business Intelligence Reporting Tools. Proyecto Eclipse.

## 6.4. Conclusiones

La evolución del terminal financiero da una perspectiva muy interesante de la evolución de las plataformas cliente. Es importante señalar como en esta evolución confluyen plataformas físicas, sistemas operativos, comunicaciones y orientación funcional.

La última versión del Terminal Financiero es un ejemplo claro de la importancia de ser capaz de evolucionar conviviendo con el pasado. La solución desarrollada, el terminal basado en Eclipse/RCP, permite la convivencia transparente de nuevas aplicaciones Java, con nuevas aplicaciones Web, con Ofimática, entornos colaborativos y fundamentalmente con el anterior terminal, del que inicialmente también heredó el núcleo escrito en C, con la capacidad de desarrollar procesos de negocio que integren en la interfaz de usuario cualquiera de esas tecnologías, compartiendo datos de forma transparente para el usuario y evitando la necesidad de teclear el mismo dato más de una vez.

Es importante señalar que las entidades que optaron por un terminal basado en navegador, tuvieron dos opciones: hacerlo en un sólo paso (big-bang), asumiendo los plazos de una migración extremadamente voluminosa, o mantener una convivencia larga entre el sistema anterior, generalmente de tipo carácter, y el sistema nuevo, duplicando en muchos casos los puestos de trabajo de la oficina y complicando enormemente la operativa en las mismas.

## Capítulo 7

# Incorporación a Internet: Webs y soluciones de Banca Electrónica

### 7.1. Objetivos

El objetivo de este capítulo es analizar cómo se produjo la incorporación de las entidades financieras a Internet. Se hace especial énfasis en las aplicaciones de Banca Electrónica en sus distintas vertientes, fundamentalmente particulares y empresas. El capítulo trata de poner en valor la importancia de una arquitectura basada en servicios para conseguir una reutilización de la operativa existente en otros canales.

### 7.2. Contexto general

Aunque la aparición de la World Wide Web es de principios de los años 90<sup>1</sup>, las entidades financieras no se incorporaron de forma masiva a esta tecnología hasta la parte final de esa década. Al final del siglo XX, en 1998, el número de usuarios de Internet en el mundo era superior a 70 millones y el número de sitios web del orden de 1,5 millones. En España el número de personas que en ese momento tenían acceso a la red era de cerca de 1,4 millones y el número de personas que usaban un ordenador de tipo personal era de alrededor de 5 millones<sup>2</sup>.

---

<sup>1</sup>En 1990 desaparece ARPANET y en 1991 se establecieron en el CERN los principios de la World Wide Web

<sup>2</sup>Fuente EGM.

El concepto genérico de Comercio Electrónico, a pesar de los ejemplos conocidos de éxito como Amazon (1995) o algunos intermediarios financieros virtuales como E-Trade o Charles Schwab, no tenía en aquél momento el desarrollo que hoy tiene. La primera banca electrónica en España habría que situarla en 1997.

### 7.3. Planteamiento

La aproximación a Internet en el caso que analizamos se hizo estableciendo varios frentes de actuación:

1. Acceso a Internet.

El objetivo era facilitar a los usuarios internos la posibilidad de consultar sitios web. Se trataba por lo tanto de resolver la necesidad de acceder a información externa. Los problemas relacionados con este acceso se centraron en dos ámbitos: seguridad y productividad. Los dos problemas llevaron inicialmente a una necesidad importante de monitorización del uso de la plataforma.

2. Correo Electrónico.

Inicialmente el correo electrónico se planteó únicamente para uso interno, sin salida al exterior. En el momento en que se habilitó el acceso a Internet de los usuarios internos se decidió también abrir el correo al exterior.

3. Establecimiento de un sitio web corporativo.

La primera Web fue de tipo institucional, para pasar rápidamente a la preocupación por los aspectos de publicidad, comerciales y a la oferta de servicios. El sitio web se fue especializando con el tiempo intentando enfocarlo cada vez más al tipo de cliente que va a utilizarlo: empresa, particular, autónomo, cliente directo etc. El diseño inicial fue independiente de los servicios que se ofrecían por las Bancas Electrónicas, aunque siempre había un enlace a las mismas. La Web fue rediseñada varias veces buscando mejorar la personalización hacia el cliente, siempre que éste estuviese previamente identificado, e integrándose de modo cada vez más transparente con las aplicaciones de Banca Electrónica, de modo que el cliente no necesitase nunca una segunda identificación.

#### 4. Banca Electrónica.

Es la aplicación o conjunto de aplicaciones Internet por antonomasia de las entidades financieras y constituye el núcleo de lo que en general se denomina Comercio Electrónico para este tipo de organizaciones, sin olvidar las tarjetas, especialmente las de crédito. Así como el planteamiento de un sitio web es muy similar para cualquier tipo de empresa, la Banca Electrónica es estrictamente operativa bancaria. Por su importancia, por la necesidad de construirla sobre sistemas existentes y por su relación con la parte II de esta tesis se ha considerado dedicarle una sección específica en la siguiente sección de este documento.

#### 5. Aplicaciones Internet para usuarios finales.

Dentro de esta categoría se sitúan las aplicaciones que las entidades financieras distribuyen a sus clientes con propósito variado. Se trata tanto de soluciones de contabilidad doméstica, que se comunican con la entidad fundamentalmente para actualizar posiciones y movimientos de cuentas, como de aplicaciones para empresas para trabajar con recibos, efectos comerciales o transferencias. En algunos casos no se distribuía la aplicación sino que simplemente se habilitaba el envío de información por medio de una transmisión de ficheros. Un ejemplo importante fue un acuerdo entre cerca de 30 entidades para desarrollar un paquete propio, denominado "Efectivo 98", que recogía información de las posiciones del cliente en las entidades participantes y gestionaba localmente la información de cuentas y recibos en un formato de contabilidad doméstica.

## 7.4. La Banca Electrónica

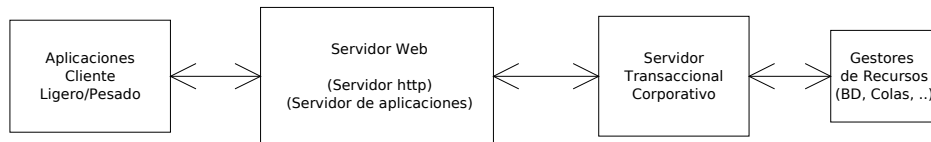
Las aplicaciones de Banca Electrónica ofrecen al usuario final la posibilidad de realizar operativa bancaria sin necesidad de acudir a las oficinas o a los cajeros. Es por ello que el reto más importante para su desarrollo fue el de la reutilización de la operativa existente.

En el momento en el que se plantearon estas aplicaciones existía ya un Servidor Financiero transaccional diseñado para atender al Terminal Financiero de las oficinas. Con este servidor se había resuelto la operativa con clientes presenciales, o de oficina, y posteriormente había sido reutilizado en el canal de cajeros, integrándose con sus protocolos<sup>3</sup>. La situación que se planteó con la Banca

---

<sup>3</sup>La Parte II de esta tesis trata este punto

Electrónica fue similar, se trataba de dar servicio a los clientes pero ahora utilizando tecnología diferente ya que aparecieron nuevos elementos técnicos como Servidores de Páginas y detrás de ellos Servidores de Aplicaciones, configurando típicamente arquitecturas cliente-servidor de tres niveles que en este caso, por existir además el Servidor Financiero de tipo transaccional que resolvía la lógica funcional de las aplicaciones y el acceso a las bases de datos, habría que hablar de al menos cuatro niveles [27].



**Figura 7.1:** Arquitectura de 4 niveles

La primera presión para instrumentar una Banca Electrónica en una entidad financiera vino generalmente del ámbito de las empresas, fundamentalmente las más grandes que a su vez suelen tener mayor desarrollo informático propio, este fue el caso en la experiencia que aquí se describe.

Las aplicaciones de Banca Electrónica de Empresas tienen características diferenciadoras en al menos dos aspectos:

1. Gestión de autorizaciones

La diferencia estriba en el concepto de bastanteo. Cuando se trata de clientes particulares el usuario de la Banca Electrónica es único y tiene atribuciones para operar en nombre propio, sin embargo en las aplicaciones para clientes empresa los usuarios operan en nombre de sus empresas, haciendo uso de los apoderamientos que les fueron otorgados. De esta forma para hacer un tipo de operación pueden necesitarse dos firmas y para otra operación diferente podrían hacer falta esas mismas u otras, en ocasiones condicionado también por el importe de las operación. Esta particularidad complica el concepto de contrato para operar en Banca Electrónica de Empresas que debe recoger toda esa información que es necesario considerar en el momento en el que se ejecuta una operación. Además es usual en este tipo de aplicaciones que las operaciones queden en un estado de espera hasta que registran el complemento de firma si éste fuese necesario.



## 2. Operaciones masivas

La segunda característica es la necesidad de envío masivo de operaciones (ej. transferencias o recibos) para lo que fue necesario incorporar en estas aplicaciones la posibilidad de enviar o recibir ficheros con este tipo de operaciones, así como efectuar las validaciones necesarias para que éstas sean conformes con la normativa de intercambio establecida por el CSB<sup>4</sup>. Estos ficheros se caracterizan por incorporar, además de las operaciones, una estructura compuesta por registros de cabecera de identificación y registros finales de totalización.

La primera aplicación de banca electrónica instalada fue la de Banca Electrónica de Empresas. No se trató de un desarrollo interno, fue adquirida en 1998, se ejecutaba en servidores Unix que conectaban con el transaccional mediante el protocolo APPC<sup>5</sup>. El proyecto de implantación fue por lo tanto un proyecto de integración; dado que los servicios a ofrecer en Banca Electrónica tenían diferente granularidad que los existentes, planteados inicialmente para el canal oficina/cajeros, se diseñó un elemento intermedio en el gestor transaccional, bautizado como "distribuidor", cuyo papel era orquestar la invocación de servicios elementales bajo el ámbito de una única transacción, garantizando de esta forma la integridad transaccional global.

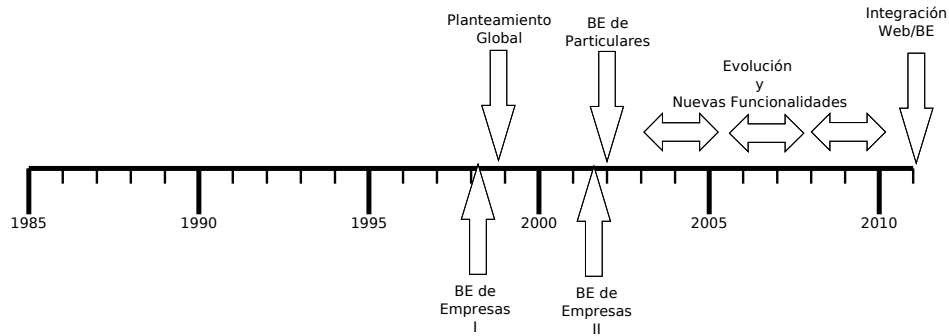
Posteriormente se definió una arquitectura propia para aplicaciones de Banca Electrónica utilizando tecnología Java, haciendo uso de los conceptos típicos de la plataforma: Contenedores, JSP, Servlets etc., pero manteniendo la integración con el transaccional y la filosofía de los denominados "distribuidores". Sobre esta arquitectura se desarrolló la Banca Electrónica de Particulares. Este desarrollo se hizo en 2002/2003, incorporando paulatinamente mayor funcionalidad en la medida en que ésta era solicitada por el mercado y se disponía de la aplicación de servicio correspondiente.

La última versión de la Banca Electrónica (2012) se hizo utilizando la tecnología desarrollada para el Terminal Financiero explicada en el capítulo anterior. Para ello fue necesario ampliar la plataforma de desarrollo construida para el terminal financiero Eclipse/RCP en dos direcciones. En primer lugar para soportar el desarrollo de aplicaciones basadas en navegador y en segundo lugar para contemplar aplicaciones basadas en servidor, introduciendo de esta forma un nivel más en la arquitectura. La primera de las dos evoluciones se utilizó para construir la Banca Móvil, ya que se optó en este caso por aplicaciones basadas en el navegador del

---

<sup>4</sup>CSB: Consejo Superior Bancario.

<sup>5</sup>APPC: Application Program to Program Communication. Protocolo de la familia SNA



**Figura 7.2:** Evolución web y Banca Electrónica

dispositivo frente a aplicaciones nativas del mismo. La segunda se utilizó para realizar la Banca Electrónica en su última versión integrada con la Web corporativa.

La decisión de utilizar en dispositivos móviles aplicaciones basadas en navegador tuvo que ver fundamentalmente con el objetivo de tratar de independizarse de los dispositivos. Indudablemente esto tuvo el precio de la flexibilidad de uso, que es sin duda mejor en las aplicaciones nativas. Se trató de una decisión importante, apostar por el lenguaje HTML y su evolución, pero controvertida ya que la presión por utilizar en sus últimas capacidades los denominados "smartphones" fue muy fuerte por parte de los usuarios, principalmente por personas de alto nivel en la empresa que generalmente eran los primeros en disponer de este tipo de dispositivos. Sin embargo, si se piensa en términos de generalización y facilidad de gestión, el desentenderse de los dispositivos no gestionados por la entidad fue coherente con la solución utilizada en la Banca Electrónica, también instalada sobre navegadores frente a la opción de soluciones distribuidas a los clientes. En cualquier caso la alternativa no habría sido desarrollar de forma específica para cada dispositivo, pues eso llevaría a centrarse en los líderes del mercado y estar limitados ante nuevos contendientes como por ejemplo Windows, sobre todo si se tiene en cuenta que el número de usuarios en ese momento era muy reducido. Como alternativa se evaluó también la instalación de una plataforma propia de desarrollo multidispositivo y con distribución de software lo que sin embargo también hubiese estado en contra de la costumbre de los usuarios de bajarse ellos mismos las aplicaciones desde los denominados "markets" o "stores" existentes para estos dispositivos.

## 7.5. Conclusiones

La incorporación de las entidades financieras a Internet se produjo con distintos planteamientos. Algunas entidades, las menos, se posicionaron claramente desde el principio en este canal de distribución, la mayoría, sin embargo, realizó una aproximación de tipo defensiva, tratando de no canibalizar a sus propios clientes de oficinas que en este nuevo escenario tendrían más fácil el cambio de entidad. Una opción intermedia fueron las llamadas bancas telefónicas que posibilitaban la operación remota pero a través de un operador que mantenía el vínculo con la entidad.

En cualquier planteamiento que se siga, más o menos radical, existen dos facetas muy diferentes: la presencial de la Web y la operativa de las Bancas Electrónicas. Como ya se indicó el planteamiento Web no es específico de las entidades financieras aunque pueda tener algunas características propias. El que es sin duda diferencial es el de las Bancas Electrónicas por la importancia que tiene la reutilización de servicios.

Es importante señalar que en esta experiencia el planteamiento inicial de independencia entre el Terminal Financiero y el Servidor Financiero fue la base para disponer de una arquitectura de servicios, que si bien no fueron diseñados en origen como servicios reutilizables sino como servicios ajustados a los diálogos establecidos con el Terminal Financiero, pudieron ser fácilmente adaptados y orquestados para poder ser utilizados en las distintas Bancas Electrónicas: Particulares, Empresas, Banca Móvil, etc.



## Capítulo 8

# Gestión documental

### 8.1. Objetivos

En este capítulo se trata de analizar las distintas facetas de la gestión documental en las entidades financieras. Se deja fuera de este análisis la gestión documental interna, como pueden ser el almacenamiento de expedientes de los empleados o el de los contratos con los proveedores, para centrarse en los aspectos que han tenido un mayor impacto económico, concretamente en la distribución y en el almacenamiento de información hacia oficinas y hacia clientes.

### 8.2. Las información documental en las entidades financieras

El tratamiento de la información documental, entendiendo como tal la de naturaleza no estructurada o semiestructurada, ha sido siempre fundamental en las entidades financieras. La información documental que manejan las organizaciones puede clasificarse en dos grandes grupos:

1. Información producida por los propios sistemas.
2. Información capturada.

La diferente naturaleza de cada tipo de información exige soluciones diferenciadas en muchos casos específicas al problema a resolver [48].

El primer grupo se divide a su vez en dos subclases. La primera de ellas es la información de consumo interno, representada por los clásicos informes o listados.

La segunda es la información destinada a los clientes tal como información sobre saldos, movimientos, información para renta o patrimonio, o la información que se produce en el ámbito de la propia oficina, en forma de justificantes o contratos, como consecuencia de las operaciones que éstas realizan.

En el segundo grupo está la información que se captura de clientes, fundamentalmente en las oficinas. Dentro de este tipo de información aparecen los documentos de identificación, firmas, documentos de fé de vida, balances de empresas, documentos de constitución etc.

El tratamiento informatizado de cada una de estas categorías de información, así como sus características de uso y almacenamiento y la capacidad existente de regenerarla, es diferente. Si se observa en el tiempo la mayor antigüedad la tienen los listados, a continuación los comunicados a clientes, después la información generada en oficina y por último la captura externa de información. En medio aparecen otro tipo de necesidades como las derivadas del uso de Internet que llevó a hacer disponibles las comunicaciones a clientes en ese canal. La diversidad de formatos, orígenes y los distintos momentos del tiempo en que se disponía de la información hicieron que la solución fuera diferente para cada uno de ellos, de acuerdo también con la capacidad técnica existente en cada momento.

### **8.3. Planteamiento y evolución**

Cronológicamente el primer problema abordado fue el del almacenamiento de los listados de tipo informe que en sí mismos constituían información histórica. En el momento en el que esto se planteó las aplicaciones no estaban mayoritariamente sobre bases de datos relacionales, al menos las más importantes y de mayor volumen de información, y no era asumible la construcción de históricos sobre las bases de datos jerárquicas existentes por razones de coste y rendimiento. El problema se resolvió inicialmente (1994) mediante un producto orientado al almacenamiento y distribución de información formateada, el concepto de informe o "report" en la literatura anglosajona.

Una cuestión diferente la representaban los listados de uso diario o control, la mayor parte para uso de oficinas o de servicios centrales. La distribución de listados en línea fue resuelta inicialmente por dos caminos. Se instaló una plataforma de distribución basada en un producto comercial que permitía almacenar estos listados y consultarlos en pantalla y por otro lado se desarrolló una aplicación específica para las oficinas. La aplicación de las oficinas permitía distribuir al Terminal Financiero los informes que las oficinas debían de recibir en el día para actuar

de forma inmediata; un ejemplo puede ser el listado de los descubiertos en cuenta. La diferencia entre ambos sistemas es que el primero es de tipo reactivo, mientras que el segundo es proactivo tratando de mover a la revisión inmediata.

En cuanto a las comunicaciones masivas a clientes, la gestión documental se planteó con un doble objetivo: optimizar el coste de distribución y mejorar la comunicación con los clientes. El idea fundamental era reducir la frecuencia con la que se enviaban sobres a los buzones de los mismos, muchas veces conteniendo un único documento. Para solucionar este problema se instaló una aplicación de distribución de correo que interceptaba las comunicaciones dirigidas a los clientes; las aplicaciones dejaron de realizar la impresión directa para enviar la información a esta aplicación, que la almacenaba en un repositorio de distribución de correo, identificando, mediante un concepto de "línea técnica"<sup>1</sup> que la precedía, datos tales como cliente, contrato o tipo de documento. La aplicación de distribución de correo mantenía un repositorio con estos documentos que recibía de las aplicaciones, produciendo grupos por destinatario en base a criterios fundamentalmente de períodos de retención y número máximo de documentos por sobre. La aplicación citada, aunque inicialmente fue adquirida, tuvo que ser adaptada en la propia instalación, sufriendo además una evolución interna muy importante. La impresión se producía en formato AFP<sup>2</sup>, de acuerdo con el parque de impresoras instalado, y también era en ese formato se almacenaban los documentos al ser capturados directamente de las colas de impresión. El formato AFP optimiza el espacio al separar los recursos compartidos, por ejemplo las imágenes de marca o los textos comunes, de los datos propios de documento y poder ser los primeros compartidos.

Con la aparición de las aplicaciones de Banca Electrónica se hizo necesario poner este tipo de documentos a disposición de los clientes utilizando el concepto de "Correspondencia Virtual". A tal efecto se instaló un repositorio documental, en el año 2005, en el que al mismo tiempo que se almacenaba la información en el repositorio de distribución de correo, se cargaba en este repositorio para su acceso en línea, indexado por cliente y contrato. Para la presentación en Internet de estos documentos fue necesaria la conversión al vuelo del formato AFP al formato PDF.

El tratamiento de la información histórica evolucionó en sintonía con la migración de las aplicaciones a tecnología relacional. Las aplicaciones pasaron a responsabilizarse directamente de sus propios datos históricos diseñando en ellas tablas para tal fin. Esto fue posible no solamente por el cambio a base de datos relacional, si no por el aumento de la capacidad y la velocidad de acceso del almacenamiento secundario. En el año 2006 se sustituyó el producto de almacenamiento y distribución de listados por otro que mejoró sensiblemente las

---

<sup>1</sup>Registro que precede al documento con información para su clasificación

<sup>2</sup>AFP: Advanced Function Printing

consultas sobre la información tipo informe, permitiendo a los usuarios internos acceder a la misma de forma más flexible, seleccionando datos o eliminando columnas de la presentación, o también localizar información por algún criterio relevante. A pesar de que el nuevo planteamiento priorizó el almacenamiento de la información histórica sobre bases de datos relacionales, continuó existiendo información en listados que no precisaba ser almacenada como información histórica y que continuó siendo almacenada como listados.

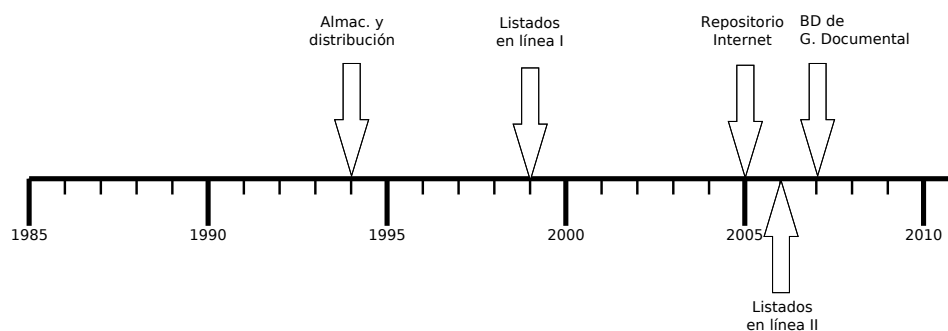
Las siguientes cifras dan una idea de la magnitud del problema:

- Número diferente de listados: del orden de 5.000
- Número de documentos para enviar a clientes en un mes: alrededor de 2,5 millones.
- Número de documentos que se trasladan a la Correspondencia Virtual al mes: 0,5 millones aproximadamente. No todos los documentos que se envían a clientes pasan a la Correspondencia Virtual ya que no todos los clientes tienen habilitado el servicio.

La información impresa directamente en oficinas presenta su propia problemática. La forma de tratarla está supeditada al tipo de plataforma desplegada en las mismas, el Terminal Financiero explicado en el capítulo 6. El modelo elegido para la gestión de la impresión en oficinas fue el de un servidor específico de composición de documentos. El entorno separa el diseño de los documentos y los formatos de su utilización en tiempo de ejecución, los documentos pueden ser estáticos o dinámicos, en cuanto a su estructura, y de tamaño variable, pudiendo llegar a decenas de páginas en el caso de algún contrato. En tiempo de ejecución se invoca el servicio correspondiente que alimenta el modelo o formato seleccionado con los datos a incorporar. El modelo puede incorporar su propia lógica de ejecución dando lugar al concepto de formato dinámico.

Por último para la información capturada de clientes, como pueden ser: documentos de identificación, declaraciones, balances de empresas etc., que es de naturaleza más variable y menos controlable en cuanto a su formato y estructura, se decidió utilizar una base de datos documental de propósito general (2006/2007). Es importante señalar que en el momento en el que se tomó esta decisión ya existían otros repositorios documentales en la instalación, fundamentalmente el relacionado con la información accesible por los clientes mediante la Banca Electrónica, por lo que de nuevo se plantea un problema importante de reutilización, migración y coexistencia.





**Figura 8.1:** Evolución temporal de los sistemas documentales

## 8.4. Proyecto de base de datos documental

Se estableció como requisito fundamental conseguir la independencia entre las aplicaciones y los gestores documentales utilizados. Para ello se definieron dos elementos:

- Los servicios documentales.
- El catálogo documental.

Los servicios documentales, definidos como servicios web independientes de la plataforma que necesite invocarlos, establecen la funcionalidad necesaria para manejar un repositorio documental tanto en consulta como en actualización definiendo una interfaz que aísla a las aplicaciones del conocimiento del gestor documental concreto con el que se está trabajando. De esta forma el repositorio accedido para trabajar con un tipo de documento puede variar en el tiempo sin que las aplicaciones que los usan se vean afectadas.

El catálogo documental, construido sobre una base de datos relacional, define cada tipo de documento y lo clasifica en su categoría. Es en este repositorio donde se definen los metadatos necesarios para cada tipo de documento así como en qué repositorio está ubicado, convirtiéndose en el "Diccionario" de la Gestión Documental.

La definición de los servicios documentales representa un problema técnico que necesita del conocimiento de las APIs de acceso a los distintas plataformas integradas y su dificultad es estrictamente esa: dificultad técnica. Lo verdaderamente complejo e interesante es el planteamiento del repositorio de metadatos documentales, el que se ha denominado catálogo documental. Su construcción exige un análisis detallado de los distintos tipos de documentos que trata una empresa,

sus características, su clasificación -necesaria para su indexación - y su forma de uso, así como características de versionado, seguridad, ciclo de vida, formato de almacenamiento, formatos de presentación etc.

## 8.5. Conclusiones

Todos los aspectos comentados en relación con la Gestión Documental son importantes, los últimos son los que tienen el mayor atractivo tecnológico por abordar cuestiones como el acceso a los documentos a través de Internet o la utilización de tecnología de almacenamiento y acceso de bases de datos documentales de propósito general. Sin embargo es necesario hacer hincapié en la gran importancia que, desde el punto de vista económico, tuvo para la entidad la distribución de listados a oficinas en línea y la distribución del correo físico a clientes de forma unificada. La distribución en línea de los listados supuso, además de un ahorro muy grande de papel, una reducción muy importante en toda la logística de distribución, en el almacenamiento físico para su distribución, en el espacio necesario para su archivado en oficinas y también en la manipulación que exigía todo el proceso. Por otra parte el correo unificado a clientes redujo de forma sustancial el coste del franqueo de los sobres, además de hacer disminuir también el número de ellos, el número de carátulas de identificación, el número de sobres que se enviaban con un único documento y de aumentar con todo ello la satisfacción de los clientes al colaborar a reducir la inundación de comunicados que se produce en sus buzones.

Es importante resaltar la importancia de los metadatos al abordar un proyecto documental. En los sistemas documentales, de la misma forma que en los operacionales o que en los orientados a la información, las definiciones precisas, la estandarización y la estructuración correcta son las claves para el éxito. Es fundamental la existencia de un catálogo documental, en el mismo sentido en el que se puede hablar del modelo conceptual de datos al definir una base de datos convencional. Realmente en un proyecto de base de datos documental existen dos niveles de metadatos, el primero es el que define el esquema de la propia base de datos documental, en cierto modo lo que es su modelo lógico, definido por medio de clases y sus relaciones; el segundo es el catálogo documental mencionado, que resuelve aspectos de negocio y que podría ser único o existir catálogos diferentes para distintos propósitos.

También es fundamental el planteamiento de independencia entre las aplicaciones y el gestor documental. La capa de servicios documentales, apoyados en el catálogo documental, aísla de la tecnología concreta de cada gestor documental que se utilice y abre de esta forma el camino de evolución hacia otras tecnologías que pudieran presentarse en el futuro.

## Capítulo 9

# Arquitectura de integración de aplicaciones y BPM

### 9.1. Objetivos

La integración de aplicaciones presenta dos escenarios claramente diferenciados: integración interna e integración externa. En este capítulo se parte de la situación inicial, la que existía en 1986, para analizar desde ese punto la evolución de las necesidades de integración, cómo se planteó en el caso que se analiza y finalmente cómo se llegó a los entornos de integración de propósito general y a los sistemas BPM<sup>1</sup>.

### 9.2. Planteamiento general

La necesidad de integrar aplicaciones es prácticamente tan antigua como la implantación de la informática en las empresas. En un principio las aplicaciones nacen aisladas para resolver problemas concretos pero en seguida aparece la necesidad de integración, es frecuente que la primera necesidad de integración se presente con los Sistemas Contables, a los que las aplicaciones de negocio tienen que pasar información de sus operaciones. Este tipo de problemas se resolvían inicialmente usando intercambio de ficheros. Dado que estas integraciones tenían lugar sobre el mismo ordenador la situación estaba bien controlada, siempre que existiese una planificación de trabajos adecuada.

---

<sup>1</sup>BPM: Business Process Management.

La situación en el tiempo se complicó en dos direcciones:

1. Sistemas distribuidos.

Los sistemas distribuidos aparecieron de dos formas. Por un lado los usuarios comenzaron a tener ordenadores de tipo PC que necesitaban información de los sistemas centrales para realizar sus simulaciones y análisis. De otra parte la distribución de las aplicaciones, en plataformas diferentes al ordenador central, hizo aparecer la necesidad de intercambio de información entre los sistemas.

2. Intercambios con el exterior.

El intercambio de información con el exterior es un aspecto de la máxima importancia en la informática de las entidades financieras, se pueden mencionar desde los intercambios con la Administración del Estado, realizados con un sin fin de organismos de ámbito local o nacional, hasta los intercambios, cada vez más frecuentes, con clientes o la integración en redes financieras como puede ser el ejemplo de la red SWIFT<sup>2</sup>.

Estas necesidades fueron apareciendo a lo largo del tiempo y las soluciones que se dieron a las mismas dependían de la magnitud del problema, su criticidad, la tecnología disponible, la situación de cada aplicación, el momento del tiempo en el que era necesario resolverlas y hasta el plazo para hacerlo. El último paso de esta evolución lo representan los entornos BPM que los fabricantes hoy presentan como la solución a todos los problemas de la informática en las empresas.

Cuando una organización se plantea una solución BPM[35], lo hace en base a una o más de las siguientes motivaciones:

- Formalizar de algún proceso definido.
- Tener un mejor control de los procesos.
- Mejorar de productividad.
- Unificar de modos de operar.
- Mejorar el soporte del cumplimiento regulatorio.
- Flexibilizar los sistemas.

---

<sup>2</sup>SWIFT: Society for Worldwide Interbank Financial Telecommunication.

### 9.3. Planteamiento en la Entidad Financiera

Como todos los aspectos señalados anteriormente se consideraron relevantes, sería difícil destacar a uno de ellos frente a los demás cuando se decidió acometer la implantación de un entorno de integración en la reflexión que se hizo cuando se abordó el último proyecto de Terminal Financiero. Nos situamos en el año 2006 y es un momento en el que suena con fuerza el concepto de "Proceso de Negocio", cuyo soporte fue uno de los requisitos para el nuevo terminal. Es el momento en el que el mercado empieza a inundarse de soluciones catalogadas con el nombre genérico de plataformas BPM.

Frente a la posible adquisición de una solución comercial de BPM, se decidió realizar un desarrollo interno utilizando como punto de partida la plataforma creada para el Terminal Financiero Eclipse/RCP. Este terminal contemplaba en su diseño inicial procesos comerciales simples que resolvían flujos de interacción sencillos sin contemplar ni la persistencia, ni la colaboración. Un ejemplo de proceso comercial sería el que resuelve la contratación encadenada de dos productos, por ejemplo una cuenta y una tarjeta, sin que el operador tenga que acceder al segundo tras haber finalizado con el primero. La plataforma permitía definir de forma externa, en modo gráfico, estos flujos y construir su guía operativa (Cap. 6, pág. 41), si se consideraba necesario.

La decisión tomada fue la siguiente:

1. Definir un primer nivel de integración en la interfaz de usuario.

La tecnología Eclipse/RCP, con sus conceptos de perspectivas y vistas, fue la base para esta integración. Este nivel de integración definió lo que en aquel momento denominamos "integración a nivel de la presentación" que utilizaba el concepto de "escritorio integrado".

2. Instalar un entorno técnico de integración de aplicaciones (EAI).

En este caso se seleccionó una plataforma comercial que cubría los requisitos planteados y especialmente el de integración con la red financiera SWIFT.

3. Evolucionar el Terminal Financiero para soportar Procesos de Negocio en sentido amplio.

Se trataba de evolucionar el concepto inicial de proceso comercial simple al concepto más general de proceso que incluyese persistencia y colaboración. La

persistencia permite abandonar un proceso para retomarlo en otro momento del tiempo, mientras que la colaboración permite asignar la siguiente actividad de un proceso a un usuario o rol diferente. El entorno de desarrollo de terminal se modificó para poder definir estos procesos y poder ser desplegados en el propio terminal o en un servidor de aplicaciones, en cuyo caso la colaboración podría establecerse entre distintos canales, interaccionando en el proceso clientes y personal de la entidad. Un ejemplo de esto último puede ser un préstamo solicitado por un cliente en Banca Electrónica, que la oficina tramita aportando información y que un departamento central de riesgos debe aprobar, para por último comunicar de nuevo con el cliente.

Esta forma de plantear el problema resultó satisfactoria pues con este Gestor de Procesos de Negocio, construido en el ámbito del Terminal Financiero, se pudieron acometer los procesos más importantes, generalmente ligados a la concesión de operaciones de activo. Posteriormente, en el año 2011, se construyó una versión ligera, que se denominó "Gestor de Trámites", que permitía la recogida en oficinas de la información básica para realizar determinadas operaciones, información que se transmitía a un equipo centralizado, interno o externo a la entidad, para continuar la ejecución del proceso, descargando a la oficina del trabajo administrativo que suponía la introducción de operaciones para poder dedicar más tiempo a la gestión comercial.

## 9.4. Conclusiones

La existencia de un entorno de integración abrió nuevas posibilidades a los intercambios externos facilitando la apertura de la instalación a los servicios web, además de una mejor integración con la red financiera SWIFT. Este tipo de servicios se podrían haber habilitado desde el propio gestor transaccional pero por motivos de rendimiento, asociados a la carga que supone el procesamiento de los mensajes XML, se consideró más efectivo y flexible incorporar su tratamiento dentro de esta plataforma, convirtiéndola a en adelante en el nexo de conexión entre los procesos de negocio y los servicios por estos solicitados.

Una aplicación muy interesante del entorno de integración fue la que se diseñó para gestionar los intercambios de ficheros. En una entidad financiera el volumen de ficheros que se intercambian, tanto interna como externamente, es muy elevado, lo que exige una gestión cuidadosa. El sistema, que se construyó apoyado en el entorno de integración y en el Diccionario de Datos, permite la definición y monitorización de estos intercambios. El Diccionario de Datos resuelve la casuística de los formatos y el entorno de integración los protocolos de transmisión, el direccionamiento y las

transformaciones entre formatos. El sistema se apoya en un cliente web desde el que un usuario controla todo lo que recibe y envía; como características importantes se pueden citar los aspectos relacionados con el tratamiento de la caducidad de envíos, la posibilidad de múltiples destinatarios, la capacidad de administración y monitorización, y la interfaz para la definición de los distintos intercambios, cuyos parámetros se almacenan en una base de datos desde la que se gestiona el sistema.





## Capítulo 10

# Sistemas Informacionales y "Data warehouse"

### 10.1. Objetivos

En este capítulo se explica cómo se introdujeron los sistemas informacionales en la empresa hasta llegar al concepto que hoy conocemos con el nombre de "Data warehouse". No se entra en el detalle de los modelos diseñados ya que el foco fundamental está en los problemas de la gestión de un entorno de este tipo y en la importancia de la Administración de Datos en este contexto.

### 10.2. Situación general

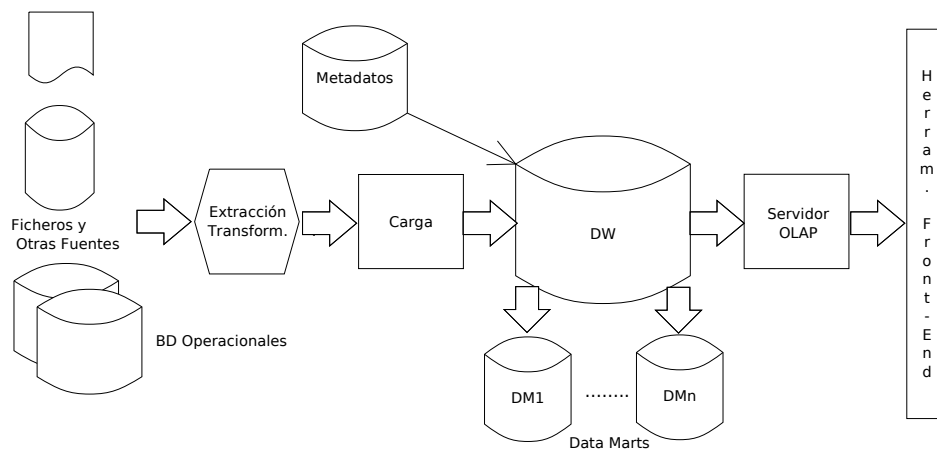
Los sistemas orientados a la información -"Sistemas Informacionales"- aparecen de forma muy temprana en las organizaciones que tienen un gran volumen de datos y en las que los factores de concurrencia y tiempo de respuesta son críticos. La necesidad de garantizar el rendimiento de los sistemas hizo que se separase claramente el uso operacional del uso orientado al análisis de la información. Fue precisamente en este último ámbito donde las empresas comenzaron a incorporar de forma temprana tecnología de bases de datos relacional frente a los sistemas inicialmente existentes, mayoritariamente basados en los modelos jerárquico o en red. A finales de los años 80 aparecieron los llamados Centros de Información con el objetivo de unificar y distribuir información, posteriormente proliferó el uso de computadoras personales en las organizaciones y nacieron las informáticas departamentales, que trataban de resolver, cada una a su manera, sus necesidades de análisis de información. Las herramientas utilizadas eran de tipo básico,

generalmente entornos locales de desarrollo y base de datos o herramientas de ofimática, sobre todo hojas electrónicas que alcanzaban niveles de complejidad muy alto. Los usuarios se animaron con el poder que les daba el tratamiento de la información en sus manos y comenzaron a desarrollar sus propias aplicaciones que en algún caso sobrepasaban sus propias necesidades de información para pasar a cubrir aspectos operativos.

El punto al que se llegó con estos planteamientos empezó a ser problemático, tanto por el nacimiento de aplicaciones que empezaban a tener alguna criticidad, como por el hecho de que los análisis que efectuaban de la misma información distintos departamentos eran inconsistentes, tanto por planteamiento como por problemas de interpretación de los datos. Las empresas comenzaron entonces un proceso de centralización de la información, inspirado en los trabajos de Inmon [42] [43] que establecieron los fundamentos de lo que hoy conocemos como almacenes de datos o "data warehouse".

### 10.3. Planteamiento en la Entidad Financiera

Por los problemas anteriormente enunciados se decidió la necesidad de un planteamiento corporativo. La tecnología jugaba ya a favor, tanto por el rendimiento alcanzado por las bases de datos relacionales como por la aparición de un número importante de herramientas especializadas. El proyecto de implantación se estableció con un sólido fundamento en la experiencia acumulada en los sistemas operacionales con la filosofía de Administración de Datos así, además de considerar las características que definen este tipo de entornos como: orientación a temas, integración, tratamiento del tiempo y no volatilidad, se hizo especial énfasis en el tratamiento de los metadatos, compartiendo el Diccionario, ya poblado por las aplicaciones operativas, y extendiéndolo en todo aquello que significase información nueva a integrar, externa, interna, o agregados. Además, siguiendo de nuevo la experiencia acumulada en los sistemas operacionales, se tuvo especial cuidado con los aspectos de recuperación y de seguridad de acceso. En la figura (10.1) se presenta la arquitectura del sistema.



**Figura 10.1:** Arquitectura del Data Warehouse

De la experiencia de implantación de un Data Warehouse se pueden señalar dos aspectos críticos:

1. El problema de la heterogeneidad de las fuentes de datos a tratar.

Aún partiendo de un entorno con una administración de datos robusta, siempre existen aplicaciones que, por su antigüedad o procedencia, no respetan los criterios establecidos por ésta. Situaciones como traducir códigos para unificar valores, formatos de fechas, valores libres que deben de transformarse a códigos, cálculo de datos derivados o acumulativos, corrección de errores o generación de claves en destino son aspectos cruciales que no se pueden infravalorar al abordar un proyecto de esta naturaleza.

2. La necesidad de integración con la planificación global.

Es necesario que las extracciones de datos desde las aplicaciones se sitúen siempre en los puntos apropiados de los flujos de ejecución y que participen de los procesos de recuperación establecidos para los mismos, de forma que las posibles repeticiones de trabajos no afecten a la calidad de la información del sistema.

En un proyecto fin de carrera de la propia facultad dirigido por el doctorando [23] se hace un análisis pormenorizado de los problemas que se plantearon en la instalación que se usa de referencia y del tipo de soluciones arbitradas. De entre ellos resaltar las siguientes como lecciones aprendidas más importantes:

- Metadatos

Su importancia es trascendental, no sólo para los equipos de desarrollo sino también para los propios usuarios finales de la información.

- Almacenamiento intermedio

Los procesos de depuración y carga pueden ser muy laboriosos. Los resultados intermedios deben ser almacenados para poder ser analizados y efectuar las correcciones que sean necesarias.

- Integridad referencial

Su uso debe de ser cauteloso pues puede incrementar notablemente el coste. En ocasiones puede ser preferible trasladar las validaciones a los propios procesos de carga.

- Rendimiento y optimización

El crecimiento en el tiempo de los Almacenes de Datos puede ser espectacular. Es necesario ser muy vigilante con la asignación de espacios y con la duración de las consultas. Los conocimientos sobre optimización en bases de datos son fundamentales.

- Métodos de captura

Generalmente es conveniente utilizar una mezcla de los mismos, utilizando métodos estáticos como: captura estática, por marca de tiempo, por comparación de ficheros, o por métodos incrementales apoyados en disparadores o ficheros de log.

- Cambios en las dimensiones

La referencia es al problema conocido como "dimensiones de cambio lento". Se trata de cambios en las mismas que pueden afectar a información almacenada anteriormente. La solución a este tipo de problemas ha sido generalmente la aplicación de técnicas de bases de datos temporales a las propias dimensiones.

## 10.4. Conclusiones

De la misma forma que el Terminal Financiero es la plataforma fundamental para las oficinas y por lo tanto la más importante en la relación directa con el cliente, los sistemas informacionales constituyen la plataforma mejor valorada por los servicios centrales, complementando y en ocasiones sustituyendo a los sistemas de gestión tradicionales caracterizados generalmente por su falta de flexibilidad. Las herramientas de usuario final, trabajando sobre los modelos desarrollados, potencian y flexibilizan la función de los analistas de negocio, dotándoles de mayor flexibilidad ante situaciones siempre cambiantes.

Resaltar especialmente la importancia que tienen en un proceso de implantación de un Data Warehouse los metadatos. Sin una definición cuidadosa de los datos y una estandarización, los modelos se convertirían en inservibles rápidamente, pues se producirían inevitablemente inconsistencias y duplicidades generando desconfianza en los usuarios; de alguna forma se repetiría la situación a la que se había llegado en los Sistemas Departamentales y que motivó la potenciación de la centralización de los modelos de información. La existencia de un Diccionario de Datos consistente para las aplicaciones operativas marcó el camino que se debía seguir en los sistemas informacionales, refrendando la función de la Administración de Datos y del Diccionario de Datos.



# Capítulo 11

## Lecciones aprendidas

Además del análisis efectuado sobre cada uno de los temas tratados y de las lecciones que de los mismos se han extraído, en este capítulo se establecen algunas conclusiones globales que se consideran importantes en la evolución de los sistemas y que en este caso están respaldadas por la experiencia profesional vivida.

### **Lo nuevo y lo antiguo**

Lo más sencillo es siempre construir desde cero pero no siempre es posible.

El caso planteado de construcción de los sitios Web pertenece a la categoría de "lo nuevo". Es un planteamiento desde cero que por lo tanto tiene pocas ataduras. El problema más importante en estos casos es tener claro qué se quiere hacer, seleccionar la tecnología, dotarse de capacidades en la misma y aplicarla. La aplicación de Banca Electrónica sería una situación intermedia entre "lo antiguo" y "lo nuevo", en este caso confluye la necesidad de una nueva tecnología con la de integrarse con los servicios de negocio existentes. Por último el Terminal Financiero es un ejemplo en el que la necesidad de apoyarse en "lo antiguo" se hace más patente, tanto en lo que se refiere a la reutilización de la aplicación C existente y sus paneles de entrada de datos, cifrados en aquel momento en más de 4.000, como en la de los servicios del núcleo, que en este caso se hizo utilizando JNI<sup>1</sup>, o de las transacciones que constituían los servicios de negocio.

---

<sup>1</sup>JNI: Java Native Interface.

## Los cambios de tecnología

Para las organizaciones la tecnología es un medio y no un fin.

Las organizaciones asumirán los cambios tecnológicos siempre que económicamente sean justificables, bien directamente, por reducir costes de propiedad, mantenimiento o utilización, o bien indirectamente, por ofrecer nuevas posibilidades al negocio que le permitan adquirir algún tipo de ventaja competitiva o reducción de coste en su propia actividad. En cualquier caso es necesario tener en cuenta también el riesgo del cambio, las pequeñas mejoras tecnológicas deben de valorarse cuidadosamente en ese sentido antes de abordarlas. Un ejemplo lo tenemos en algo tan simple, aparentemente, como evolucionar la versión de un procesador de texto u hoja electrónica que está siendo utilizado en toda una organización de forma estandarizada. Frecuentemente el nivel técnico de utilización de este tipo de productos es muy básico, limitándose los usuarios a usar aquél 20 % que resuelve el 80 % de las situaciones; la nueva versión con frecuencia incluirá mejoras que serán seguramente muy poco utilizadas, sin embargo el coste del cambio en forma de uso, formación e incompatibilidades puede ser muy alto. El ejemplo puede ser aplicable a muchas otras situaciones en las que hablamos de versiones de bases de datos o de cualquier otro producto o sistema software.

## La resistencia al cambio

Los usuarios sobre todo y también los técnicos no aceptan siempre los cambios.

Es un punto que si bien puede tener alguna relación con el anterior nos referimos a él en otro sentido. En este caso se trata del problema que se presenta al incorporar una tecnología o solución que pierde funcionalidad o facilidad de uso con la existente. Un ejemplo de esta situación se planteó con la alternativa de utilizar un navegador para el Terminal Financiero que aún siendo una solución técnica más moderna ofrecía sin embargo una experiencia de usuario peor. Nunca debemos menospreciar la resistencia de un usuario a la pérdida de funcionalidad, ya que el usuario será un freno importante en la implantación de un nuevo sistema si entiende que pierde funcionalidad o facilidad de uso. Un usuario difícilmente aceptará argumentos de tipo técnico.



## El compás

### La importancia de la Planificación Estratégica de los Sistemas de Información.

Los supuestos tecnológicos de cambio que hemos utilizado como ejemplo no suceden aisladamente, se superponen en el tiempo y es necesario acompasarlos. En una instalación compleja la Planificación Estratégica de los Sistemas de Información es una herramienta fundamental, por encima de la planificación de cada proyecto individual.

Las plataformas hardware, los entornos de comunicaciones, el software de base y las aplicaciones evolucionan. Las organizaciones, por razones de coste y obsolescencia, evolucionan sus sistemas, pero deben hacerlo siempre desde una perspectiva global, en línea con planteamientos del tipo de los que propone Zachman<sup>2</sup> [71] [72] y que desarrolló con sentido práctico Spewak [68]. Los responsables de la evolución tecnológica en las organizaciones tienen que saber mantener un equilibrio entre la oferta tecnológica existente y sus posibilidades de integración en la organización, estableciendo en muchos casos el tiempo y la pausa necesarios: el compás.

## Los principios y las consecuencias

### Es fundamental analizar las decisiones que se toman hasta el final

Con frecuencia se adoptan decisiones que un primer análisis parecen acertadas pero que si profundizamos más en sus consecuencias podemos encontrarnos con situaciones no previstas y no deseadas. El ejemplo del tratamiento de los domicilios en el Fichero Central de Clientes puede servir como ejemplo, el planteamiento era conceptualmente correcto pero la realidad es la que es y en este caso obligó a hacer un excesivo uso de domicilios no formateados que, si bien estaban contemplados en el diseño, no era la mejor opción por perderse la componente de localización geográfica existente.

---

<sup>2</sup>Zachman propone una visión bi-dimensional para los sistemas de una organización, en columnas sitúa lo que denomina las arquitecturas que inicialmente eran datos, funciones y red, y en filas los niveles que empiezan en el nivel conceptual y terminan en los aspectos físicos.



## Parte II

# La persistencia de las soluciones



## Capítulo 12

# Introducción

El objetivo de la segunda parte de la tesis es el contrario del de la primera. Así como en la primera parte se analiza la evolución informática en una empresa, en esta segunda se analizan los factores para la duración en el tiempo de los desarrollos software. Para ello se ha elegido un proyecto relevante que sirve como ejemplo de la importancia de la utilización de principios sólidos de diseño para garantizar la evolución y la durabilidad de los sistemas.

El papel del tiempo aquí es doble. Si bien es cierto que el sistema elegido ha tenido que adaptarse a los cambios que el tiempo introdujo sobre el entorno, también ha sido capaz de persistir en su funcionamiento durante más de 25 años sin modificar su diseño. Los cambios en el entorno han sido muchos, tanto desde el punto de vista del hardware y de las comunicaciones, como desde el punto de vista del software en una doble vertiente, el software de base -fundamentalmente la tecnología de la base de datos- y las aplicaciones con las que se integra.

Se trata por lo tanto de un segundo objetivo de esta tesis en el que se demuestra la importancia que tiene la realización de un diseño que sea capaz de plantear de forma adecuada los elementos claves del problema a resolver, independizándose en lo posible de la plataforma tecnológica, en línea con las propuestas de patrones de integración empresarial hoy más utilizados [31]. Uno de los aspectos del diseño que ha contribuido en mayor medida a su éxito ha sido su planteamiento como entorno de integración, que si bien no se trata en este caso de una plataforma de propósito general sí tiene en cuenta los aspectos más relevantes para este tipo de entornos como se describen en los textos de Hohpe [36] y de Manouvrier [53]. Es también relevante señalar que las referencias citadas son publicaciones de los años 2003 y 2008 respectivamente, mientras que este sistema se diseñó en el año 1986.

Una versión simplificada del sistema que se describe, la integración de una entidad financiera en una red de cajeros, lo he empleado como base para un ejercicio práctico durante los últimos años en la asignatura de Arquitectura Cliente-Servidor. En ella los alumnos construyen una versión reducida del mismo, pero con suficiente riqueza como para percibir que se están enfrentando a un problema real y poder manejar distintas tecnologías cliente-servidor.

Sobre este mismo tema se han desarrollado tres proyectos de fin de carrera [62] [2] [32], cada uno utilizando una tecnología diferente: Java/Corba, COM/MTS y Java/EJB. En el último año se ha realizado un cuarto proyecto [1] que introduce la variante de utilizar un integrador, en este caso Spring-Integrator [29] [69].

Esta parte se ha estructurado en los siguientes capítulos:

- Introducción.

Plantea los objetivos y el porqué se ha elegido este sistema en concreto.

- Antecedentes.

Se explica el contexto del proyecto y se describe la plataforma.

- El proyecto concreto.

Se recogen en este capítulo las especificaciones más significativas del proyecto, sin entrar en detalles no relevantes para el propósito de esta tesis.

- El diseño planteado.

Se explican las decisiones de diseño que sin lugar a dudas han sido el determinante más importante de la larga vida del sistema.

- Arranque y recuperación.

La criticidad del sistema hace que los aspectos de arranque y recuperación sean fundamentales en él.

- Operación y monitorización.

Se detallan los aspectos más relevantes para la correcta operación y monitorización del sistema. De nuevo es la criticidad del sistema la que hace tomar una importancia especial a este capítulo.

- Evolución del sistema.

Se detallan los cambios en comunicaciones, en el software de base, especialmente la base de datos, y en las aplicaciones con las que se integra el sistema.

- Algunos datos de volúmenes.

Se presentan algunas estadísticas sobre la cantidad de mensajes que se procesan y la distribución en el tiempo de los mismos.

- Lecciones aprendidas.

Se analizan los aspectos de diseño que han permitido la durabilidad de este sistema.

La naturaleza descriptiva de los capítulos en los que se ha dividido esta parte de la tesis hace que en ellos no existan conclusiones, reservándose las mismas para el último de ellos, el de "Lecciones aprendidas" en el capítulo 20.





## Capítulo 13

# Antecedentes

### 13.1. Contexto de desarrollo del proyecto

El proyecto que se utiliza como referencia se desarrolló en el marco de la migración de la plataforma Bull a la plataforma IBM citado en la primera parte de esta tesis en el capítulo 4. Se trata de un proyecto singular, de naturaleza más técnica y con necesidades de integración más relevantes que el resto de los proyectos acometidos en la migración.

El problema consistió en la integración de un Servidor Financiero con una red de cajeros externa basada en una topología de consorcio<sup>1</sup>. Este tipo de red es la misma que el texto clásico de Rumbaugh sobre OM/T<sup>2</sup> [65] utiliza en alguno de sus ejemplos de análisis. La topología supone que los cajeros están conectados, independientemente del banco<sup>3</sup> al que pertenezcan, a un ordenador central que denominaremos "ordenador del consorcio" y que éste recibe las peticiones de los cajeros para reenviarlas a cada banco adherido, dependiendo de cual sea el banco emisor de la tarjeta que está operando. El esquema general puede verse en la figura 13.1.

Es responsabilidad de cada uno de los bancos la implementación de la comunicación con el consorcio de acuerdo con unos formatos y un protocolo preestablecidos que en adelante denominaremos FAP<sup>4</sup>. Además cada banco debe de realizar la

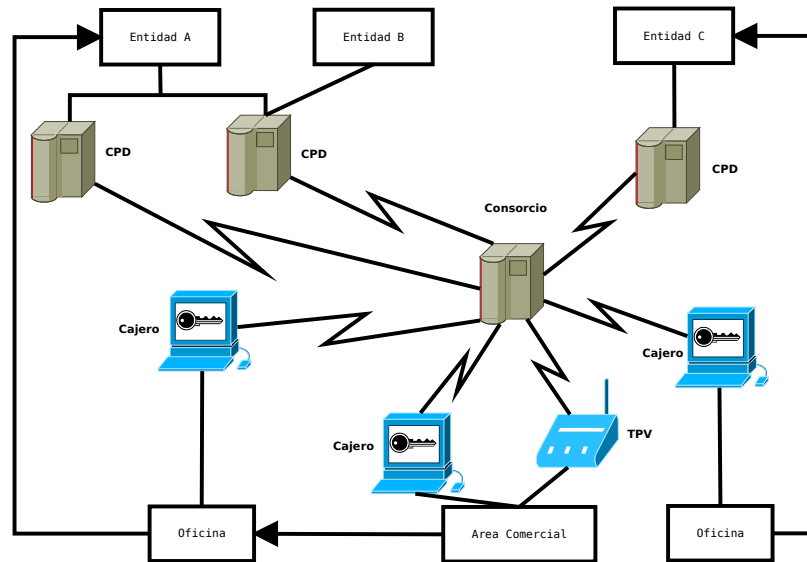
---

<sup>1</sup>consorcio: agrupación de bancos que comparten una red de cajeros conectados a un ordenador central.

<sup>2</sup>OM/T: Object Modeling Technique.

<sup>3</sup>En esta parte de la tesis utilizaremos el término banco en lugar de entidad financiera para simplificar el lenguaje.

<sup>4</sup>FAP: Formats and Protocols.



**Figura 13.1:** Topología de la red

integración con sus sistemas internos, en este caso el Servidor Financiero mencionado en la primera parte, de acuerdo con sus planteamientos internos de integración. Los formatos definidos en este FAP para los campos de los mensajes obedecen a los estándares ANSI [3] e ISO [44] establecidos para este tipo de operaciones.

La red de cajeros está integrada a su vez con otras redes, nacionales e internacionales, de forma que cuando una tarjeta de una entidad de esta red opera en un cajero externo a la misma las redes se comunican. La entidad emisora de la tarjeta recibirá la solicitud de la operación, en forma similar a las que recibe de su red propia, en un formato más de los definidos en el FAP. Es frecuente que estas operaciones tengan cruces de divisas.

El diseño elegido, que se detalla en el capítulo 15, fue el de una arquitectura de tres niveles, según el esquema de la figura 13.2, donde el primer nivel gestiona el protocolo, el segundo nivel es el Servidor Financiero en sentido estricto y el tercer nivel lo constituye el acceso a los gestores de recursos, en este caso gestores de base de datos.



**Figura 13.2:** Arquitectura de tres niveles

La ubicación de la gestión del FAP en el esquema de comunicación establecido puede verse en la figura 13.3.

## 13.2. Plataforma tecnológica

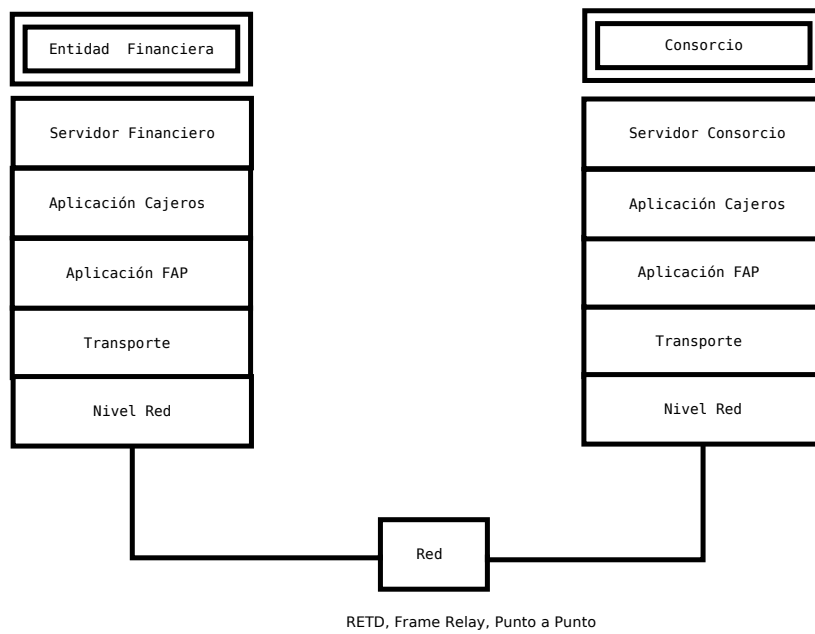
Desde el punto de vista de las comunicaciones para soportar la transmisión de datos se utilizaba la Red Especial de Transmisión de Datos (RETD) de Telefónica.

La comunicación física se establecía por medio de una unidad de control de comunicaciones modelo IBM 3725. El software de esta unidad consistía en el programa de control de comunicaciones NCP, sobre el que se instalaba el ISARD<sup>5</sup>, para la integración de la RETD en la arquitectura SNA, trabajando en el modo que aparece documentado como ICA<sup>6</sup>. El método de acceso de comunicaciones era VTAM<sup>7</sup>, que por un lado conectaba con los terminales, en este caso la línea RETD con el consorcio, y por el otro con el monitor de teleproceso sobre el que se implementaron tanto el FAP como el Servidor Financiero.

<sup>5</sup>ISARD: Interface SNA a Red de Datos.

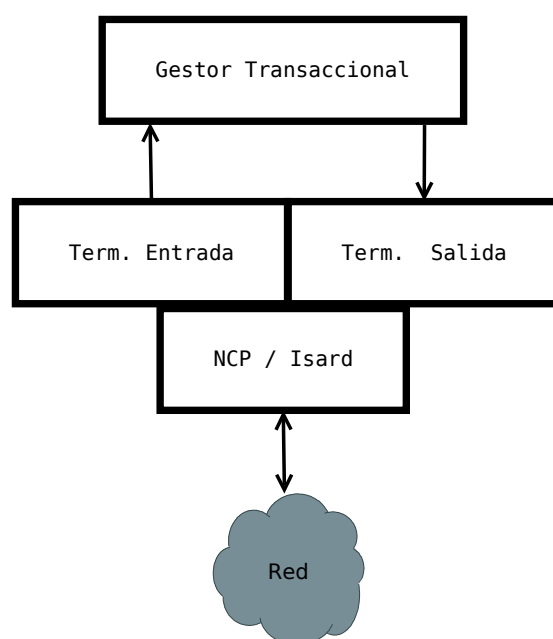
<sup>6</sup>ICA: Interconexión de Centros de Abonado.

<sup>7</sup>VTAM: Virtual Telecommunications Access Method (IBM). Control de red SNA.



**Figura 13.3:** Esquema de comunicación

Cada conexión por medio del sistema ICA exigía la definición de dos terminales, uno para entrada y otro para salida. Estos terminales se definían también en el CICS y constituían los elementos lógicos por donde se recibían o enviaban los mensajes intercambiados entre el banco y el consorcio (figura 13.4). En la terminología RETD el banco y el consorcio son dos Centro de Cálculo de Abonado (CCA).



**Figura 13.4:** Comunicación técnica

Los mensajes recibidos se presentaban al CICS con un identificador predefinido, incorporando, además de los datos del protocolo y de la aplicación, un código informativo del estado de la propia comunicación. Cuando lo que se pretendía era enviar a través de esta conexión los mensajes se prefijaban con un código de tres posiciones también predefinido.



## Capítulo 14

# El proyecto concreto

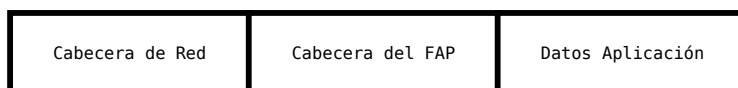
Para explicar el alcance del proyecto desarrollado y poder valorar el diseño que se explica en el siguiente capítulo, es necesario en este punto explicar con más detalle el compromiso de integración establecido a nivel de aplicación entre un banco y el consorcio, esto es, el protocolo que contempla tanto las reglas del diálogo como los formatos de los mensajes que se intercambian entre el consorcio y el banco al que genéricamente hemos denominando FAP.

### 14.1. El FAP

El FAP está concebido para independizar la capa de servicios de la aplicación, que soporta la funcionalidad del negocio, del transporte utilizado, admitiendo que éste último pueda estar orientado a la conexión o no. A pesar de que originalmente nace pensado para redes X.25, no hay ninguna limitación en que pudiera utilizarse Frame Relay, líneas punto a punto o por ejemplo el protocolo de transporte UDP de TCP/IP que no garantiza entrega ni secuencias de los mensajes. La justificación de lo enunciado está en que el propio FAP define procedimientos de recuperación de mensajes y mecanismos de secuencia que garantizan tanto la entrega como la secuencia correcta de los mismos [16]. El FAP tampoco condiciona el tipo de aplicación servidora; el diseño que se presenta es de hecho independiente de la aplicación, aunque nos centraremos en la problemática de su utilización en relación con la aplicación de la Red de Telebancos y Terminales Punto de Venta (TPV) que denominaremos por simplicidad aplicación de Cajeros.

Los mensajes que se intercambian tienen tres partes significativas: la cabecera de red, la cabecera del FAP y los datos de la aplicación (fig. 14.1). Se considera en

adelante, ya que no es objeto de este trabajo, que el protocolo de transporte ya ha eliminado la cabecera propia del mismo.



**Figura 14.1:** Estructura de los mensajes

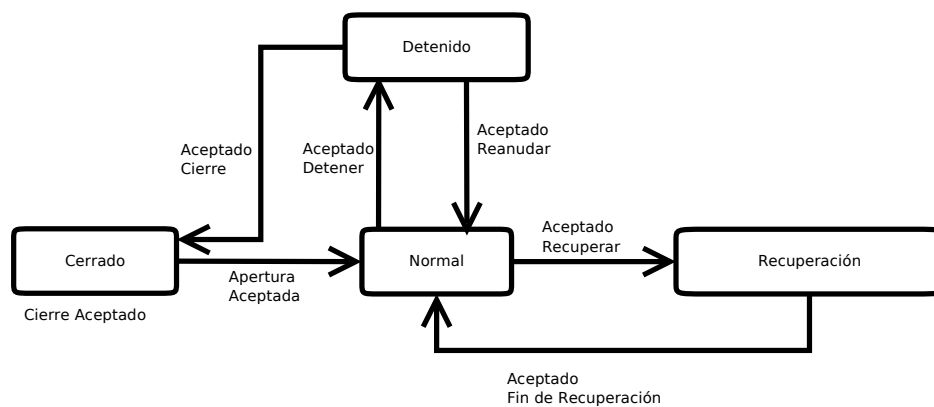
El FAP define dos tipos de mensajes:

- Mensajes de control
- Mensajes de datos

El protocolo tiene definido un concepto de estado que está gobernado por los mensajes de control (fig. 14.2). También existe un concepto de sesión, delimitada entre una apertura explícita y un cierre de la misma. Durante la sesión se produce el tráfico normal de datos de la aplicación, así como también pueden intercambiarse mensajes de control que modifican el comportamiento temporal de la misma al poder cambiar el estado. Los mensajes de control se utilizan para cuestiones como abrir sesión, detener tráfico, iniciar un procedimiento de recuperación o cerrar sesión. Para poder establecer una sesión, el servidor del consorcio debe estar previamente levantado y podrá aceptar o denegar el inicio de sesión.

Todos los mensajes, ya sean de control o de datos, tienen definido en el protocolo una respuesta. Los mensajes de control, dependiendo de su tipo, pueden tener su origen en el consorcio o en la entidad financiera, mientras que los mensajes de datos tienen siempre su origen en el consorcio pues son generados desde solicitudes que se reciben desde los cajeros (tabla 14.1). Las peticiones recibidas por el consorcio desde los cajeros a través del FAP, son enviadas a la entidad financiera que esté definida como destinataria para la tarjeta que hubiera iniciado la operación en el cajero.



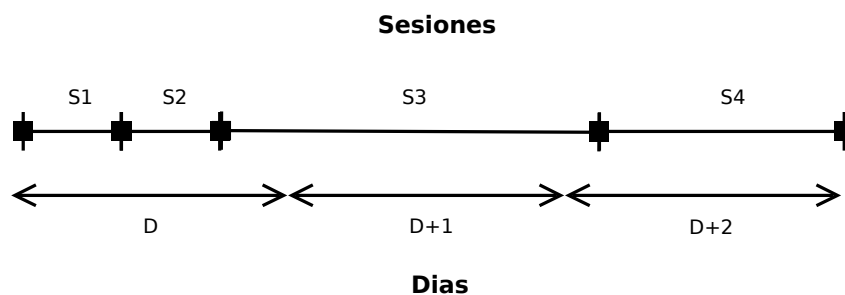


**Figura 14.2:** Estados del FAP

Tipo de mensaje	Origen
Solicitud Inicio de Sesión	entidad
Aceptación Inicio de Sesión	consorcio
Solicitud Detener Tráfico	entidad o consorcio
Aceptación Detener Tráfico	consorcio o entidad
Solicitud Reanudar Tráfico	entidad o consorcio
Aceptación Reanudar Tráfico	consorcio o entidad
Solicitud Recuperar Tráfico	consorcio
Aceptación Recuperar Tráfico	entidad
Solicitud Fin Recuperar Tráfico	consorcio
Aceptación Fin Recuperar Tráfico	entidad
Solicitud Cierre Sesión	entidad
Aceptación Cierre Sesión	consorcio

**Tabla 14.1:** Mensajes de control

La responsabilidad de abrir y cerrar una sesión es siempre de la entidad financiera que se conecta al consorcio. La duración de la misma puede ser variable, podría haber varias sesiones en un día o una sesión extenderse a lo largo de más de un día. La entidad financiera ajustará normalmente su duración para sincronizarla con su planificación de procesos, especialmente los procesos contables y de control y los de cierre de las aplicaciones integradas. En la figura 14.3 se representan las posibles duraciones de las sesiones.



**Figura 14.3:** Duración de sesiones

Un banco puede tener más de una sesión con el consorcio. Sucede este hecho cuando existen varios centros de proceso de datos operativos o cuando se quiere hacer algún tipo de separación, bien sea geográfica o también por la necesidad de separar sesiones en diferentes monedas, por ejemplo euros y pesetas, como sucedió en el cambio al euro. El FAP, en su cabecera, identifica la entidad y el número de sesión. Se puede solicitar un inicio de sesión siempre que no exista ya una sesión establecida para esa entidad y con ese mismo número. Si ese fuese el caso se tendría que solicitar previamente el cierre de la sesión anterior ya que no se permiten sesiones simultáneas para la misma entidad financiera con el mismo número.

Un parámetro fundamental en el establecimiento de la sesión, que decide el banco, es el número de terminales o canales lógicos que se van a utilizar durante la sesión. Este parámetro limita el número máximo de peticiones concurrentes que admitirá durante la sesión y no es modificable dinámicamente; para hacerlo sería necesario cerrar la sesión y abrir una nueva que tuviese el número de canales deseado.

Establecida la sesión es el consorcio quien inicia el tráfico de datos hacia el banco ya que los mensajes de datos siempre tienen su origen en los cajeros y es el consorcio quien los dirige a cada banco. Cada petición del consorcio a la entidad financiera podrá enviarse por cualquiera de los canales disponibles, entendiendo que un canal

está libre si se cumple una de las condiciones siguientes:

1. No fue utilizado todavía durante la sesión.
2. La última petición que se realizó por el mismo ya ha sido contestada por el banco.

Como consecuencia de lo anterior, si en un momento del tiempo el número de peticiones que recibe el consorcio hacia una entidad superase el límite que imponen los canales, sería responsabilidad de éste su encolamiento. Este funcionamiento en modo de "pool" de conexiones lógicas, protege la carga que puede llegar al servidor de la entidad financiera. El banco responde a las peticiones que le llegan, como máximo tantas concurrentes como canales definidos, sin necesidad de que se respete el orden de recepción de las mismas; es decir, el orden en el que se responde a cada canal no es relevante, aunque sí el orden de los mensajes dentro de cada canal.

Los mensajes de datos tienen un identificador único dentro de la sesión, compuesto por el número del canal y el número de secuencia del mensaje dentro del mismo. Este número de secuencia se inicializa a uno, en cada canal, al comenzar la sesión y el consorcio lo progresa en cada nueva petición.

## 14.2. Mensajes de control

Los mensajes de control definidos cubren las siguientes situaciones:

1. Solicitud de apertura de sesión.

Define los parámetros de la sesión, fundamentalmente el número de canales a utilizar.

2. Solicitud de cierre de sesión.

El cierre de sesión lo solicita siempre el banco que es quien la abrió. Los mensajes de cierre de sesión incorporan información estadística a efectos de cuadro global: número de operaciones de cada tipo, importe total por tipo, etc. En el caso de haber alguna disconformidad entre la información del consorcio y la del banco, el FAP no prevé la conciliación sino que ésta tendría que hacerse por fuera del sistema, generalmente usando procesos "batch" de comparación y análisis detallado de las operaciones de cada extremo, que a tal efecto habrán sido intercambiadas por medio de una transmisión de ficheros.

### 3. Solicitud de detener/reanudar tráfico.

La solicita el banco en el caso de plantearse algún problema en sus sistemas. El banco deberá seguir atendiendo mensajes de datos en tanto no reciba la respuesta de conformidad, momento en el que el tráfico queda efectivamente detenido, no pudiendo el consorcio enviar nuevas peticiones mientras la entidad no solicite la reanudación del tráfico.

Para poder solicitar un cierre de sesión es necesario haber conseguido previamente detener el tráfico. Las peticiones que el consorcio reciba de los cajeros cuando no hay tráfico, bien por que el banco lo ha detenido o porque todavía no se hubiera establecido sesión, pueden ser atendidas directamente por el consorcio contra el riesgo establecido para cada tarjeta. En el momento en el que se establezca la conexión, si en el parámetro correspondiente de sesión hubiera sido establecido así, el consorcio las enviará en el tráfico normal de peticiones pero incorporando un indicador de que ya fueron atendidas en el cajero, de forma que el banco lo pueda tener en cuenta en su proceso de autorización que, en este caso, es sobre un hecho consumado.

### 4. Solicitud de recuperación/fin de recuperación.

Lo solicita normalmente el consorcio en el caso de producirse algún problema con el nivel de transporte, por ejemplo un time-out. El banco está obligado a aceptar siempre la recuperación. En estado de recuperación el consorcio puede efectuar dos tipos de envíos:

- a) El último mensaje de datos que ya hubiera enviado por cada canal y de los que no ha recibido respuesta.
- b) Mensajes de datos no enviados pero donde el canal a utilizar está libre.

Si el mensaje que recibe el banco es nuevo y conforme con el estado del canal lo procesará como una solicitud en tráfico normal. Si recibe en un canal la última pregunta que ya había entrado por el mismo, caso de reenvío, lo procesará como una nueva solicitud, si no le consta la respuesta, o reenviará al consorcio la respuesta que ya tuviere. De esta forma al estar los mensajes identificados por el número de canal y el número de secuencia se asegura la idempotencia de las solicitudes, no procesándose en los sistemas del banco nunca más de una vez una solicitud (ej. un reintegro o una transferencia). Los mensajes en recuperación se identifican mediante un bit que lo indica.

Un detalle mayor de la estructura de los mensajes intercambiados se puede ver en el apéndice B.

### 14.3. Estados del FAP

Los estados del FAP se resumen en el diagrama de la figura 14.4.

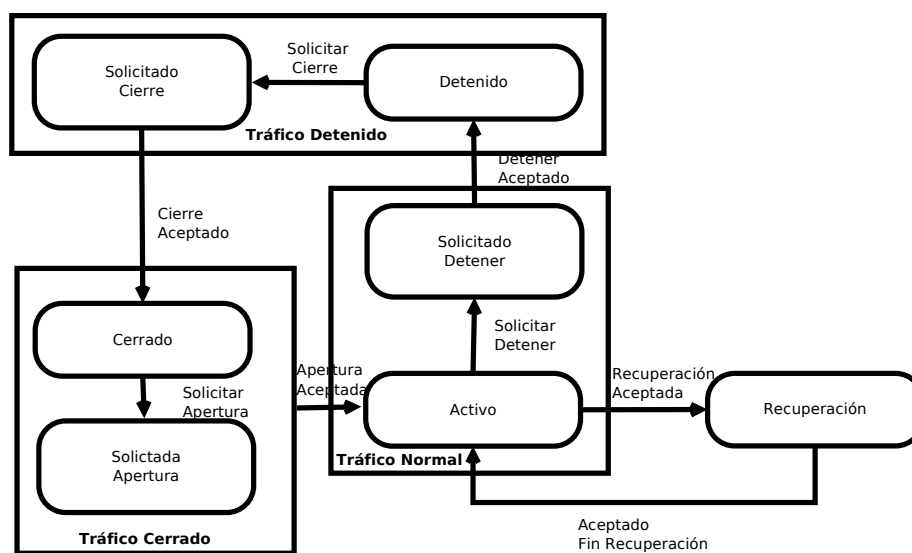


Figura 14.4: Estados FAP en la Entidad Financiera

## 14.4. Operaciones

El FAP está definido para poder soportar múltiples aplicaciones y el diseño realizado para el mismo lo ha tenido en cuenta. Sin embargo, en el ámbito de esta tesis, nos limitamos a la problemática del soporte de la aplicación de Cajeros en la que los formatos de los mensajes están de acuerdo con las normas anteriormente citadas ANSI X9.2-1980 [3] e ISO DIS 7580 [44].

Los mensajes de aplicación establecidos tienen dos objetivos fundamentales:

1. Imputación instantánea de las operaciones -básicamente reintegros, ingresos y traspasos- y operaciones de punto de venta en las cuentas de los clientes.
2. Permitir a los clientes obtener información puntual de sus saldos y movimientos.

El conjunto detallado de operaciones implementadas en esta integración ha sido:

- Cajeros-Telebancos
  - Reintegros.
  - Anulaciones de reintegros.
  - Abonos Euro Assistance.
  - Traspasos entre cuentas asociadas a la tarjeta.
  - Anulaciones de traspasos.
  - Ingresos en efectivo.
  - Anulaciones de ingresos.
  - Consultas de saldos de las cuentas asociadas a una tarjeta.
  - Consultas de movimientos de las cuentas asociadas a una tarjeta.
- POS - Comercios (Terminales Punto de Venta)
  - Ventas.
  - Anulaciones de ventas.
  - Devoluciones.
  - Anulación de devoluciones.
  - Abonos a establecimientos.
  - Consulta de saldo de cuentas asociadas a una tarjeta de establecimiento (tarjeta telecuenta).
  - Consulta de últimos movimientos de cuentas asociadas a una tarjeta telecuenta.

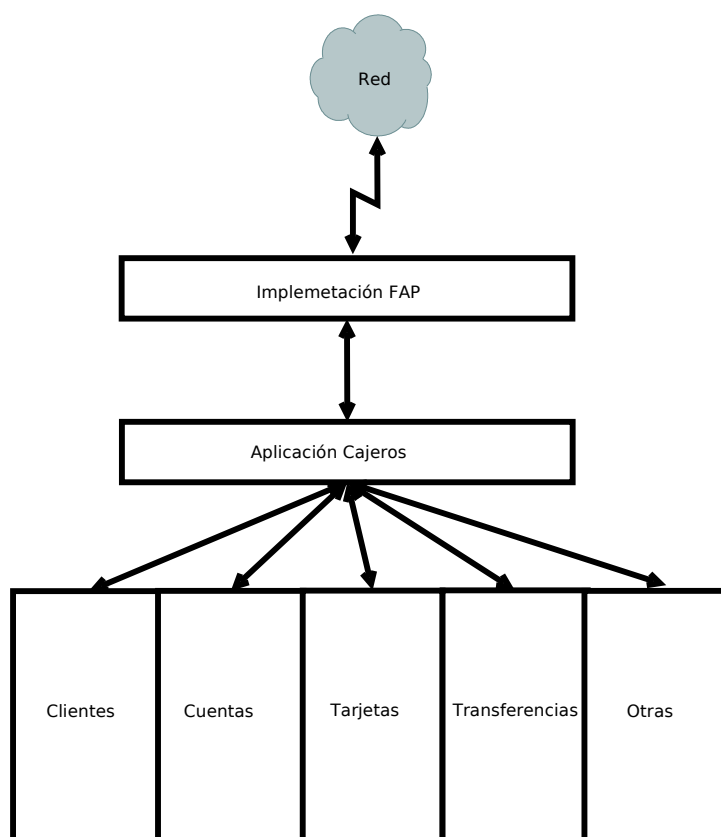


Figura 14.5: Integración funcional

## 14.5. Integración con aplicaciones

Para el soporte interno de la funcionalidad que exigen las operaciones de negocio anteriormente definidas, fue necesaria la integración con las aplicaciones de negocio correspondientes residentes en el Servidor Financiero, aplicaciones que se estaban rediseñando dentro del proyecto de migración explicado anteriormente. En concreto las aplicaciones integradas fueron: Clientes, Tarjetas, Cuentas Corrientes, Cuentas de Ahorro y Cuentas de Crédito, además de aplicaciones de servicio común como Totales, Diarios, Gestión de Logs o Gestión de Errores (fig. 14.5). También fue necesaria la integración "batch" resuelta mediante intercambio de ficheros y la utilización del planificador.

Dado que el planteamiento de la integración con la red de cajeros es fundamentalmente de funcionamiento en línea, se resolvió principalmente dentro del entorno del servidor transaccional que estaba soportado por CICS.

Desde el punto de vista de la aplicación de Cajeros la integración funcional se produjo de la siguiente forma:

1. Todos los mensajes hacia el servidor del banco incorporan la identificación de la tarjeta que está operando.
2. La aplicación servidora valida la tarjeta, comprueba que no tiene limitaciones para operar (ej. nivel de riesgo, robo, pérdida, etc.) y que tampoco existen circunstancias del cliente que no permiten la operativa (ej. alarmas).
3. Dado que una tarjeta puede tener asociadas hasta tres cuentas, una cuenta corriente, una cuenta de ahorro y una cuenta de crédito, se identifica la cuenta o cuentas afectadas para poder invocar el servicio correspondiente de la aplicación concreta. Es de notar que las tarjetas no tienen información sobre las cuentas salvo su existencia, la tarjeta conoce si tiene cuenta corriente, si tiene cuenta de ahorro o si tiene cuenta de crédito. Es la aplicación de Cajeros quien conoce las cuentas reales, permitiendo de esa forma que puedan ser cambiadas sin que las tarjetas físicas se vean afectadas. El usuario es quien determina en el cajero sobre cual de las cuentas asociadas a la tarjeta va a operar; en el mensaje que envía el cajero al consorcio se identifica con que tipo de cuenta desea operar.
4. Finalmente se invoca el servicio correspondiente de la aplicación final, en función del tipo de operación solicitado: reintegro, traspaso, etc.

Es responsabilidad de cada aplicación habilitar los servicios transaccionales necesarios para soportar la operativa que en muchos casos consiste en la reutilización de servicios funcionales de bajo nivel compartidos entre el canal oficina y el canal cajero.

La forma detallada en la que se realizó la integración desde el punto de vista técnico se explica en el capítulo 15 dedicado al diseño y constituye el núcleo de esta parte de la tesis.



## Capítulo 15

# El diseño planteado

Una vez detallado el entorno de aplicación y el FAP, se explicarán las opciones de diseño elegidas y el impacto que las mismas han tenido en la robustez del sistema y en su durabilidad, que le han permitido soportar cambios tecnológicos importantes en la plataforma -sistema operativo, arquitectura de comunicaciones, gestor de base de datos, monitor transaccional- y la evolución de las aplicaciones con las que se integra.

La aplicación se configuró en base a los siguientes subsistemas :

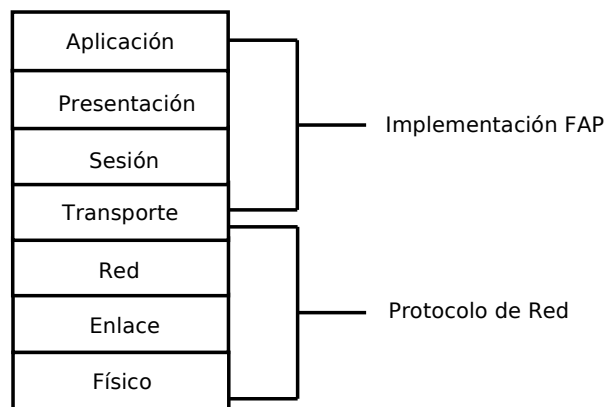
1. Subsistema de Entrada de Mensajes.
  - Control del FAP
  - Transformación de formato
2. Integración Servidor Financiero.
3. Subsistema de Emisión de Mensajes.
4. Terminal de Control y Consola de Operación.
5. Auditoría.
6. Integración con la Planificación de Procesos.

## 15.1. Subsistema de Entrada de Mensajes

### 15.1.1. Control del FAP

Recibe mensajes del consorcio, tanto de control como de datos, así como del terminal de control. Su responsabilidad está en el tratamiento de la cabecera del FAP y el control del estado de la sesión.

Se optó por estructurarlo de acuerdo con los niveles superiores de la pila de comunicaciones OSI [45], en concreto los niveles de transporte y sesión son los que aplican a este subsistema.

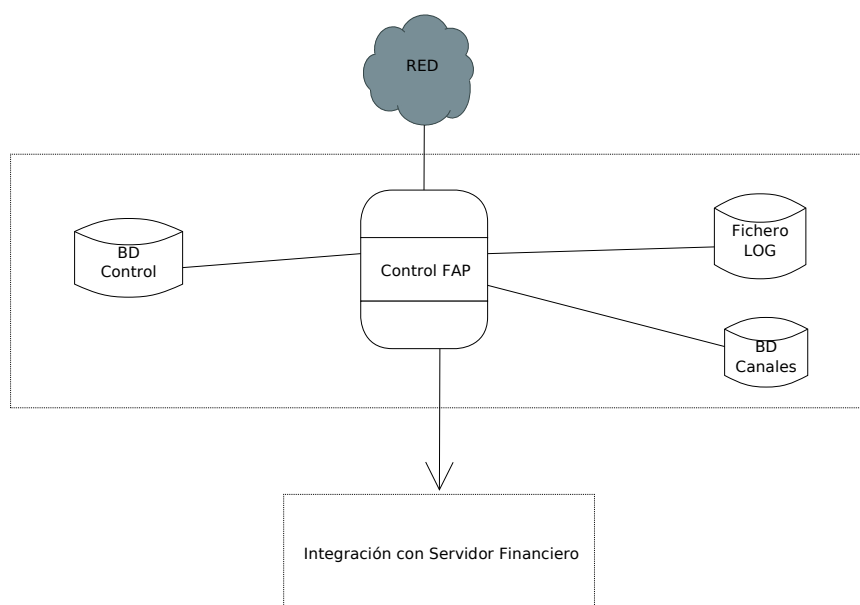


**Figura 15.1:** Pila de comunicaciones OSI

Desde el punto de vista del transporte el subsistema de entrada es responsable de controlar que la secuenciación de los mensajes sea correcta. En cuanto a la capa de sesión, en el sentido que le da OSI y no en el del concepto de sesión del propio FAP, se responsabiliza del tratamiento de la cabecera del mensaje, validando que sea conforme con la sesión del FAP establecida. En este nivel se tratan cuestiones tales como los parámetros de la sesión: identificación del consorcio, identificación del banco y número de sesión establecida, y fundamentalmente el tipo de mensaje recibido: en cuanto a si es de control o de datos y a su validez en el estado actual del diálogo.

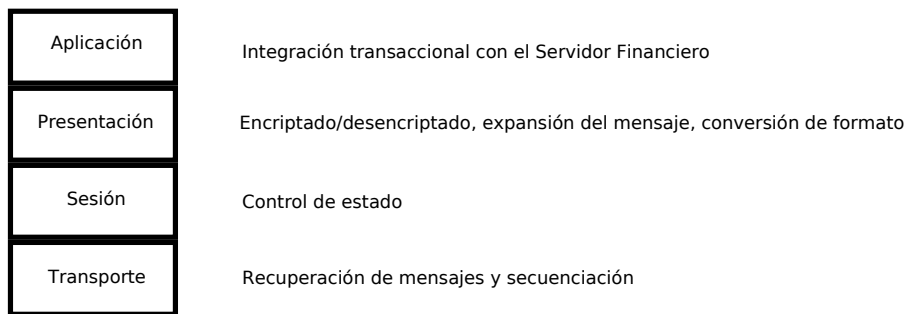
Si el mensaje no fuese correcto, de acuerdo con los criterios anteriores, se rechaza con el código de error correspondiente cediéndolo al subsistema de emisión de mensajes para su devolución.

Si se tratase de un mensaje de control válido, lo que se especifica por el bit correspondiente en la cabecera del FAP, el canal lógico será cero y el número de secuencia del mensaje no será necesario validarlo por el receptor, ya que se trata de un identificador del propio mensaje generado por el emisor sin tener que respetar secuencia alguna. Este número se replica en los mensajes de respuesta a efectos de poder ignorar respuestas atrasadas si se cruzasen peticiones. Los mensajes de control, en el caso de ser aceptados, provocan el cambio de estado correspondiente, tanto en memoria como en el sistema de persistencia, a efectos de recuperabilidad, y se registran en el "log".



**Figura 15.2:** Subsistema de entrada de mensajes

Si el mensaje es de datos y es correcto, se almacena en la base de datos, y se pasa a la capa de presentación responsable de validar el resto del mensaje convirtiéndolo previamente a un formato válido para el entorno de programación utilizado. Cabe recordar que los tipos de datos definidos para los mensajes no tienen un tratamiento directo en el lenguaje de programación Cobol que se usaba en la instalación.



**Figura 15.3:** Proceso cabecera FAP y OSI

### 15.1.2. Transformación de formato

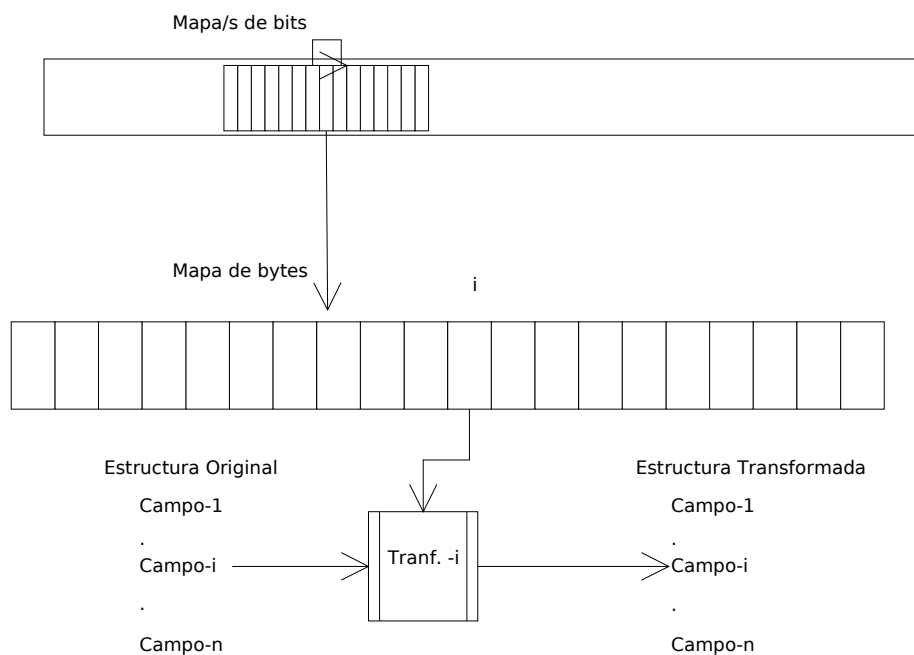
Se responsabiliza de la presentación en el sentido que le da OSI.

Una vez validado el mensaje se transforma mediante un procedimiento automático con los siguientes pasos:

1. Conversión del mapa de bits a mapa de bytes válido para el entorno de programación objetivo.
2. Expansión automática del mensaje y transformación al formato adecuado para invocar el servicio de negocio.

Para la realización de esta transformación y la correspondencia final entre formatos se diseñó un concepto diccionario en la propia aplicación. En el diccionario se define para cada bit posible del mapa las características del campo correspondiente. Este diccionario puede considerarse como una extensión del diccionario corporativo ya que permite generar automáticamente las estructuras de datos, en este caso en formato Cobol, necesarias para poder aplantillar cada tipo de mensaje. El diseño admite extensiones de modo que si el sistema destino manejase formatos diferentes, sería cuestión de invocar los módulos de conversión necesarios para el mismo y generar desde el diccionario las estructuras correspondientes donde se aplantillarían los datos del mensaje original transformados.

El algoritmo de transformación recorre el mapa de bits y en función del diccionario interno invoca las rutinas de bajo nivel escritas en ensamblador [70][37] para transformar el formato de los campos y obtener las estructuras Cobol correspondientes (fig. 15.4).



**Figura 15.4:** Mapas, estructuras y transformaciones

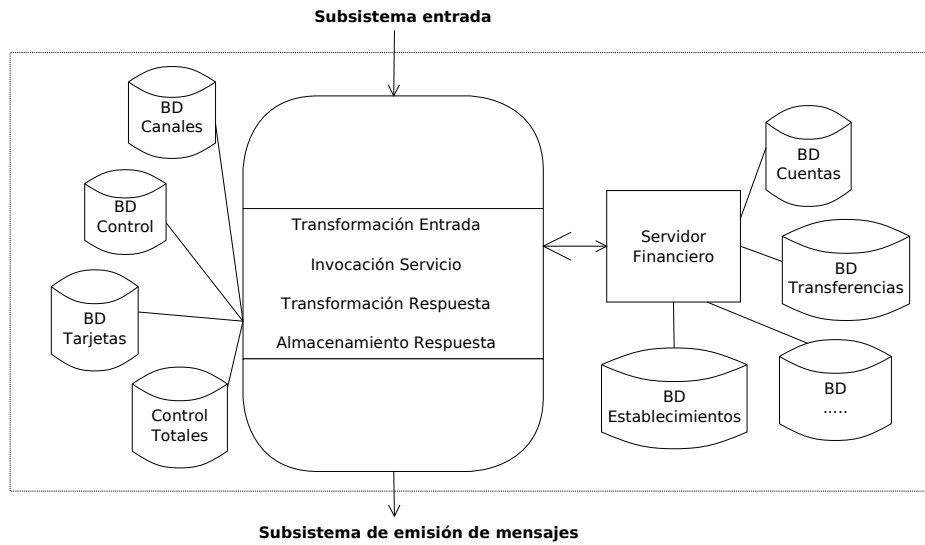
El mecanismo de expansión de mensajes establecido flexibiliza en gran medida la implementación realizada del FAP, admitiendo con gran facilidad la definición de nuevos campos en los mensajes, o nuevos mensajes, que pudiese necesitar esta u otra aplicación que utilizase el mismo FAP.

Todo el proceso realizado en el subsistema de entrada de mensajes se efectúa en el ámbito de una única transacción, garantizando de esta forma la consistencia entre el estado de la sesión y el almacenamiento del mensaje recibido en la base de datos de canales. Este diseño podríamos decir que sigue el patrón "singleton" garantizando que existe un único mensaje en proceso en esta fase. El proceso es suficientemente rápido, por actuar fundamentalmente en memoria, como para que esto no suponga un cuello de botella en la entrada de mensajes.

La transacción termina invocando de forma independiente una nueva transacción, ésta ya de integración en el ámbito de negocio que consideramos dentro del subsistema de integración con el Servidor Financiero.

## 15.2. Integración con el Servidor Financiero

Es responsable de la integración con las aplicaciones que constituyen el Servidor Financiero del banco, donde residen los servicios de negocio a invocar.



**Figura 15.5:** Subsistema de integración

Su funcionalidad consiste en:

1. Transformar el mensaje expandido al formato que requiere el servicio a invocar.
2. Invocar los servicios de las aplicaciones de negocio correspondientes.
3. Recoger y analizar la respuesta.
4. Invocar el módulo de transformación que de acuerdo con el tipo de mensaje genera el mensaje de salida hacia el consorcio, incorporando el mapa o mapas de bits necesarios y los campos en el formato que establece el FAP. Es el proceso inverso al explicado en el punto segundo del subsistema de entrada.
5. Almacenar la respuesta generada en la base de datos de canales para garantizar la recuperación de mensajes y la idempotencia de los mismos.
6. Invocar al subsistema de emisión de mensajes.

El subsistema de integración trabaja también en modo transaccional garantizando la atomicidad entre las actualizaciones que pudieran haber realizado los módulos de negocio invocados y el almacenamiento de la respuesta generada. Este mecanismo es el que garantiza en recuperación la idempotencia de los mensajes pues al tener almacenada la respuesta simplemente se reenviaría ésta en caso de repetición de una pregunta por un canal en estado de recuperación.

### **15.3. Subsistema de Emisión de Mensajes**

Su misión es el envío de mensajes al consorcio. Recibe el mensaje a enviar de cualquiera de los otros subsistemas y lo envía al consorcio.

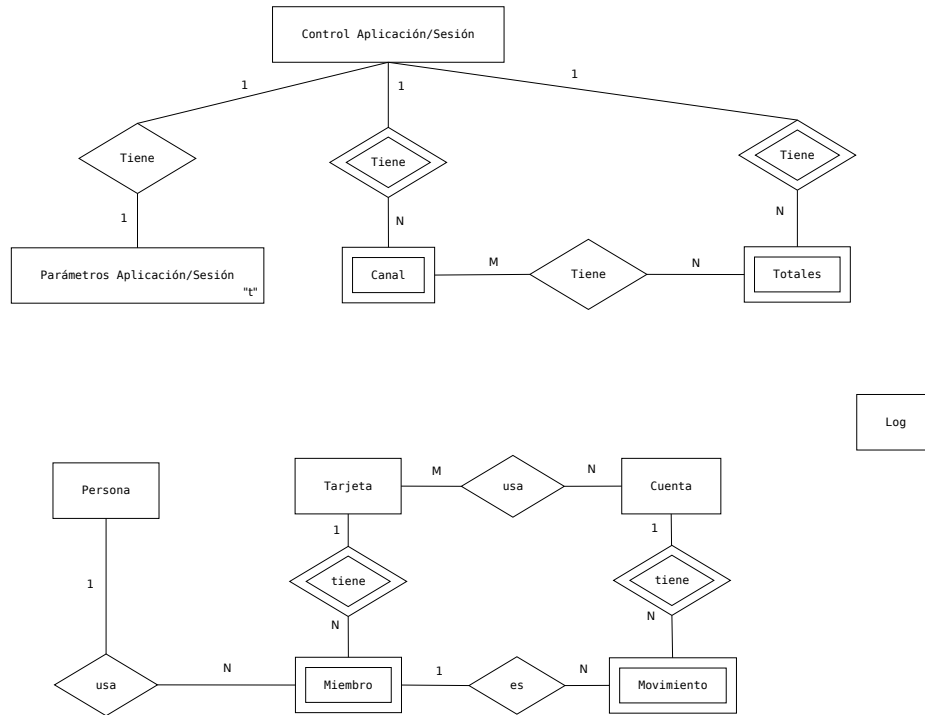
Se diseñó como un elemento independiente para permitir la conexión con cualquier aplicación, independientemente del canal de comunicación o del protocolo utilizado.

### **15.4. Bases de datos**

Para el correcto funcionamiento de los subsistemas descritos hasta ahora fue necesario el diseño de unas bases de datos y ficheros que garantizaran la recuperabilidad, así como la facilidad de mantenimiento y de administración. Las bases de datos necesarias para este diseño fueron:

1. Parámetros del FAP.
2. Control de Aplicación/Sesión.
3. Control de Totales.
4. Control de Canales Lógicos.
5. Fichero de Registro de Mensajes (Log).

La decisión de utilizar tecnología de base de datos se tomó en base a los requisitos de integración transaccional y capacidad de recuperación ante fallos que éstas ofrecían. La tecnología de base de datos utilizada inicialmente fue de tipo jerárquico, utilizando el producto IMS/VS DB, también conocido como DL/I, de IBM, que era la utilizada en la instalación en el momento en que se acometió el proyecto. Con posterioridad se migró a tecnología relacional, utilizando el gestor de base de datos DB2 de IBM.



**Figura 15.6:** Modelo conceptual global

El esquema conceptual global se presenta en la figura 15.6, detallando a continuación las que afectan al control del FAP y a la recuperación.

#### 15.4.1. Parámetros Aplicación/Sesión

Almacena la información necesaria para poder establecer una aplicación/sesión. Identifica el origen y destino necesarios para la cabecera del FAP así como el identificador de la aplicación, tipo de participación en el diálogo, número de canales a utilizar, opciones de "log", parámetros de conexión con el subsistema de salida de mensajes, identificación de terminales a efectos de mensajes de error e impresión y parámetros de tiempos de espera para intentar re-arranques automáticos. Es de tipo temporal pues a pesar de su naturaleza estática, que no permite modificar los parámetros de una sesión establecida, sí permite establecer parámetros a una fecha futura.



### 15.4.2. Control de Aplicación/Sesión

Se establece por aplicación/sesión. Permite identificar si una sesión es de producción o de pruebas y almacena dinámicamente los datos de estado así como marcas de tiempo de los últimos mensajes de control tratados en emisión o en recepción.

### 15.4.3. Control de Totales

Es una base de datos que almacena los totales por tipo de operación para intercambiar en el mensaje de cierre. No todas las aplicaciones la necesitan. Tiene dos niveles, el primer nivel recoge los acumulados globales mientras que el segundo totaliza a nivel de canal. Se establece por aplicación/sesión.

### 15.4.4. Control de Canales

Por cada aplicación/sesión y canal, mantiene la información necesaria para controlar la secuencia correcta de los mensajes, la ocupación o no del canal y la información del último mensaje de petición aceptado en el mismo y la respuesta en el caso de que ya se hubiese producido. Esta respuesta es la que se utiliza en recuperación en el caso de recibir el mismo mensaje de entrada por el mismo canal.

### 15.4.5. Registro de mensajes ("log")

Registra todos los mensajes recibidos o emitidos, los de error siempre y los de datos dependiendo del nivel de grabación especificado en la base de datos de parámetros. Físicamente se implementó con un fichero VSAM/RRDS<sup>1</sup>, fichero de tipo directo, preformateado, con funcionamiento circular, en el que el primer registro apunta al primer registro libre. Este fichero está también controlado por la base de datos de parámetros en cuanto a su tamaño y frecuencia de descarga. El fichero está replicado de forma que al llenarse conmuta a un segundo fichero, descargando el primero. En el momento en el que éste segundo se llena pasa a reutilizarse el primero. Después de un cierre, antes de comenzar una sesión, los ficheros de "log" están vacíos.

El automatismo en el funcionamiento del mismo libera al operador de intervenciones manuales. El hecho de que se trate de un fichero de acceso directo habilita su consulta en línea desde el terminal de control.

---

<sup>1</sup>VSAM/RRDS: Virtual Storage Relative Record Dataset.

## 15.5. Terminal de Control y Consola de Operación

La tecnología utilizada en estos desarrollos fue de terminales no inteligentes que lanzan transacciones CICS, utilizando el protocolo 3270 y diseño de la interfaz con BMS<sup>2</sup>.

Se diseñaron las interfaces necesarias para :

1. Mantenimiento de los parámetros del FAP
2. Consola de Operación y Monitorización

La importancia de este subsistema está directamente relacionada con la criticidad que tiene una aplicación de servicio de cajeros. Cada aplicación que utilice esta implementación del protocolo puede parametrizarse desde aquí, así como ver su estado y último mensaje de control aceptado o emitido. El Terminal de Control permite acceder al fichero de log y navegar por el mismo.

La consola de operación ofrece la posibilidad de emitir manualmente aquellos comandos que estén permitidos en el estado que tenga la sesión de una aplicación.

Las acciones posibles son:

1. Solicitar Inicio de Sesión.
2. Solicitar Detener Tráfico.
3. Solicitar Reanudar Tráfico.
4. Solicitar Cierre de Sesión.
5. Cambiar a modo off-line.
6. Comprobar Actividad.

## 15.6. Integración con la Planificación de Procesos

La planificación de procesos es un elemento de vital importancia en la instalación. La importancia y criticidad de los procesos "batch" es tanta como la del propio proceso en línea. Todos los procesos de contabilización, obtención de presupuestos, control de gestión, liquidaciones, campañas comerciales o comunicaciones a clientes son fundamentalmente "batch".

---

<sup>2</sup>BMS: Basic Mapping Support

El sistema que nos ocupa también tiene su parte "batch", quizá no tan importante como otros, y por lo tanto la necesidad de integrarse con la planificación global. Desde el punto de vista de este sistema la integración se produce fundamentalmente en dos situaciones :

1. Arranque del teleproceso.
2. Cierre del teleproceso tras un cierre de sesión.

El arranque del teleproceso puede ocurrir después de haber pasado por un cierre ordenado o como consecuencia de una caída. En cualquiera de los dos casos el primer elemento a considerar es la base de datos de Control de Sesión donde, a través de los estados definidos, se conoce la situación de cada aplicación.

En un ciclo de ejecución normal, el arranque del teleproceso habrá estado precedido por un cierre ordenado tras el que el planificador habrá lanzado los procedimientos de extracción de totales y borrado de los mismos para habilitar la apertura de una nueva sesión. El propio arranque del teleproceso detectaría esta situación, establecería el área de control en memoria de la aplicación y subiría a la misma la información sobre parámetros de la sesión y estado, existentes en las bases de datos de parámetros y control respectivamente, a continuación trataría de lanzar de modo automático la solicitud de arranque de sesión, que en caso de no conseguirse sería necesario relanzarlo mediante la consola de operación.

Si el arranque del teleproceso se produjese tras una caída del mismo, la situación puede ser diferente ya que en el momento de la caída podría estar activa una sesión. En este caso la actuación respecto a las tablas de memoria sería la misma pero no se lanzaría automáticamente la solicitud de inicio de sesión.



## Capítulo 16

# Arranque y recuperación

La disponibilidad exigida al sistema hizo que cobraran una importancia muy relevante los aspectos relacionados con el arranque y la recuperación. El diseño de esta operativa se basó en varios juegos de estados establecidos en distintos ámbitos y relacionados entre sí.

### 16.1. Juego de estados

Se definieron cuatro tipos de estados:

#### 1. Estado de la comunicación física.

Determina el estado en relación con el protocolo de transporte utilizado. Representa la situación de la comunicación física.

Contempla dos estados :

- Abierta
- Cerrada

La integración con el protocolo de transporte es la que informa de este estado, implementada inicialmente por medio del ISARD. No puede haber físicamente tráfico si se encuentra cerrado.

#### 2. Estado de la aplicación servidora.

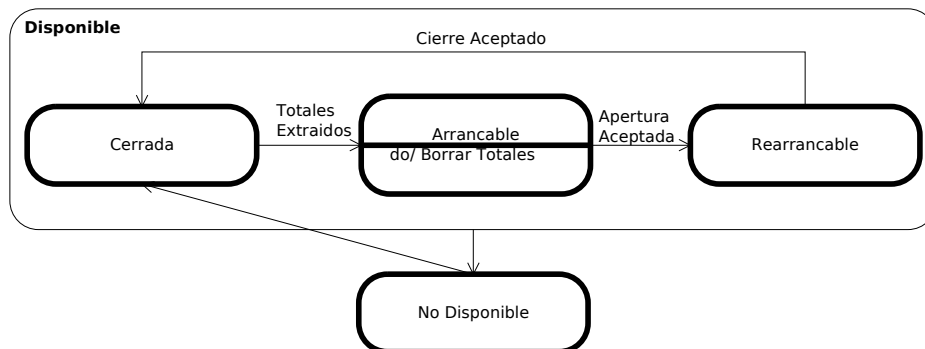
En este caso el estado hace referencia a la disponibilidad o no de la aplicación servidora de la entidad financiera, que es donde en última instancia se realizan las operaciones de negocio solicitadas por medio del FAP.

Puesto que el primer nivel de integración "batch" se establece en base a los

totales registrados durante una sesión, éstos juegan un papel fundamental en la determinación de este estado. Se relaciona por tanto con el siguiente tipo de estado que es el que establece la situación concreta de los mismos. Se ha planteado como un estado independiente para tener la posibilidad de inhibir la aplicación.

Se contemplan los siguientes estados :

- Arrancable: El FAP estará cerrado y los totales a cero.
- Cerrada: El FAP debe estar cerrado, pero los totales no han sido extraídos.
- Rearrancable: El FAP estará en un estado distinto de cerrado. Se alcanza después de haber abierto una sesión con éxito.
- No Disponible: Causas de fuerza mayor que obligan a hacerlo.



**Figura 16.1:** Estados de la aplicación

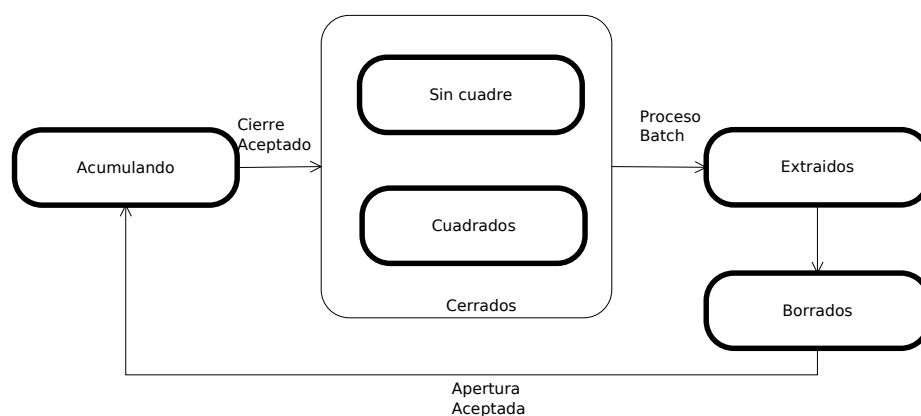
### 3. Estado de totales.

Están relacionados con la integración contable a través del planificador que controla el proceso por lotes o "batch".

Los estados que pueden presentar los totales son :

- a) Acumulando: Es el estado normal durante una sesión, la aplicación estará por tanto en estado de "Rearrancable".
- b) Cerrados: Se alcanza tras haber obtenido un cierre de la sesión.
- c) Cuadrados: Si además de cerrada la sesión ha habido cuadro en el intercambio de datos de cierre.

- d) Extraídos: Han sido ya tratados por el proceso batch contable.
- e) Borrados: En disposición de solicitar un arranque de una nueva sesión.



**Figura 16.2:** Estados de los totales

#### 4. Estados del FAP

Los estados del FAP ya fueron explicados anteriormente (capítulo 14. fig. 14.4. pág. 93).

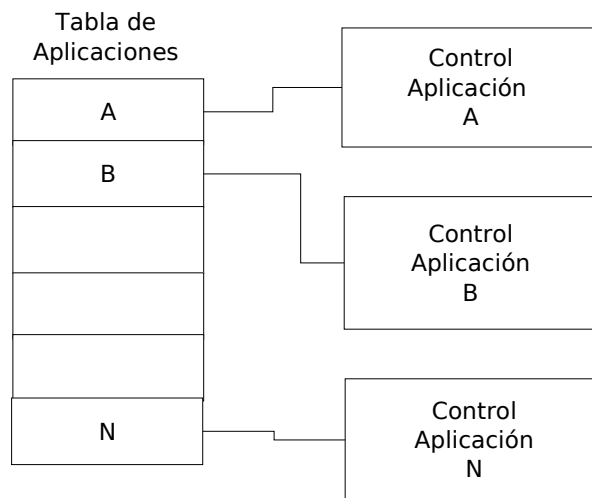
## 16.2. Arranque del sistema

Se describe en primer lugar el arranque inicial. Para poder realizar un inicio de sesión el servidor del consorcio debe de estar en estado "Disponible". Se efectúa a petición del banco, en modo manual, por medio de un comando del operador o lanzado por el planificador.

Es prerequisite que la aplicación esté en estado "Arrancable" y la comunicación "Abierta". Se efectúa a nivel del FAP, el consorcio, normalmente, lo aceptará quedando la aplicación en modo "Rearrancable" y el estado del FAP en situación de "Tráfico Normal".

En el arranque del sistema se inicializa la base de datos de control y la base de datos que da soporte a los canales en la que se recogen todas las peticiones y respuestas que se reciben o envían por cada uno de ellos.

En el momento del arranque del sistema la base de datos de control del FAP se sube a memoria donde es direccionada por los módulos del FAP que la usan. El modo más habitual de uso es simplemente en consulta, fundamentalmente a efectos de control de estado, actualizándose, con replicación en la base de datos real, únicamente cuando se produce un cambio de estado en el FAP. El diseño se realizó de esta forma teniendo en cuenta factores de rendimiento y control de concurrencia. El rendimiento se ve mejorado al no tener que acceder a la base de datos en disco continuamente, sino solamente en aquellas actualizaciones que afectasen a una posible recuperación. El control de concurrencia se hace actuando en combinación con el diseño transaccional que protege la actualización concurrente de la misma.



**Figura 16.3:** Tablas en memoria

### 16.3. Recuperación del sistema

Una vez arrancada la sesión, el control del protocolo de diálogo lo realiza el diseño implementado para el FAP. Si por cualquier motivo fuese necesario arrancar de nuevo la aplicación, por ejemplo ante una caída del ordenador, el estado de la aplicación estaría almacenado en la base de datos de control del FAP, la aplicación estaría en estado "Rearrancable" por lo que no se produciría la inicialización de la



base de datos de control del FAP, ni la de control de los canales. Este arranque implicaría simplemente el volver a establecer el área de memoria correspondiente a esta base de datos de control del FAP, quedando la aplicación en el estado que tuviera antes de la caída. El tráfico se reanuda ya que el consorcio estaría intentando una recuperación por no recibir respuesta durante el tiempo de caída a los mensajes emitidos.

Como mecanismo adicional para garantizar la disponibilidad del sistema, existe una tarea que periódicamente analiza la situación de cada uno de los canales lógicos establecidos, asegurándose de que en al menos en uno de ellos exista tráfico. Actúa en modo de alarma de "time-out", dependiente de un parámetro de la sesión. En el caso de que salte la alarma por "time-out", se informa al operador en la consola de control de la aplicación. La justificación de este mecanismo está en que lo normal es que en cualquier momento del día, indudablemente menos por la noche, debe de existir tráfico desde los cajeros y si no existe el operador del sistema debe ser informado para que pueda comprobar que no existe alguna anomalía de funcionamiento que exija su intervención.



## Capítulo 17

# Operación y monitorización

Uno de los aspectos fundamentales que hay que tener en cuenta cuando se diseña un sistema crítico son las facilidades de operación y monitorización. El sistema que se está describiendo es en efecto un sistema crítico tanto por la sensibilidad de su funcionalidad, movimientos de dinero que realizan los clientes de forma desatendida, como por el volumen de operaciones que gestiona (capítulo 19).

Los operadores del sistema deben poder conocer en todo momento cualquier anomalía de funcionamiento que se produzca y para ello tenían que disponer de las herramientas necesarias para poder monitorizar e intervenir si fuese necesario. La idea básica era doble: conocer el estado de funcionamiento y tener a su disposición todas las posibilidades de operación que el protocolo admitía.

### 17.1. Consola de operación

Se diseñó una consola de operación y monitorización, implementándola mediante servicios transaccionales CICS y pantallas 3270 con BMS. La consola permite que el operador conozca el estado de la sesión de una aplicación, el tráfico que está procesando y puede interactuar con la aplicación, emitiendo comandos de control para ser validados por el controlador del FAP y, en su caso, enviados al consorcio a través del subsistema de envío de mensajes. La funcionalidad soportada contempla:

1. Consulta de aplicaciones habilitadas.
2. Consulta de parámetros y estado.

3. Modificación de parámetros de cada aplicación.
4. Emisión de mensajes de control del FAP.
5. Recepción de respuestas a mensajes de control.
6. Recepción de mensajes informativos de la aplicación.
7. Consulta de tráfico de cada canal establecido.
8. Consulta del "log" de mensajes.

El detalle de los mensajes de control que puede emitir el operador desde esta interfaz son:

1. Solicitud de inicio de sesión.
2. Solicitud de interrupción de tráfico.
3. Solicitud de reanudación de tráfico
4. Solicitud de re-arranque.
5. Solicitud de cierre de sesión.
6. Solicitud de cambio en el modo de transmitir las operaciones realizadas sin conexión.

La importancia de un mecanismo ágil a disposición de los operadores ha sido otro punto considerado como fundamental en el diseño de la solución. La criticidad del sistema, debida al tipo de operativa que soporta, exige que en cualquier momento que sea necesaria una intervención pueda realizarse por el personal encargado. En este caso son los operadores de consola los que pueden actuar en la aplicación, teniendo a su disposición las herramientas necesarias para controlar cada una de las aplicaciones que se hayan establecido usando el FAP.

Hay que recordar que en este caso hablamos únicamente de la aplicación de Cajeros, pero posteriormente, sin necesidad de modificar en punto alguno el diseño, se incorporaron otras aplicaciones como Gestor de Tesorería, Terminal Desplazado o en su momento varias instancias de la aplicación de Cajeros para contemplar sesiones diferentes para euros o pesetas.

## Capítulo 18

# Evolución del sistema

Como todos los sistemas, y de modo especial los de mayor permanencia, el sistema sufrió distintas adaptaciones en el tiempo (fig. 18.1), principalmente por los siguientes motivos:

- Migración a BD Relacionales.

El diseño original de esta aplicación fue realizado para bases de datos jerárquicas, sufriendo por lo tanto dos tipos de migración relacional. La primera migración fue la correspondiente a cada una de las aplicaciones con las que se integra, que se hizo al ritmo que marcaron las propias aplicaciones, fue de poco impacto por la forma de invocar los servicios de las aplicaciones mediante paso de mensajes. La segunda fue la de las bases de datos de la propia aplicación (2003): Parámetros, Control, Canales Lógicos, fue de mayor impacto afectando también a los subsistemas de monitorización y recuperación.

- Rediseño de las aplicaciones con las que se integra.

Fundamentalmente las aplicaciones de Cuentas, Tarjetas, Clientes y Transferencias.

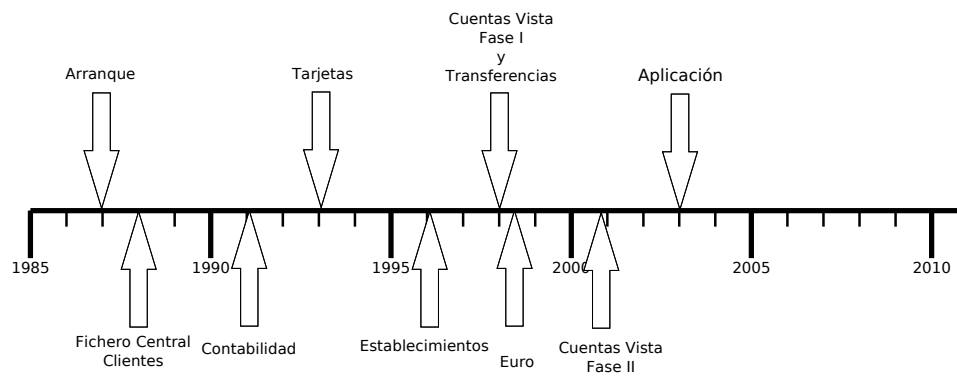
- Cambios transversales.

Los más importantes fueron sin duda el cambio al Euro y el Año 2000. Con el cambio al Euro fue necesario mantener más de una sesión que hasta el

momento no había sido preciso. El segundo de ellos no afectó por contemplarlo el diseño original.

- Cambios de versiones en el software de base.

Sistema operativo, base de datos, gestor transaccional, comunicaciones etc.



**Figura 18.1:** Evolución del sistema

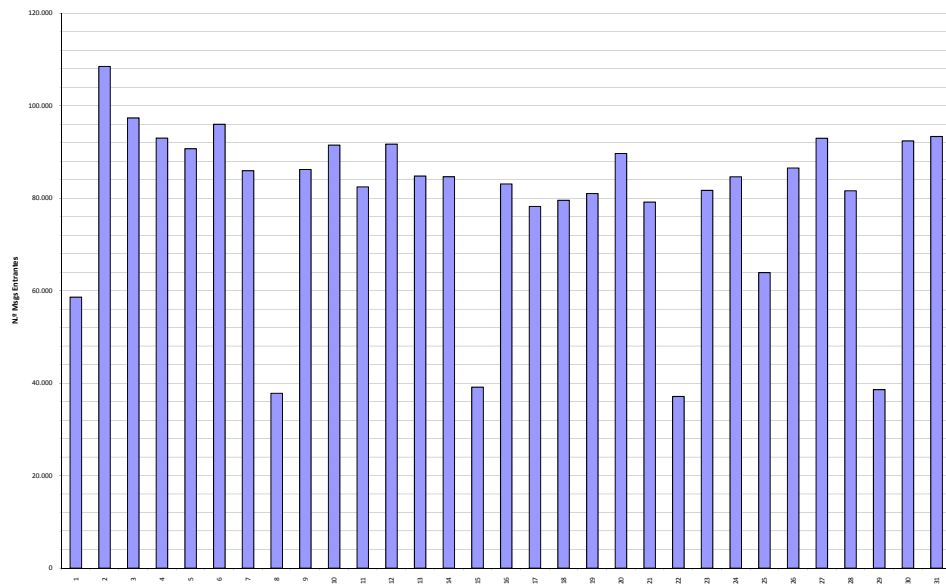
## Capítulo 19

# Algunos datos de volúmenes

La criticidad de la aplicación que se ha descrito puede entenderse mejor a la vista de algunos datos reales. En el año 2012, en el que la crisis ha reducido de forma importante la actividad comercial y con ello el número de operaciones de telebancos y terminales de punto de venta, el sistema trataba del orden de 2,5 millones de mensajes de entrada mensualmente y una cifra equivalente de mensajes de salida.

Si extrapolamos estos datos a los 25 años de vida de la aplicación, admitiendo el error que ello pueda suponer, el sistema habría procesado cerca de 750 millones de mensajes entrantes y generado las respuestas correspondientes.

Los siguientes gráficos representan los mensajes recibidos en un mes en dos perspectivas diferentes. El primero lo hace por días y en él se puede ver la caída correspondiente a los domingos. El segundo los agrupa, independientemente del día, según el cuarto de hora en el que se recibieron, teniendo de esa forma una distribución horaria de la entrada de mensajes. En este segundo gráfico se aprecia la concentración en las horas centrales y a última hora de la tarde.



**Figura 19.1:** Mensajes entrantes por día



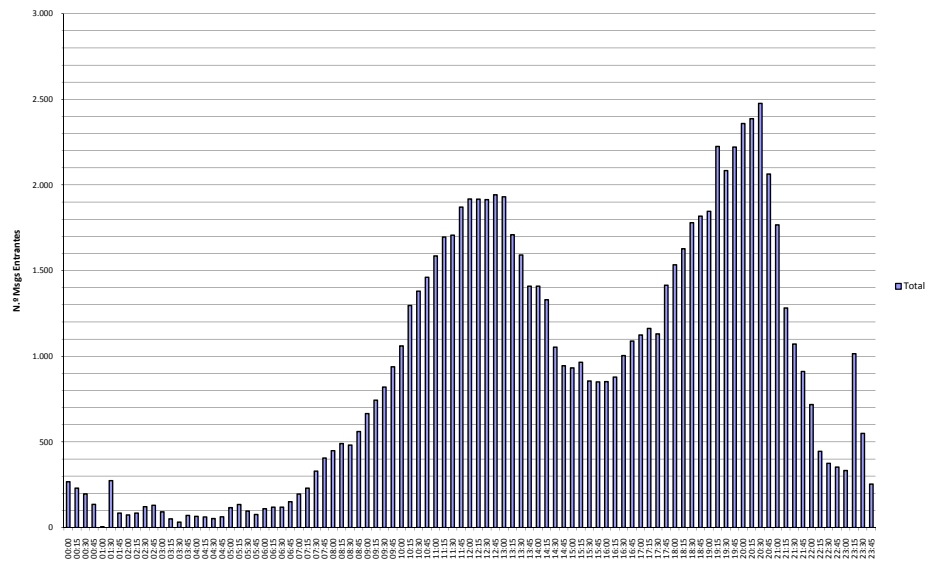


Figura 19.2: Mensajes entrantes cada cuarto de hora



## Capítulo 20

# Lecciones aprendidas

La fecha en la que se realizó este diseño (1986), la capacidad de la plataforma utilizada y el entorno sobre el que se implementó, son condicionantes importantes que podrían haber limitado el alcance de la solución y su persistencia en el tiempo, sobre todo si se piensa desde la perspectiva actual con la potencia de los equipos, las herramientas y los entornos hoy disponibles.

Sin embargo, no restando importancia a esos factores se resaltan en estas conclusiones aquellos aspectos del proyecto que, aunque ya fueron tratados los capítulos anteriores dentro de su contexto, suponen las aportaciones más importantes en esta experiencia concreta. El análisis se hace desde tres puntos de vista: el del diseño, el del tratamiento de los requisitos no funcionales y el de la integración.

### 20.1. Visión del diseño

- Diseño transaccional.

Fue fundamental desde varias perspectivas: coordinación de recursos, control de concurrencia, integración con las aplicaciones de negocio y también para facilitar la evolución en el tiempo de las plataformas de ejecución y del software de bae disponible en cada momento. Uno de los aspectos más significativos fue la coexistencia de bases de datos relacionales y jerárquicas actualizadas en la misma transacción.

- Bases de datos y recuperabilidad.

Las bases de datos, como gestores de recursos coordinados por el gestor transaccional, juegan un papel básico en la capacidad de recuperación y reorganización del sistema, garantizando la persistencia de la situación del diálogo y de la información sobre mensajes existente en cada canal, de forma que se pueda volver a actualizar la información en memoria a la situación que tenía en el momento del fallo.

- Separación del acceso a las bases de datos.

El acceso a las bases de datos fue desde el principio modularizado, aislándolo de la propia lógica de la aplicación. Este tipo de diseño independiza a la propia aplicación del gestor de bases de datos y es lo que ha permitido en el tiempo cambiar radicalmente la tecnología de bases de datos sin impacto significativo en el resto del diseño. Podríamos en cierto modo decir que se ha simulado el comportamiento de un DAO<sup>1</sup>.

- Modularización y estructuración en base a capas.

En este punto cabe señalar el esfuerzo en el diseño de asignar responsabilidades bien definidas a cada módulo, favoreciendo la conexión de los mismos. Es de especial relevancia la separación del tratamiento de los mensajes recibidos del consorcio siguiendo el planteamiento de las capas superiores de OSI, asignando responsabilidad concreta a módulos diferentes en cada nivel.

- Utilización del diccionario de mensajes conforme a los mapas de bits.

El formato complejo de los mensajes a tratar, tanto en cuanto a su número como a su composición: longitud variable, composición dinámica en base a estructura de mapa de bits, y el anticipar su evolución en cuanto a otras aplicaciones que utilizaran este mismo protocolo de intercambio, o la aparición de nuevos campos en mensajes de las aplicaciones existentes, exigió intentar flexibilizar al máximo esta parte del diseño.

El diccionario de mensajes resultó crucial ya que permite la definición externa de los mapas de bits y las características de cada campo. Desde este diccionario se generan las estructuras de datos necesarias para tratar los mensajes de entrada y salida, así como las estructuras necesarias para el tratamiento por

---

<sup>1</sup>DAO: Data Access Object.

el entorno de programación, jugando también esta parte un papel típico de los entornos de integración en su faceta de transformación y correspondencia de estructuras.

- Utilización del diccionario corporativo.

El diccionario corporativo, que como se ha indicado no fue parte de este diseño aunque sí coetáneo, fue utilizado como repositorio para la definición de todos los datos y de las estructuras de datos que maneja la aplicación. Fundamentalmente estructuras de base de datos y de ficheros, cuyos esquemas o formatos de registros en los casos de los ficheros se generan automáticamente en modo pasivo desde esta información. El diccionario refuerza el concepto de Administración de Datos que exige un comportamiento riguroso en cuanto al nombrado y definición de cualquier tipo de dato.

- Control de concurrencia a nivel de protocolo.

El planteamiento de única transacción en la integración con el protocolo de transporte, junto con la base de datos de control del diálogo en memoria y el acceso único a la misma, garantizan el tratamiento correcto de la concurrencia de las peticiones. El diseño permite que un único mensaje puede estar en fase de tratamiento de la cabecera y que tantos como canales lógicos disponibles pueden estar siendo procesados por la aplicación y el servidor financiero con el que se integra.

## 20.2. Vision de los requisitos no funcionales

Resulta interesante la visión del diseño desde una perspectiva FURPS<sup>2</sup> [33], pudiendo señalar cómo se tuvieron presentes, además de los requisitos funcionales intrínsecos al problema, condicionantes de otro tipo cuya importancia ha sido constatada en el tiempo:

- Usabilidad.

Este punto es necesario entenderlo dentro de las limitaciones que supone la tecnología 3270. Es crucial fundamentalmente en la consola de operación, donde se establecieron principios de navegabilidad que permiten ir profundizando hasta llegar al detalle del mensaje original.

- Fiabilidad.

La criticidad del sistema estuvo siempre en el foco. A diferencia con el canal oficina, único en aquél momento, el canal de telebanco o cajero supone la interacción directa del cliente final con el Servidor Financiero, mientras que en las oficinas es siempre un empleado de la misma quien interacciona con el sistema. Además el sistema trata operaciones que suponen flujo de dinero, la mayor parte de las veces físico, al que los clientes son muy sensibles.

- Rendimiento.

Es un condicionante aplicable a cualquier sistema, en este caso muy relacionado con el anterior.

- Adaptabilidad y soporte

La evolución soportada y el poco esfuerzo que fue necesario para realizarla validan el diseño desde la óptica de este requisito.

---

<sup>2</sup>FURPS: Funcionality, Usability, Reliability, Performance, Supportability.

- Implementación.

El diseño se ajustó a las capacidades de la plataforma, tratando de hacer un aprovechamiento óptimo de la tecnología disponible pero con un planteamiento de arquitectura que consideramos fue muy importante para facilitar su evolución. Es una cuestión importante y que tiene que ver tanto con la conceptualización de los problemas como con la importancia de aislar los aspectos técnicos propios de la tecnología utilizada.

- Operación.

Integración con la planificación por lotes corporativa y con la consola de operación.

### 20.3. Visión de la integración empresarial

El diseño realizado también puede analizarse desde una perspectiva de integración. El problema planteado se puede encuadrar en este tipo de soluciones si se tienen en cuenta sus características de sincronización de datos, intercambio externo a la propia empresa e integración de aplicaciones con formatos de datos y protocolos diferentes.

Este tipo de problemas generan habitualmente dificultades especialmente en lo que se refiere a la integración con sistemas heredados cuando existe la exigencia de mantener las interfaces con los mismos. Analizado desde esta óptica el enunciado puede entenderse como la necesidad de abrir su funcionalidad al exterior sin que las aplicaciones se vean modificadas. El problema tiene por tanto características tanto de una integración interna de aplicaciones (EAI) como de una integración con el exterior de la empresa (B2B)[66].

Así dentro del diseño realizado se pueden observar características -soluciones de arquitectura en este caso- que se pueden encuadrar dentro de los patrones que hoy se consideran relevantes cuando se habla de integración [36][73][53], como:

- Construcción de mensajes.

Se hizo con separación clara entre cabeceras y cuerpo, extrayendo y transformando el cuerpo para establecer la integración con el Servidor Financiero y responsabilizando a cada subsistema de tratamiento de una parte del mensaje.

- Canales de mensajes.

La base de datos de control del FAP, al registrar los mensajes recibidos y los mensajes a enviar, actúa como una cola de mensajes que permite la comunicación asíncrona, independizando los extremos. Se apoya en la propia especificación del FAP y en los procedimientos de recuperación que este define.

- Enrutamiento de mensajes.

Se hace en función del tipo de mensaje, invocando los servicios de negocio relevantes (ej. reintegro, transferencia, etc.) o el módulo de tratamiento interno correspondiente si se tratase de un mensaje de control.

- Transformación de mensajes.

La diferencia entre los tipos de datos que define el FAP y los tipos de datos del entorno de programación, hicieron imprescindible esta transformación que solventó con apoyo de un diccionario propio definido para el FAP. Existe además una segunda transformación que es la necesaria para construir el mensaje que permite invocar el servicio de negocio correspondiente.

- Gestión y monitorización.

En el capítulo 17 se explicaron tanto la consola de operación como las funcionalidades de monitorización. También se cuidaron aspectos como la detección de falta de operación, cuellos de botella, caídas etc., arbitrando mecanismo de información al operador en el caso de detectarse.



## Parte III

# La Administración de Datos



## Capítulo 21

# Introducción

En esta tercera parte de la tesis se describe una función, la de Administración de Datos, a la que generalmente no se le presta en las organizaciones la atención y dedicación que merece. La función de Administración de Datos es mucho más que simplemente una función, con más rigor se puede decir que es una filosofía de trabajo, horizontal en todo el desarrollo informático, y sobre todo la pieza clave para gestionar los datos como un recurso y como un factor de producción, tal como se planteó en la introducción de esta tesis.

El motivo de la elección de esta función como una parte de la tesis es que constituye un elemento diferencial de la historia que se ha descrito y que ha sido uno de los elementos más importantes para haber podido llevar a cabo un desarrollo de sistemas integrado y con la capacidad de evolución que se le ha exigido.

Se describe también en esta parte el Diccionario de Datos que ha sido la plataforma técnica utilizada para soportar la función de Administración de Datos.



## Capítulo 22

# Administración de Datos

La magnitud del proyecto de migración planteado en el año 1985 (capítulo 4) aconsejó la utilización de un Diccionario de Datos. El planteamiento fue más allá y se implantó una filosofía y una función, la de Administración de Datos [26] [20]. También en Martin [54, cap. 21] se describen las funciones de los Administradores de Datos y la de los Administradores de Base de Datos, presentando los papeles diferenciados de figuras como el analista, el diseñador de base de datos, el supervisor de base de datos y el oficial de seguridad. En Mairet [52] se centra más en la función específica del Administrador de Datos que es la que aquí se trata.

### 22.1. Principios de la Administración de Datos

Los principios básicos perseguidos en la implantación de una función de Administración de Datos fueron:

- Garantizar definiciones y formatos únicos para los datos, potenciando su uso compartido e integración.
- Estandarizar la nomenclatura de los datos.
- Garantizar el acceso global y unificado a los datos.
- Incrementar el valor de la información.
- Mejorar los procesos de desarrollo en base a la reutilización.
- Mejorar la calidad de la información.
- Posibilitar el análisis del uso y el impacto sobre los datos.

- Establecer políticas y procedimientos de seguridad sobre los datos.
- Evitar redundancias.
- Mejorar la documentación global de los sistemas.
- Potenciar el uso de los Modelos Conceptuales de Datos.
- Integrar la Administración de Base de Datos en este contexto, fundamentalmente en lo que concierne a los Modelos Lógicos de Datos.

## 22.2. El valor de la información

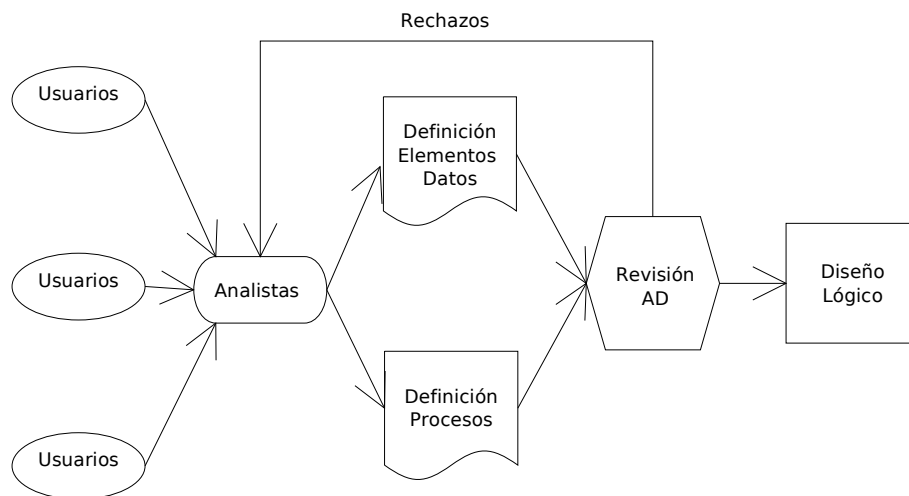
La existencia de una función de Administración de Datos es uno de los factores que potencian los planteamientos IRM<sup>1</sup>, cuyo objetivo es la gestión de la información como un activo. Los principios fundamentales del IRM se resumen en las siete leyes que siguen [56]:

1. La información es de uso compartido.
2. El valor de la información aumenta con el uso. Este principio se basa en que el mayor coste de la información está en los procedimientos de captura, almacenamiento y mantenimiento, mientras que el uso incrementa relativamente poco el coste.
3. El valor de la información decrece con el tiempo. Se pueden distinguir tres estados de la información, el primero sería el operacional, el segundo sería el informacional y el tercero sería el legal.
4. El valor de la información aumenta con la precisión. Existe además un punto de inexactitud por debajo del cual la información carece de valor.
5. El valor de la información aumenta al integrarla.
6. Más no es siempre mejor.
7. La información es autogenerativa. El análisis y la sumarización produce nueva información valiosa; no sucede lo mismo con otros activos como puede ser el propio dinero.

---

<sup>1</sup>Information Resource Management.

La función de Administración de Datos se sitúa de forma central en los procesos de captura de requisitos, análisis y también en el diseño de las bases de datos y de otras estructuras de información como pueden ser registros de ficheros, mensajes etc. La Administración de Datos estrecha la colaboración entre los analistas y los responsables de los proyectos, estableciendo un punto de control en sus definiciones y modelos para adecuarlos a los estándares y usos de la instalación, reforzando así también la utilización de modelos conceptuales de datos que en este caso estaban basados en una variante del modelo entidad relación propuesto por Chen [13] denominada ELKA [61].

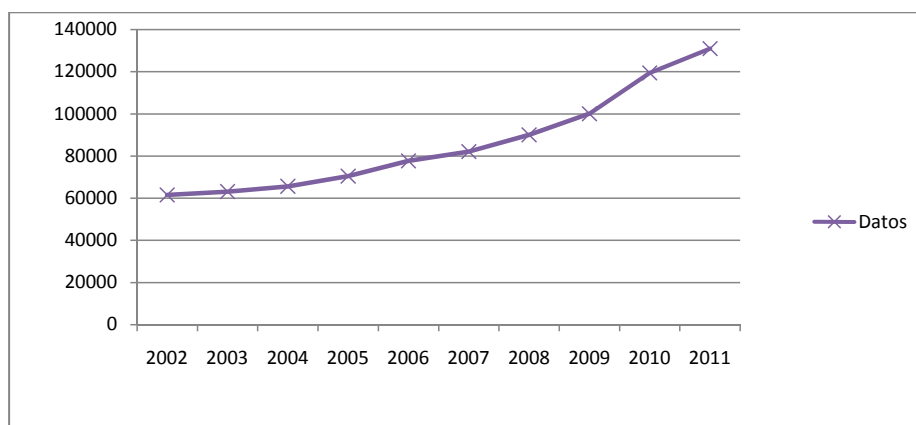


**Figura 22.1:** Administración de datos y diseño de base de datos

### 22.3. Resultados obtenidos

El foco mantenido en la Administración de Datos hizo posible que se pudiera gestionar de forma adecuada la cantidad tan importante de datos que se manejan en una instalación de una Entidad Financiera, que consistían en más de 100.000 definiciones de datos, permitiendo una gran eficiencia ante retos tan importantes como los que en su momento supusieron para una instalación de este tipo el cambio del Año 2000 o la incorporación al Euro.

El volumen de elementos de datos definidos es importante como se puede apreciar en el gráfico. Con cifras de 2012 observamos que hay inventariados más de 130.000 elementos de datos y de ellos más de 56.000 estaban definidos en el año 1999. El efecto del Año 2000 exigió localizar el uso de más de 2500 grupos de datos con formato de tipo DD-MM-AA y casi 200 datos de fechas definidos como elemento de datos único en formatos DDMMAA o AAMMDD. Los datos correspondientes a la introducción del euro, con sus implicaciones de moneda y decimales, son algo mayores pero del mismo orden de magnitud.



**Figura 22.2:** Evolución del volumen de datos

El uso de un diccionario constituye en sí mismo una lección aprendida por la importancia que ha tenido en el tiempo su utilización. En el año 2012 sigue vigente, después de transcurrir más de 25 años, habiéndose posicionado como el principal elemento de documentación y comunicación entre los equipos de los distintos proyectos.



Otro aspecto fundamental de la Administración de Datos lo constituyó el establecimiento de una norma de estandarización de los nombres de los datos. En este caso se estableció una estructura para los mismos en base a tres tipos de componentes: una palabra de clase o tipo de dato (ej. importe, cantidad, texto, fecha, código,..), hasta tres calificadores (ej. primera-adquisición) y por último la entidad de pertenencia del dato, así el nombre completo de un dato podría ser "importe-primer-a-adquisición-inmueble".

En el ejemplo anterior "importe" clasifica el dato dentro de una de las categorías permitidas, "inmueble" es la entidad a la que pertenece y "primera" y "adquisición" son calificadores. Es de notar que también se han eliminado preposiciones, artículos, contracciones y en general todas aquellas palabras propias del lenguaje natural que no aportan significado adicional.

Además del nombre estandarizado de cada dato en el diccionario se almacenan otras características relevantes como quién lo creó y cuándo fue creado, descripción detallada, notas de uso y validación, formato y nombres para su uso en plataformas que limiten el tamaño de los mismos como fue el caso del IMS que limitaba los nombres a ocho caracteres.

La utilización de una normativa para nombrar y definir los datos hace que la probabilidad de que dos personas diferentes nombren de la misma forma un dato se incremente de forma muy importante, estableciendo de esta forma un vocabulario común que favorece tanto la comunicación entre los integrantes de un proyecto como el mantenimiento de las aplicaciones. La identificación de los elementos de datos y de los programas de las aplicaciones afectados por el Año 2000 y por la Implantación del Euro, se vio enormemente simplificada y acelerada por esta normativa, mostrando su importancia en la resolución del impacto frente a los cambios que los sistemas de información sufren con el tiempo.



## Capítulo 23

# Diccionario de Datos

### 23.1. Concepto

El idea básica de lo que es un "Diccionario de Datos" se centra en los siguientes conceptos:

- Es un repositorio de toda la información sobre el recurso «datos».
- Contiene metadatos.
- Provee medios para documentar datos.

Las funciones básicas que debe de soportar un diccionario de datos son:

- Documentación de datos.
- Análisis de impacto.
- Obtención de documentación.
- Generación de código relacionado con los datos (ej. definición de ficheros o base de datos).

Generalmente se establece una distinción entre el concepto de diccionario y de directorio de datos, reservando éste último para los que son internos a un gestor de base de datos. Los catálogos de las bases de datos relacionales entrarían en esta última categoría.

Los diccionarios tienen por lo tanto una vocación multiplataforma sin la cual estarían limitados a un entorno y no podrían contener información de todos los datos de una organización.

## 23.2. El entorno

El diccionario de datos utilizado fue desarrollado por la empresa británica MSP [7]. La empresa no existe en la actualidad por haber sido adquirida por ASG (www.asg.com) que incorporó estos productos dentro de su línea de Repositorios de Metadatos, junto con ASG-Rochade y ASG-becubic, bajo la denominación de Manager Products. El Diccionario parte de un metamodelo [9] donde están establecidos los tipos de miembro base<sup>1</sup> y las relaciones existentes entre los mismos. Estos tipos base junto con las denominadas "unidades seleccionables" constituyen el modelo básico del diccionario, pudiendo el usuario definir ("encode") objetos pertenecientes a los mismos. Estas definiciones residen en el núcleo, denominado ControlManager [6][8]. Además existe la posibilidad de usar los denominados tipos extendidos que permiten en base a jerarquías preestablecidas incorporar los tipos de datos que contemplan sin necesidad de definiciones adicionales. Estas jerarquías son EDPS (Extended Data Processing Structure), SAS (Structured Analysis Structure) y SDS (Structured Development Structure) y sus diagramas se presentan al final del documento (apéndice C. pág. 165. Por último existe la posibilidad de extender los tipos de miembros base definiendo los tipos necesarios para una instalación o simplemente ajustando el lenguaje al entorno.

La definición de los tipos de usuario se hace por medio de las facilidades de administración que permiten definir lo que se conoce como una tabla UDS<sup>2</sup> consistente en:

- Una jerarquía para los tipos de miembro.
- La definición de los tipos de miembro, modelados en base a los tipos de miembro base. Ej. PROGRAMA asimilado a PROGRAM o ENTIDAD a GROUP.
- Opcionalmente tipos de miembro grupo que habilitan referencias colectivas.
- Los tipos de atributo nuevos.
- Opcionalmente los grupos de atributos de nuevo a efectos de referencia colectiva.

El diccionario contempla un concepto de "status"[5][4] bajo el que se puede discriminar el trabajo en distintos entornos, típicamente los de desarrollo, aceptación, integración y producción (fig. 23.1). Aunque el trabajo se realiza en un "status", se controla la no existencia de duplicidades o se avisa de la posible necesidad de

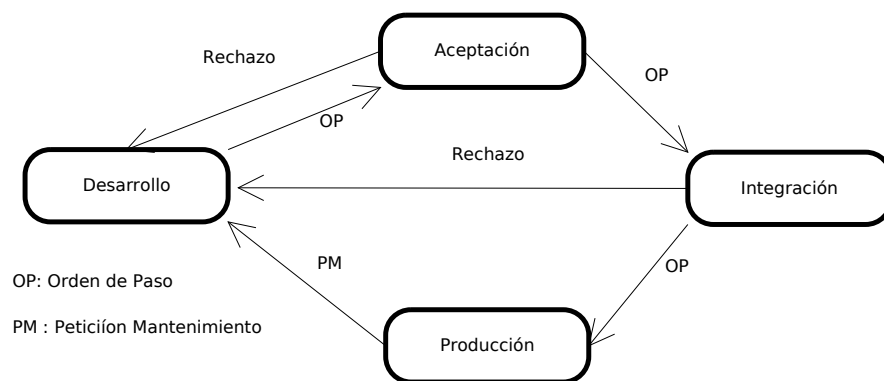
---

<sup>1</sup>SYSTEM, PROGRAM, MODULE, FILE, GROUP, ITEM.

<sup>2</sup>UDS: User Defined Syntax.

extender la acción a otros "status", lo que se permitirá o no en función de los privilegios establecidos.

Todas los órdenes de paso entre entornos se realizan y controlan por medio del diccionario que comprueba la existencia de los objetos necesarios en el estado adecuado e invoca directamente los procesos necesarios en cada entorno.



**Figura 23.1:** Diccionario y entornos

### 23.3. Modelo de información

El modelo de información del Diccionario se presenta en la figura 23.2. En el diagrama aparecen representados los distintos tipos de entidad definidos en el mismo y las relaciones más importantes que internamente se establecen mediante cláusulas del tipo: contains, refers, uses etc. Del modelo de información global se derivan los modelos de información para cada elemento documentado que podríamos asimilar al concepto de vistas del modelo relacional. El detalle de los mismos puede verse en el apéndice D (pág. 169).

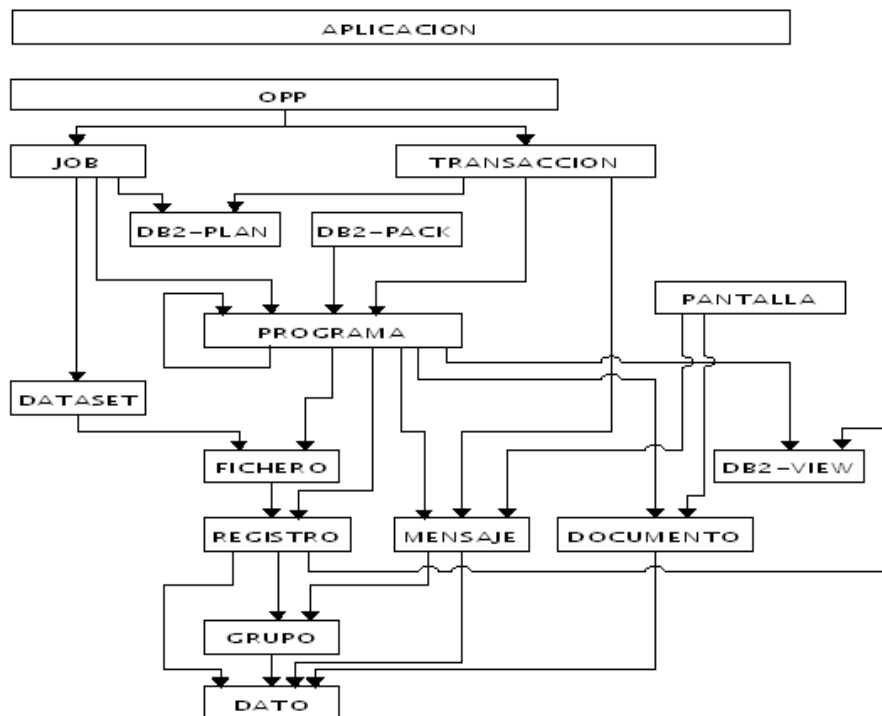


Figura 23.2: Modelo de información del Diccionario

## 23.4. Funciones soportadas

Sobre el soporte básico del diccionario, que permite consultas de la información almacenada según el modelo definido en base a un lenguaje 4GL, se han desarrollado una colección importante de funciones para soportar la metodología, los estándares y las normas de calidad establecidas que configuran un entorno de desarrollo del que se ha podido constatar en el tiempo su validez. Se presenta a continuación un resumen de las más importantes:

- Generación de fuentes de copias o "includes".
- Control de compilación en función del entorno.
- Control de coherencia y validación de objetos pasados a producción.
- Generación de DDL para base de datos relacionales con aplicación de normativa de la instalación.
- Importación de programas y estructuras de terceros.
- Generación automática de código de utilidades para descargas de base de datos.
- Generación automática de documentación a distintos niveles.
- Análisis de cadenas de ejecución ("jobs") para incorporación de documentación en el diccionario.
- Análisis de uso de elementos de datos.
- Desarrollo de una interfaz gráfica de usuario con Eclipse/RCP<sup>3</sup>.
- Automatización de la gestión del repositorio mediante una interface gráfica Eclipse/RCP<sup>4</sup>.

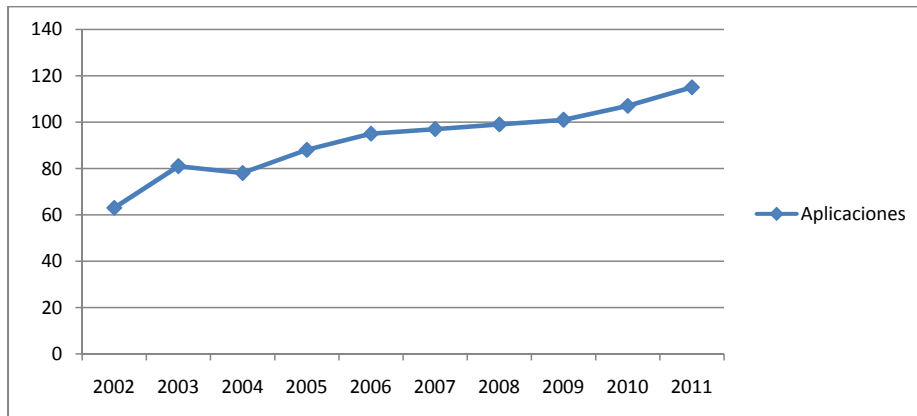
## 23.5. Datos cuantitativos

La importancia del diccionario y el papel central que juega en el desarrollo de todos y cada uno de los proyectos que se llevan a cabo en la instalación queda constatado a la vista de los siguientes gráficos, en los que se recoge la evolución del número de elementos almacenados de los tipos de objetos más relevantes:

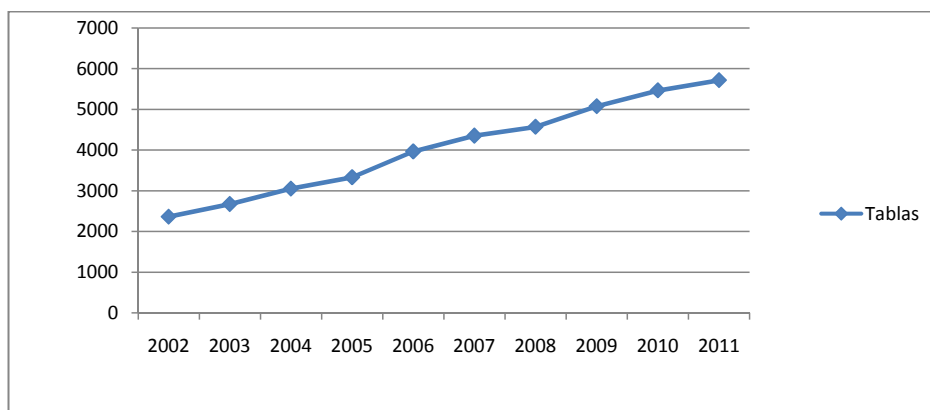
---

<sup>3</sup>Se hizo en el ámbito de un proyecto fin de carrera de la Facultad que dirigí [51].

<sup>4</sup>Proyecto dirigido por el autor del proyecto anterior [50].

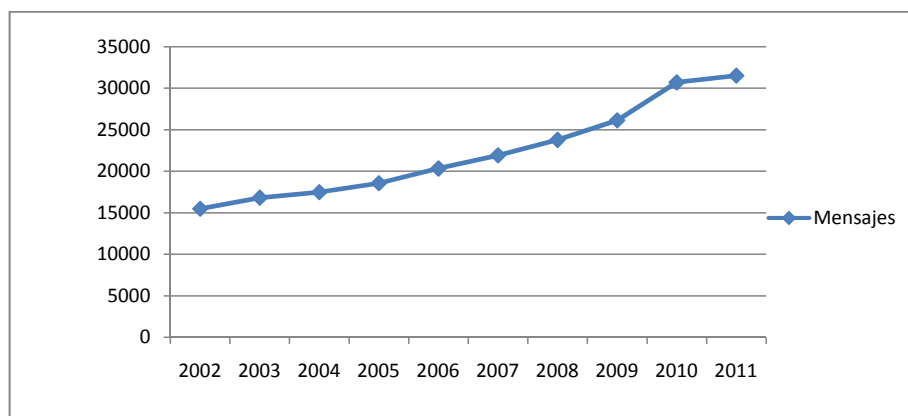


**Figura 23.3:** Evolución - Número de aplicaciones

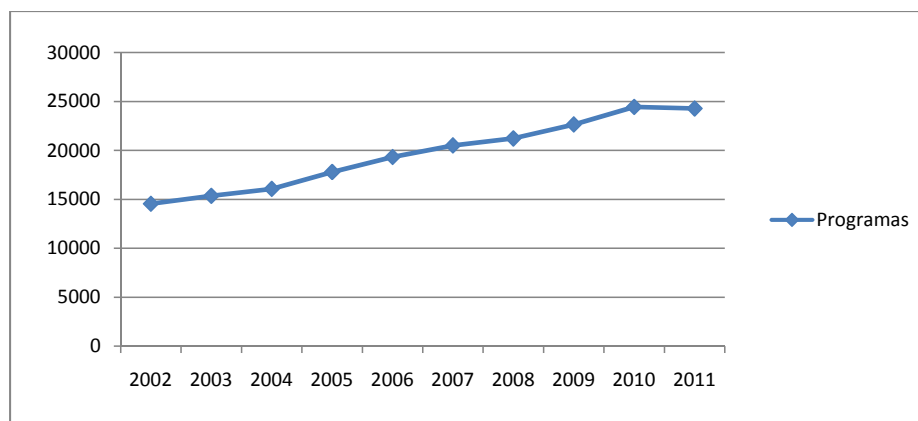


**Figura 23.4:** Evolución - Número de tablas

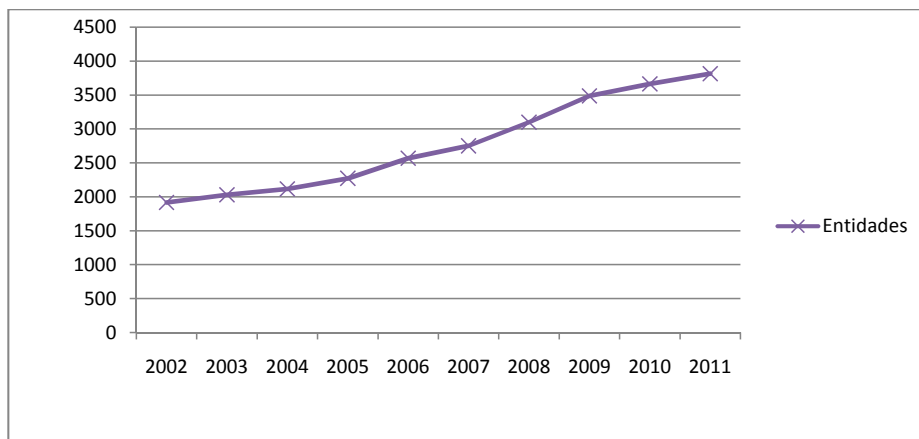




**Figura 23.5:** Evolución - Número de mensajes



**Figura 23.6:** Evolución - Número de programas



**Figura 23.7:** Evolución - Número de entidades

## Capítulo 24

# Lecciones aprendidas

La importancia de la función de Administración de Datos, el foco en el modelado de los datos, separando los niveles conceptual, lógico y físico, y la gestión adecuada, en este caso utilizando un diccionario de datos, han constituido elementos de la mayor importancia.

Las herramientas CASE<sup>1</sup> de mitad de los años 80 fueron incapaces de competir con este planteamiento pues si bien presentaban una interfaz gráfica atractiva, en un momento en el que el diccionario no la tenía, quedaban empequeñecidas tan pronto tenían que competir con lo que demandaba la función de administración de datos y con la integración conseguida entre el diccionario y las plataformas de ejecución. Indudablemente para una instalación nueva el planteamiento podría ser diferente, pero habría entonces que desarrollar la funcionalidad necesaria para soportar la Administración de Datos y la integración con el entorno de ejecución.

Llama la atención sin embargo la poca importancia que generalmente se le da en el desarrollo de proyectos. Una prueba evidente son los propios proyectos de fin de carrera de la facultad donde con frecuencia la atención que se presta al modelado conceptual es muy pobre y las definiciones de datos simplemente no existen, dando por sentado que cualquier nombre de entidad o atributo tiene el mismo significado para cualquier persona que para el autor del proyecto. Un ejemplo que la experiencia profesional me ha permitido vivir y que puede resultar ilustrativo es el de la definición de lo que es "cliente" en una organización; hay situaciones en que un cliente es una persona, otras veces el conjunto de titulares de un contrato y aún conceptos más amplios como unidades familiares o grupos de empresas. Los sistemas necesitan una gestión adecuada de los datos para poder garantizar su evolución. Los

---

<sup>1</sup>CASE: Computer Aided Software Engineering.

nombres correctos y las definiciones precisas son de la mayor importancia, sin ellos el deterioro de las aplicaciones por el mantenimiento está garantizado.

Parte IV

Resumen final



## Capítulo 25

# Conclusiones

La velocidad a la que suceden los cambios en la Informática hace que se preste poca atención a conocer y entender su evolución. El objetivo principal de este trabajo ha sido analizar esa evolución, utilizando como referencia una empresa del ámbito financiero durante un periodo importante de tiempo, 25 años.

El periodo de tiempo es significativo pues representa prácticamente la mitad de la existencia de los ordenadores en la empresa, ya que si bien los primeros ordenadores en la empresa datan de 1947, el comienzo de su auge hay que situarlo en los años 60. El entorno empresarial que se ha utilizado como referencia es también significativo por la importancia temprana que ha tenido la informática en las entidades financieras, ya que este tipo de empresas han sido con frecuencia pioneras en la incorporación de tecnología informática y en ellas la información es un factor de producción de la mayor importancia.

Tal como se planteó en la introducción, en este trabajo se han abordado distintos problemas relacionados con la evolución de la tecnología informática en la empresa utilizando dos perspectivas: la evolución tecnológica y la durabilidad de los diseños. En la primera de las dos perspectivas se ha realizado un análisis de la problemática de la incorporación y evolución de distintas tecnologías; en la segunda se ha analizado un sistema que ha sido capaz de sobrevivir a esos cambios tecnológicos.

El análisis efectuado se ha realizado sobre escenarios reales. No se trata por tanto de planteamientos experimentales realizados por el doctorando, sino de datos obtenidos y experiencias vividas en su participación directa en esos escenarios durante los 25 años tratados.

La formación en las Facultades de Informática peca de aislar al alumno del contexto real de aplicación de los conocimientos que adquiere. Los planteamientos académicos son con frecuencia irreales, presentando situaciones que no tienen ni pasado ni herencia. En esta tesis se ha pretendido hacer una aproximación a esa realidad profesional presentando problemas reales de incorporación de la tecnología, conceptualizándolos dentro de un marco de aproximación académica.

Las conclusiones del análisis realizado se han reseñado al final de cada una de los escenarios tecnológicos tratados, incorporando además un capítulo con las lecciones aprendidas al final de cada una de las tres primeras partes de este documento (capítulos 11, 20 y 24). Estas conclusiones no se van a repetir en este resumen, remitiendo al lector a su ubicación en cada una de las partes de este trabajo, pero sí se van a señalar algunos temas por su naturaleza más horizontal:

- Migraciones entre plataformas.

Cuando se realiza una migración son muchas las consideraciones que hay que tener en cuenta, cada problema es distinto y el punto de partida también es diferente; esto hace variar el modo de acometerlas, su plazo de ejecución y sobre todo, la forma en la que se hereda de la situación anterior y como se coexiste con ella. El análisis minucioso de las alternativas posibles y la selección del modo de hacerlo son aspectos críticos, no debiendo olvidar nunca las lecciones aprendidas que se expusieron en el Capítulo 11.

En esta tesis se han visto ejemplos de este problema en los casos de: la migración inicial de plataforma central, presentada en el capítulo 4, la migración relacional, tratada en el capítulo 5, y la migración de tecnología en el caso del Terminal Financiero en el capítulo 6.

- Procesos de "batch".

Los procesos "batch" son imprescindibles en las instalaciones de las características de la que aquí se ha utilizado como referencia. Es preciso resaltar de nuevo la necesidad de una planificación de los procesos a realizar y la importancia de la integración con el teleproceso, cada vez más compleja por las exigencias de disponibilidad de los sistemas las 24 horas del día y todos los días de la semana. Ejemplos de esta situación se han visto en el Data Warehouse, tratado en el capítulo 10, y en el sistema de Cajeros, que fue objeto de la Parte II.



- Multiplicidad de entornos.

En este caso se crearon para desarrollo, aceptación, integración y producción. Habría que citar además el de formación, orientado tanto a oficinas como a centros que impartían formación especializada. Fue fundamental la automatización del movimiento de objetos entre los distintos entornos. También hay que resaltar de nuevo la dificultad del mantenimiento de entornos de pruebas coherentes.

- Sistemas legados.

Nos referimos a los problemas de la evolución y de la incorporación de nuevas tecnologías cuando tienen que convivir con sistemas anteriores. Este tema no se ha planteado como un tema específico, sino que ha sido común a cada uno de los capítulos tratados en la Parte I e inherente a la Parte II. El Terminal Financiero (capítulo 6), la migración a bases de datos relacionales (capítulo 4), la gestión documental (capítulo 8) y la integración con la Red de Cajeros (Parte II), son ejemplos de ello.

Además de los temas que acabamos de señalar es importante reflexionar sobre la motivación y el riesgo asociado a la evolución. Los enunciados analizados en la primera parte de esta tesis, responden a distintos escenarios y a diferentes motivaciones de evolución. El tipo de riesgo que se asume en una migración o en una incorporación de una nueva tecnología es diferente dependiendo de muchos factores: nivel de la decisión, alcance, duración, madurez de la tecnología, etc. Uno de los riesgos que con frecuencia se infravaloran es el de la aceptación del cambio por los usuarios finales. Cuando una migración se produce por un planteamiento estratégico este tipo de riesgo está mitigado, pero cuando la decisión es de tipo técnico es necesario tener especial cuidado en su gestión.

Los escenarios de evolución tecnológica tratados obedecieron a distintas motivaciones y tuvieron por lo tanto riesgos asociados diferentes:

- Migración de plataforma.

Fue una decisión de tipo estratégico, con alcance global y plazo definido. A pesar de tratarse de una decisión estratégica, tomada como tal por la dirección de la entidad financiera, no estuvo exenta de riesgo por la envergadura del proyecto y por su naturaleza transversal.

- Migración relacional.

Obedeció a una decisión tecnológica. El alcance fue muy amplio al afectar a todas las aplicaciones, sin embargo se pudo planificar en distintos proyectos con plazos diferentes. La duración total fue de algo más de 15 años.

- Evolución del Terminal Financiero.

En este caso existieron dos motivaciones, la tecnológica, como sucedió en la migración de OS/2 a Windows al quedar el primero descatalogado, y la funcional, como fue el caso de la última versión basada en la plataforma Eclipse/RCP.

- Internet - Banca Electrónica.

Fue una decisión de negocio. El abrir un nuevo canal comercial en modo pasivo, como pueden ser las Webs, o como canal operativo, como es el caso de la Banca Electrónica, es siempre una decisión que parte del ámbito del negocio.

- Integración y gestión de procesos de negocio.

En este caso la motivación tiene dos niveles. Existiendo un trasfondo funcional, la decisión, en el caso de la plataforma de integración, fue de tipo técnico. Sin embargo los procesos de negocio corresponden a una motivación claramente organizativa en la que el objetivo fundamental era la eficiencia. En ninguno de los dos casos se presentaba una situación de riesgo importante.

- Gestión Documental.

Fue una evolución con especial impacto económico y con fácil justificación del retorno de la inversión. Afectó a la organización del trabajo en oficinas y a los procesos de las mismas.

- Sistemas Informacionales.

Respondió a un problema antiguo que las mejoras tecnológicas, la capacidad de almacenamiento y la capacidad de proceso, hicieron viable.

Por último insistir en estas conclusiones finales en el papel central que tiene que tener la Administración de Datos en las organizaciones. La importancia de la Administración de Datos y del Diccionario de Datos como plataforma de soporte, se ha visto en prácticamente todos los temas planteados. Ese fue el motivo que llevó a dedicarle una parte específica (Parte III). Proyectos como el Año 2000 o la migración al Euro serían mucho más complicados si no hubiese habido una Administración de Datos eficiente, que se preocupase de la clasificación de todos los datos y de la estandarización de los nombres, y sin un Diccionario de Datos que los tuviera perfectamente identificados e inventariados.



## Apéndice A

# SADAT: Sistema Astano de Diseño Automático de Tuberías

### A.1. El proyecto

El primer proyecto citado, SADAT, fue desarrollado entre los años 1974-1979, momento en el que la actividad del astillero ASTANO se centraba en la construcción de petroleros de gran tonelaje. Coincide con la época del cierre del Canal de Suez momento en el que se hizo necesario incrementar la capacidad de los buques para así rentabilizar los transportes desde Oriente Medio que en ese momento tenían que doblar el cabo de Buena Esperanza para salvar el continente africano.

El sistema SADAT tenía como objetivo la optimización del proceso de diseño y del montaje de tuberías en la cámara de máquinas de un buque, cámaras que podían tener decenas de metros en cualquiera de sus dimensiones, estableciendo varios niveles o pisos hasta alcanzar cerca de los 40m. en altura. El objetivo del proyecto era reducir los tiempos de diseño, tiempo empleado en la construcción de planos, y sobre todo los contratiempos que se podían producir en la fase de montaje debidos a interferencias entre elementos.

El planteamiento de la solución consistió en el modelado mediante figuras geométricas de la cámara de máquinas de un buque y de los elementos a incorporar en ella para detectar anticipadamente, mediante procesos automáticos de tipo algorítmico, posibles interferencias en el montaje. El proyecto tuvo una primera fase estrictamente en la que las interferencias se detectaban mediante procesos "batch". Se proyectaba la ubicación de los elementos: tubos, válvulas, medidores, motores etc. mediante planos bi-dimensionales dibujados en papel. En un siguiente

paso los elementos, ya posicionados en el espacio, se recubrían mediante una figura geométrica sencilla extraída de un catálogo predeterminado compuesto por cubos, paralelepípedos, pirámides, pirámides truncadas, cilindros, conos, conos truncados, superficies toroidales, etc. A continuación se ejecutaba el proceso de validación consistente en la detección de interferencias entre estas figuras geométricas que recubrían a los objetos reales. Este proceso estaba basado en un conjunto de algoritmos de detección de interferencias que permitían determinar si dos figuras tenían intersección o no. La propia cámara de máquinas se modelizaba como un figura más y se subdividía en subespacios menores. Cada figura era validada contra el resto de las figuras ya posicionadas contenidas en subespacios que tuviesen interferencia con la figura a posicionar. Es curioso señalar que actualmente en bases de datos espaciales se hacen planteamientos parecidos de simplificación cuando se pretende buscar proximidades [34].

En una segunda fase del proyecto se introdujo un elemento técnico singular, muy avanzado para el momento, consistente en un Sistema Gráfico Interactivo sobre el que visualmente se hacía el diseño tridimensional que periódicamente era transferido al sistema de proceso por lotes de detección de interferencias. Se partía de un fichero de elementos a posicionar constituido por los recubrimientos de los objetos reales y se posicionaban manualmente sobre un espacio tridimensional, aplicando funciones de rotación, escalado (zoom) etc. para facilitar el posicionamiento visual sin interferencias. Para este propósito se utilizó la plataforma gráfica vectorial Adage cuyas características técnicas por su singularidad se detallan en la sección siguiente.

La tercera fase, en la que ya no tuve la oportunidad de participar por haber dejado la empresa, consistiría en sustituir la validación de interferencias en modo de proceso por lotes por la validación en línea directamente en el Sistema Gráfico.

El proyecto resultó apasionante para una persona de formación matemática, ya que el peso más importante del sistema estaba en la parte geométrica, tanto desde el punto de vista del sistema de detección de interferencias, como desde el del desarrollo de la aplicación de diseño asistido sobre una interfaz de usuario muy evolucionada técnicamente que permitía la interacción con el diseño con elementos tales como diales, que suministraban valores en el rango  $(-1.0;1.0)$  interpretables como escalados, rotaciones etc., botones (switches) cuya pulsación suponía un evento a tratar, lápiz óptico con capacidad de detectar vectores en pantalla, tableta de datos con la que se emulaba el comportamiento de un ratón, dispositivo que no era común como lo es hoy en aquel momento de la tecnología, y finalmente teclado. Hay que pensar que entonces, año 1977, estábamos en un momento en el que la programación se realizaba fundamentalmente escribiendo el código en papel, que posteriormente se perforaba sobre fichas que eran leídas por el ordenador, no existiendo todavía el concepto de ordenador personal en la empresa. El equipo utilizado y buena parte de los algoritmos de transformación realizados se explican en los textos de Rogers-Adams [63] y Newman [57].

## A.2. El sistema gráfico Adage

El equipo gráfico interactivo utilizado en el proyecto estaba configurado con los siguientes elementos:

- Procesador DPR4 con memoria de 32K.
  - Sistema Operativo AMOS/2
  - Lenguajes: Adept Assembly y Fortran IV con compilador gráfico
  - Editor de Texto en pantalla
- Procesador Gráfico GS/340.
  - Microcódigo Grafx
  - Zoom y 3-D
- Pantalla CRT de 23" circular.
- Disco intercambiable de 81 millones de bits.
- Impresora/Plotter electrostática.
- Lector de cinta perforada.
- Teclado alfanumérico.
- 32 Switches de Función.
- Data Tablet de 11".
- 6 Diales de Control Variable.
- Light Pen.
- 2 Controles de pedal.
- Comunicación con el IBM/370 para transmisión de ficheros.

Además de su utilización en el proyecto SADAT de Diseño de Tuberías mencionado, también fue utilizado en el Sistema Nesting de optimización de recortes. En este sistema el problema consistía en recortar piezas planas, de forma variable, de una chapa plana con el mayor aprovechamiento del material. El sistema presentaba un catálogo de piezas a obtener que manualmente se distribuían, en modo gráfico, sobre la figura que representaba la plancha de material tratando de maximizar el porcentaje de aprovechamiento. El resultado obtenido se volcaba sobre una cinta perforada que era leída por las máquinas automáticas de oxicorte para hacer el corte real. Con anterioridad a la utilización de este sistema el trabajo se realizaba en escala 1:10 con recortes en papel que se distribuían manualmente.





## Apéndice B

# Proyecto Cajeros: Estructura de los mensajes

En este apéndice se describe de forma genérica la estructura de los mensajes, sin entrar en detalle de longitudes ni de formato de campos.

### B.1. Cabeceras

La estructura de los mensajes obedece a un formato típico de cabeceras y cuerpo.

#### B.1.1. Cabecera de control

Una vez eliminada la parte del mensaje que corresponde a protocolo de red, la primera parte del mensaje la constituye la cabecera del FAP estructurada de la siguiente forma :



**Figura B.1:** Cabecera del FAP

La información de la figura corresponde a :

**OO** Origen del mensaje

**OD** Destino del mensaje

**NS** Identificador de sesión

**NA** Identificador de aplicación servidora

**IOP** Identificador de operación

**NC** Número de canal lógico

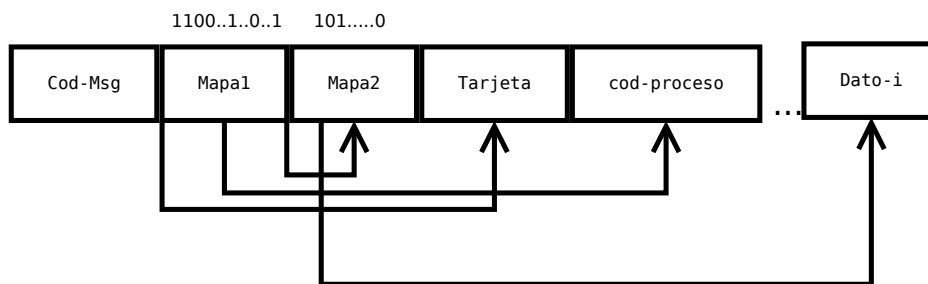
**NS** Número de secuencia

**CE** Código de error

**BI** Bits indicadores :

- Tipo de mensaje: control/datos
- Sentido del mensaje: pregunta/respuesta
- Indicador de recuperación
- Otros...

### B.1.2. Cabecera de datos



**Figura B.2:** Cabecera de datos

## B.2. Cuerpo

Desde el punto de vista de la aplicación, una vez que ha sido extraída del mensaje la cabecera de control del FAP, los mensajes se estructuran en tres partes: un código de mensaje, un mapa de bits y tantos campos de datos como se informan en el mapa de bits. Cada bit del mapa hace referencia a un campo concreto, por ejemplo el saldo de la cuenta, con un formato y longitud concreta. El formato puede ser:

1. Numérico empaquetado sin signo (dos dígitos numéricos en cada byte).
2. Alfanumérico EBCDIC.
3. Uno o más conjuntos de ocho bits de información.
4. Código numérico EBCDIC.

Los mensajes tienen un código compuesto que en su primera parte identifica la clase del mensaje y en la segunda el mensaje concreto dentro de la clase. Para identificar la operación concreta puede existir además, dentro de los campos de datos, un dato de tipo de proceso que incluye el tipo de operación.

Los mapas de bits, segunda parte del mensaje tras el código, se componen de grupos de un número fijo de bits. El número de grupos puede ser uno o más, identificándose que hay más de uno mediante el último bit del grupo que indica si hay más grupos o no. El mapa de bits es un mecanismo que permite ajustar el tamaño del mensaje a la información que realmente porta, evitando espacios vacíos o separadores de campos. Si el mensaje tuviese un único mapa de bits y éste tuviese activados solamente los bits 10, 23 y 34, indicaría que a continuación del mapa vendrían tres campos cuyas longitudes y formatos serían los definidos para los campos 10, 23 y 34. Los mensajes son así de longitud variable y es el mapa de bits quien regula la información que soportan.

La tercera parte del mensaje la constituyen los campos de datos, que dependen de cada mensaje y son de formato predefinido. La información del mapa de bits sería en ese sentido redundante si no pudiese darse la circunstancia de la existencia de campos opcionales.

Otra característica interesante en la definición de mensajes es la existencia en algunos de ellos de un MAC<sup>1</sup>. Su uso es opcional y depende de acuerdo previo entre las partes: consorcio y banco. El MAC se calcula en base a claves previamente intercambiadas y se trata como dos campos más, controlados por el mapa de bits, uno para informar de la clave utilizada por su número relativo y el segundo para contener el resultado del propio MAC.

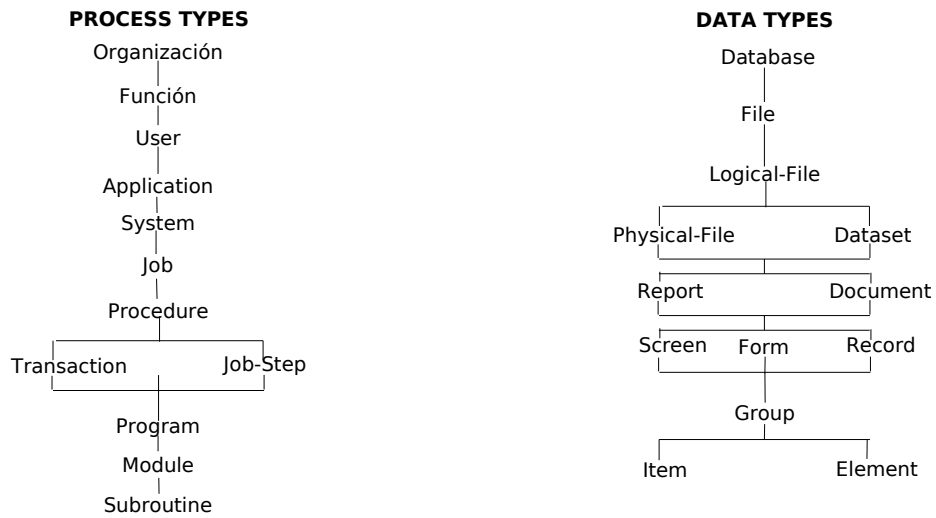
<sup>1</sup>MAC: Message Authentication Code.



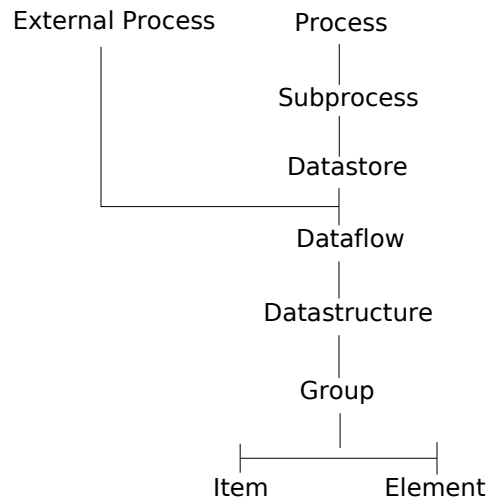
## Apéndice C

# Diccionario de Datos: Jerarquías

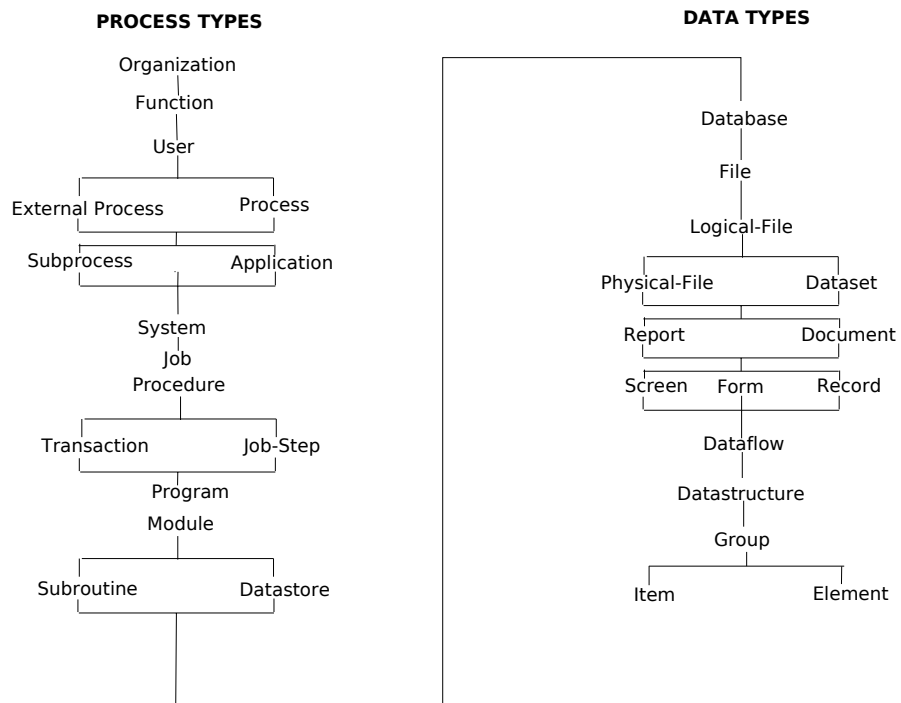
Las jerarquías que se detallan en este apéndice son jerarquías por defecto para modelos de información preestablecidos. Como ya se explicó en el capítulo correspondiente al Diccionario de Datos, en la Parte III de este documento, el modelo de información del diccionario es definible y en la instalación utilizada como referencia no se usó ninguno de estos tres modelos que responden a formas de documentar en metodologías de desarrollo ampliamente utilizadas.



**Figura C.1:** Jerarquía EDPS (Extended Data Processing Structure)



**Figura C.2:** Jerarquía SAS (Structured Analysis)



**Figura C.3:** Jerarquía SDS (Structured Development Structure)



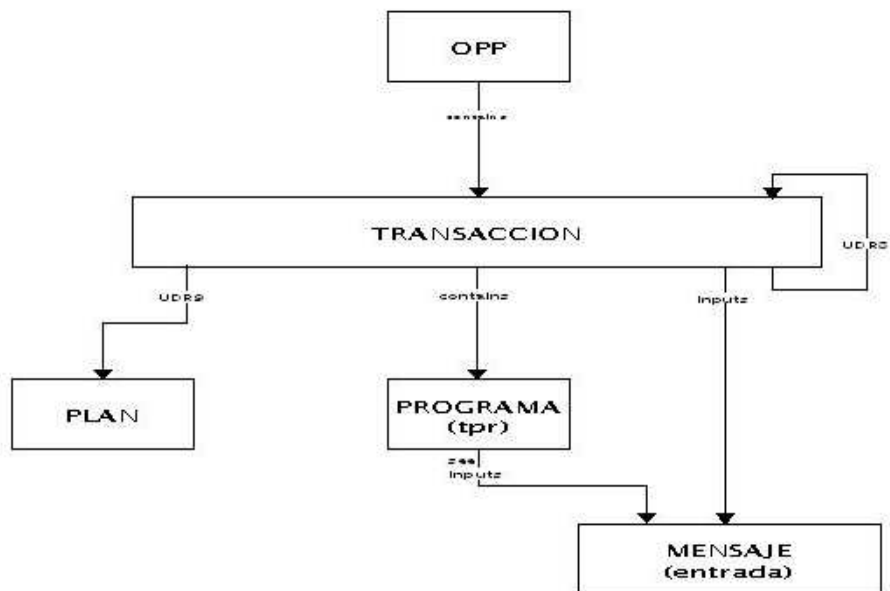


## Apéndice D

# Diccionario de Datos: Modelo de información

En este apéndice se presenta el detalle de la estructura de información definida para la instalación que se referencia. Tomando como base el modelo global presentado en el capítulo correspondiente al Diccionario de Datos (fig. 23.2 pág. 142) y las relaciones definidas en el metamodelo establecido, se detallan en las figuras siguientes las relaciones existentes para cada tipo de objeto documentado en el diccionario.





**Figura D.2:** Documentación transacciones

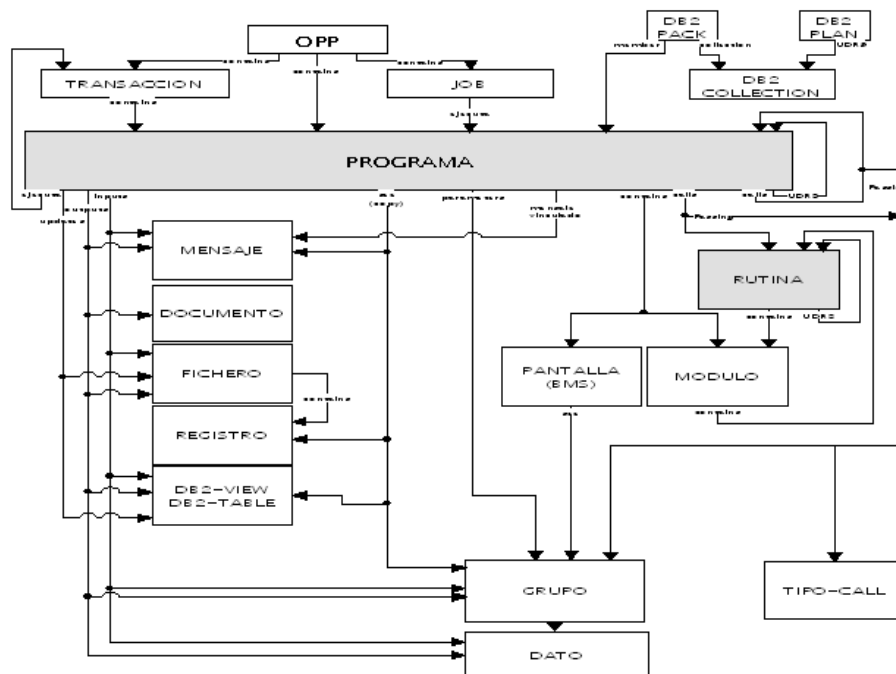


Figura D.3: Documentación programas

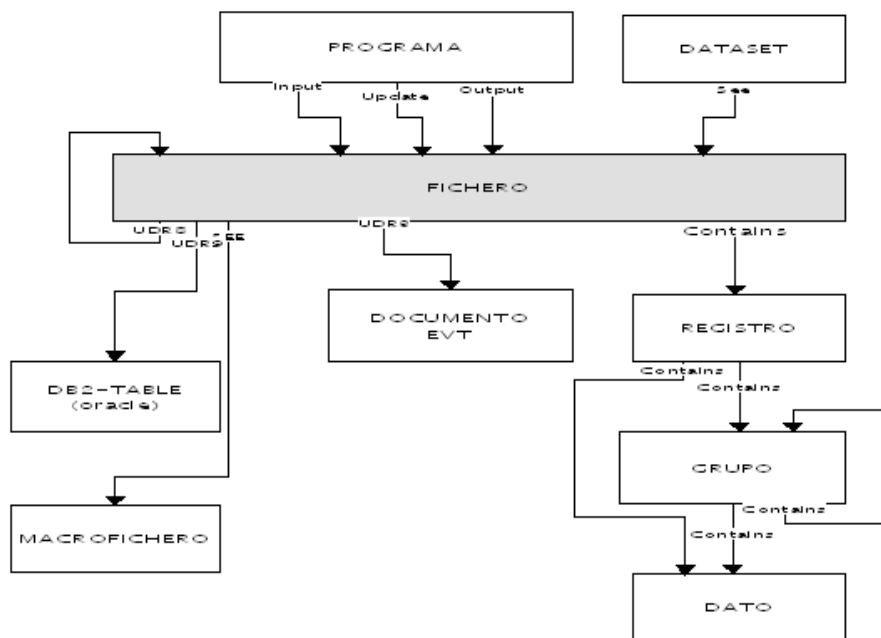


Figura D.4: Documentación ficheros

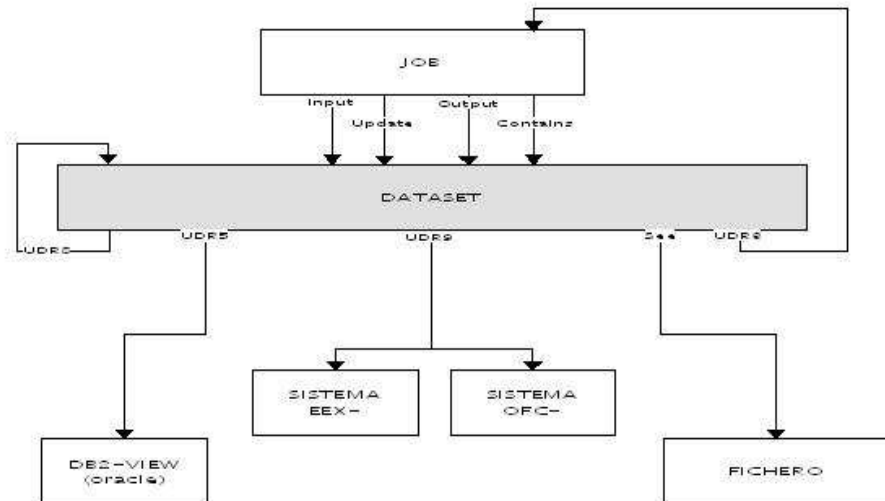


Figura D.5: Documentación datasets

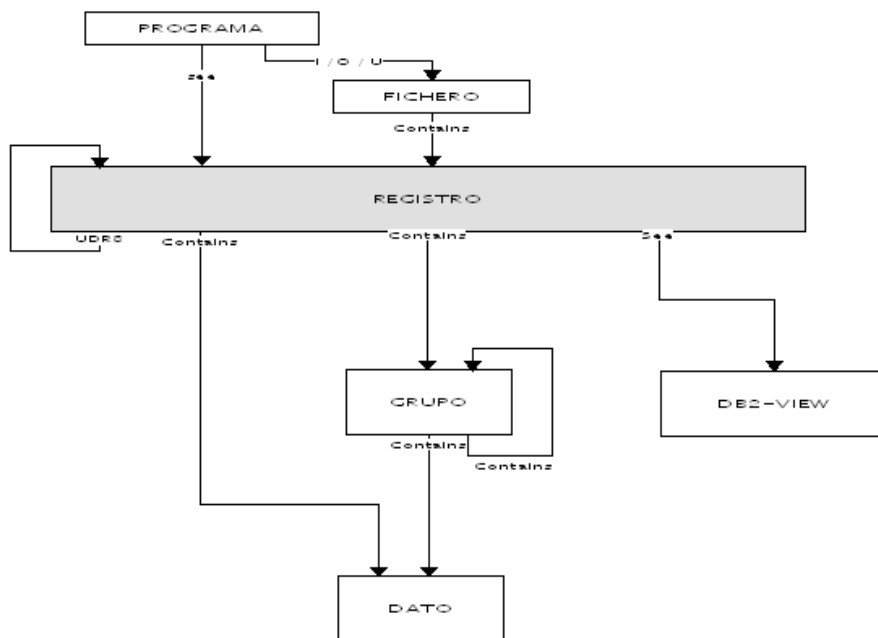


Figura D.6: Documentación registros





## Apéndice E

### Siglas y acrónimos

**Actuate** Actuate Corporation ([www.actuate.com](http://www.actuate.com)). Empresa de software que fundó y colidera el proyecto de código abierto Eclipse BIRT (Business Intelligence and Reporting Tools). Disponen de productos comerciales que extienden las capacidades de BIRT.

**Adage, Inc. Analogic Digital Age.** Empresa ya desaparecida, radicada en Boston (EE.UU), que fabricó algunos de los sistemas gráficos interactivos más potentes de los años 70.

**AEAT** Agencia Española de Administración Tributaria ([www.aeat.es](http://www.aeat.es)).

**AFP** Advanced Function Printing. Formato de documentos utilizado por IBM orientado a la impresión.

**APPC** Application Program to Program Communication. Protocolo de la familia SNA para comunicaciones peer-to-peer.

**ASG** Allen Systems Group ([www.asg.com](http://www.asg.com)).

**ASTANO** Astilleros y Talleres del Noroeste. Empresa constructora de grandes buques y plataformas situada en la ría de Ferrol. Actualmente forma parte de Navantia.

**A2A** Application to Application. Concepto de integración de aplicaciones dentro del ámbito de una empresa [53].

**BIRT** Business Intelligence and Reporting Tools. Sistema de generación de informes de código abierto basado en Eclipse que se integra con aplicaciones Java/Java EE. El desarrollo original fue de la empresa Actuate.

- BMS Basic Mapping Support.** Interfaz entre el Servidor Transaccional CICS y los programas de aplicación que corren en él para soportar flujos de datos bajo protocolo 3270. Resuelve el formateo y presentación de los datos.
- BPM Business Processing Management.** Aproximación que aglutina la técnicas, herramientas y servicios de integración necesarios para modelizar, ejecutar y monitorizar procesos con foco en su optimización. Es tanto una filosofía como actualmente una categoría de productos software.
- B2B Business to Business.** Concepto de integración de aplicaciones traspasando la frontera de una empresa [53].
- CCA Centro de Cálculo de Abonado.** Concepto utilizado en la RETD que define un origen o destino de una comunicación.
- CERN** European Organization for Nuclear Research ([www.cern.ch](http://www.cern.ch)).
- CICS Customer Information Control System (CICS).** Servidor Transaccional de IBM fundamentalmente en los grandes ordenadores del fabricante (mainframe) bajo sistemas operativos z/OS o z/VSE. Tradicionalmente se le denominó como Monitor de Teleproceso.
- Cobol Common Business Oriented Language.** Lenguaje de propósito comercial definido bajo los auspicios del Departamento de Defensa de EE.UU y estandarizado por ANSI en 1968.
- CSB Consejo Superior Bancario.** Actualmente englobado en la AEB (Asociación Española de Banca Privada) ([www.abbanca.es](http://www.abbanca.es)).
- Datamanager** Software de Diccionario de Datos original de la Empresa MSP hoy integrado en ASG.
- DB2** Sistema de Gestión de Bases de Datos Relacionales de la empresa IBM.
- DDL Dynamic Link Library.** Concepto de librería de programas compartidos en los sistemas operativos Windows y OS/2.
- DL/I Data Language I.** Sistema de Gestión de Bases de Datos Jerárquicas de la empresa IBM. Se utiliza en el mismo sentido que IMS cuando éste se refiere al Gestor de Base de Datos y no al monitor de teleproceso.
- DMIV/TP** Monitor de teleproceso en los ordenadores Honeywell-Bull.
- DPS** Familia de ordenadores del fabricante Honeywell-Bull. En su línea de grandes ordenadores estaba el DPS-8 mencionado.

- EAI** Enterprise Application Integration. Hace mención tanto a una filosofía como a la categoría de soluciones que la sustentan, tanto comerciales como de software libre.
- EBCDIC** Extended Binary Coded Decimal Interchange Code (EBCDIC). Sistema de codificación de caracteres de 8-bit que aparece en los años 60 en los grandes sistemas de IBM y que fue adoptado posteriormente por otros fabricantes.
- Eclipse** Comunidad para individuos y organizaciones que desean colaborar en el desarrollo de software de código abierto (open source). Además del conocido entorno de desarrollo se citan en este trabajo dos proyectos adicionales RCP y BIRT.
- ELKA** Element Key Attribute. Modelo Conceptual de Datos propuesto por Hughes Aircraft y la Universidad de California (UCLA) [61]
- FURPS** Functionality, Usability, Reliability, Performance, Supportability. Modelo, inicialmente propuesto por Hewlet-Packard, para clasificar los atributos de calidad, funcionales y no funcionales, del software.
- GCOS** Sistema Operativo de los ordenadores Honeywell-Bull.
- ICA** Interconexión Centros de Abonado. Modo de conexión con la RETD bajo arquitectura SNA en el que se definen dos terminales lógicos, uno para entrada de mensajes desde la red y otro para salida de los mismos.
- IDMS** Integrated Database Management System. Sistema de Gestión de Bases de Datos con modelo en red que tiene su origen en los trabajos de C. Bachman (IDS Integrated Data Store) y que fue utilizado por distintos fabricantes, entre ellos Honeywell-Bull.
- IMS** Information Management System. Base de datos jerárquica de IBM, también monitor de teleproceso. (ver DL/I).
- IRM** Information Resource Management. Filosofía de diseño, gestión y control de los recursos necesarios para procesar información de modo compartido entre usuarios y aplicaciones. Los elementos fundamentales son: Recursos del negocio, Sistemas Físicos y Lógicos, y Datos.
- ISARD** Interface SNA a R de Datos.
- JNI** Java Native Interface. Framework para la integración de la plataforma Java en modo bi-direccional con aplicaciones nativas de la plataforma de ejecución escritas generalmente en C, C++ o Assembler.
- J2EE** Java Platform, Enterprise Edition o Java EE. Plataforma Java para desarrollo de aplicaciones.

- Host** En general sería una computadora conectada a una red que provee servicios a la misma. También se aplica al concepto de ordenador principal en sistemas centralizados.
- Job** En el sistema operativo OS/MVS de IBM se trata de un trabajo que se envía al sistema para su ejecución. Las características del mismo se definen en un lenguaje de script denominado JCL (Job Control Language).
- MAC** Message Authentication Code. Valor que se calcula mediante la aplicación de una función hash con clave secreta intercambiada previamente entre el origen y el destino.
- Microsoft Office** Conjunto de aplicaciones integradas del fabricante Microsoft para tareas de Ofimática.
- Mips** Millones de instrucciones por segundo. Medida de potencia para comparar ordenadores con características similares.
- MSP** Manager System Products. Empresa originalmente británica fabricante de soluciones de repositorio. Actualmente es propiedad de ASG que mantiene una familia de productos de Diccionario con ese nombre.
- MVS** Multiple Virtual Storage. Sistema Operativo de IBM (1974) usado en sus sistemas S/370 y S/390.
- NCP** Network Control Program. Programa de control de comunicaciones que se instala en la unidades de comunicaciones de los sistemas IBM.
- NCR3270** Protocolo del fabricante NCR utilizado para comunicar sus Terminales Financieros
- Nixdorf** Fabricante especializado en equipos de oficina, tanto ordenadores como periferia ([www.wincor-nixdorf.com](http://www.wincor-nixdorf.com)).
- .Net** Plataforma de Microsoft para desarrollo de aplicaciones.
- Notes** Conjunto de aplicaciones colaborativas desarrollados inicialmente por la empresa Lotus y posteriormente adquiridos por IBM.
- Olivetti** Fabricante de informática especializado en equipos de oficina, tanto ordenadores como periferia ([www.olivetti.com](http://www.olivetti.com)).
- PDF** Portable Document Format. Formato de representación de documentos que los independiza de las aplicaciones y plataformas original de la empresa Adobe Systems ([www.adobe.com](http://www.adobe.com))
- PDM80** Prototype Development Methodology. Metodología definida por la empresa Computomata International Corporation.

- PIN** Personal Identification Number. Código de seguridad asignado a una persona. Se usa como clave de seguridad en el uso de las tarjetas y generalmente es el mismo para todas las tarjetas de una persona en una entidad.
- PROTOCOLO 3270** Protocolo para manejo de terminales de visualización definido por IBM [1972]. Se trata de un protocolo orientado a bloques que minimiza el número de interrupciones de entrada/salida. Actualmente se utiliza fundamentalmente en emuladores de terminal al no fabricarse los mismos.
- OS/2** Sistema operativo diseñado por IBM para Computadores Personales (PC) originalmente de 16 bits para pasar en su versión 2.0 (1992) a 32 bits [60].
- RCP Rich Client Platform**. Plataforma basada en el IDE de Eclipse para desarrollo de interfaces de usuario de "cliente rico".
- RETD Red Especial de Transmisión de Datos**.
- RM/T Relational Model Extended**. Evolución del modelo relacional propuesta por Codd [14].
- RSAN Red Secundaria de Alto Nivel** dentro de la RETD. Sustituída más adelante por Iberpack X-25.
- SADAT Sistema Astano de Diseño Automático de Tuberías**.
- SADT Structured Analysis Design Technique**. Metodología para el diseño y especificación de sistemas que usa diagramas SA [64] que fueron la base para IDEF0 [59].
- SNA System Network Architecture**. Arquitectura de red propietaria definida por IBM en 1974.
- SOA Service Oriented Architecture**. Paradigma para organizar y usar medios distribuidos que pueden ser propiedad y estar bajo control de diferentes dominios [58].
- SWIFT Society for Worldwide Interbank Financial Telecommunication** ([www.swift.com](http://www.swift.com)).
- UDS User Defined Syntax**. Facilidad del diccionario de datos de MSP para la definición de nuevos tipos por parte del usuario.
- Windows** Familia de Sistemas operativos nacidos con los ordenadores personales (PC) propietario de la empresa Microsoft.
- VSAM (RRDS) Virtual Storage Access Method**. Sistema introducido en los años 70 por IBM para manejo de archivos. Soporta ficheros secuenciales (SSDS), indexados (KSDS) y directos (RRDS) [49].

**VTAM** Virtual Telecommunication Access Method. Programa producto de IBM que desde un mainframe controla una red SNA.

**XML** Extended Markup Language. Lenguaje de etiquetas para codificación de documentos en formato legible tanto para un humano como para un procesador.

# Bibliografía

- [1] Domingo Manuel Amado Candamo. *Implementación de un protocolo de cajeros sobre una tecnología de integración*. Facultade de Informática - Universidade da Coruña, 2012.
- [2] J. Diego Amor Sánchez. *Servidor Final de una Red de Cajeros Automáticos con tecnología Java/Corba*. Facultade de Informática - Universidade da Coruña, 2002.
- [3] ANSI. *ANSI X9.2-1980 (interchange message specification for debit and credit card message exchange among financial institutions)*, 1980. URL [www.cs.tufts.edu/nr/cs257/archive/alfred-spector/spector85cirrus.pdf](http://www.cs.tufts.edu/nr/cs257/archive/alfred-spector/spector85cirrus.pdf).
- [4] ASG-Manager-Products. *Status Concepts*, 1987.
- [5] ASG-Manager-Products. *Advanced Status*, 2011.
- [6] ASG-Manager-Products. *Control Manager User's Guide*, 2011.
- [7] ASG-Manager-Products. *Dictionary/Repository User's Guide*, 2011.
- [8] ASG-Manager-Products. *System Administrator Guide*, 2011.
- [9] ASG-Manager-Products. *User Defined Syntax Version 2.7.3 MPR2100-273-UDS*, 2011.
- [10] Gary D. Brown. *Advanced ANS COBOL with Structured Programming*. John Wiley and Sons, 1977.
- [11] Cii-Honeywell Bull. *Data Management IV, Data Base Administrator, Reference Manual*, 1978.
- [12] Edward H. Carr. *¿Qué es la Historia?* Ariel, 1961-2010.
- [13] P. Chen. *The Entity-Relationship Approach to Logical Data Base Design*. QED Information Sciences, Inc., Wellesley, Mass., 1977.

- [14] E. Codd. Extending the database relational model to capture more meaning. In *ACM TODS*, 4, n<sup>o</sup>. 4, 1979.
- [15] Douglas E. Comer. *How To Write a Dissertation*, 2000. URL <http://www.cs.purdue.edu/homes/dec/essay.dissertation.html>.
- [16] Douglas E. Comer and David Stevens. *Internetworking with TCP/IP (Volume III)*. Prentice Hall International Edition, 1996.
- [17] Codasy Systems Committee. *Selection and acquisition of Database Management Systems*, 1976.
- [18] Computomata-International-Corporation. *PDM-SS User Manual Ver. 2.5*, 1988.
- [19] Alfonso F. Cárdenas. *Sistemas de Administración de Bases de Datos*. Limusa, 1983.
- [20] Robert M. Curtice. *Standards and Procedures for the Data Base Administration Functions*, 1978.
- [21] C. J. Date. *An Introduction to Database Systems. Vol II*. Addison Wesley, 1982.
- [22] Colin J. Date, C. J.; White. *A Guide to DB2*. Addison Wesley, 1982.
- [23] M.L. Díaz Pousa. *Data Warehouse - Estudio del mantenimiento evolutivo y caso práctico en una entidad financiera*. Facultade de Informática - Universidade da Coruña, 2008.
- [24] E.W. Dijkstra. Letters to the editor: go to statement considered harmful. In *Communications of the ACM*, 11 (3), 1968.
- [25] E.W. Dijkstra. The humble programmer. In *Communications of the ACM*, 15 (10), 1968.
- [26] William R. Durrel. *Data Administration*. McGraw Hill, 1985.
- [27] Jeff E. Estefan. *Exploring Open Software Standrds for Enterprise e-business Computing - ibm-Redbooks*, 2000.
- [28] FIC. *Facultade de Informática de A Coruña*, 2012. URL <http://http://gl.wikipedia.org>.
- [29] Mark Fisher, Marius Bogoevici, et al. *Spring Integration Reference Manual, 2.1.3.RELEASE.*, 2012.
- [30] Josep Fontana Lázaro. *La historia*. Salvat - Biblioteca GT, 1973.



- 
- [31] Martin Fowler. *Patterns of Enterprise Application Architecture*. Addison-Wesley Signature Series, 2003.
- [32] Manuel García Galdo. *Servidor Final de una Red de Cajeros con tecnología Java/Java Beans*. Facultade de Informática - Universidade da Coruña, 2003.
- [33] Robert Grady and Deborah Caswell. *Software Metrics. Establishing a Company-wide program*. Prentice Hall, 1987.
- [34] Gilberto Gutiérrez and José Paramá. Finding the largest empty rectangle containing only a query point in large multidimensional databases. In *Proceedings of SSDBM 2012*, LNCS 7338, pages 316–333. Springer, 2012.
- [35] Michael Havey. *Essential Business Process Modeling*. O Reilly, 2005.
- [36] G. Hohpe and B. Woolf. *Enterprise Integration Patterns: Designing, building, and deploying messaging solutions*. Addison Wesley, 2003.
- [37] IBM. *IBM System/360 Reference Data*, 1970.
- [38] IBM. *OS/VS2 MVS Overview (GC28-0984-0)*, 1978.
- [39] IBM. *IMS/VS DB Implantación de BD Físicas*, 1989.
- [40] IBM. *Designing Client-Server Applications with CICS and a GUI: A Practical Guide, ibm-Redbooks*, 1992.
- [41] IBM. *Object-Oriented Interface Design*, 1992.
- [42] W.H. Inmon. *Building the Data Warehouse*. QED Technical Publishing Group, 1992.
- [43] W.H. Inmon and R.D. Hackathorn. *Using the Data Warehouse*. Jon Wiley-QED, 1994.
- [44] ISO. *ISO DIS 7580 (Identification Cards - Card originated messages content for financial transactions)*, 1980.
- [45] ISO. *ISO 7498-1:1994 (Information Tecnology - Open Systems Intercommunication Basic Reference Model)*, 1994.
- [46] Ivar Jacobson, P. Jonsson, M. Christerson, and G. Overgaard. *Object-Oriented Software Engineering. A Use Case Driven Approach*. Addison-Wesley Publishing Company, 1992.
- [47] Ivar Jacobson, Grady Booch, and James Rumbaugh. *The Unified Software Development Process*. Addison-Wesley, 1999.

- [48] Setrag Khoshafian and A. Brad Baker. *Multimedia and Imaging Databases*. Morgan Kaufman, 1996.
- [49] Dave Lovelace, Rama Ayyar, Alvaro Sala, and Valeria Sokal. *VSAM Desmytified, ibm-Redbooks*, 2003.
- [50] Bruno López Lagares. *Automatización de la gestión del repositorio de información corporativo (Diccionario de Datos)*. Facultade de Informática - Universidade da Coruña, 2010.
- [51] Angel López Pazos. *Diseño e implementación de un software para la interacción con el Diccionario de Datos de ASG*. Facultade de Informática - Universidade da Coruña, 2007.
- [52] C. Mairet. *Data Administration: Managing an Important Resource*. Jardine, D. (ed) - North Holland Publishing Co., 1977.
- [53] Bernard Manouvrier and Laurent Menard. *Application Integration*. iSTE, 2008.
- [54] James Martin. *Principles of Data Base Management*. Prentice Hall, Englewood Cliffs, N.J., 1976.
- [55] Hiroshi Maruyama, Kent Tamura, and Naohiko Uramoto. *XML and Java - Developing Web Applications*. Addison Wesley, 1999.
- [56] Daniel Moody and Peter Walsh. *Measuring the Value of Information Assets*. Simson Bowles - Associates, Melbourne, Australia, 1997.
- [57] William M. Newman and Robert F. Sproull. *Principles of Computer Graphics*. McGraw-Hill Book Company, 1997.
- [58] OASIS. *Reference Model for Service Oriented Architecture 1.0, Committee Specification*, 2006.
- [59] Computer Systems Laboratory of the National Institute of Standards and Technology (NIST). *IDEFØ Function Modeling Method*, 1993.
- [60] Robert Orfali and Dan Harkey. *Client/Server Programming with OS/2 2.0. 2ª ed.* Van Nostrand Reinhold, 1992.
- [61] T.L. Ramey, Brown R.R., Melkanoff M.A., and Rodríguez Ortiz G. *ELKA INFORMATION MODELS - Hughes Aircraft Company - University of California*, 1983.
- [62] Alfonso Rivero Cebrián. *Servidor Final de una Red de Cajeros con tecnología COM/MTS*. Facultade de Informática - Universidade da Coruña, 2001.

- 
- [63] D. F. Rogers. *Mathematical Elements for Computer Graphics*. McGraw Hill, 1976.
- [64] D. T. Ross. Structured analysis (sa): A language for communicating ideas. *IEEE Transactions on Software Engineering*, pages SE-3 (1), 1977.
- [65] J. Rumbaugh, M. Blaha, W. Premerlani, F. Eddy, and W. Lorensen. *Object-Oriented Modeling and Design*. Prentice Hall International Editions, 1991.
- [66] Arthur B. Sculley, W. Blaha, William, and A. Woods. *B2B Exchanges*. ISI, 1999.
- [67] Robert M. Solow. Growth theory and after. *The American Economic Review*, pages Vol 78 N°3, 307,317, 1988.
- [68] Steven H. Spewak. *Enterprise Architecture Planning*. Jon Wiley, 1993.
- [69] SpringSource. *SpringSource Spring Framework*, 2012. URL <http://www.springsource.org>.
- [70] Rina Yarmish and Joshua Yarmish. *Assembly Language Fundamentals*. Addison Wesley, 1979.
- [71] John A. Zachman. A framework for information systems architecture. *IBM Systems Journal*, pages Vol 6, N° 3, 1987.
- [72] John A. Zachman. Extending and formalizing the framework for information systems architecture. *IBM Systems Journal*, pages Vol 31, N° 3, 1992.
- [73] Rafael Zancan. *An easy-to-maintain Model-Driven Engineering Approach*. The Distributed Group, Universidad de Sevilla, 1991.



