Medical Electronics Manufacturing Spring 2000

The Internet Appliance: Medical Instrument of the Future

Medical devices are ready to be connected to the Internet—perhaps readier than their manufacturers know.

Edward F. Steinfeld

Medical instruments have evolved from simple controllers, whose only user interface was an on-off switch, into programmable devices with a color screen and a variety of input controls. The new devices have internal computers with the computational power of the desktop PC of just a few years ago. Until very recently, most medical instruments plugged into the wall or, occasionally, another device. Each instrument had its own man-machine interface (MMI). Devices seldom required interconnection in the past, and when the need was apparent, the cost was prohibitive. Even if the interconnection cost could be justified, MMIs unique to each device posed an obstacle.

All that is changing. Several technologies have come together to create intelligent, remotely controlled, easy-to-use, interconnected devices called Internet appliances.

Intelligent Devices

The technologies that define the intelligent device include the 32-bit microcomputer, the network protocol, and the Web browser as the accepted MMI. They are increasingly available as low-cost commodities. One technology, the physical interconnect, is still evolving. Many software interconnects exist, but no others have become as ubiquitous as the Web browser and the Transmission Control Protocol/Internet Protocol (TCP/IP) network standard.

The 32-bit microcomputer. The heart of the intelligent device is the programmable computer. In the past 10 years, companies such as Motorola, Intel, ARM, IBM, and Texas Instruments have developed the microcomputer into a reliable, low-cost, and easily integrated component. It ranges in complexity from the 4-bit device used in microwave ovens to the 32-bit microcomputer in desktop PCs. The 32-bit chip has the power to control a device, provide a useful MMI, and act as a node in a network of interconnected devices. Volume production and advances in fabrication have reduced its cost from more than $1000 to as little as $9.

Network protocols. Some 30 years ago, the need to interconnect computers called for the design of reliable and expandable communication protocols. One emerging from a Department of Defense project became the worldwide standard for computer interconnects. The TCP/IP protocol can now be found in devices as diverse as cellular telephones, weather-modeling computers, and home computers connected to America Online. The software to create this protocol has become a commodity, like the 32-bit computer.

Adding a network connection is a software and a hardware function. The hardware protocol used most commonly is Ethernet. An Ethernet interface either can be integrated with the microcomputer, as is the case with NetSilicon and Connect One products, or can be an additional chip or chips. The TCP/IP protocol runs on top of the hardware-connect protocol. Most TCP/IP software stacks are not designed for embedded devices. One of a small number of vendors specializing in TCP/IP for Internet appliances is Treck.

The man-machine interface. The mechanism by which a person interacts with an intelligent device is the MMI. In the case of the Web browser, this is called the graphical user interface, or GUI.

The first MMI for electronic devices was the labeled on-off switch. Later, more-complex devices with some intelligence designed in had interfaces with switches, rotary knobs, and even a text display. Today's MMI might be a color display showing both text and graphics and accepting user input via a key panel, keyboard, or touch-sensitive panel. Devices are being interconnected with other devices, and the choice of MMI is moving toward the Web browser GUI. The 32-bit microcomputer and TCP/IP networking protocol have made this possible. As Web technology extends its commercial reach and more electronic devices incorporate Web-related software, the Web browser is becoming the standard means of communicating with computing devices.

Creating the Internet Appliance

In order for a medical instrument to use the browser as its MMI, the instrument must become an Internet appliance. Two pieces of software are required to make an electronic device into an Internet appliance: the embedded Web server, providing a way to store and transmit the graphical Web pages that are displayed on the user's browser, and the HTML (hypertext markup language) compiler, for inserting dynamic display information into those pages.

Embedded Web components. Tools to design HTML-formatted Web pages are available from a number of sources. Allegro Software Development Corp. is one of a few companies that offer tools for adding the embedded intelligence necessary to convert electronic devices into Internet appliances. Its products include the RomPager embedded Web server, which includes an HTML compiler; RomPager Soft Pages, which enables new pages to be added on the fly to the embedded Web server; and RomPager Java Graphlets, to facilitate display of graphs and charts and periodic updates of the user display. (Table I provides examples of embedded Internet tools.)

Product Name	Description	ROM (Kbyte)	RAM (Kbyte)	Standards Supported
RomPager	Embedded Web server—full featured; communicates with all Web browsers; includes a Web application toolkit with compile-time HTML parsing and compression.	13–29	3–5 (per request)	HTTP 1.0/1.1; HTML 2.0, 3.2, 4.0; RFC 1867, RFC 1945, RFC 2068, RFC 2069, RFC 2616, RFC 2617.
RomPager SoftPages (RomPager option)	Run-time HTML parser.	15	5	HTML 2.0, 3.2, 4.0.

RomPager Light	Embedded Web server—small footprint; communicates with all Web browsers.	7–12	2–4	HTTP 1.0/1.1; RFC 1945, RFC 2068, RFC 2616.
RomWebClient	Embedded HTTP client—sends to and receives from all Web servers.	10	2	HTTP 1.0/1.1, RFC 1945, RFC 2068,RFC 2616.
RomPOP	POP3 client—receives Internet e-mail.	12–18	10–20	POP3; MIME attachments; RFC 822, RFC 1939, RFC 2045, RFC 2046.
RomMailer	SMTP client—sends Internet e-mail	4–8	3	SMTP; MIME attachments; RFC 821, RFC 822, RFC 2045, RFC 2046, RFC 2110, RFC 2111, RFC 2112.
RomDNS	Domain Name Services client	3	5–20	RFC 1034, RFC 1035.
RomPager Java Graphlets	Java applets for dynamic data presentation.	15–100	0	Most popular browsers.
RomXML	Embedded XML parser/framer.	10	3	W3C XML.

The advantage of using toolkits such as these is that the instrument manufacturer need know little about Internet or Web technology. The primary links between the instrument software and the code generated by the Web server toolkits are pointers to variables and buffers. Web page designers have to know the names and functions of the variables in the instrument. Using these variables in the Web page allows the display of dynamic data and the input of control values.

Once the electronic device manufacturer has incorporated a 16- or 32-bit microcomputer and a network protocol, has selected the physical interconnect method, and has incorporated the embedded Web server, the additional step necessary to Web-enable the device for intelligent interactivity is relatively easy.

E-mail capability. Intelligent medical devices can be made independently interactive by the addition of e-mail software. For example, a drug-dispensing device could notify the pharmacy by e-mail when the medicine was running low. Toolkits can add the e-mail function either to an embedded Web server or to a medical device that lacks a Web server but does include the hardware basics of microcomputer, network interface, and network protocol. Allegro's RomPOP e-mail client receives and RomMailer SMTP client sends e-mail. (These are not mail readers or general mail servers.) As with incorporating Web server toolkits, the instrument manufacturer needs only minimal knowledge of Internet mail.

Embedded e-mail can now be found in networked computer printers. A printer can notify the appropriate person via e-mail when toner is low. When the RomPager embedded Web server is installed in the same printer, the browser becomes the tool to manage the machine. Jobs can be sent to it and operational data—pages printed, the number of letter-size sheets remaining in the drawer—viewed. Likewise, functions can be added to medical devices for their remote management.

Advantages of Embedded Web Servers

How expensive is it to network independent intelligent devices? How difficult to develop the Internet appliance? How much will this added technology affect the price of the product?

Much new technology comes at the cost of lengthening the learning curve for product developers, overcoming user resistance to learning yet another way to do things, and finding new methods to distribute the technology. And perhaps new use limitations will appear that were not seen in the old technology.

Embedded Web servers involve none of these problems. Quite the contrary. Medical device manufacturers for years looked for an easy path to a graphical user interface and a common application programming interface. Even when one was found, the interface for each operating system and hardware platform had to be rewritten. Then there was the problem of distribution. How could manufacturers ensure that the correct GUI was sent to the user? And what happened when a new user or computer was added to the network? Web server technology solves these problems.

Potential customers already use them. Will customers know what to do with an Internet appliance? The end-user or client interface for the embedded Web server is the ubiquitous Web browser that millions of people use every day. There may be nothing new for the customer to learn.

The browser might be Netscape, Internet Explorer, or Mosaic running on a Macintosh or IBM-compatible PC, a handheld device with Windows CE, or any other computer. The Web browser has become a universal GUI.

Integration cost is minimal. The implementation cost of an embedded Web server is so low that it might not even be a budget item. Prospective purchasers of Internet appliances get Web browsers "free" as part of an operating system or an application suite. A product with an embedded microchip could already be talking to a network, and thus would have both the network protocol and the hardware interface. The additional memory required for an embedded Web server is minimal, 7 to 30 Kbyte.

The cost to incorporate Internet and Web technology into an instrument is chiefly that of the Web server toolkit which, depending on the functionality needed, can vary from $25,000 to $50,000. This is normally a one-time expense. Most toolkits are licensed royalty free.

A typical programmer can integrate a Web server into instrument code in less than a week. Most Web server toolkits support common commercial TCP/IP stacks and even have *make* files already generated for instant integration. If the instrument uses a commercial kernel, or real-time operating system (RTOS), similar *make* files will exist. The Web server toolkits perform much of the integration and relieve the instrument manufacturer of the need to write Internet applications.

No custom GUI is needed. How about development costs? When the customer provides the Web browser, the universal GUI, containing all the client software required for its implementation, the device manufacturer does not have to purchase or design a graphical interface, saving money and months of development time.

As for introducing the embedded server, an Internet appliance can be prototyped by means of tools as simple as Microsoft Word or FrontPage. The markup language, HTML, is like text-formatting languages with which developers may be familiar. With such Web page development tools as FrontPage and Dreamweaver, the format-defining tags used in HTML coding will not be visible. These new tools are inexpensive enough to encourage experimenting to discover which best meets development needs.

Writing the final dynamic HTML pages is like writing any application. The designer uses the standard HTML tools to create pages that are compiled and then linked with C functions to insert dynamic variable information. The HTML pages are compiled into C source code and used like any other C program. All the page designer needs to know is the variable names used in the instrument code.

Distribution costs nothing. Every time an Internet appliance is accessed, a new copy of the user interface is distributed to the user by the Web server. No longer must the product manufacturer cut new disks, print new installation guides, or install software for users. In connecting with a Web-enabled device, the user receives the current display graphics. The manufacturer installs any software changes only on the embedded computing device by means of, say, the Allegro RomPOP POP3 e-mail client feature of the RomPager embedded Web server.

To access the Web-enabled instrument, a new user types in the URL for the device (usually including the device name and patient room number and stores it in either the bookmarks or the favorites section of the browser). The device might include a security module to be updated for each new user. The manufacturer does not have to update the user's machine. The user interface is coded within the device, and the number of clients is nearly unlimited. File upload, as defined by RFC 1867, is supported by the RomPager server, and Microsoft and Netscape browsers later than 4.0 can also be used to install changes. RFC (Request for Comment) is a numbered series of documents that specify how the different parts of the Internet work.

The front panel becomes superfluous. The front panel is the most costly part of many medical instruments. Panels are fitted with controls and switches and, in many cases, a text or video display. Front panels are usually hand wired, which makes them prone to wiring errors. The costs of panel components, of the labor to wire the panel and all its controls, and of testing and repairing the panel add up.

Web technology enables the front panel to be replaced by the user's desktop or notebook PC. A virtual panel can be displayed on a Web browser and can even be customized for each lab or user. And not only can the front panel be replaced with HTML pages, it can be viewed remotely. Three different products can be generated from a single instrument design: the basic instrument, the instrument with an embedded Web server, and the instrument with a simplified actual front panel and the full front panel accessible via the Web browser (see Figure 1).

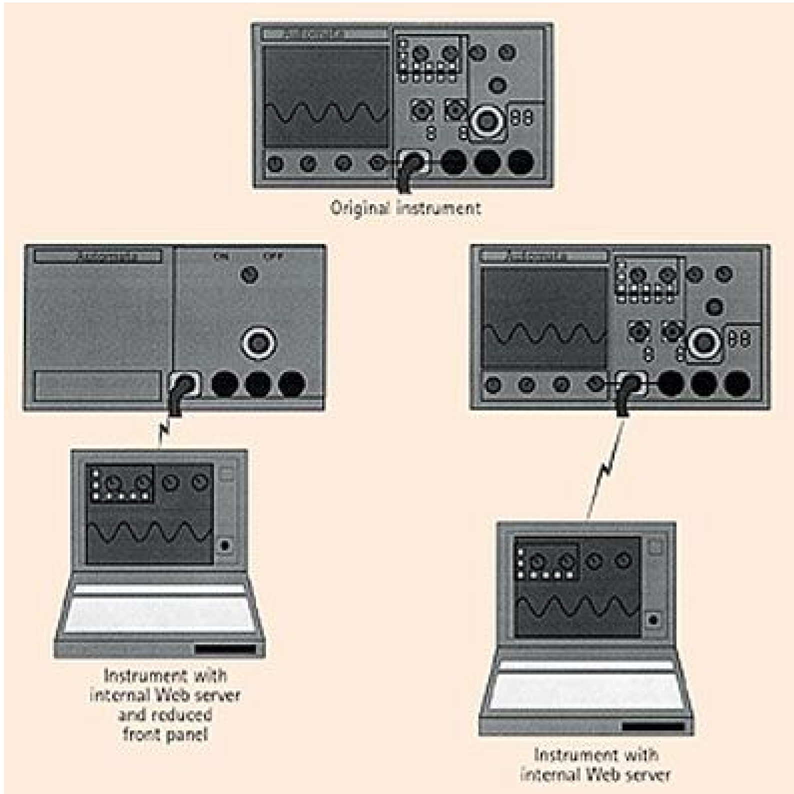

Figure 1. *Three products can be created from a single design with the use of Web technology*

The cost of the additional programming, network interface, Web server, and possible additional device memory requirement for an on-line front panel may be less than the cost of manufacturing real panels. The ability to customize a browser-displayed front panel gives the manufacturer the advantage of a new product and the user the advantage of an interface that is easy to use and accessible from a convenient remote location.

Conclusion

The information technology revolution that is the Internet is part of daily life for increasing millions of people. Wiring intelligent medical devices to the Web—equipping them with e-mail capability and enabling them to interact with remote operators through their own Web pages—is a natural step for the healthcare industry to take in exploiting its communications potential. Medical device manufacturers will come to see that to create an Internet appliance is to create a better instrument. This is a technology that will be easy to implement and easy to sell.

Edward F. Steinfeld is an embedded-computing market consultant for Automata International Marketing (Westford, MA), with more than 25 years of experience in real-time and embedded computing. He can be reached at stein@ma.ultranet.com or http://www.go-embedded.com.