

MIDAS: Posibilitando el acceso a fuentes de información heterogéneas de manera sencilla.

Andrés Sanchis (ansanes@fis.upv.es), José Alberto Maldonado (jamaldo@upvnet.upv.es), Pere Crespo (pedcremo@fis.upv.es), Carlos Angulo (cangulo@fis.upv.es), Montserrat Robles (mrobles@fis.upv.es)

Resumen

En un entorno hospitalario como el del Hospital Universitario La Fe, hospital de referencia de la Comunidad Valenciana con numerosas especialidades y más de 1800 médicos facultativos repartidos entre los cuatro hospitales que engloba (general, infantil, maternal y rehabilitación), los diferentes departamentos han incorporado las tecnologías de la información a sus procesos progresivamente, normalmente sin una política conjunta y eligiendo los sistemas que se han considerado más adecuados para satisfacer sus necesidades en cada momento. Esta manera de proceder ha ocasionado que en la actualidad nos encontremos ante una situación donde conviven un conjunto heterogéneo de sistemas de información, planteando cada uno de ellos, una problemática diferente respecto al acceso a sus registros. La necesidad de tener una vista unificada de la historia clínica electrónica del paciente nos lleva, por tanto, ha desarrollar aplicaciones que han de hacer frente a esta heterogeneidad. El objetivo del proyecto MIDAS es crear la infraestructura necesaria para posibilitar un acceso uniforme por parte de las aplicaciones a las distintas fuentes de información clínica.

1. Introducción

El proyecto MIDAS nace como colaboración entre el grupo de informática médica del BET de la Universidad Politécnica de Valencia y los servicios de informática del hospital la Fe para intentar dar solución a la demanda por parte del personal sanitario de una aplicación que proporcione el acceso a una vista unificada de la historia clínica de los pacientes para, de esta forma, agilizar algunos de los procesos de los diferentes departamentos como el acceso a los resultados de las pruebas de laboratorios o al historial de pruebas radiológicas que se le

han realizado a un determinado paciente. Se ha aprovechado la experiencia en el desarrollo de aplicaciones para el acceso y compartición de información clínica basados en estándares que el grupo de sistemas de información hospitalarios ha adquirido con el proyecto PANGEA [1][2]. Así, se han utilizado algunos de los conceptos utilizados en este proyecto y se han aplicado a MIDAS, sistema objeto de este artículo.

2. Integración de datos

Podemos entender por integración de datos el problema de combinar datos que residen en diferentes sistemas y proporcionar a los usuarios una vista unificada de éstos. La integración de datos ha sido, y es, un campo de investigación muy activo. En la literatura informática se pueden encontrar diversas arquitecturas y metodologías [3]. MIDAS, como sistema de integración de datos, sigue una arquitectura basada en mediadores y adaptadores [4]. En esta arquitectura varias fuentes de datos están “envueltas” por una capa de software denominada adaptador o *wrapper* el cual traduce entre el lenguaje, modelos y conceptos de la fuente de datos y el lenguaje, modelos y conceptos del mediador. Los mediadores obtienen la información a partir de una o más fuentes de datos o de otros mediadores, es decir, de las componentes que están por debajo de él, y proporcionan información a las componentes que están por encima, en nuestro caso, los usuarios externos del sistema.

Los sistemas de integración basados en mediadores son utilizados cuando la necesidad de acceder a información actualizada y reciente es crítica o cuando es imposible o difícil instanciar todos los datos almacenados en las fuentes. Esto es así en el caso de la integración de información clínica distribuida. El usuario final del sistema de integración, por ejemplo el médico que atiende al paciente, debe tener acceso a toda la información

disponible del paciente hasta la fecha. La información clínica sobre un paciente puede ser muy voluminosa (en el caso más extremo historia clínica electrónica completa), por tanto, no es recomendable acceder a toda ella sino que es más conveniente el consultar sólo aquella que es necesaria. Por otro lado, el esquema global de un mediador se suele construir siguiendo una metodología “top-down” de acuerdo con las necesidades de información de los usuarios globales. Es por esto que se puede entender a los mediadores como servicios que se construyen y se ponen a disposición de los clientes.

3. El Sistema MIDAS

El sistema MIDAS se encarga de atender cualquier petición de información clínica-administrativa sobre los pacientes, por tanto se sitúa entre los usuarios finales (aplicaciones) y las fuentes de datos del hospital. La comunicación entre las aplicaciones cliente y MIDAS está basada en el uso de un conjunto de servicios ofertados por el servidor que son accesibles para cualquier cliente.

Los usuarios del sistema reciben la información clínica-administrativa de los pacientes por medio de un mensaje. El mensaje de respuesta contiene un documento XML con la información solicitada. Por tanto, es necesario poder definir plantillas de mensajes donde se especifique la información a la que se accederá y como se estructurará en el documento respuesta que se servirá a las aplicaciones clientes. Estas plantillas se han de definir de manera sencilla y debe existir un mecanismo mediante el cual se puedan incluir nuevas plantillas en el servidor en tiempo de ejecución. Con este propósito se ha definido un lenguaje basado en XML para la definición de las plantillas de mensaje que ha sido bautizado como MAD. Con este lenguaje es posible especificar la siguiente información: fuente o fuentes de datos que contienen la información necesaria para crear instancias de mensajes, información que se va a extraer de esas fuentes y el etiquetado y estructura del documento XML resultante (MAD permite crear estructuras anidadas sin límite de profundidad), parámetros que acepta el mensaje y las combinaciones válidas de éstos. Por ejemplo, para acceder a los análisis de laboratorio de los pacientes, definiríamos un mensaje en el cual la única fuente de datos relevante sería la base de datos del laboratorio, la información a extraer sería los campos de la base de datos que contienen el

tipo y resultado de las pruebas y estructuraríamos el mensaje de respuesta en dos bloques con etiquetas “bioquímica estándar” y “inmunología” cada una de ellas con sus respectivas pruebas. Los parámetros que aceptaría el mensaje en este caso serían el número de historia clínica del paciente y el código del hospital, que se corresponden con la información necesaria para identificar los análisis correspondientes a un paciente.

Ejemplo 1. Definición de un mensaje Midas:

```
<mapping nombre="nombreMapping" fuente="fuente"
etiqueta="etiquetaprincipal">
  <from>tabla 1, ..., tabla N</from>
  <columna nombre="campo1" etiqueta="eti1" />
  <columna                               nombre="campo2"
etiqueta="eti2.subeti1" />
  .
  .
  <columna nombre="campoN" etiqueta="etiN"
/>
</mapping>
<filtros>
  <filtro param="param1" campo="campo1" operador =
"igual" />
  <filtro param="param2" campo="campo2" operador =
"mayorque" />
</filtros>
<parametros>
  <parametro nombre="param1"/>
  <parametro nombre="param2"/>
</parametros>
```

Ejemplo 2. Petición.

```
<petición>
<nombre_mensaje>mensaje1</nombre_mensaje>
  <parámetros>
    <parametro
nombre="param1">valorparam1</parametro>
    <parametro
nombre="param2">valorparam2</parametro>
  </parámetros>
</petición>
```

En la ejemplo 1 se muestra una plantilla genérica para un mensaje. El ejemplo 2 muestra una petición realizada por un cliente. En ella se especifica el nombre del mensaje que deseamos consultar y los parámetros correspondientes con sus respectivos valores.

A partir del valor de los parámetros de la petición realizada por un cliente se genera una consulta SQL sobre la base de datos a la que referencia el atributo *fuente* de la etiqueta <mapping>. Esta consulta SQL es una operación select donde la cláusula SELECT incluye los campos que aparecen en los atributos *nombre* de los diferentes elementos <columna>, la cláusula FROM tiene el valor del elemento <from> y por último, la cláusula WHERE se construye a partir del elemento <filtro> relacionando el

valor del

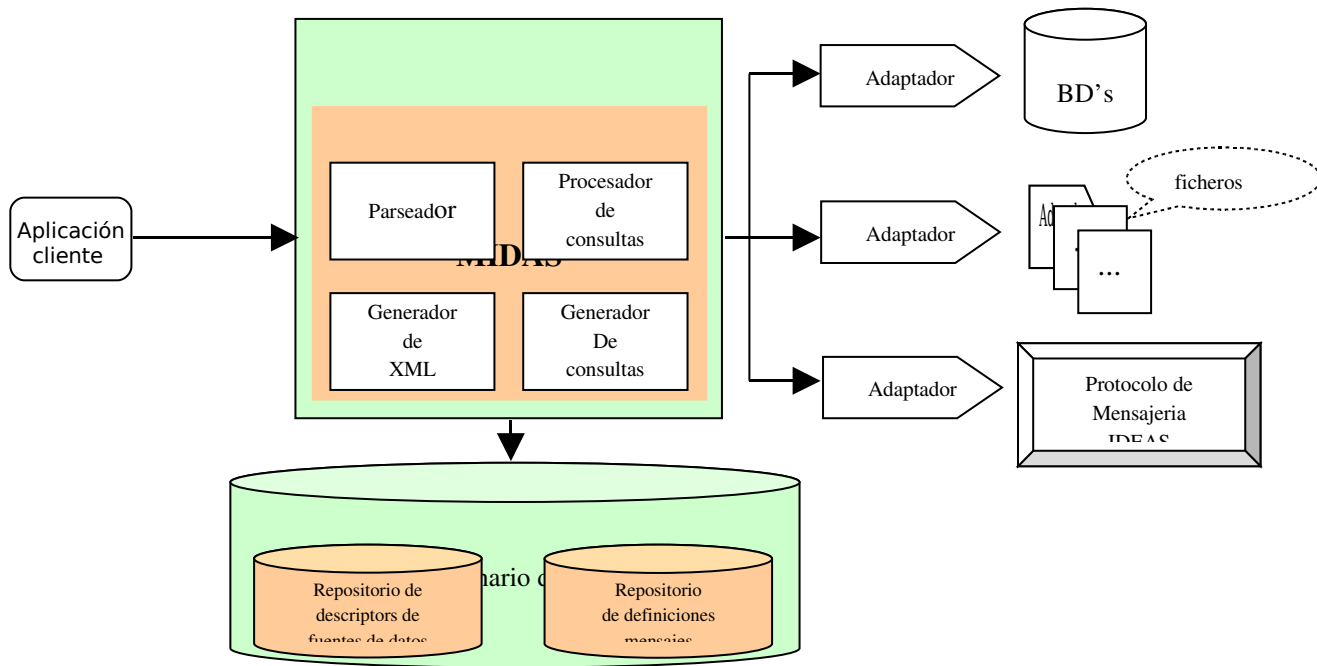


Figura 1. Arquitectura del sistema MIDAS

parámetro de la petición cuyo nombre coincide con el atributo *nombre* con el campo especificado en el atributo *campo* mediante el operador especificado por el atributo *operador*. Para el ejemplo la consulta SQL sería:

```
SELECT campo1, ..., campoN FROM tabla1, ..., tablaN
WHERE valorparam1=campo1, valorparam2 > campo2.
```

Después de ejecutar la consulta, se construye el documento XML cuya etiqueta raíz coincide con la especificada en el atributo *etiqueta* del elemento <mapping> y contiene las etiquetas indicadas en el atributo *etiqueta* de los diferentes elementos <columna> con el valor obtenido tras la ejecución de la consulta. El atributo *etiqueta* puede contener un valor compuesto como vemos en el ejemplo 1 para la columna correspondiente al campo 2, en este caso, el documento XML contendrá una jerarquía de elementos y el valor obtenido se le asignará al elemento hoja. Esta sería la estructura del documento XML respuesta:

```
<etiquetaprincipal>
  <eti1>valor_campo1</eti1>
  <eti2>
    <subeti1> valor_campo2</subeti1>
  </eti2>
  .
  .
  <etiN> valor_campoN</etiN>
</etiquetaprincipal>
```

Cabe destacar que en un mismo mensaje se pueden especificar diferentes fuentes de información de las cuales obtener datos. Por ejemplo, en el mensaje que ofrece MIDAS para el acceso a los informes de radiología, la información referente al informe como puede ser el médico que lo redactó, la fecha y la hora de redacción, etc., se obtiene accediendo a una base de datos relacional mientras que el texto del informe se obtiene de otra fuente de datos diferente, en este caso, un archivo de texto que se accede a través de un servidor FTP. Esto se consigue utilizando diferentes elementos <mapping> en el mismo mensaje y enlazándolos entre sí.

Se han evaluado algunas herramientas existentes en el mercado para la publicación en XML de datos contenidos en fuentes de datos, sobre todo relacionales. Los problemas que se han encontrado en estas soluciones es que normalmente no permiten la creación de mensajes que accedan a múltiples fuentes de datos o bien no permiten acceder a fuentes de datos que no sean relacionales. Entre las herramientas que han sido evaluadas está Jallora, de la compañía Hit Software [5]

Para procesar las plantillas y generar los documentos respuesta, el sistema hace uso de un diccionario de datos que contiene las definiciones de los mensajes y la descripción de las fuentes de datos a las que el sistema ha de conectarse. Estas fuentes de datos pueden ser de

diferente naturaleza. Para cada una de ellas, el repositorio contiene la información necesaria para el acceso, por ejemplo, direcciones de red, adaptador a utilizar para el acceso a datos, información de autenticación en los sistemas donde se alojan, etc. En el caso del hospital La Fe la mayoría de la información esta almacenada en bases de datos relacionales aunque también se pueden encontrar informes en archivos de texto que son accesibles a través de servidores FTP, imágenes en sistemas PACS o datos que se obtienen a través de servicios de mensajería. Para acceder a estas fuentes se han diseñado adaptadores específicos.

En la figura 1 podemos ver un esquema de la arquitectura del sistema. Como se puede observar MIDAS tiene cuatro componentes básicos. Cuando se inicia el sistema el parseador analiza las plantillas de mensajes disponibles en el diccionario de datos y crea una estructura interna (árbol) que simula la estructura del mensaje respuesta, en la que cada nodo se correspondería con un elemento XML, ya sea este simple o compuesto. Si es un elemento simple, el nodo correspondiente es una hoja del árbol y tiene asociada una expresión con la cual el sistema podrá asociarle el valor correspondiente cuando se obtenga el resultado de ejecutar la consulta pertinente. Un elemento compuesto, únicamente define cómo estructurar los datos en el mensaje respuesta y puede tener asociada una plantilla de consulta para obtener la información a la que hacen referencia sus nodos descendientes. Cuando el usuario del sistema realiza una petición de instanciación de uno de los mensajes para un caso concreto (p.ej. datos demográficos de un paciente o resultados de un análisis), el generador de XML interpreta el árbol generado previamente y haciendo uso del generador de consultas crea las consultas necesarias para obtener la información requerida. A continuación, estas consultas son lanzadas y gestionadas por el procesador de consultas. Una vez obtenidos los datos el generador de XML construye con ellos el mensaje respuesta que se envía finalmente al cliente.

4. Métodos

Teniendo en cuenta como prerequisite que el sistema ha de ser accesible para diferentes aplicaciones que se ejecutan en distintos sistemas operativos y arquitecturas el uso de tecnologías abiertas se convierte en la opción más adecuada para la comunicación de la información. De las tecnologías disponibles se opta por XML [6] para la construcción de los mensajes y la definición tanto de estos como de las fuentes de datos del sistema y de Servicios Web[7] para la comunicación de la información.

Básicamente un Servicio Web implementa una colección de operaciones que pueden ser invocadas a través de la red utilizando mensajería estándar basada en XML. Estas operaciones se especifican en un fichero WSDL al cual se puede acceder a partir de una URL y que especifica las diferentes operaciones del servicio, los parámetros de entrada y de salida de estas operaciones, los tipos de datos relacionados así como información relativa a autenticación, seguridad, etc. A partir de la información contenida en este archivo descriptor podemos crear clientes del servicio web e integrarlos en nuestras aplicaciones.

Normalmente, las aplicaciones que exponen Servicios Web se despliegan en servidores de aplicaciones. En nuestro caso hemos utilizado el contenedor de servlets TOMCAT de la fundación APACHE[8] con la librería AXIS para la publicación de servicios web de la misma fundación. Estas tecnologías están teniendo una gran difusión y existen gran cantidad de librerías y utilidades para su uso en la mayoría de los entornos de desarrollo existentes. Gracias a esto podemos elegir entre una gran variedad de plataformas a la hora de implementar aplicaciones que hagan uso de la arquitectura MIDAS. En nuestro caso, para el diseño del visor de historia clínica, se ha utilizado JAVA como lenguaje de programación, pero existen también dentro del hospital clientes que obtienen información de MIDAS desarrollados en la plataforma .NET de Microsoft.

5. Conclusiones y trabajo futuro

La implantación de MIDAS se ha realizado satisfactoriamente en el hospital y se ha demostrado la viabilidad de este tipo de arquitectura en un entorno con sistemas de información tan heterogéneos. Una de las principales ventajas del sistema es la posibilidad de tener un acceso uniforme y centralizado a las distintas fuentes de información clínica, lo cual facilita enormemente entre otras cosas el desarrollo de nuevas aplicaciones, el mantenimiento y la auditoria de acceso a los datos. Las ventajas del sistema serian aún más importante si el etiquetado y estructura de los mensajes siguiesen algún estándar para la comunicación de información clínica como el EN13606 del Comité Técnico 251 del Comité Europeo de Normalización (CEN/TC251) [9]. Como tareas futuras se ha planteado añadir mecanismos de encriptación y autenticación más robustos y diseñar herramientas gráficas que faciliten la configuración de la aplicación y del diccionario de datos.

Referencias

- [1] Maldonado JA, Robles,M, Crespo P. *Integration of distributed healthcare records: publishing legacy data as XML documents compliant with CEN/TC251 ENV13606*.Proceedings of the 16th IEEE Symposium on Computer Based Medical Systems. IEEE Computer Society Press, pp. 213-218 (2004).
- [2] Maldonado, J.A., Robles, M., Cano, C., Crespo, P. *Integración de sistemas de información hospitalarios: utilización del estándar de arquitectura de historia clínica electrónica ENV13606 del CEN/TC251*, **34**, 44-50 (2002).
- [3] Bouguettaya, A., Benatallah, B., Elmagarmid, A. *Interconnecting heterogeneous information systems*. Kluwer Academic Publishers (1998).
- [4] Wiederhold, G. Mediators in the Architecture of Future Information Systems. *Computer* **25(3)**, 38-49 (1992).
- [5] http://www.hitsw.com/products_services/xml_platform/allora.html
- [6] <http://www.w3.org/XML>
- [7] <http://www.w3c.org/2002/ws>
- [8] <http://www.apache.org>
- [9] <http://www.centc251.org>