

# MELSIS: un marco de trabajo para la construcción de Sistemas de Ayuda a la Toma de Decisiones en problemas de monitorización

Picaza J. M.\*, Sobrado F.J.\*, García J.J.\*, Ocariz C.\* y Aldamiz-Echevarria L.<sup>+</sup>

**Grupo ERABAKI**

Laboratorio de Informática Médica

\* Departamento de Lenguajes y Sistemas Informáticos. Facultad de Informática (UPV/EHU). Donostia-San Sebastián.

<sup>+</sup> Hospital de Cruces. Osakidetza-Servicio Vasco de Salud. Baracaldo (Bizkaia)

**Palabras clave:** Sistemas de Ayuda a la Toma de Decisiones, Aplicaciones Web, Ontologías, Tecnología Java.

## RESUMEN:

Los Sistemas de Ayuda a la Toma de Decisiones (DSS) son de gran interés para las organizaciones pero hay una serie de aspectos que hacen que su construcción conlleve una gran complejidad, son los siguientes: 1) la cantidad de variables y su interrelación; 2) la calidad requerida en el diseño de bases de datos, modelos, sitio Web de acceso; 3) construir o disponer de módulos y componentes web; 4) las necesidades de gestión del proyecto que su tamaño exige. Para superar dichas dificultades es imprescindible disponer de una infraestructura más poderosa que la de cualquier entorno de trabajo. MELISIS pretende ser un marco de trabajo flexible que, además de integrar las partes comunes a diferentes casos de monitorización, esté preparado para el cambio y permita llevar una gestión tan concienzuda como las exigencias de las aplicaciones requieran.

## 1. Introducción

La toma de decisiones concernientes a sistemas complejos a menudo supera nuestras capacidades cognitivas debido a la cantidad de variables involucradas y a sus sutiles interdependencias. También es conocido que el juicio intuitivo humano y su toma de decisiones está lejos de ser óptimo, viéndose deteriorado con la complejidad y el estrés. Debido a la importancia que las decisiones de calidad tienen en muchas situaciones, ofrecer una ayuda que mitigue las deficiencias anteriores constituye una de las líneas principales de la ciencia a lo largo de la historia. Disciplinas como la estadística,

economía y la investigación operativa han desarrollado varios métodos para la realización de elecciones racionales. Más recientemente, estos métodos, se han visto potenciados por aportaciones de la informática, psicología e inteligencia artificial en forma de programas de ordenador, tanto como herramientas aisladas como entornos integrados para la toma de decisiones complejas o Sistemas de Ayuda a la Toma de Decisiones (DSS). El concepto de DSS es extremadamente amplio y su definición varía dependiendo del punto de vista del autor. Para evitar la exclusión de cualquiera de los tipos de DSS existentes, podríamos definirlo como un sistema informático interactivo que ayuda a los usuarios a juzgar y decidir sobre actividades.

Los DSS han ganado popularidad en ámbitos como los económicos, de ingeniería, militares y médicos. Esto es así por su utilidad para ayudar a la toma de decisiones precisas y óptimas a partir de grandes cantidades de información: proveyendo de diferentes fuentes de información, acceso inteligente al conocimiento relevante, estructuración del proceso de decisión o dando una aproximación heurística a problemas intratables formalmente. La apropiada aplicación de herramientas de ayuda a la decisión incrementa la productividad, eficiencia, efectividad y competitividad. Como consecuencia, haciendo más segura la planificación, organización e inversión.

Visto lo anterior conviene hablar de cómo la ayuda a la toma de decisiones puede ser

soportada por un programa informático. En la sección de antecedentes haremos una más detallada introducción en el problema señalando las dificultades de la toma de decisiones y la necesidad del uso de modelos, los componentes con que cuenta un DSS, los DSSs de análisis de decisiones, la importancia de las interfaces de usuario y el impacto de la Web, y finalmente las ventajas de estándares como XML y RDF. En la sección tercera, revisaremos algunas soluciones existentes en el dominio de la gestión de proyectos y los intentos de construcción de herramientas que constituyen marcos de trabajo útiles en cualquier dominio, en éstos últimos prestaremos una especial atención a la tecnología empleada. Por último, en la sección cuarta, presentaremos las características de MELSI (Monitorizazio Erabakietan Laguntzeko SIstemen Sortzailea / Generador de Sistemas de Ayuda en las Decisiones de Monitorización) como marco de trabajo (*framework*) para la construcción de DSSs para problemas de monitorización.

## **2. Antecedentes.**

### **2.1 Las decisiones y su modelado.**

Decidir significa elegir entre varias alternativas. Para dar una solución más completa, es conveniente construir las alternativas (con una clara presentación de las mismas y su justificación), indicar que se necesita una decisión, e incluso, señalar la elección más recomendable justificándola adecuadamente.

#### **2.1.1 El juicio humano y la toma de decisiones.**

Estudios teóricos sobre toma de decisiones racionales, principalmente en el contexto de la teoría de la probabilidad y la teoría de la decisión, han ido acompañadas de investigación empírica sobre si el comportamiento humano se ajusta a la teoría. Numerosos estudios empíricos indican que el juicio sobre las situaciones y la toma de decisiones humanas están basados en estrategias intuitivas (heurísticas) en vez de en reglas de razonamiento teóricamente sólidas. Con ello se pretende reducir la carga cognitiva. El juicio y elección humanos sin ayudas, exhiben sistemáticas violaciones de los axiomas de la

probabilidad (sesgos). Además de la evidencia de laboratorio, hay varios estudios de rendimiento de expertos en marcos realistas, mostrando que es inferior incluso a modelos lineales simples (para una visión informal de las evidencias disponibles y citas de la literatura, ver [DAW88]).

#### **2.1.2 El modelado de decisiones.**

La superioridad de los modelos, incluso lineales simples, sobre el juicio intuitivo humano sugiere que un modo de mejorar la calidad de las decisiones es descomponer el problema de la decisión en componentes simples que sean bien entendidos y definidos. El estudio de un sistema complejo construido con tales componentes puede ser ayudado por una técnica formal teóricamente sólida. Esto permite aplicar conocimiento científico que puede ser transferido a otros problemas (o incluso dominios) permitiendo analizar, explicar y argumentar acerca del problema de decisión.

En las diferentes herramientas de modelado desarrolladas, el conocimiento acerca del sistema es representado por ecuaciones o reglas lógicas, mejoradas con una representación explícita de la incertidumbre. Una vez que un modelo ha sido formulado, pueden usarse una variedad de métodos matemáticos para analizarlo. La toma de decisiones bajo incertidumbre, introduce mejoras en base a aproximaciones estadísticas, tales como análisis de la confianza, simulación, y toma de decisiones basada en estadísticas. La mayoría de estos métodos han sido realizados en las universidades y pueden encontrarse en libros de texto de gestión.

#### **2.1.3 Componentes del modelado de decisiones.**

Mientras matemáticamente un modelo consiste en un conjunto de variables y una especificación de interacciones entre ellas, desde el punto de vista de la toma de decisiones, un modelo y sus variables representan los siguientes tres componentes: una medida de preferencias sobre los objetivos de la decisión, opciones de decisión disponibles, y una medida de incertidumbre

sobre variables que influyen la decisión y los resultados. El proceso se soporta en la asunción de que una buena decisión es la que resulta de utilizar un buen proceso de decisión que considera todos los factores importantes y es explícito acerca de las alternativas de decisión, preferencias e incertidumbre.

Es importante distinguir entre buenas decisiones y buenos resultados. Ayudar a la toma de decisiones significa ayudar en el proceso de decisión para que puedan tomarse las mejores decisiones. Puede esperarse que las mejores decisiones lleven a los mejores resultados.

## 2.2 Los Sistemas de Ayuda a la Toma de Decisiones (DSS).

Los DSS son sistemas informáticos interactivos que ayudan a los usuarios a juzgar y elegir actividades. Ofrecen almacenamiento y recuperación de datos pero mejorando las funciones de acceso y recuperación tradicionales con ayudas para la construcción del modelo y el razonamiento basado en el modelo. Ayudan en el estructuración, modelado y resolución del problema.

Áreas de aplicación típicas de los DSSs son: la gestión y planificación de negocios, atención sanitaria, militar, y cualquier área de gestión en el que se encuentren situaciones de toma de decisión complejas. Se usan principalmente para decisiones estratégicas y tácticas en la gestión de nivel superior (baja frecuencia y consecuencias potenciales altas) en los que el tiempo dedicado es amortizado generosamente a largo plazo.

Los DSSs tienen, tres componentes fundamentales:

- a) Sistema de Gestión de las Bases de Datos (SGBD). Son los bancos de datos para los DSSs. Separan al usuario de los aspectos físicos de la estructura, contenidos y procesamiento de la base de datos.
- b) Sistema de gestión de la base del modelo (SGBM). El papel del SGBM es análogo al del SGBD. Su función principal es dar independencia entre los modelos específicos usados en el DSS y las aplicaciones que los usan. El objetivo es transformar datos en información útil para

la toma de decisiones. Por lo no estructurado de muchos problemas, debería de asistir también en la construcción del modelo.

- c) Sistema de generación y gestión del diálogo (SGGD). El principal producto de la interacción con un DSS es la profundización en el problema. Como los usuarios son a menudo gestores sin formación informática, se necesitan interfaces intuitivos y fáciles de usar. Estos interfaces ayudan en la construcción del modelo y en la interacción con el mismo, proporcionando una visión y unas recomendaciones. Para una mayor sencillez en la exposición, en adelante nos referiremos a SGGD como "interfaz de usuario".

Aunque existe una variedad de DSSs, los tres componentes anteriores forman parte de muchas de sus arquitecturas y juegan un papel prominente en sus estructuras. La interacción entre ellos es la que se señala en la Fig. 1.

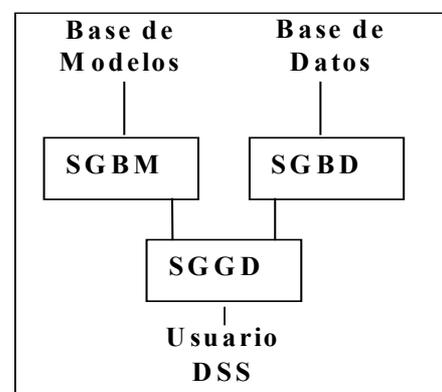


Fig. 1. Componentes de un DSS

### 2.2.1 DSSs de análisis de decisiones.

Esta emergente clase de DSSs aplica los principios de la teoría de la decisión, teoría de la probabilidad, y análisis de las decisiones a sus modelos de decisión. La teoría de la decisión es una teoría axiomática de toma de decisiones construida sobre un pequeño conjunto de axiomas para una toma de decisiones racional. Expresa incertidumbre en

términos de probabilidades y preferencias en términos de utilidades; se combinan usando la operación de esperanza matemática.

Lo atractivo de la teoría de la probabilidad, como un formalismo para soportar la incertidumbre en DSSs, radica en su solidez y sus garantías relativas a rendimiento a largo plazo. La teoría de la probabilidad es vista a menudo como el estándar de oro de la racionalidad para razonar bajo incertidumbre. El análisis de las decisiones es el arte y la ciencia de aplicar la teoría de la decisión a los problemas del mundo real. Incluye un abundante número de técnicas para la construcción de modelos, tales como métodos de elicitación de la estructura del modelo y distribuciones de probabilidad que permiten minimizar el error humano, métodos para la comprobación de la sensibilidad del modelo a la imprecisión de un dato, cálculo del valor de obtener información adicional, y presentación de resultados. Estos modelos están bajo un continuo examen por parte de psicólogos que trabajan en el dominio de la teoría de la decisión del comportamiento y han probado que hacen frente razonablemente bien a los peligros relacionados con los sesgos humanos a la hora de juzgar.

La generación del modelo dinámico es un intento de automatización de la parte más laboriosa de la toma de decisiones, la estructuración.

### **2.3 Interfaces de usuario para DSSs.**

Mientras que la calidad, confianza que ofrecen las herramientas de modelado y las arquitecturas internas son importantes, el aspecto más crucial, por mucho, de los DSSs lo constituyen sus interfaces. Interfaces incómodas, poco claras y que requieren habilidades inusuales, raramente son aceptadas en la práctica. El resultado más importante de una sesión con un DSS es el poder profundizar en el problema de decisión. Si a esto añadimos un sistema que esté basado en principios normativos, podrá ser utilizado como tutor, pudiéndose esperar que el usuario aprenda del modelo de dominio la forma que tiene de razonar en el tiempo, y pueda mejorar su propia forma de pensar. Un buen interfaz debería soportar:

1. La construcción y el análisis del modelo. Para llegar a hacer que el modelo en el que se basa el razonamiento del sistema sea transparente

al usuario. Ésta fase de construcción del modelo requerirá de varias iteraciones.

2. Razonamiento acerca de la estructura del problema por la adición de cálculos numéricos. Si los cálculos numéricos son importantes en la ayuda a la decisión, razonar acerca de la estructura del problema lo es más. Un gráfico causal realzará el interfaz de usuario dando mayor información.
3. La elección y optimización de variables de decisión. Muchos DSSs tienen una estructura inflexible, las variables a ser manipuladas son determinadas durante la construcción del modelo, esto resta posibilidades de interactividad.

#### **2.3.1 Desarrollo de Aplicaciones Web de ayuda a la decisión basadas en modelos.**

Los DSSs basados en la Web pueden ayudar a recuperar, analizar y visualizar datos estructurados provenientes de grandes bases de datos multidimensionales, modelos, documentos multimedia, datos no estructurados, o facilitar la comunicación y toma de decisiones en grupos de trabajo distribuidos. Todo ello de forma interactiva y accesible para múltiples usuarios. Las aplicaciones Web suponen una revolución capaz de ofrecer soluciones no disponibles hasta ahora.

La consecución de estas soluciones requiere de una sólida arquitectura basada en tecnologías de gestión de datos bien establecidas, particularmente tecnología de bases de datos.

Por otra parte, las aplicaciones Web deben ser diseñadas para el cambio, no sólo de sus contenidos, sino que también de sus requerimientos y arquitecturas. Entonces, su desarrollo necesita ser organizado en procesos bien definidos, pensados para beneficiarse de la ingeniería del software, en los que la automatización de tareas repetitivas esté basada en la reutilización de componentes.

### 2.3.2 Herramientas para el desarrollo de aplicaciones Web y sitios Web que manejan grandes volúmenes de datos.

Existe un impresionante número de herramientas de distintas gamas para la construcción de aplicaciones para la Web. Sin embargo, una revisión cuidadosa de sus características revela que la mayoría de las soluciones se concentra en la implementación, prestando poca atención a proceso completo de diseño de una aplicación Web, en [FRA99] puede encontrarse un estudio detallado sobre el tema.

Los marcos de trabajo de aplicaciones Web que nos interesan en nuestro problema son los utilizables para la construcción de aplicaciones Web que publican y mantienen grandes volúmenes de datos, por lo que las necesidades de disponer de herramientas que cubran todo el ciclo de vida del desarrollo del proyecto resulta determinante si queremos diseñar una solución capaz de adaptarse a los cambios. En [FRA99] podemos encontrar una comparativa de herramientas desarrolladas, o en fase de construcción, que intentan hacer frente a este objetivo: [Araneus](#), [Autoweb](#), [Strudel](#), [Web Architect](#), y [W3I3](#). Además del ciclo de vida, dan soluciones que facilitan, en alguna medida, la automatización de procesos, el modelado de abstracciones, la reutilización y generación de componentes, la definición de arquitecturas y la ayuda al uso.

Algunas de estas herramientas podrían ser utilizadas, extendiéndolas, en nuestro trabajo de construcción de un DSS genérico para la gestión de proyectos software. Sin embargo, dado el impresionante desarrollo de la tecnología Java y las importantes aportaciones en cuanto a software abierto de la [Apache Software Foundation](#) a través del proyecto [Jakarta](#), principalmente con los proyectos Struts y Tomcat, sin olvidar [Sun](#) y [HP Bluestone Software](#), así como la posibilidad de contar con componentes desarrollados por terceros en forma de librerías de etiquetas (*Tag libraries*) nos inclinamos por ésta tecnología para el desarrollo de nuestro marco de trabajo.

### 2.3.3 La importancia de XML y RDF para la construcción de ontologías.

La World Wide Web es posible debido a que un conjunto ampliamente establecido de estándares

garantiza la interoperabilidad a varios niveles. Aunque hasta ahora, la Web ha sido diseñada directamente mediante el trabajo humano, en el futuro, será necesario usar herramientas que permitan construir sistemas flexibles capaces de ofrecer servicios inteligentes tales como: agentes de búsqueda, filtros de información, *brokers* de información, etc. Lo que requiere una mayor funcionalidad e interoperabilidad que los sistemas actuales. Esta línea ha recibido un fuerte impulso con la estandarización por parte de W3C de XML (eXtensible Markup Language) y RDF (Resource Description Framework). Es un hecho que todos los desarrolladores de software están adoptando XML para sus problemas de interoperabilidad, el uso de RDF para la construcción de ontologías de dominios lleva un camino similar. Para una visión más completa del papel de las ontologías en la arquitectura de las aplicaciones Web del futuro ver [DEM00].

## 3. Alternativas de desarrollo y características del dominio de aplicación.

La complejidad de las soluciones requeridas nos obliga a contar con un equipo humano numeroso y a disponer de herramientas que ofrezcan soluciones efectivas para todo el ciclo de vida del proyecto, en todas sus fases y disciplinas, incluido el mantenimiento. Pensando primero en el problema de ingeniería que se nos presenta, además de una metodología de proceso, es preciso utilizar herramientas para el análisis, diseño, pruebas, gestión, comunicación entre miembros del proyecto, etc. A continuación revisaremos las alternativas para la toma de decisiones en el desarrollo del software y las de nuestro dominio de aplicación habitual: la Informática Médica.

### 3.1 DSSs para gestión de proyectos vía Web.

También en el desarrollo de un DSS la toma de decisiones adquiere una gran importancia, es preciso tomar multitud de decisiones para el cumplimiento de los plazos y la adecuada conclusión del proyecto. Un buen proceso de

toma de decisiones nos ayudará a sobrellevar la complejidad de la construcción de un DSS.

### 3.1.1 Soluciones existentes.

Herramientas integradas para envío de mensajes como Lotus Notes y herramientas de ayuda al diseño como TeamFusion se comportan adecuadamente en sus respectivos dominios. Pero cuando un gestor de proyectos necesita organizar tareas, asignar recursos y hacer el seguimiento del rendimiento; debe usar otros recursos, especialmente en los proyectos software. El software de gestión de proyectos facilita las labores administrativas relacionadas con el grupo de trabajo implicado, facilitando la planificación y la estimación de costos de aspectos críticos de análisis. El saber si un proyecto está cumpliendo los plazos previstos es la cuestión para la que se utilizan herramientas como Microsoft Project, Primavera Systems SureTrak Project Manager o la PS6 de Scitor Inc.; sin embargo, es poco más lo que se consigue de ellas. Están pensadas para ser usadas por profesionales de la gestión y no por miembros del equipo del proyecto, no están pensadas para la colaboración en tareas. Esta clase de herramientas son, por supuesto, indispensables para las labores administrativas, ofreciendo (tanto Microsoft Project como SureTrak Project Manager) la publicación Web de los datos del proyecto y el correo electrónico integrado.

Como resultado, los miembros del equipo de desarrollo del proyecto, deben usar una mezcla de herramientas no integradas que soporta una parte del trabajo global: gestión del proyecto, comunicación, colaboración en tareas compartidas, soporte para grupos dispersos, reutilización de código, etc. Están comenzando a aparecer herramientas que pretenden integrar todo el proceso. Estas nuevas herramientas, destinadas a facilitar la construcción de aplicaciones Web, están basadas en la arquitectura cliente-servidor para el acceso compartido a datos y la utilización de diferentes estándares. Por la forma de uso, es necesario que dispongan de una gestión integrada del proceso. De entre estas nuevas herramientas podemos mencionar:

1. [Mesa/Vista Project Manager](#), incluye comunicación basada en Web que puede

operar con diversas herramientas de gestión (Microsoft Project) y diseño (Rational Rose) y tiene algunas de las funcionalidades de Lotus Notes y Microsoft Outlook como control de acceso y gestión de documentos. En cambio, no dispone de soporte para gráficos, control de recursos, o estimaciones presupuestarias. En definitiva, ofrece un nivel de servicios horizontal.

2. [WebProject](#), se acerca más a una gestión de proyectos tradicional ampliamente extendida con una completa implementación Java (con sus ventajas e inconvenientes) y un diseño inteligente basado en la Web. Los aspectos de colaboración y comunicación se parecen a los de Microsoft Project. Por otra parte, ofrece un excelente soporte para la toma de decisiones distribuida y la definición de proyectos, tareas y recursos vía Web, así como la comunicación con otras herramientas utilizadas en la gestión de proyectos.

Tal y como sucede en otros ámbitos de la informática, las soluciones de mayor calidad existentes son cerradas y de un costo que sólo está al alcance de grandes instituciones. Las soluciones abiertas disponibles que utilizan tecnología Java están adquiriendo una gran importancia, basta recordar el impacto de proyectos como Apache, Tomcat, Protégé-2000, etc.

### 3.2 Características de los DSSs necesarios en la Informática Médica y alternativas.

La Informática Médica, que podríamos definir como el uso de los ordenadores y de la información en la atención sanitaria, educación e investigación médica; presta una atención creciente a los problemas de tratamiento tanto de pacientes internos como de externos por el costo económico que generan y por el interés de la medicina preventiva. Distintas razones (mayor uso de tecnología, movilidad de población, especialización, utilización de sistemas informáticos, incremento de costos, mejoras en

hardware, y mejoras en métodos) han potenciado la Informática Médica hasta ser considerada como una nueva disciplina académica.

La medicina es el dominio de aplicación de técnicas informáticas más importante, los aspectos más tratados son los siguientes:

- Técnicas de adquisición de datos.
- Almacenamiento de los múltiples tipos de datos que se utilizan: textos, imágenes, etc.
- Vocabularios unificados médicos (*UMLS*)
- Organización de datos, registros médicos.
- Interfaces estándares (*HL7*) y seguridad.
- Lenguajes de recuperación de datos.
- Adquisición del conocimiento (*Protégé*).
- Representación del conocimiento (*Ontologías*).
- Aplicación del conocimiento necesario (decisiones, alertas, etc.)
- Educación y gestión de la enseñanza.
- Protocolos médicos, conocimiento compilado.
- Telemedicina (línea de mayor prioridad)
- Bioinformática (genoma humano).
- Arquitectura de sistemas unificados (*IAIMS*).

Se ha construido una gran cantidad de aplicaciones de las que un buen número han pasado a ser de uso real [COI99].

Aunque el diagnóstico sea el tipo de problema al que se le ha dedicado más atención, en los últimos años, se están haciendo grandes esfuerzos en la monitorización o seguimiento del tratamiento médico debido a su gran incidencia en los presupuestos de los servicios de salud. Esto requiere una infraestructura informática más importante, especialmente si se quiere acceder a las enormes ventajas de la Telemedicina. MEL SIS pretende ser un marco de trabajo genérico aplicable a problemas de monitorización (no sólo en medicina) por lo que en adelante nos centraremos en éste tipo de problema.

El conocimiento médico necesario suele estar estructurado para una mejor aplicación por un amplio número de médicos, esta estructuración puede llegar a ser de la máxima credibilidad y constituir un protocolo médico. La existencia de conocimiento de amplio consenso facilita la construcción de sistemas basados en el conocimiento.

#### 4. El marco de trabajo MEL SIS.

La construcción de DSSs aplicables a problemas de monitorización de la complejidad que los problemas reales exigen necesita de:

- 1) Una infraestructura software que permita la rápida construcción de prototipos por la reutilización de la arquitectura, módulos y componentes utilizados en casos anteriores. Sólo posible si se cuenta con un marco de trabajo de la máxima flexibilidad.
- 2) Un servidor capaz de responder con la celeridad requerida a las peticiones que lleguen. Toda la carga de proceso tendrá lugar en él, lo que disminuirá los costos de soporte de usuarios.
- 3) La aplicación de las tecnologías más recientes y mejor valoradas considerando conjuntamente la eficiencia y la calidad técnica. Esto último debe posibilitar la integración de:
  - a) Análisis, diseño e implementación con una documentación clara y automatizada.
  - b) Componentes software propios o de terceros para la interfaz de usuario y procesos repetitivos.
- 4) Soporte para la construcción del modelo de decisión de forma que:
  - a) Facilite la adquisición del conocimiento mediante formularios específicos.
  - b) La incorporación de conocimiento definido por nosotros o terceros en forma de ontologías.
- 5) Soporte para la construcción del modelo de decisión de gestión de proyectos capaz de mejorar sus prestaciones en función de la información conseguida en casos anteriores. El esfuerzo que pretendemos hacer en ésta línea intenta poner las bases para la introducción de una cultura de mejora constante, dar soporte a la ingeniería de procesos a fin de cumplir los

requisitos del negocio y la optimización de recursos. Todo ello, para facilitar el camino hacia la mejora procesos software siguiendo modelos estándares como ISO 9000-3, CMM, BOOTSTRAP o SPICE

#### 4.1 Características generales.

Se pretende que el sistema que implemente el marco de trabajo anterior ofrezca las siguientes características generales que puedan utilizarse como infraestructura software para sucesivos casos:

1. Dado que, en el desarrollo de proyectos software, deben coordinarse grupos de trabajo geográficamente dispersos, el sistema tomará la arquitectura de una aplicación Web.
2. Arquitectura y funcionalidad parametrizada suficientemente flexible como para poder incluir en cualquier momento, sin necesidad de detener el servidor Web, nuevas características de gestión: datos a controlar, elementos de control, variables complementarias, así como procedimientos de estimación, gestión y visualización.
3. Rápida construcción de un sitio Web con todos los recursos de gestión de proyectos y almacenamiento de toda la información de un proyecto: fases, artefactos, modelos, bibliografía, herramientas, relación de miembros del proyecto, *forums* de discusión, etc.
4. Comoquiera que los usuarios pueden pertenecer a comunidades lingüísticas diferentes, el sistema podrá ser utilizable en diferentes idiomas con la sola duplicación y traducción de los literales de un fichero.
5. Existirá un conjunto de grupos de usuario predefinidos, ampliable, y una opción de gestión de grupos y cuentas de usuario. Existirán grupos de: desarrolladores, expertos, administradores y usuarios.
6. La funcionalidad será específica y ampliable para cada grupo pudiéndose introducir restricciones de uso para cuentas concretas de ese grupo.

7. Todas las bases de datos usadas por el sistema podrán sufrir cambios en su estructura y ser puestos en servicio en cuanto se les transfiera la información de las antiguas.
8. Control de sesiones en curso y un histórico de sesiones para analizar el uso del sistema por desarrolladores y, posteriormente, usuarios.
9. Definición de ontologías explícitas [HES97] de representación, genéricas (abstracciones temporales), de dominio (gestión de proyectos), y de aplicación (p. e. el tratamiento del asma). Los resultados se guardarán en formato RDF(S) para su posterior reutilización.
10. La información que da soporte a la toma de decisiones responderá al modelo conceptual de árbol de decisión utilizándose modelos construidos partiendo de la teoría de la decisión.

En lo que se refiere a la parte de gestión proyectos software que lleva incorporada, pretendemos ir incorporando, en los próximos años y en colaboración con otros grupos de investigación, aspectos más ambiciosos como:

1. Mejora de las técnicas de estimación de coste de los proyectos software.
2. Definición y diseño de un sistema de métricas para la valoración de Proyectos de Desarrollo de Software.
3. Evaluación de procesos y comparación de los procedimientos de evaluación.
4. Análisis, construcción y evaluación de un sistema de planificación de software utilizando métodos metaheurísticos
5. Estrategias de visualización de proyectos software para proporcionar mapas visuales.
6. Aplicación de algoritmos metaheurísticos para la fase de prueba del software.
7. Nuevos procedimientos para verificación y validación de software.

## 4.2 Arquitectura y componentes.

El marco de trabajo MELISIS cuenta, básicamente, con los componentes mencionados en la Fig. 1. Al tratarse de una aplicación Web en la que conviene evitar el costo de soporte de usuarios, tan elevado en el dominio de aplicación médico, hemos concentrado todo el proceso en el servidor (*thin client*). Además, dadas la complejidad añadida por el uso de la red y la indiscutible importancia que las ontologías tienen en éste tipo de aplicaciones, hemos tenido que incorporar otros marcos de trabajo, módulos y componentes desarrollados en otras instituciones.

Como puede verse en la Fig. 2, actualmente, los componentes del DSS MELISIS se materializan en lo siguiente:

1. Las bases de datos, distribuidas en cuatro, que están destinadas a:
  - a. Contener todos los parámetros del sistema.
  - b. Almacenar los datos del dominio de aplicación.
  - c. Mantener los datos de gestión del proyecto.
  - d. Guardar las ontologías.
2. La base de modelos, formada por un conjunto de ontologías desarrolladas mediante la herramienta Protégé-2000, incorporada en MELISIS con todos los *plug-ins* disponibles para facilitar:
  - a. La definición de ontologías.
  - b. Su visualización.
  - c. Incorporación de axiomas.
  - d. Generación de formularios de adquisición del conocimiento.
3. El interfaz de usuario se ha construido a partir del marco de trabajo Struts que proporciona una infraestructura software construida como un módulo Web [SUN00] formado por diferentes componentes Web [SUN00].

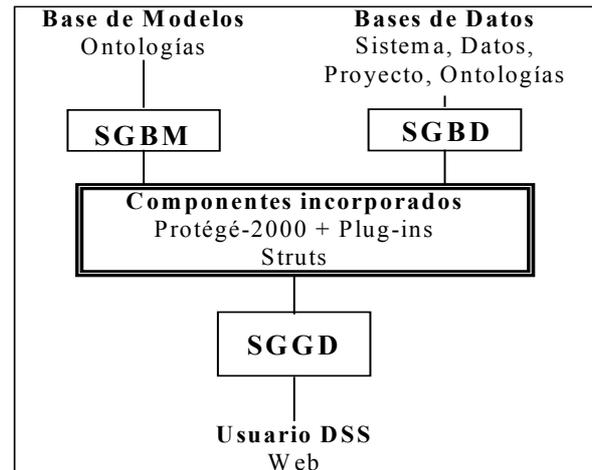


Fig. 2. Componentes de MELISIS

## 4.3 Proceso de desarrollo.

En las iteraciones llevadas hasta el momento, se ha seguido, básicamente, la metodología de proceso *Rational Unified Process (RUP)* añadiendo algunos documentos y diagramas relativos al manejo de conocimiento tomados de la metodología CommonKADS.

La parte de que podríamos identificar como de ingeniería del software se encuentra en la fase de transición de la metodología RUP, y la parte relacionada con la ingeniería del conocimiento en la fase de elaboración. Los modelos de análisis, diseño e implementación correspondientes a la primera parte han sido definidos mediante la herramienta *Racional Rose Enterprise* usando tecnología Java. El uso módulos Web y componentes Web con *taglibs* [SUN00] está resultando ser de gran ayuda en la estructuración interna del sistema en construcción. Para la implementación final de los prototipos de ésta aplicación Web hemos contado de contenedores de *servlets* abiertos como Tomcat (Apache) y Total-e-Server (HP Bluestone Software) y *drivers JDBC* de tipo 4. Tal y como indica la metodología RUP y reclaman las circunstancias de contar con un equipo de trabajo geográficamente disperso hemos construido un sitio Web que incluye todos los entregables del proyecto.

En la segunda parte se está utilizando la herramienta Protégé-2000 para la definición de ontologías y formularios específicos para la

adquisición del conocimiento, el resultado está siendo representado en formato RDF(S) con la idea de que sea reutilizado, guardándolo también en la base de datos de ontologías y en formato CLIPS para ser ejecutado usando el motor de inferencia de Jess y FuzzyJess; en el caso de que estas últimas herramientas no cumplieren con las expectativas, disponemos de la herramienta EHSIS, construida por nosotros, que integra CLIPS y FuzzyCLIPS con otros módulos de comunicaciones, desarrollo de interfaces y programación orientada a eventos que ha sido suficientemente probada en el desarrollo del DSS Diabetes-I construida en los últimos años.

Se han considerado otras tecnologías de implementación, .NET y ASP principalmente, pero han sido desechadas por no disponerse de versiones estables y por no seguir el formalismo de orientación a objetos respectivamente.

Otras herramientas como las de SAS y SPSS han sido desechadas por su elevado coste y por ser cerradas.

#### 4.4 Resultados.

Se ha construido un primer prototipo del DSS Arnasa. Este sistema pretende ser una ayuda integral, entre otros, para lo siguiente:

1. El tratamiento del asma permitiendo realizar el proceso de tratamiento a distancia con el acceso y actualización, por parte de pacientes y médicos, de todos los datos necesarios y el intercambio de información.
2. Acceso a información relativa al asma.
3. Investigación médica a partir de los datos acumulados.

Actualmente, usando las bases de datos del sistema y de datos del dominio, permite visualizar y actualizar los datos temporales producidos en un periodo de estudio de la evolución de un paciente asmático. Para la visualización, existen varias alternativas:

1. Usando el formato del documento que habitualmente se usa para el intercambio de información entre pacientes y médicos.

2. Gráficamente, permitiendo profundizar en los subperiodos de interés interactivamente.
3. Valiéndonos de la estadística descriptiva, visualizar gráficamente y en forma de texto la información del periodo elegido con la posibilidad de profundizar en periodos unidades de tiempo inferiores interactivamente.

Actualmente, además de ampliar la funcionalidad anterior, estamos desarrollando las ontologías del dominio en base a la información contenida en [NAE97] y con la inestimable ayuda de los médicos del Hospital de Cruces de Baracaldo (Vizcaya) que participan en el proyecto.

#### 4.5 Conclusiones.

La enorme revolución tecnológica producida en los últimos años, principalmente en lo relacionado con Internet, nos ha obligado a un gran trabajo de valoración de tecnologías a emplear en nuestro propósito de construir un marco de trabajo de gran flexibilidad no sólo en lo referente a la ingeniería del conocimiento sino también en lo concerniente a la ingeniería del conocimiento. Además de encontrar herramientas que han demostrado ser de gran utilidad, la decisión de optar por la tecnología Java está demostrando ser acertada y es capaz de afrontar con garantías los retos con los que nos vamos encontrando en la construcción de MELSI, no pueden olvidarse las posibilidades de reutilización de código y la disponibilidad de software abierto de esta tecnología.

Algo parecido puede decirse de la opción de disponer de Protégé-2000, aunque en este área no puede decirse que existan estándares de garantía (excepción hecha de XML y RDF), la calidad de la herramienta y el amplio número de usuarios a nivel mundial junto con el numeroso equipo que en la Universidad de Stanford trabaja en su entorno nos indican que pueden ser una buena elección. Además al generar resultados en el formato CLIPS se adecua muy bien a nuestra línea habitual de trabajo.

## 5. Agradecimientos

Nuestros trabajos de investigación y desarrollo han podido realizarse gracias a la financiación recibida de la Universidad del País Vasco/Euskal Herriko Unibertsitatea (UPV 141.226-TB153/98).

## REFERENCIAS

- [COI99] Coiera E. [Artificial Intelligence Systems in Routine Clinical Use](#). 1999.
- CHJ00] Chee C.L., Jarzabek S. and Paul R. *F-metric: a WWW-based framework for intelligent formulation and analysis of metric queries*. The Journal of Systems and Software 43 (1998) 119-132.
- [DRF00] Druzdzal M. J. and Flynn R.R. *Decision Support Systems*, Encyclopedia of Library and Information Science, Allen Kent (ed.), New York, Marcel Dekker, Inc., 2000.
- [DAW88] Dawes R.M. *Rational Choice in an Uncertain World*. Hartcourt Brace Jovanovich, Publishers, San Diego, CA, 1988.
- [DEM00] Decker S., Melnik S., Van Harmelen F., Fensel D., Klein M., Broekstra J., Erdmann M. And Horrocks I. *The semantic Web: The Roles of XML and RDF*. IEEE Internet Computing, Sept/Oct 2000, pp. 63-74.
- [FRA99] Fraternali P. *Tools and Approaches for Developing Data-Intensive Web Applications: a Survey*. ACM Computing Surveys, 31(3); 227-263, 1999.
- [FRP99] Fraternali P. and Paolini P. [Model-Driven Development of Web Applications: the Autoweb System](#), ACM Transactions on Office Information Systems vol. 18 (4), 2000.
- [FES99] Fernandez M., Suciú D. and Tatarinov I. [Declarative Specification of Data-Intensive Web Sites](#) In USENIX Conference on Domain-Specific Languages , 1999
- [GAC99] Gachet A. [A Framework for Developing Distributed Cooperative Decision Support Systems – Inception Phase](#). Decision Support Systems Group, Department of Computer Science, University of Fribourg, Switzerland.
- [HES97] Van Heist G., Schreiber T. and Wielinga B. [Using Explicit Ontologies in KBS Development](#). International Journal of Human-Computer Studies. Vol. 46, (2/3), 183-292, 1997.
- [KAP01] Karacapilidis N. I. and Pappis C. P. [A framework for Group Decision Support Systems: Combining AI tools and OR techniques](#). GMD – German National Research Center for Information Technology (Germany) and University of Piraeus (Greece).
- [NAE97] NAEPP. [Practical Guide for the Diagnosis and Management of Asthma](#). National Asthma Education and Prevention Program. National Heart, Lung, and Blood Institute. National Institutes Of Health. 1997.
- [SUN00] Sun Microsystems, Inc. [Building Web Components: Forte™ for Java™, Internet Edition, 2.0](#) Sun Microsystems, Inc. Diciembre de 2000.