E-mail See Electronic Mail.

Embedded Systems

Less than 50 years ago, an individual computer filled an entire room; the only way such a machine would be found in a car or on an airplane was as cargo. Eventually, computers shrank in size and weight and could be used as *embedded systems*. Embedded computers are generally used as controllers in other, even more complex systems. They tend to be "invisible," since the end user is unaware of their existence; the user does not need to know that a **microprocessor** is controlling an oven or a fax machine.

Embedded systems have been expanding in ability and complexity. For example, for guidance and navigation of early U.S. piloted spacecraft such as *Gemini* and *Apollo*, there were computers amid the rate gyros, accelerometers, inertial measurement units, and other items that worked together to perform the flight control functions. By the Space Shuttle era, the advent of microprocessors meant that embedded computers could spread to several dozen systems, with the flight control system being just one of them. Embedded computers are now present in aircraft (flight control, navigation, engines), automobiles (antilock brakes, sound systems), and in the home (microwave ovens, audio systems, video recorders, and cell phones).

Programming and networking embedded systems are becoming increasingly difficult, due to the diversity and range of operating systems that must interconnect. One solution is a programming language that could be executed across all platforms. In the 1990s, **Sun Microsystems** introduced **Java**, mainly as a programming language for the **World Wide Web**, but it could also be used in embedded systems. The very popular Java language supports concurrency through threads, which are implemented and scheduled by the underlying Java virtual machine. At this time, the use of only a few threads yields tolerable performance; this performance will improve with improved hardware.

It is the portability of Java that is the largest source of encouragement for its use in embedded systems. Sun's vision involves the use of embedded processors that identify themselves when connected to a network. A new **printer**, for example, could provide all computers in the network with the appropriate drivers, making it unnecessary to update the software at each node. **Jini**, the protocol proposed by Sun, and should make it possible to build complex networks of embedded systems.

FURTHER READING

Ganssle, Jack G. *The Art of Programming Embedded Systems.* San Diego, Calif.: Academic Press, 1992.

Heath, Steve. *Embedded System Design.* Woburn, Mass.: Butterworth-Heinemann, 1997.

Simon, David E. *An Embedded Software Primer.* Reading, Mass.: Addison-Wesley, 2000.

—James Tomayko

Emoticons

Emoticons are combinations of **keyboard** characters (or **ASCII** glyphs) used in communications between computer users to provide information about the writer's feelings and state of mind (see table). Many emoticons depict facial expressions and are also called *smileys*. Western emoticons are usually interpreted sideways—in other words, the page has to be turned mentally 90 degrees clockwise. Japanese emoticons, by contrast, do not have to be rotated.

There is a psychological explanation for the need for emoticons. Since vocal inflections and facial expressions are absent from online communication, other clues have to be added to the message to avoid misunderstandings. Sometimes a remark is intended to be ironic and has to be annotated as such without too much trouble. Emoticons are "online emotion"; they substitute body language by extending the online language.

The invention of the smiley is usually credited to Scott Fahlman, who posted a notice to the **bulletin board system** at Carnegie Mellon University around 1980. He later wrote: "I wish I had saved the original post, or at least recorded the date for posterity, but I had no idea that I was starting something that would soon pollute all the world's communication channels." From that original posting a seemingly never-ending flow of

EMOTICONS: A PARTIAL LIST

WESTERN EMOTICONS

:-)	Smiley
;-)	Winking
:-(	Unhappy
:-o	Surprise
:-#	Censored
:-*	Kiss
:-}	Crooked smile
:'-(	Crying
(-:	User is left-handed
:-X	User's lips are sealed
:——}	You lie like Pinocchio
:#)	Drunk smiley
:?)	Philosopher
`@;;;;;;;;;;	Caterpillar
<:~_)—	Mouse
@(_)~~~	Hot coffee
:-)(-:	Just married!
:-)x===>	A tie

JAPANESE EMOTICONS

(^_^)	Ear-to-ear smile
*^_^*	Blushing smile
(^.^)/	User is waving hello
___m_oo_m__	Spying over the wall
m(_ _)m	Deep bow used for apologizing or expressing thanks
>——(^_^)——<	A hug

emoticons has arisen. Emoticons have long been controversial among **Internet** users. Purists resent their "cute" nature and argue that one's words should speak for themselves. Preferred in some circles are acronyms, such as LOL for "Laughing Out Loud" and HTH for "Hope This Helps." (See **chat rooms** for more.)

Microsoft patented a form of user interface that may render emoticons obsolete in chat rooms. Its Microsoft Chat program depicts a chat as an interactive comic strip. The chatters select the emotions of the cartoon characters from a predefined menu. Comic-style ballons display the text of the conversation. The second release of the program even interprets common smileys typed on the keyboard and produces the appropriate emotion in the cartoon figure.

FURTHER READING

Sanderson, Doherty, and David Sanderson. *Smileys*. Sebastopol, Calif.: O'Reilly, 1993.

Turkle, Sherry. *Life on the Screen: Identity in the Age of the Internet*. New York: Simon and Schuster, 1997.

—*Raúl Rojas*

Encryption and Cryptography

Cryptography is the science of digital security and a branch of mathematics. Historically, it has been limited to encryption—the creation of secret messages—but since the mid-1970s it has been expanded to include other security-related techniques, such as digital signatures.

Encryption is as old as communication. Early written languages were a form of encryption: only those who knew the language could read the messages. Examples of encryption can be found throughout history, from 1900 B.C.E. in Egypt, through ancient Rome, India, medieval Europe, and the Renaissance, to World Wars I and II and the present day. Encryption has taken a wide variety of forms: For example, the Navajo language was used as a form of encryption during World War II.

Technically, secret languages and word substitution are called *codes*. Letter substitution (or, on computers, bit substitution) is called a *cipher*. Modern encryption concerns itself entirely with ciphers; codes are used only in very specialized situations.

Encryption can be viewed as a sender (called Alice), a receiver (called Bob), and an eavesdropper (called Eve). Alice wants to send Bob a message so that Eve can't read it. In some circumstances, Alice could meet privately with Bob, but if they can't, cryptography allows Alice to encrypt and send the message over an insecure communications link (radio, or the **Internet**). Even though Eve can intercept the message, if she can't decrypt it, she can't read it.

The method of encryption is called an **algorithm**. An example of a simple algorithm is the *Caesar cipher*. To encrypt a message, replace every letter with the letter three to the right (wrapping around Z). So A would be D and Y would be B. The encryption of HELLO would be KHOOR. To decrypt, replace every letter with