

21

PROTEUS WEB: LA META INFORMACIÓN AL SERVICIO DE LA ADMINISTRACIÓN

Raúl Fernández Antoranz

Jefe de sección de sistemas informáticos. Responsable de proyecto
Subdirección General de Sistemas de Información. Dirección General de
Recursos Humanos, Servicios e Infraestructura. Ministerio de la Presidencia

1. SITUACIÓN ACTUAL

En la Subdirección General de Sistemas de Información del Ministerio de la Presidencia se plantea la siguiente situación:

- El **área de desarrollo** se define actualmente como un área con características propias que debe contar con personal especializado y dedicado en exclusiva al desarrollo de software para atender las crecientes demandas de las distintas unidades del Ministerio así como las demandas de otras unidades externas al propio Ministerio de la Presidencia a las cuales se debe dar servicio, ya sea debido a las funciones coordinadoras que ejerce nuestro departamento ministerial, sobre todo teniendo en cuenta que su titular viene ostentando desde hace ya algún tiempo los cargos de Vicepresidencia Primera del Gobierno y Portavoz del Gobierno, como a las solicitudes de colaboración que con frecuencia se producen (por ejemplo para actos singulares y de vital importancia para el desarrollo de la función de Gobierno y representatividad del Estado Español).
- El **personal** con que cuenta actualmente la Subdirección, como ocurre en muchos centros de la Administración, no es suficiente para cubrir la demanda actual. Esta situación no es ajena al área de desarrollo.
- Una singularidad de las demandas que recibe el área de desarrollo es que provienen de personas muy cercanas a la **función de gobierno** lo que significa que su resolución y rápida atención es en casi todos los casos prioritaria.
- En la Subdirección General de Sistemas de Información se cuenta desde hace largo tiempo, entre otras muchas aplicaciones, con un sistema de información que permite la consulta de multitud de bases de datos, con todo tipo de información. Entre dichas bases de datos se encuentran las bases de datos de actividad parlamentaria y legislativa, de publicaciones en el Boletín Oficial del Estado, de Seguimiento de las Comisiones de Secretarios de Estado y Subsecretarios y de Consejos de Ministros, etc. La aplicación a la que me refiero se denomina Proteus.

2. ¿QUÉ ES PROTEUS?

En los años 80 el Ministerio de la Presidencia disponía ya de un sistema de consulta de bases de datos con el objetivo de facilitar la consulta y gestión de la información almacenada en diferentes bases de datos. Dicho sistema se denominaba SIIPG y estaba soportado en un equipo propietario.

Posteriormente, entre los años 1997 y 1998, se procede a la migración del sistema de consultas SIIPG a un entorno Microsoft con repositorio de datos SQL Server. Pasa entonces a denominarse Proteus y con ese nombre ha llegado hasta nuestros días. Proteus está actualmente en producción y goza del reconocimiento de los usuarios de distintos departamentos ministeriales que lo utilizan.

El objetivo de Proteus es proporcionar un interfaz unificado de acceso a la información, tanto estructurada (almacenada en bases de datos) como no estructurada (ficheros externos a la base de datos – lo que podríamos denominar documentos –). Permite la consulta, visualización y manipulación de la información (inserción, modificación y borrado), así como la generación de salidas en distintos formatos.

El sistema Proteus está realizado en Visual Basic 6.0, con arquitectura cliente/servidor, disponiendo de un cliente muy pesado. Esta configuración de cliente pesado dificulta la implantación y despliegue de nuevas versiones de la aplicación. Sin embargo, este proceso de actualización de versiones está solventado de manera ingeniosa mediante la ejecución en cliente de dos programas que permiten actualizarse uno a otro. De este modo no es necesaria la presencia física de una persona para realizar dicha actualización y es el propio sistema el que al detectar una nueva versión de la aplicación actualiza la copia en el cliente. Por desgracia, esta solución no resuelve los problemas al 100% y la atención a usuarios por problemas en el cliente consume muchos recursos de tiempo y personal.

Proteus es una aplicación ampliamente conocida no sólo en el Ministerio de la Presidencia, sino también en el resto de departamentos ministeriales. Actualmente Proteus gestiona del orden de 180 bases de datos vivas. Es una aplicación conocida, alabada, criticada, pero sobre todo muy usada. Incluso el aspecto de las posibles críticas que reciba (y recibirá Proteus) no debe tomarse en sentido negativo. Es más, para mí no lo es; y no lo es porque no se puede criticar aquello que no se usa, y si Proteus se critica es porque se usa; y si se usa es porque es útil.

Y son los dos últimos puntos los que conforman la posición de partida de una nueva versión de 'nuestro' Proteus; quizás debería ser más correcto decir que conforman la posición de partida de un nuevo producto: Proteus Web.

Porque Proteus Web:

- Nace de Proteus cliente/servidor.
- Nace de la necesidad de actualización tecnológica de Proteus.
- Nace con la experiencia de Proteus, de lo bueno (de lo que se le alaba) y de sus carencias o mejoras (de lo que se le critica).
- Pero nace también con ansias de independencia. Quiere que todo aquel que confió en Proteus, que vio en Proteus un apoyo a su trabajo diario, encuentre lo mismo en Proteus Web. Pero a la vez quiere que todos aquellos que encontraron carencias, deficiencias o simplemente limitaciones en Proteus, las vean superadas en Proteus Web. Y para configurarse como solución de futuro a ambos perfiles se muestra como un producto distinto.

3. ANTECEDENTES EN PROTEUS

Para abordar el desarrollo de Proteus Web se ha realizado un concienzudo repaso de las funcionalidades que ofrece Proteus para a continuación incorporar las funcionalidades demandadas, a las que por un motivo u otro no se ha podido dar respuesta con el sistema actual.

FUNCIONALIDADES DE PROTEUS:

El actual sistema de información Proteus presenta las siguientes funcionalidades:

- **Consulta a bases de datos:** Proteus permite la consulta de información en una base de datos de entre todas aquellas a las que un usuario tenga acceso.
- **Consulta libre a bases de datos:** Se trata de una consulta por contenido en la que no se especifica el campo en el que se espera encontrar la información (o en el que se desea que aparezca) sino que es válida la aparición en cualquiera. Se dispone del comodín %,

que se puede traducir por ningún, uno o varios caracteres. A diferencia del tipo de consulta visto en el punto anterior, existen dos variantes de la búsqueda libre:

- *Búsqueda libre en una base de datos:* Esta era la única opción en el apartado anterior.
- *Búsqueda libre en todas las bases de datos* a las que el usuario tiene acceso.
- **Consultas predefinidas:** Se trata de consultas que desarrolla el equipo de Proteus, a petición del administrador de la base de datos.
- **Ordenación:** Proteus permite la ordenación de los resultados de una consulta con un criterio distinto al que presenta por defecto (el orden cronológico de introducción de la información en la base de datos).
- **Obtención de informes:** Proteus cliente/servidor permite la obtención de diversos informes:
 - *Informes estándar:* Son informes “por defecto”, es decir, están activos para cualquier base de datos. Presentan la salida en formato Microsoft Word. Permite guardar como favorito un determinado informe (con unos determinados campos).
 - *Informes personalizados:* Son informes hechos ad-hoc para una base de datos en concreto, a solicitud del administrador de la misma. El informe propiamente dicho se presenta en Microsoft Word o Microsoft Access, aunque existen también algunos informes que generan ficheros XML.
- **Gestión de “imágenes”,** es decir, de ficheros almacenados en el sistema de ficheros de uno o más servidores. Permite copiar desde una máquina cliente ficheros a un servidor, renombrarlos en el servidor, eliminarlos del servidor, vincularlos a un campo de la B.D.
- **Gestión de tablas de códigos:** Proteus maneja, como cualquier otra aplicación de gestión, unas tablas auxiliares o de códigos. Las tablas auxiliares que son propias de una determinada base de datos las mantienen los usuarios de esa base de datos con perfil para hacerlo. Las tablas auxiliares que, por su generalidad, afectan a multitud de bases de datos (por ejemplo, tabla de provincias) son mantenidas por la S. G. de Sistemas de Información.
- **Gestión del glosario de términos** por base de datos: Proteus crea y mantiene dinámicamente una lista de términos significativos que aparecen en el contenido de los campos de la base de datos. Esta glosario se utiliza para efectuar búsquedas en esos campos y para la búsqueda libre, haciendo uso también de la lista de palabras vacías (definida al igual que el glosario para cada base de datos).
- **Interfaces de consulta:** Proteus ofrece la posibilidad de elegir entre tres interfaces de consulta: consulta libre, consulta estándar y consulta avanzada
- **Interfaces de visualización:** Proteus ofrece la posibilidad de elegir entre tres interfaces de visualización, permitiendo además personalizarlos en cuanto a los campos de salida que el usuario quiere que le muestre: presentación en tabs o pestañas, presentación en árbol, presentación clásica, presentación navegador y presentación naveg-tabs.
- **Actualización (gestión) de la información:** Proteus facilita la operativa de altas, bajas y modificaciones de información. Permite la actualización uno a uno (presenta la posibilidad de configurar por usuario el formulario que permite esta operativa), así como la actualización masiva de información. La última posibilidad debe manejarse con mucho

cuidado pues una operación inadecuada puede llegar a causar un desastre en los contenidos de la base de datos.

- **Configuración por usuario y base de datos:** Proteus permite configurar, para cada usuario y base de datos, ciertos aspectos, tales como interfaces de consulta y visualización preferidos, formularios de actualización, consultas, ordenaciones e informes preferidos.

PECULIARIDADES DE PROTEUS:

Dos de las características fundamentales de Proteus cliente/servidor son:

- **Visión jerárquica:** Proteus se basa en una visión jerárquica de la realidad. Toda la información tiene una estructura jerárquica en una determinada base de datos, por lo que siempre será posible localizar al “padre” de la información buscada.
- **Estructura (metainformación):** Proteus logra llevar a cabo sus funcionalidades y gestionar una multiplicidad de bases de datos gracias a que se apoya en una tabla que se denomina “estructura”. Dicha tabla no contiene otra cosa más que metainformación, es decir, información acerca de la información recogida en la base de datos.

4. OBJETO DEL PROYECTO PROTEUS WEB

Proteus es un dios griego del mar que tenía la virtud de cambiar de formas, por lo que en ocasiones se le denominaba como el dios de las mil formas. Como la divinidad griega, Proteus Web pretende mostrar muchas apariencias gestionadas por un único aplicativo (por un código común).

Proteus Web nace inicialmente de la idea de abordar la migración a entorno Web del sistema de información Proteus para el manejo de bases de datos.

Aprovechando dicha migración se va a flexibilizar todo el sistema, ampliando las capacidades y funcionalidades actuales.

El nuevo sistema debe ser compatible con el actual Proteus cliente/servidor y con las bases de datos bajo él definidas. Para asegurar esta compatibilidad se proveerán los mecanismos necesarios, aunque no se desecha la posibilidad de realizar la migración de las estructuras actuales de metadatos a las nuevas de Proteus Web.

5. ENTORNO TECNOLÓGICO

Se ha escogido como plataforma de desarrollo y explotación la plataforma .NET de Microsoft. El desarrollo se realizará en ASP .NET. Como lenguaje de script para la ejecución en cliente se adoptará JScript .NET. El lenguaje para el procesamiento de las páginas ASP en servidor será Visual C# .NET.

Las bases de datos se almacenarán en Microsoft SQL Server, aunque se tratará de que el sistema gestor de bases de datos (en este caso SQL Server) no determine ni limite la funcionalidad del sistema de información.

No se requerirá ningún software en el cliente, a excepción claro está del navegador que éste utilice para visualizar las páginas Web. En principio y, salvo que se encuentre dificultad en ello (lo

que no debería ocurrir al utilizar la plataforma Microsoft .NET), no se restringirá la explotación de la aplicación a la utilización de un determinado navegador.

Está por determinar el formato de salida de los informes. En cualquier caso, el informe se generará en el servidor, sin necesidad de ningún software en cliente. Se permitirá asimismo la obtención de los informes en formato HTML y XML. Se está evaluando el generador de informes entre Crystal Reports o Microsoft SQL Server Reporting Services.

6. EQUIPO DE DESARROLLO DE PROTEUS WEB

Para el desarrollo de Proteus Web se ha optado por un equipo mixto con participación de personal interno y externo, pero con el control en manos de personal de la S. G. de Sistemas de Información. De esta manera se busca que el conocimiento de la aplicación resida siempre en personal propio, aspecto que consideramos muy importante para la fase de mantenimiento.

7. METAS DE PROTEUS WEB

Los objetivos que se quieren cubrir con Proteus Web son:

- **Migración** del actual sistema de información Proteus cliente/servidor a entorno Web.
- **Superación de las limitaciones** que impone el actual sistema.
- **Usabilidad** del nuevo sistema: Uno de los retos al abordar Proteus Web es alcanzar al menos la misma aceptación que tiene en la actualidad Proteus cliente/servidor. Se va a hacer hincapié en la usabilidad del sistema de cara al usuario final. Para ello se buscará la parametrización al máximo de todas las posibilidades de Proteus Web, haciendo un sistema configurable en el mayor porcentaje posible. Con esta idea, se facilitará, entre otras cosas, la personalización de la interfaz.
- **Rapidez** en la obtención de resultados de una consulta: Esta es una de las claves del éxito de Proteus cliente/servidor.
- Mejorar la **gestión** y el **control** del sistema en producción: Como se ha indicado Proteus nace de un sistema propietario, SIIPG, y ha ido evolucionando al tiempo que se han ido incorporando nuevas bases de dato y usuarios. Este crecimiento, quizás mayor del esperado, hace que las tareas de gestión y control sean más difíciles, al no estar preparado inicialmente para la situación presente. Al abordar Proteus Web se debe prestar mucha atención a todas las labores de administración, gestión y control de la actividad del sistema, usuarios, errores, ...
- Incrementar la **seguridad**.
- Sistema más **flexible** que permita una mejor adaptación a las necesidades presentes y futuras de los usuarios.
- Incorporación de **nuevas funcionalidades**, ampliamente demandadas. Entre ellas:
 - **Búsqueda en documentos externos** a la base de datos (información no estructurada).
 - **Perfiles de usuario**.
 - Más protagonismo al **propietario** y **administrador** de cada base de datos.
 - Ordenaciones de recuperación de datos.

- Posibilidad de **firma electrónica**: La necesidad de incorporación de certificados de usuario y firma electrónica se puede resumir en “adaptarse o morir”. No sería concebible abordar la nueva aplicación sin tener presentes estos aspectos.
- **Eliminar las necesidades de instalación en cliente**. Al mismo tiempo se facilita el acceso al sistema, ya que se puede acceder desde cualquier ordenador desde el que el usuario se pueda validar y autenticar en el sistema, sin otro requisito que tener instalado un navegador Web.
- **Generación de aplicaciones** soportadas en Proteus Web. Esta es la parte más ambiciosa del proyecto. Como las aplicaciones de gestión, o gran parte de ellas, se basan en altas, bajas, modificaciones y consultas, junto con alguna regla más o menos sencilla o más o menos compleja (por ejemplo un botón que ejecute una acción más compleja que una simple consulta o actualización de datos), se plantea la posibilidad de, en base a metainformación, poder parametrizar estos aspectos “variables” de las aplicaciones de gestión. ¿No es posible generar “en caliente” una presentación diferente para distintas aplicaciones sencillas de gestión?
- **Optimización de recursos** en el grupo de trabajo de Proteus Web de modo que se pueda dar el mayor servicio a nuestros usuarios con el mínimo coste (en tiempo de desarrollo y en mantenimiento).

8. FUNCIONALIDADES DE PROTEUS WEB

Proteus Web recoge todas las funcionalidades de Proteus, con el objetivo de ampliarlas, añadiendo, matizando o introduciendo los siguientes puntos:

- **Definición de un lenguaje estándar de consulta** que maneja el usuario y que le permite interrogar a Proteus acerca de la información almacenada en cualquiera de sus bases de datos. Proteus ofrece al usuario un lenguaje de consulta para que éste realice preguntas al sistema. A la vista de la experiencia obtenida con Proteus, de impresiones obtenidas en los cursos de formación (en los que el feedback de los alumnos hacia los profesores constituye una inestimable fuente de opinión), y de análisis exhaustivos realizados por el propio personal de desarrollo que conforma el equipo de Proteus y de Proteus Web, se ha redefinido el lenguaje de consulta de modo que:
 - Sea preciso y no ambiguo.
 - Sea sencillo y fácil de aprender.
 - Usuario y sistema se entiendan perfectamente en base a muy escasas y sencillas reglas de escritura, de manera que no haga falta saber nada de ordenadores ni de bases de datos ni mucho menos de organización de la información. Lo único que se debe conocer es en qué universo de conocimiento nos movemos y que información se va a manejar.
- **Definición de perfiles** entre los usuarios que acceden a la información de una determinada base de datos. Se consolida y se potencia así la gestión de los permisos de usuarios en función de su pertenencia a determinados roles o perfiles. Se definen tres tipos de roles de usuarios:

- **Rol de administrador:** Es el usuario administrador de una determinada base de datos. Este rol, si bien está ya definido en Proteus, no está implementada la funcionalidad específica asignada a él. En Proteus Web vamos a implementar completamente dicha funcionalidad, de modo que el administrador pueda realizar ciertas tareas sin depender del personal de informática. De esta manera pretendemos conseguir dos objetivos:
 - Delegar en el administrador ciertas tareas rutinarias de gestión (se entiende gestión, no técnica, pero sí de contenidos), con lo que la persona que ejerza este rol se sentirá más implicado con la aplicación, a la vez que con más responsabilidad sobre la labor que realiza.
 - Eliminar el cuello de botella que en picos de trabajo se puede producir en el personal de la S. G. de Sistemas de Información, tanto de desarrollo como de atención a usuarios.
- **Rol de usuario de actualización:** Se trata de usuarios que, sin tener los permisos que el administrador, sí van a poder actualizar información en la base de datos.
- **Rol de usuario de consulta:** Se trata de usuarios cuya actividad con la base de datos va a estar limitada a la extracción de información. Dentro de este rol se incluyen los que podemos denominar usuarios “invitados” a participar de los contenidos almacenados en la base de datos.
- **Establecimiento de políticas de seguridad.** ACID (Autenticación, Confidencialidad, Integridad, Disponibilidad). Con un doble objetivo:
 - Poder definir niveles de seguridad independientes para cada base de datos, perfectamente integrados en Proteus Web. Como ya se ha mencionado, Proteus tiene la particularidad de ofrecer acceso mediante un mismo aplicativo a multitud de bases de datos. Como es lógico la información contenida en todas esas bases de datos es de lo más dispar, como también lo son los requerimientos de seguridad de cada una de ellas. Es por ello que se pretende en Proteus Web gestionar de manera flexible el nivel de seguridad alcanzado para cada base de datos. Evidentemente, el punto máximo de seguridad lo demandan las bases de datos que puedan contener o llegar a contener datos de carácter personal, protegidos por la Ley Orgánica de Protección de Datos. Proteus Web debe ser capaz de garantizar los niveles de seguridad requeridos en estos casos.
 - Poder gestionar los niveles de seguridad definidos: De nada nos sirve adaptarnos a todos los niveles de seguridad necesarios en cada momento si no se puede realizar una gestión eficaz de dicha seguridad.
- **Personalización de la interfaz de Proteus Web,** tanto a nivel de interfaz de consultas como de actualización de información. Proteus en su versión cliente/servidor ya recoge esta posibilidad, que ahora se consolida y amplía. Para quienes no conozcan el actual sistema de información, baste decir que presenta la posibilidad de manejar tres interfaces diferentes de introducción de la consulta en Proteus y cinco interfaces distintos de presentación de los resultados. Además, cada usuario puede guardar su configuración preferida, es decir la combinación de interfaz de consulta y de resultados que prefiera, para cada base de datos (un mismo usuario puede tener personalizado una combinación de interfaces en una base de datos y otra diferente en otra base de datos). Además, se pue-

den configurar los campos de salida que un usuario quiere ver (de entre los que tenga permiso). Se establecen varios tipos de personalización de interfaz:

- Personalización de la interfaz de Proteus Web para *cada base de datos*: Se podrán personalizar ciertos aspectos de la interfaz por base de datos. Por ejemplo, si entramos a consultar información de la base de datos de información parlamentaria podríamos tener como fondo de pantalla una imagen del Congreso de los Diputados; en cambio si accedemos a la base de datos de Biblioteca podríamos ver de fondo una imagen de estanterías con libros. Parece una tontería sin ningún sentido, salvo el “que bonito” que algunos podrían exclamar. Sin embargo, consideramos que es útil, porque se trata de un impacto visual que advierte al usuario casi de manera inconsciente dónde se encuentra. Hay que tener presente que si bien Proteus muestra en todo momento el nombre de la base de datos en la que el usuario está trabajando mediante la aparición del nombre de la misma en un lugar determinado de la pantalla, lo habitual es que nadie, ni nosotros mismos, el personal de desarrollo, se fije en él. Este hecho, unido a la similitud de campos que presentan muchas bases de datos (por ejemplo, dos bases de datos de registros, cuyos campos para la gestión del registro correspondiente son prácticamente los mismos, incluso con el mismo nombre) y a que un mismo usuario puede tener acceso a multitud de bases de datos, creemos que justifican la personalización del interfaz, de cara a incrementar la usabilidad del sistema y la satisfacción de los usuarios, así como a minimizar la posibilidad de error durante el trabajo con el sistema.
- Personalización de la interfaz de Proteus Web para *cada usuario*: Se trata de recoger el testigo de Proteus en lo relativo a los interfaces preferidos por cada usuario (incluyendo el detalle de campos mostrados).
- **Incorporación de mecanismos de firma electrónica y certificados digitales de usuario.** La tendencia actual a la utilización de certificados de usuario para acceder a la información y/o los servicios hace que se recoja en Proteus Web esta posibilidad. Lo mismo ocurre con la firma electrónica.
- **Incorporación de funciones de gestión documental.** Incluye:
 - Búsquedas en *documentos externos* a la base de datos, es decir, búsqueda en información no estructurada. Actualmente, mediante Proteus se gestiona y consulta tanto información estructurada (almacenada en bases de datos) como información no estructurada (almacenada en ficheros externos al S.G.B.D.). Ambas posibilidades se gestionan de manera unificada con Proteus. Sin embargo, carece de la posibilidad de búsqueda en el interior de documentos. Esto provoca, por ejemplo, que en una base de datos como la de disposiciones publicadas en el Boletín Oficial del Estado se pueda buscar por el contenido de los campos de catalogación de las disposiciones, pero no en el texto exacto y completo de la disposición. Ésta es una demanda ampliamente reivindicada por algunos usuarios de Proteus y echada de menos, aunque no solicitada explícitamente, por muchos otros. Conscientes de ello, Proteus Web recogerá esta funcionalidad. Está por determinar cómo se plasmará esta capacidad documental, y estamos en fase de evaluación de las distintas alternativas para llevarlo a cabo, siempre con la con-

dición de integrar la búsqueda documental dentro de Proteus Web y no de manera inconexa. Las posibilidades analizadas son:

- Integración con un API de desarrollo de algún gestor documental, de modo que bajo el control de nuestro sistema se realicen llamadas al motor de búsquedas, recogiendo los resultados, quizás incluso con la necesidad de un procesamiento posterior de la salida obtenida.
- Desarrollo dentro de Proteus Web de un motor de búsqueda ligero, haciendo uso del ingenio (para nada escaso) de los integrantes del equipo de desarrollo de Proteus Web. Somos conscientes que con esta alternativa no podremos alcanzar el nivel de funcionalidad que nos proporcionaría un gestor documental, pero esto mismo es lo que se ha realizado en Proteus cliente/servidor (con ciertas limitaciones, por supuesto) y los resultados en cuanto a potencia de búsqueda y rendimiento son notables y así lo reconocen las personas que trabajan habitualmente con el sistema.
- Búsquedas en información almacenada en *campos de base de datos*, es decir, búsqueda en información estructurada. Se añade a la búsqueda facilitada por un gestor de bases de datos, la búsqueda documental (por ejemplo, contemplando distintas formas verbales, sinónimos, proximidad entre términos, ...). La viabilidad o no de incorporar esta funcionalidad depende de la opción seleccionada (como se ha comentado en el punto anterior).
- Incorporación de la funcionalidad proporcionada por un *tesauro*, con la posibilidad de búsqueda en función de términos relacionados en el tesauro. En Proteus cliente/servidor se recogen varias bases de datos que conforman sendos tesauros, algunos de ellos muy bien diseñados y mantenidos por los responsables de los mismos. Sin embargo, en las consultas Proteus no ofrece la potencia suficiente como para explotar y sacar partido a la información almacenada en dichos tesauros. De nuevo, si se opta por integrar en Proteus Web un gestor documental, con una alta probabilidad éste ya ofrezca la posibilidad de definición, mantenimiento y explotación de tesauros. En otro caso (el del desarrollo propio) deberemos plantearnos la posibilidad de ampliación del lenguaje de consultas de modo que permita a un usuario avanzado en la materia la ejecución de comandos que permitan utilizar los tesauros.
- **Visión relacional del mundo:** Proteus cliente/servidor impone una restricción a las bases de datos que maneja: su estructura debe obedecer a una disposición jerárquica de la información. En sus orígenes toda la información que albergaban las bases de datos que debía gestionar Proteus presentaban esta estructura jerárquica. Con el paso del tiempo empezó a plantearse la necesidad de dar cobertura a bases de datos con estructura relacional; en ocasiones, cambiando ligeramente el enfoque de la información y realizando algunas adaptaciones, dichas bases de datos podían acomodarse a una visión jerárquica. En otras ocasiones, era imposible alejarse del enfoque relacional. En estos últimos casos, o bien no era posible dar soporte con Proteus a la necesidad planteada, o bien había que dar una solución poco o nada natural para la gestión de la información. En Proteus Web, se modifica en paradigma de modo que se adopta una visión relacional del mundo. Así se puede dar solución tanto a modelos jerárquicos de la información como a

modelos relacionales. El planteamiento de la estructura relacional presenta no pocos problemas, tanto de control como de usabilidad de la aplicación de cara al usuario final.

- **Incorporación de estándares** en el desarrollo de Proteus Web. Se quiere que el desarrollo de Proteus Web se adecue a una serie de estándares, tanto en lo relacionado al proceso como al producto:
 - *Estándar del área de desarrollo de la Subdirección General de Sistemas de Información del Ministerio de la Presidencia*: Se ha definido un estándar en el área de desarrollo y se pretende con Proteus Web evaluar las ventajas y/o inconvenientes del mismo detectando puntos débiles y replanteándolo allí donde sea necesario.
 - *Estándar de páginas HTML*: Las páginas generadas por la aplicación y presentadas al usuario deben cumplir el estándar XHTML 1.0
 - *Estandar de Cascade Style Sheets 2.0 (CSS 2.0)*.
 - *Normas WAI de accesibilidad*: Las páginas deben adecuarse igualmente a las normas WAI (Web Accessibility Initiative) sobre accesibilidad, es decir, conformidad con las Directrices de Accesibilidad para el Contenido Web 1.0 (WCAG 1.0).
- *Incorporación de auditorias de información*: Nos referimos con ello al típico fichero de log de toda la vida, es decir, un lugar donde registrar la actividad que se lleva a cabo con las bases de datos. Se trata de que se pueda conocer quién ha realizado qué acción en el sistema e incluso cuándo lo hizo.
- Informes.
- **Ordenaciones de información**: Proteus cliente/servidor ofrece actualmente la posibilidad de ordenar la información de salida e incluso de guardar las ordenaciones preferidas del usuario. Pero no permite configurar por defecto para cada usuario una ordenación concreta, con posibilidad de cambiarla, de modo que no deba especificarla de nuevo (o recuperarla) al iniciar una nueva sesión en el sistema. Proteus Web va a ampliar las posibilidades ofrecidas al usuario final, tratando de optimizar el rendimiento tanto de consultas con y sin ordenación.
- **Generación de aplicaciones sencillas de gestión** de información, mediante la simple definición de la misma y de sus formularios en un catálogo de aplicaciones gestionado por Proteus Web.
- **Gestión eficaz de errores** producidos durante la explotación del sistema. Se asegurará no interrumpir el servicio ante un error inesperado durante la ejecución de la aplicación.
- **Definición de un sistema de alertas automáticas**: Se recoge así la idea de que un sistema de información es útil no sólo cuando el usuario interactúa con el sistema, sino cuando a su vez el sistema interactúa con el usuario (es el “principio del despertador”: siempre que le mire me va a proporcionar información sobre la hora que es, pero me sirve de poco si no suena la alarma a una determinada hora).
- **Ayuda**: Se proveerá de un completo sistema de ayuda on-line.
- Se debe asegurar el mismo éxito que en Proteus cliente/servidor en cuanto a la rapidez en la obtención de respuestas por parte del sistema. Debido al cambio de entorno tecnológico (ahora estamos tratando con una aplicación web) no es posible continuar con el ingenioso mecanismo de consulta de Proteus cliente/servidor. Se ha desarrollado un sis-

tema estándar de generación de consultas a las bases de datos que sea rápido, aprovechando al mismo tiempo para optimizar las ordenaciones de información que eran un punto débil de Proteus.

9. DIRECTIVAS DE DESARROLLO

Tal y como se recomienda a la hora de programar ASP .NET se separará la lógica de proceso de la página de la presentación de la misma, de modo que no se mezcle en el mismo fichero .aspx código de presentación y lógica de proceso.

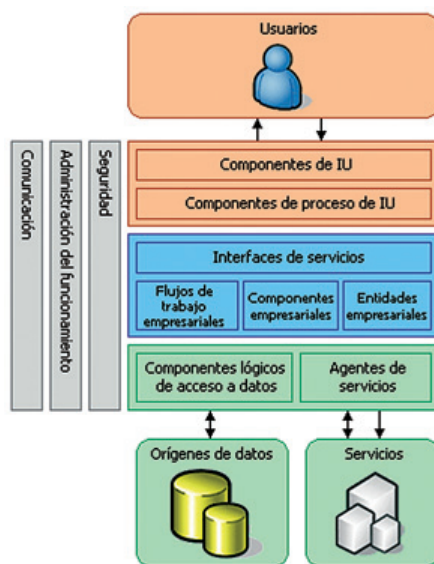
Se seguirá el estándar de nomenclatura propuesto para el área de desarrollo, consolidando dicho estándar, revisándolo y completándolo según se considere necesario a lo largo del proceso de desarrollo. Esta consolidación es importante debido a que éste será el primer proyecto que siga íntegramente las directrices marcadas por el estándar, y por tanto permitirá valorar las ventajas o inconvenientes del mismo.

Los objetos creados durante el desarrollo deberán llevar documentación embebida en los mismos (de nuevo siguiendo el estándar indicado). Al utilizar Visual C# .NET para la implementación de la lógica de proceso dicha documentación se incluirá de modo que se pueda utilizar la generación de documentación en soporte XML que el propio lenguaje proporciona.

10. ARQUITECTURA

El desarrollo de Proteus Web se basa en:

- **Estructuración de la aplicación en capas:** capa de interfase de usuario, capa de lógica de negocio, capa de acceso a datos. Se adopta el modelo en capas que propone Microsoft para su plataforma .NET adaptándolo a las necesidades propias de la organización.

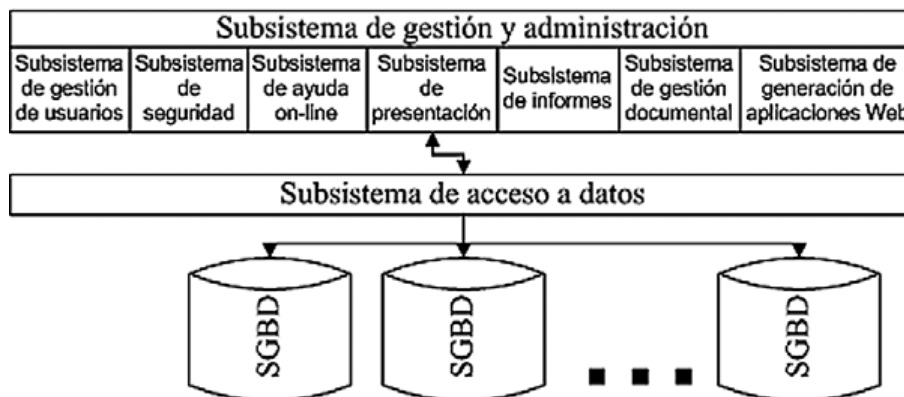


Arquitectura de capas en .NET (Fuente: Microsoft)

- El **paradigma de orientación a objetos** (clases, herencia, polimorfismo, ...).
- La utilización de **metainformación**.
- El uso de **componentes**.

De este modo se busca:

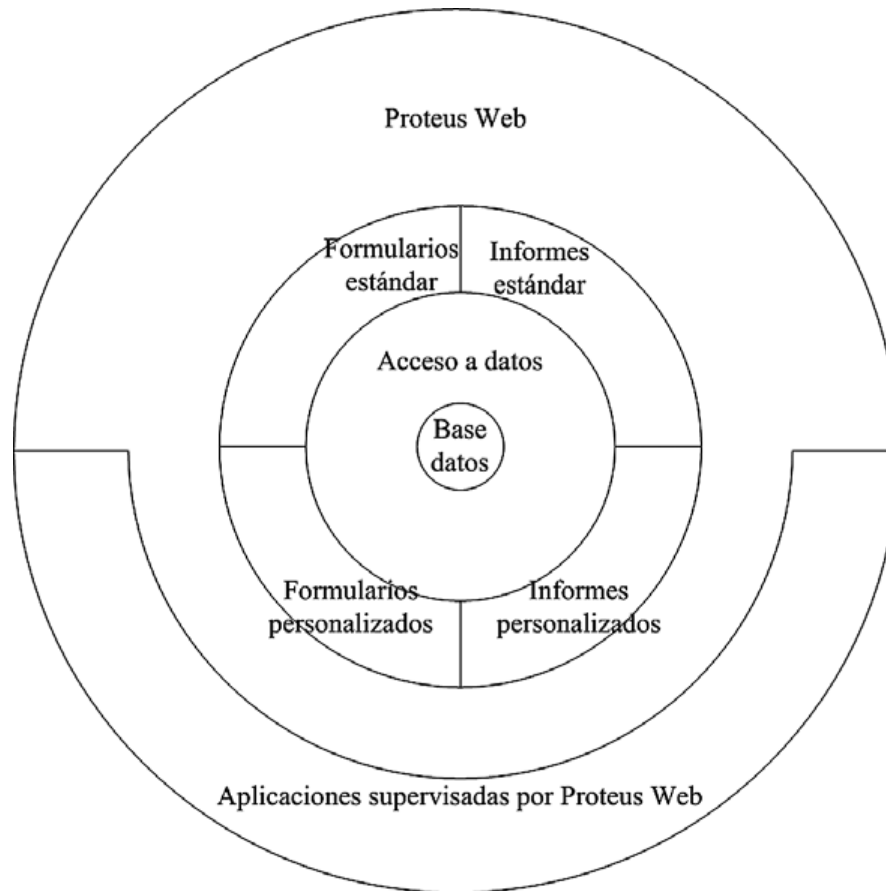
- **Independencia entre capas:** Lograr una independencia lo mayor posible entre las capas (siempre determinada por el rendimiento obtenido).
- **Reutilización de código.** Se diseñará todo el sistema con *orientación a componentes*, de forma que estos sean lo más reutilizables posible. Se trataría de trabajar con la visión de una factoría de software. El punto en contra de este planteamiento es la escasez de recursos de personal y que se necesita la implicación del mismo. Pero a largo plazo permitirá reducir el tiempo de desarrollo de nuevos sistemas, ya que una aplicación se construirá mediante el ensamblado de componentes existentes, de manera que éstos estén probados y sean fiables. Además reducirá el coste de mantenimiento, ya que un error en el funcionamiento de un componente se resolverá una vez y quedará arreglado en todos los aplicativos que utilicen ese componente.
- **Estandarización en el desarrollo.** En una ocasión alguien dijo que si le dabas un esqueleto de código donde rellenar una serie de puntos concretos, le quitabas la parte creativa de su trabajo. Y no le faltaba razón; y el avance en la informática se debe a la creación de unos u otros. Pero el esqueleto de código permitía una estandarización importante en el grupo de trabajo, de modo que fuese más sencillo de entender por todos los miembros del equipo; y a efectos prácticos se demostró rentable en la producción de programas que tenían un comportamiento y funcionalidad similar (por ejemplo, los programas de gestión de tablas auxiliares o de generación de listados en papel de la información que contienen). El tiempo de desarrollo se redujo drásticamente. Un paso más en esta línea nos la ofrece el desarrollo *orientado a componentes*. Los componentes tendrán una interfaz definida y realizarán una función concreta, que es lo que nos interesa. Se trata de potenciar el concepto de caja negra. Mientras no cambie el “qué hace” el componente, me da igual el “cómo lo hace”.



Visión parcial de la arquitectura de Proteus Web

Proteus Web presentará la siguiente arquitectura funcional, considerándose varios subsistemas:

- **Subsistema de presentación:** Se encargará de todas las funciones de presentación de la información al usuario. Tiene tanto elementos de interfaz de usuario como de lógica de negocio por lo que en la división en capas tendrá componentes en ambas.
- **Subsistema de acceso a datos:** Proporcionará las funciones necesarias para acceder a los datos (lectura, recuperación, almacenamiento, modificación), facilitando independencia del gestor de base de datos que haya por debajo. Con ello se permitirá la portabilidad de la aplicación. El objetivo es que Proteus Web pueda trabajar con varios proveedores de datos distintos (por ejemplo, Microsoft SQL Server y Oracle) de manera transparente para el usuario e incluso para las capas de interfaz de usuario y de lógica de negocio. Pero el subsistema de acceso a datos depende en gran medida de la metainformación almacenada. Por ello, existirán componentes relacionados con el subsistema de acceso a datos en la capa de acceso a datos y en una nueva capa que se situaría por encima de ella para gestionar la metainformación que dotará de flexibilidad y adaptabilidad a Proteus Web
- **Subsistema de seguridad:** Controlará todo lo relativo a usuarios y gestión de accesos. Proporcionará independencia del modo de seguridad elegido (por ejemplo, directorio activo, base de datos, ...) y dónde resida la misma. En la estructura de capas, la capa de seguridad es ortogonal a las capas de interfaz de usuario, negocio y acceso a datos.
- **Subsistema de informes:** Gestionará todo lo relativo a gestión de informes. Proporcionará independencia con el motor de generación de informes que se elija.
- **Subsistema de gestión y administración de Proteus:** Permitirá obtener estadísticas de uso, de errores producidos, tiempos medios de resolución de errores, tiempo medio entre fallos, ...
- **Subsistema de gestión documental:** Ya se ha comentado anteriormente. Está por determinar el alcance de este subsistema de gestión documental, ya que depende de la solución elegida y de la funcionalidad que la misma aporte.
- **Subsistema de reglas:** Contiene la semántica de cada una de las bases de datos. A través de él se podrán definir reglas de validación de datos y de derivación de unos datos en función de otros. Estas reglas ampliarán la potencia de los metadatos almacenados.
- **Subsistema de control de errores:** Gestiona la captura y manejo de errores que se puedan producir durante la explotación de la aplicación. Se debe llevar un log que permita hacer un seguimiento de los errores que se puedan ir produciendo en el sistema y poder obtener estadísticas e informes. Además, el sistema será tolerante a fallos de modo que se capture la ocurrencia de un error en ejecución, pero no se cancele la ejecución de la aplicación. Se trata de eliminar la ocurrencia de un 'error fatal' que causa la terminación anormal de la ejecución. Para ello, se grabarán las preferencias no permanentes (aquellas cuya vida no se extiende más allá de la sesión actual de usuario) en tablas auxiliares de base de datos. Al utilizar .NET se pueden almacenar los parámetros de sesión en otra máquina diferente para en caso de caída poder restablecer el servicio conservando los datos de la sesión del usuario.
- **Subsistema de ayuda:** Se debe proporcionar un sistema completo y eficaz de ayuda on-line

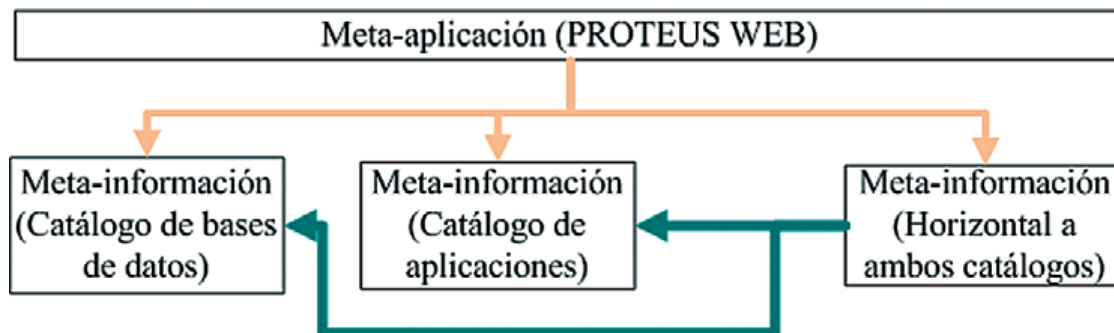


Visión global de la organización funcional de Proteus Web

Queda por comentar brevemente algo acerca del concepto que, unido al desarrollo orientado a componentes, permitirá la flexibilidad y capacidad de adaptación de Proteus Web a los cambios: la **metainformación**. Desde este punto de vista Proteus Web se configura como una metaaplicación que pretende ser capaz de gestionar:

- Un catálogo de bases de datos: Esta gestión se hará mediante metainformación acerca de las mismas.
- Un catálogo de aplicaciones de gestión: Igualmente debe hacerse en base a metainformación.
- Metadatos adicionales, horizontales a ambos catálogos.

Esta concepción de metainformación o metadatos es además un medio de independizarnos de la plataforma tecnológica (con el apoyo de la arquitectura en capas), unida a la potencia de reutilización y optimización de recursos que nos permiten los componentes conforma la base conceptual en la que se apoya Proteus Web. Como se dijo al comienzo, Proteus ya se basa en metainformación pero el diseño de aquellos “repositorios de metadatos” se ha quedado obsoleto y a día de hoy se muestra insuficiente para las demandas de los usuarios por un lado, de las normativas por otro y de la propia evolución de la tecnología.



La metainformación como elemento de Proteus Web

Definimos Proteus Web como una meta-aplicación porque:

- Es una aplicación ya que ofrece al usuario una serie de interfaces propios y una serie de funcionalidades estándar.
- Es una meta-aplicación porque, además de ofrecer funcionalidades e interfaces estándar, gestiona un catálogo de aplicaciones de gestión que corren bajo la supervisión de la meta-aplicación.

El catálogo de bases de datos contiene toda la meta-información sobre las bases de datos que gestiona Proteus, de modo que:

- Se disponga de toda la información de cada una de las bases de datos, tablas, campos, relaciones, restricciones,... en un catálogo independiente del proveedor de datos concreto.
- Se pueda proporcionar independencia del nivel de tecnología del proveedor de datos.
- Se puedan gestionar operaciones sencillas (y no tan sencillas) sobre los datos.
- Se proporciona independencia de los tipos de datos físicos que ofrece el proveedor de datos a través del catálogo. Para ello se definirán unos tipos de datos propios a nivel de catálogo que tendrán unas reglas de validación de formatos, contenidos, representación,... Esta información (los tipos de datos del ámbito de Proteus Web) se definirán en un repositorio de meta-información que se dispone horizontalmente a ambos catálogos.

El catálogo de aplicaciones (de gestión sencillas) contiene toda la meta-información sobre las aplicaciones que supervisa Proteus, de modo que:

- Se maneje la información necesaria para mostrar dinámicamente la interfaz de las aplicaciones.
- Se integre con el catálogo de bases de datos: Una aplicación de gestión accederá a una o varias bases de datos de nuestro sistema para consultar o modificar información y en ese punto utilizará la información que le proporcione el catálogo de bases de datos.

