

INTEGRACIÓN DE COMPONENTES DE HARDWARE Y SOFTWARE EN LA IMPLEMENTACIÓN DE PROTOTIPOS PARA TELEMEDICINA

Integration of components of hardware and software in the implementation of prototypes for telemedicine

RESUMEN

Este artículo tiene como finalidad presentar una propuesta para el diseño de equipos integrados de Telemedicina, es decir, un prototipo con un sistema “todo en uno”, el incluye funciones de electrocardiógrafo, encefalógrafo, pulsoxímetro y demás.

La gran ventaja de la propuesta radica en el aprovechamiento de módulos comunes, a saber: adquisición de señales (convertidores análogo digitales), filtrado (como puede ser un filtro Notch), visualización de resultados (manejo de pantallas LCD); tanto en hardware como en software lo que ofrecería grandes beneficios en reducción de costos, utilización de espacio, etc; contrario a lo que sucede actualmente en donde cada equipo especializado funciona de manera independiente.

PALABRAS CLAVES: DLL, SOAP, WSDL, Telemedicina, Servicios Web.

ABSTRACT

This article aims to present a proposal for the design of integrated telemedicine equipment, ie, a prototype system with an all-in-one, which includes functions electrocardiograph, encephalograph, pulsoximeter and others.

The great advantage of the proposal is the use of common modules, namely acquisition of signal (digital analog converters), filtration (such as a Notch filter), visualization of results (management LCDs), both hardware and software which would provide great benefits in reducing costs, utilization of space, etc., contrary to what is happening today where each team operates independently.

KEYWORDS: DLL, SOAP, WSDL, Telemedicine, Web Services.

1. INTRODUCCIÓN

Los problemas logísticos asociados tanto a la prestación de servicios médicos domiciliarios (en los cada vez más congestionadas centros urbanos), como la atención a pacientes ubicados en lugares distantes, que por su propia condición (de incapacidad de desplazamiento), demandan el creciente interés a nivel mundial en aplicar esquemas de telemedicina (entendida ésta como la habilidad para diagnosticar, trazar cursos de acción y curar a los pacientes en sitios distantes) a través de equipos de alta tecnología enlazados mediante redes de comunicaciones (de muy variados tipos) utilizando gran ancho de banda.

Por tal razón se propone un sistema que ofrece la posibilidad y conveniencia de diseñar equipos integrados que permitan la medición de muy variados tipos de

ALEJANDRO GONZALEZ OSPINA

Ingeniero Industrial.
Universidad Tecnológica de Pereira.
Candidato a Magíster en Instrumentación Física.
Universidad Tecnológica de Pereira.
softwaresalud@hotmail.com

HUGO BALDOMIRO CANO GARZÓN

Ingeniero Electricista.
Universidad Tecnológica de Pereira.
Especialista.
Convenio UTP - EAN.
Candidato a Magíster en Instrumentación Física.
Universidad Tecnológica de Pereira.
Profesor Auxiliar.
Universidad Tecnológica de Pereira.
hbcano@utp.edu.co

JOSÉ ANDRÉS CHAVES OSORIO

Ingeniero Electricista.
Universidad Tecnológica de Pereira.
Especialista en Pedagogía.
Universidad Nacional Abierta y a Distancia.
Candidato a Magíster en Instrumentación Física.
Universidad Tecnológica de Pereira.
Profesor Asistente.
Universidad Tecnológica de Pereira.
jachaves@utp.edu.co

señales fisiológicas; facilitando la adquisición, procesamiento y transmisión de las señales a fin ser tratadas en un centro de monitoreo y diagnóstico. Estos equipos operarían de tal forma que los datos se manejen a través de librerías de hardware, las cuales poseen la capacidad de atender solicitudes de servicio y/o procesamiento de otras entidades; es decir, el sistema es abierto en la medida en que puede atender solicitudes de cualquier prestador del servicio (Tanto de hardware como de software).

2. DESCRIPCIÓN GENERAL

El diseño del prototipo es producto de la interacción de varios subsistemas que en la práctica se constituyen como un arreglo de componentes configurable por software.

Fecha de Recepción: 25 de Enero de 2008

Fecha de Aceptación: 25 de Abril de 2008

La Tabla 1, presenta a manera de ejemplo la interacción de componentes de hardware comunes que pueden ser utilizados para diferentes funcionalidades del equipo. Cabe anotar que la X representa el componente de hardware que requiere cada función.

Función	Componentes de Hardware				
	Fuente	Teclado	Conversión A/D	Filtro Notch	Pantalla LCD
Electrocardiografo	X	X	X	X	X
Pulsoxímetro	X	X	X	X	X
Encefalógrafo	X	X	X	X	X
Espirómetro	X	X	X	X	X

Tabla 1. Relación entre hardware y funciones

La relación e interacción de los diferentes subsistemas se fundamenta en un modelo en Capas, tal como se observa en la Figura 1.

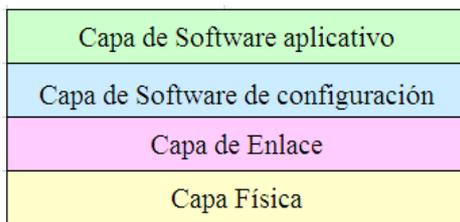


Figura 1. Modelo en Capas de la propuesta

2.1 Descripción de las capas o niveles

A continuación se realizará una descripción general de cada una de las capas del modelo propuesto.

2.1.1 Capa Física

Esta capa es la encargada del manejo de las señales eléctricas, tipo de conexiones, medios de transmisión y demás aspectos referidos exclusivamente a la transferencia de los datos.

2.1.2 Capa de Enlace

Este nivel proporciona facilidad para la interconexión de los bloques de funciones, los cuales estarán unidos por “junturas” controladas por los diferentes Drivers que permiten la articulación de los módulos.

2.1.3 Capa de Software de configuración

Nivel encargado de activar y desactivar los componentes de hardware necesarios, así como de disponer de los modos de funcionamiento, por ejemplo determinar la frecuencia de muestreo.

2.1.4 Capa de Software aplicativo

Es el nivel que interactúa con el usuario y dispone de menús que contienen funciones de cada uno de los servicios implementados en el prototipo, que incluye por ejemplo el software aplicativo para el electrocardiografo, el pulsoxímetro, el encefalografo, entre otros.

Ahora bien dado el planteamiento del trabajo, el uso en redes tipo Web es un paso natural, por tanto el tipo y tiempo de almacenamiento, y la distancia de los equipos involucrados no son temas de preocupación.

Si se hace necesario realizar un estudio de factibilidad respecto de la configuración de equipo electrónico integrado, que pueda: Adquirir, acondicionar, procesar y transmitir señales biomédicas a una central de monitoreo debidamente equipada y que cuente con personal humano calificado para ofrecer la asistencia necesaria en cada caso.

De forma análoga el sitio de monitoreo y diagnóstico centralizado debe poseer la capacidad para recibir las señales y realizar procedimientos tales como: procesamiento discreto, recomposición de la señal original, y presentación de resultados, así mismo compartir la información con sistemas expertos de monitoreo de parámetros fisiológicos.

3. UNIDAD REMOTA

Entre los Módulos del Dispositivo Integrado (MDI) que se pueden incluir en las unidades remotas están, entre otros, los de:

Electrocardiografía
Tensión y presión arterial
Espirometría
Temperatura
Temblor.

4. CENTRO DE MONITOREO Y DIAGNÓSTICO

El Centro de Monitoreo y Diagnóstico que se diseña es el lugar donde se centraliza la información de los servicios atendidos, con el fin de realizar seguimiento a cada afección particular y aplicar las respectivas medidas a cada paciente, lo anterior con los correspondientes beneficios en el mejoramiento de la calidad de vida de los usuarios, como son la disminución en los desplazamientos al centro asistencial, el descongestionamiento y rapidez en la atención, la reducción en las posibilidades de contagio de otras enfermedades, entre otras.

El sistema de monitoreo centralizado se apoya en sistemas expertos, desarrollados con técnicas de inteligencia artificial, que ayuden al trabajo del personal médico y paramédico encargado de los servicios referidos.

Teniendo en cuenta que la lectura de los diferentes parámetros fisiológicos, por ejemplo biopotenciales, deben ser sometidos a procesos muchas veces comunes a los diferentes MDI, (como es el caso de la necesidad de aplicar un filtro Notch), se presenta la inquietud acerca de cómo lograr con un mismo componente (sea de hardware o de software), responder dinámicamente de forma oportuna y eficaz a los requerimientos de los diferentes

módulos funcionales del equipo integrado que se está tratando.

La comprensión de este problema se puede abordar desde la perspectiva de una Arquitectura de software Orientada a Servicios (en inglés Service Oriented Architecture o SOA), donde los recursos y funcionalidades presentes en el sistema están disponibles para los usuarios (módulos, procesos, funciones, etc) en forma de servicios independientes, estandarizados, débilmente acoplados y altamente interoperables. Es así como está diseñada la actual estructura de Servidores Web, donde se encuentra que sus pilares son el SOAP¹ (Simple Object Access Protocol) y el WSDL² (Web Services Description Language)).

La Figura 1, muestra la relación entre hardware y Servicios web, pero deja embebida la relación existente entre los diferentes componentes de hardware.

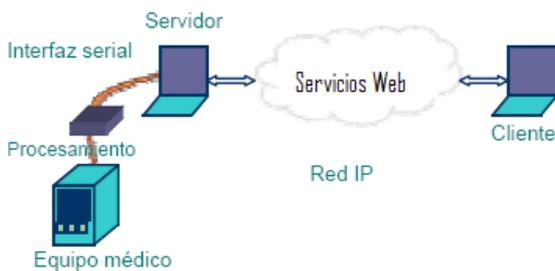


Figura 1. Relación entre hardware y Servicios web

Una de las mayores ventajas para desarrollar el proyecto sobre la base de Servicios Web, es que éstos son independientes de la tecnología subyacente y del lenguaje de programación; lo que posibilita reusabilidad de los componentes de forma que se puede fácil y rápidamente diseñar, fabricar, expandir y mantener sistemas de procesamiento de parámetros fisiológicos que pueden evolucionar y que interactúen en tiempo real con sistemas expertos de diagnóstico médico.

Lo anterior supone la definición de estándares en la construcción del hardware, tal como los que en la actualidad se utilizan para la construcción de tarjetas de circuitos para los PC. Así mismo, estos estándares definen la capa de empalme suave (Los Drivers) necesarios para la juntura de las piezas.

La Figura 2 corresponde a uno de los muchos trabajos que se desarrollan hoy en día, pero que no contemplan la desagregación de sus componentes o la adición de nuevos con tecnología “plug and play”.

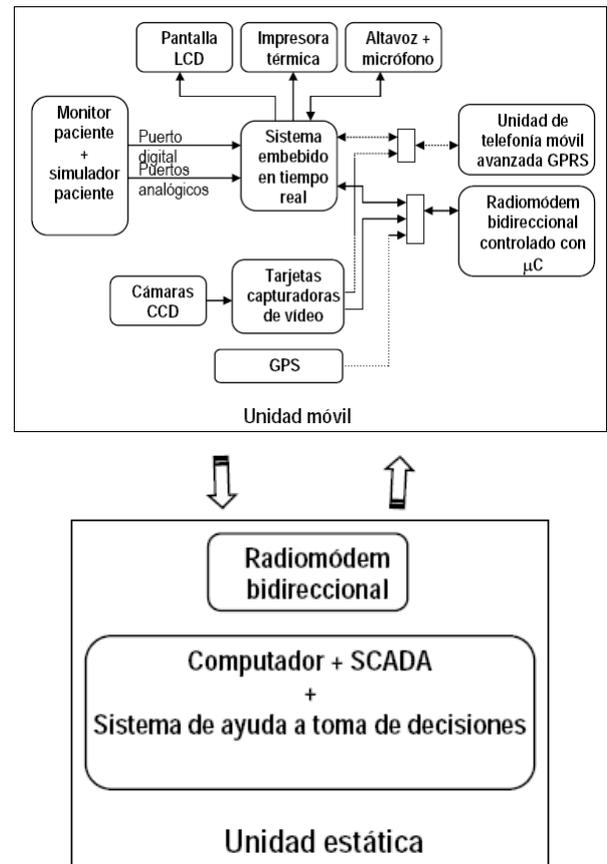


Figura 2. Esquema de sistema de diagnóstico típico actual

En la Figura 3 se observa la imagen de un pantallazo típico para el seguimiento a un paciente, con sus correspondientes señales fisiológicas.



Figura 3. Pantalla de monitoreo típica

Debido a la relevancia que tienen dos conceptos anteriormente mencionados como son “Plug and play” y “DLL`s”, se ofrece una breve definición de ambos.

1 Protocolo de acceso simple a objetos
2 Lenguaje de descripción de servicios Web

1. Las notas de pie de página deberán estar en la página donde se citan. Letra Times New Roman de 8 puntos

5. PLUG-AND-PLAY

5.1 Definición

Plug-and-play, conocida también por su abreviatura PNP, es la tecnología que permite a un dispositivo informático ser conectado a un computador sin tener que configurar (mediante jumpers o software específico; es decir sin utilizar controladores o drivers proporcionados por el fabricante), ni proporcionar parámetros a sus controladores. Para que sea posible, el sistema operativo con el que funciona el computador debe tener soporte para dicho dispositivo.

La frase plug-and-play se traduce como enchufar y usar. No obstante, esta tecnología en la mayoría de los casos se describe mejor por la frase apagar, enchufar, encender y listo.

No se debe confundir con Hot plug, (conectar en caliente) que es la capacidad de un periférico para ser conectado o desconectado cuando el ordenador está encendido.

Tampoco Plug and Play indica que no sea necesario instalar controladores (drivers) adicionales para el correcto funcionamiento del dispositivo. Plug and Play no debería entenderse como sinónimo de "no necesita drivers"³.

6. DLL

6.1 Definición

DLL es el acrónimo de Dynamic Linking Library (Bibliotecas de Enlace Dinámico), término con el que se refiere a los archivos con código ejecutable que se cargan bajo demanda del programa por parte del sistema operativo. Esta denominación se refiere a los sistemas operativos Windows siendo la extensión con la que se identifican los ficheros, aunque el concepto existe en prácticamente todos los sistemas operativos modernos.

6.2. Ventajas

Las DLLs son o pueden verse como la evolución de las bibliotecas estáticas y de forma análoga contienen funcionalidad o recursos que utilizan otras aplicaciones. Sin embargo, su uso proporciona algunas ventajas: Reducen el tamaño de los archivos ejecutables: Gran parte del código puede estar almacenado en bibliotecas y no en el propio ejecutable lo que redundaría en una mejor modularización

Pueden estar compartidas entre varias aplicaciones: Si el código es suficientemente genérico, puede resultar de utilidad para múltiples aplicaciones (por ejemplo, la MFC es una biblioteca dinámica con clases genéricas que

recubren la API gráfica de Windows y que usan gran parte de las aplicaciones).

Facilitan la gestión y aprovechamiento de la memoria del sistema: La carga dinámica permite al sistema operativo aplicar algoritmos que mejoren el rendimiento del sistema cuando se carguen estas bibliotecas. Además, al estar compartidas, basta con mantener una copia en memoria para todos los programas que la utilicen.

Brindan mayor flexibilidad frente a cambios: Es posible mejorar el rendimiento o solucionar pequeños errores distribuyendo únicamente una nueva versión de la biblioteca dinámica, nuevamente está corrección o mejora será aprovechada por todas las aplicaciones que compartan la biblioteca.

6.3 Problemas

Sin embargo, no todo son ventajas. En los sistemas Windows, las DLLs son muy comunes y muchos programas usan las mismas DLLs; pero debido a la evolución, cada una de las DLLs cambia incorporándose mejoras pero modificándolas de tal forma que dejan de ser compatibles.

Lo anterior puede producir dos efectos no deseados:

Que la instalación de un programa reemplace una DLL con una nueva versión incompatible

Que la desinstalación del programa borre una DLL compartida

En ambos casos, el resultado es que dejan de funcionar los programas que utilizaban la vieja versión. Estos problemas se denominaron el infierno de las DLLs.

Las versiones más modernas de Windows y los nuevos scripts de instalación MSI (sobre todo su característica de instalaciones residentes) abordan este problema; sin embargo, el problema persiste cuando se utilizan otros instaladores (versiones antiguas) o se realizan modificaciones manuales⁴.

7. CONCLUSIONES Y/O RECOMENDACIONES

Estandarizando el diseño del hardware para telemedicina sobre la base del Modelo Cliente / Servidor y de una Arquitectura de software Orientada a Servicios, que pueda trabajar con y como lo hacen las librerías de acceso dinámico (DLL's) en el campo del software, se podrá alcanzar el objetivo de masificar a bajos costos el uso de la telemedicina.

El hecho fundamental de "compartir componentes" trae consigo innumerables ventajas entre las que cabe resaltar disminución en los costos de operación, reducción de las áreas de trabajo, fácil manejo por medio de menús que permiten rápido intercambio entre las aplicaciones sin

³ <http://es.wikipedia.org/wiki/Plug-and-play> [5]

⁴ http://es.wikipedia.org/wiki/Dynamic_Linking_Library [6]

detrimento de la operatividad del equipo debido al bajo nivel de acople entre las funciones.

8. BIBLIOGRAFÍA

[1] <http://pvldata.com/Mauricio/index.php/archives/38>

[2] http://es.wikipedia.org/wiki/Arquitectura_orientada_a_servicios

[3] <http://www.unal.edu.co/telemedicina/bioTelemedicina.html>

[4] <http://www.saber.ula.ve/redtelemedicina/index.html>

[5] <http://es.wikipedia.org/wiki/Plug-and-play>

[6] http://es.wikipedia.org/wiki/Dynamic_Linking_Library

