

Hospitality Information Systems: Intuitive, Object-Oriented, and Wireless Technology

by
Michael L. Kasavana

Automated information system design and implementation is one of the fastest changing aspects of the hospitality industry. During the past several years nothing has increased the professionalism or improved the productivity within the industry more than the application of computer technology. Intuitive software applications, deemed the first step toward making computers more people-literate, object-oriented programming, intended to more accurately model reality, and wireless communications are expected to play a significant role in future technological advancement.

The future of hospitality information systems will be positively impacted by the evolution of intuitive, object-oriented, and wireless technologies. In the past, successful hospitality information system implementation often was a function of the effectiveness and level of training users received. The focus of providing an appropriate level of computer literacy in order to render a more effective utilization of computer technology is changing. The recent shift toward making computers more intuitive (people-literate) through graphical user interfaces, color touchscreen terminals, bar code readers, and pen-based devices offers an opportunity to significantly enhance operational controls with far less formal systems training.

Simultaneous to the development of intuitive interfaces, information system technology has begun development of an object-oriented applications framework which enables industry experts to more accurately build computer-based models capable of approximating industry transactions, transitions, and trends. In addition, management can anticipate system networking to be accomplished without cabling (wireless). The ability to configure a hospitality system using open architecture without the physical constraints of walls, floors, other obstacles, or traffic patterns provides optimal flexibility for fitting the computer to operations rather than vice versa.

GUIs Make Computers People Literate

Advanced user interfaces have become increasingly intuitive. This feature enables users to quickly become familiar with the workings of application software while requiring correspondingly less formal training. User interfaces are beginning to play a significant role in the design and implementation of hospitality information systems. The current rate of proliferation of graphical user interfaces (GUIs) ensures that they will be an integral part of the hospitality computer system landscape beyond the next decade. A GUI, deemed the first step toward making computers more people literate,¹ is a software package that operates harmoniously with a computer's standard operating system (DOS or non-DOS). A graphical user interface contains instinctive logic, consistent appearance, and on-line help functions, making it possible for users to operate GUI-driven computer application software with little or no computer training.

Historically, in order to have a computer perform a desired function, users had to enter character commands (via a keyboard) using specific formatting protocol (syntax). Such interaction was termed character user interfacing (CUI). In a CUI application, users are provided a blank screen and a system prompt (cursor). For example, in a CUI application, a user desirous of reviewing the contents of a diskette would likely type the command "DIR" beside the appropriate system prompt, and press the "Enter" key.

An alternative procedure, involving use of a GUI, provides the user a graphical toolset consisting of icons (on-screen pictures) and menu bars (command options) for program selection and execution. A GUI allows the user to control an on-screen pointer through the movement of an external component device such as a computer mouse. By pointing to the desired computer function and activating or clicking a button on the external device, the computer interprets the input as if it had been entered through a keyboard. With a GUI application, there is an obvious keystroke savings as well as a lower user dependency on knowing and/or memorizing computer command entries. In a GUI environment diskette content can be reviewed by pointing to a graphical representation (icon) of a floppy diskette and clicking the mouse. The directory is displayed as if the user had entered the command "DIR."

The development of more advanced graphical user interfaces is especially appealing to novice computer users or persons with physical handicaps. A recent research study comparing CUI and GUI applications found that GUIs were simpler to understand and preferred by users.² In addition, GUIs helped users master the technology at a much more rapid pace, with improved productivity. By implementing application software compatible with a GUI (Windows-type) environment, the user is able to rapidly select computer commands from among a set of icon options through the use of a pointing device.

Another advantage of a GUI environment is that it enables a familiar, consistent series of screens across all similarly designed GUI applications, minimizing training time since they provide an intuitive approach to computing. GUIs enable self-teaching and exploration and thereby extend learning beyond the basic capabilities of application software.³ For example, the GUI icons, toolbox features, and screen formats of a word processing program will closely approximate those used for a GUI version of an electronic spreadsheet program. This portability feature of a GUI application helps the user who is familiar with one application to quickly and intuitively adapt to another. The consistency of GUI screen design allows users to become proficient with a significantly reduced learning cycle, thereby necessitating reduced training time. This factor is likely to be of major importance in the future design and implementation of hospitality information systems.

This feature makes it easy to cross-train a front desk agent already familiar with guest registration software to serve as a reservationist. Since the screen formats and icons in both functional areas will be similar, the user should be capable of adjusting to the complexities of the reservations program with relative comfort. Related developments that further render the computer more intuitive include touch, coded, handwritten, and speech recognition.

Touch recognition involves users entering data into a computer by merely touching a location on a special screen. The user does not need a keyboard or mouse-like device to record data entry or initiate commands. A touchscreen system (TSS) has three important components: a sensor built into a specialty computer monitor that is touch-sensitive; a controller responsible for evaluating the signals initiated and transferred through the sensor; and special software which translates touch information into data or commands that can be processed by the computer. Touchscreen terminals are especially appealing to users with limited typing skills, or physical handicaps, or those interested in simplified input procedures. Since touch-sensitive screens can move large quantities of data easily, they can be especially effective in a variety of hospitality industry applications, such as food service order entry, hotel guest information systems, and multi-unit information analysis.⁴

Bar Code Scanners Speed Transactions

Bar codes serve as computerized labels that allow information to be entered automatically into a computer system, thereby speeding transactions and improving asset management. A bar code symbol's dark bars and light spaces, of varying width, represent elements of a computer binary language. A special scanning device is used to decipher the code by sweeping a small beam of light across it. The dark bars reflect less light than the spaces, and the scanner recognizes pattern variations. The scanner basically decodes the pattern into electrical signals for computer recordation. Data captured through a

bar code process can subsequently be transferred to application software for further processing.

Bar code readers scan codes and translate them into digital character information. In order to read bar codes accurately, scanners must be able to cope with a large degree of variability in the appearance of symbols.⁵ Although bar code symbols fall within certain structural guidelines, they vary in size, color, and print quality, depending on the type and texture of the material on which they appear. Sophisticated decoding algorithms minimize code recognition discrepancy or distortion through computer interpretation of the bar code's pattern. Variations in symbol size are permissible because scanners measure the relative differences between a symbol's bars and spaces, rather than comparing them to an absolute scale or predetermined dimension. Hence, although one bar code may have individual modules twice as wide as another, the scanner will decode both correctly, adjusting its reading in accord with the standard module width established for each separate symbol. Even different colors will not alter the reading so long as the colored bars reflect less of the light beam than do the spaces. If this were not the case, then the scanner would be liable to misinterpret a zero for a one, or vice versa. It is anticipated that numerous hospitality information system processes will be aided by bar code technology within the near future.

Handwriting Recognition Is Possible

The computer's ability to decipher and process input captured through hand-written documentation is possible through pen-based computing. Pen-based systems rely on input generated by an electronic pen (stylus) on a special film-like surface. When the pen writes on the pressure-sensitive display screen, the electrical voltage generated as input is evaluated and translated into pen position and data entry, similar to touchscreen sensor technology. When digitized characters are received by the computer, one of two subsequent functions will occur; input will either be converted to digital signals for further processing or it will be kept in raw form for future reference.

Whenever a user is entering data to be converted and stored as text in a predefined application, handwriting recognition software strives to match input characters with models stored in the system's database. For example, for an application to a food service order entry point-of-sale (POS) application, input characters match stored models, then the computer will be able to convert handwritten input into digital characters. In a POS process, these digitized characters can then be transmitted to remote workstation printers in a food production area. Handwritten character recognition that is clarified and defined on input can be added to the character database, along with its identity. By employing such an approach, the user can expand and customize the system's

database. Through database updates the user "teaches" the application software to recognize and process unique, individualized writing. In effect, creation of a character database enables the computer to "read" the input more quickly and efficiently. When a pen-based application does not recognize input characters, the user may need to supplement data entry through the use of a keyboard, or store the entry in its original form.

When data is captured and stored in original form, i.e., a non-digitized image, the technique is termed "inking." In the process, application software treats input as a picture and stores captured data without translation. This form of pen-based computing is especially appropriate when an application requires a signature, a notation, a commentary, or storage of some other form of an original transaction. For example, a POS credit card settlement transaction might involve the customer signing a hand-held portable display voucher. The eatery, once having the customer sign his/her name, could then store the captured signature as proof of authorization and verification. The same could be true for a guest signing at check-in. The signature provides a base of control over transaction authorization while ensuring a high level of security.

A major advantage of pen-based systems is that user productivity tends to increase. Users who for one reason or another, are unable or incapable of typing can still communicate with a computer. Also, data entry using a stylus is often much quicker and more accurate than pointing with a mouse or searching for a key on a keypad. In addition, pen-based computers are small, portable, and durable and allow users to enter data while standing, sitting, or moving. This may be of special importance to the labor intensive hospitality industry and the employment of handicapped individuals.

Digital media devices, recently introduced in the computer marketplace, may be the most promising future form of pen-based computing. Based upon revolutionary operating system technology and free-form database construction, personal digital assistants (PDAs) are basically hand-held, pen-based units that support an easy-to-use intuitive interface.⁶ In fact, even palm-sized PDAs offer pen-sensitive screens that also double as liquid crystal display (LCD) output screens. It is anticipated that pen-sensitive touchscreen surfaces will also be available.

Although small in size, PDAs possess the power to recognize handwritten and graphic input. Handwritten entries are typically converted to formatted digital text, while rough sketches can be stored as graphical maps and diagrams. In fact, some developers describe PDAs as akin to an electronic cocktail napkin, in that notes and doodles could become digital input for system storage.⁷ In addition to containing intelligence for advanced applications, PDAs may also serve as wireless communication devices, thereby enabling mobile access to a variety of media and data. PDA usage within the

hospitality industry is expected to be quite high, especially if voice communications can be added to these portable devices.

Integrated Intuitive Applications Are Next Step

Technological research is expected to produce integrated intuitive applications for the hospitality industry. Recently, graphical user interfaces and pen-based applications were integrated on an experimental basis. The GUI environment, which supports pen-based recognition applications through its programming workbench, was found to be capable of accepting and translating handwritten input. The software also included the capability for database expansion through user interaction. Due to its simplicity, and the fact that the pen provides both superior cursor-positioning and data entry abilities (compared to mouse devices), many system designers forecast that the pressure-sensitive stylus is likely to replace the mouse as the standard pointer-control device.⁸

By making handwriting recognition a standard option for many users, the pen-based market will be a significant factor in future hospitality information system applications. There are many opportunities for hospitality firms to use pen-based computing to support both their corporate and unit-level operations. For example, capturing a hotel guest's signature at check-in and then, subsequently, comparing authorization signatures at various point-of-sale outlets could help establish transaction authorization.

Electronic Mail Services Increase

The old saying "a verbal contract isn't worth the paper its printed on" is being replaced with "put it in writing...but not on paper!" The hospitality industry, like many other businesses, has become inundated with volumes of paperwork. Solutions to this paperwork shuffle include electronic mail (E-mail), voice mail (V-Mail), and facsimile mail (F-mail) service. The age of electronic mail is expected to increasingly become a major factor for business correspondence at and between corporate and unit-level operations.

Similar to the high volume of paperwork, telephone traffic has also led to the identification of an alternative to live attendant services. The future potential of voice mail (V-mail) systems, which can answer, direct, and respond to incoming calls, is only just beginning to be recognized by the hospitality industry.

Combinations of the best features of many telecommunication techniques are helping point vendors toward a refinement and promotion of facsimile mail (F-mail). Mathematically, the equation of E-mail plus V-mail may well equal F-mail, which is similar in concept to V-mail in that it is based upon an electronic store and forward principle. The F-mail mailbox (store) can be used to receive a transmission while the user can then direct printing at any fax machine or remote printer (forward). F-mail and V-mail can share the same mailbox and, hence, are referred to as fully

compatible and fully integrated systems. The main advantage of F-mail is that it enables the receiver to direct the printing of a document to a remote location.

A facsimile transmitter enables the user to send print, graphics, or photographs by telephone. The user can simply place an original document into a fax machine (source), and then dial a FAX machine at the desired location (destination). The receiver can then be placed into an acoustic coupler and the transmission accomplished. In a business environment like the hospitality industry that thrives on speed, fax may well be termed the fast food of communications. A document can be distributed in minutes, not hours, days, or weeks.

Hospitality information systems will continue to support enhanced electronic mail services such as E-mail and V-mail, but may also adopt F-Mail. F-mail can be extremely important for persons who travel frequently, for multi-unit management, and for maintaining confidentiality of communications.

Speech Recognition Functions Can Be Helpful

The area of automated speech recognition (ASR) depends on verbal interfacing, the use of voice to issue commands (voice control) and elicit computer verbal responsiveness. Since people can talk much faster than they can type (200 wpm speaking versus 45 wpm typing), verbal interfacing is an efficient alternate data entry mode which can be especially beneficial for non-typists as well as persons with disabilities or persons employed in work environments in which typing is difficult or infeasible.

For the computer to understand spoken commands, it must be capable of translating speech into action. This translation process involves two sub-processes: speech processing and digital signal processing. Speech processing involves the translation of analog (continuous) voice signals to digital (discrete) signals the computer understands (speech recognition). When a message is recorded, the analog signals of voice are translated into digital signals and entered into the system (input phase). When the computer needs to send a message to the user, the process is reversed (output phase) in that the digital signals are transformed into analog sounds the listener can understand.

Once the computer receives a digital signal packet, it must analyze its content to interpret and act upon its directive. Digital signal processing involves analysis of the frequency and acoustics of the signal of the message received. Evaluation is accomplished using specialized digital signal processing (DSP) chips, normally housed on an add-in computer-system board. Once a digital signal is analyzed it is compared to prerecorded sound signals, called models or templates, stored in an ASR database. Speech models are created when the user records certain words or phrases in the computer's memory. Speech models are captured using a specialty headset or microphone. If a

match between a received command and a speech model is found, the computer is then directed to take a specific course of action. There are basically two types of speech recognition systems: small vocabulary and large vocabulary systems.

Small vocabulary systems are very limited in that they contain only several hundred words. By contrast, large vocabulary systems may contain up to several thousand words. In small vocabulary systems, the speech model database is typically created by the user with the system using template matching for entire words or phrases. Such systems are classified as speaker-dependent; each person using the system must train it (record their own personal database) with his or her voice. In large vocabulary systems, speech model databases are created by the system supplier and included as part of the system software that is shipped with the system. Broad application models for large vocabulary systems are developed using a wide variety of male and female voices with regional accents. From such a diverse pool, the system is capable of constructing a broad-spectrum speech model based on an average voice. Based upon method of construction, such systems are called speaker-independent. A speaker-independent system does not require its users to personally train the system to their particular speech patterns. Due to the extensive nature and scope of their database, large vocabulary systems often pause slightly between words. This is unlike speaker-dependent systems which work without hesitation or interruption.

Regardless of vocabulary type, automated speech recognition systems match speech against word models in their database. Although they are carrying out actions based on the contents of those models, it's not the same as the computer directly interpreting questions or comments.

Object-Oriented Applications Are Alternative

In nearly every computer programming language, software combines with hardware to accomplish a desired task. Early programming languages involved storing data in a computer file and then giving the computer a sequential set of instructions for handling it. Hence, programming involved two separate processes: data and procedures. For nearly two decades computer programming has experienced extraordinary revolutionary and evolutionary growth. Unfortunately, many software developers believe that traditional programming languages are no longer adequate since they render the task of programming more complex and time-consuming than need be.

One of the major weaknesses of the conventional programming languages is that they are often very difficult to debug, alter, or modify. Traditional programming techniques rely on an imperative approach consisting of sequential, step-by-step instructions. In an imperative language, a command is given to the computer to perform

some task. The basic underpinning of an imperative program is that as the program progresses, it will pass through many states of control, all of which is understood by the software engineer. Given the complexities involved in newer hardware systems and the shortcoming of numerous programming techniques, an attractive alternative to imperative style programming has emerged: object-oriented programming.⁹ Object-oriented programming languages, some of which have been in existence since the late 1980s (e.g., Simula, Smalltalk, C++, etc.), are beginning to receive a disproportionate amount of attention in the information systems community. An object-oriented programming language involves the construction of software in modules, called objects, each of which contains data and a set of data manipulation operations.

Management information systems have traditionally relied on computers to conduct simulations of business events for the purpose of identifying inefficiencies. The problem with such an approach is that traditional sequential programming does not always lend itself to effective simulation. Since sequential programming is based upon a series of step-by-step instructions linked together to form a subroutine, which can make practical simulations more difficult. In other words, the act of making even slight adjustments in a traditional simulation often requires reconfiguration of much of the original program. This is not true for object-oriented programming.

Object-Oriented Programming Is Powerful Force

Although a long-time alternative for sequential programming techniques, object-oriented programming has only recently been recognized as a powerful future force. Object-oriented programming flows from the idea that since people perceive the world as being composed of objects, computers ought to address objects, too. Objects are typically identified as independent entities which may or may not interact with one another. Unlike sequential programming which represents a continuous stream of instructions, object-oriented programs organize data and procedures into small chunks (objects), each of which denotes a discrete element or aspect of the situation being simulated. Program chunks are then assembled, disassembled and/or rearranged by the programmer (similar to building blocks). Each chunk can be tested or modified individually of another chunk. Given that each object is independently established and programmed, contributes to the simplicity of development of even large and complicated models. Objects can also be imported between programs with little or no modification requirements.

While many information systems have been oriented toward artificial intelligence and expert system technology, object-oriented programming techniques appear to possess the potential to revolutionize application software. For example, consider a computer

model of a home delivery pizza outlet. While the production of a pizza (for delivery) involves a series of individual tasks, determining the best strategy by which to produce the final product can often be highly complex and perplexing. Object-oriented programming of an assembly-line like pizza production kitchen would involve development of linked objects. In developing a computer simulation of such production, a hierarchy of related objects is identified. From among these objects the programmer links the program parts (objects) much like the food service operator links personnel and job tasks.

Every object in a program is itself a mini-program, complete with procedures to accomplish a specific task. In order to simulate an actual process, objects must be capable of interaction. In order to interact, objects share instructions directed toward specific procedure capabilities. One strength of object-oriented programming is the fact that messages are sent between distinct objects. There is no permanent link written into the fabric of the program, like there would have to be in a sequential program. Altering or removing an object within an application has no effect on other objects, so long as all requisite messages get responses. In essence, objects exchange status information, which helps them determine the course of events and helps reveal the efficiency of a process being simulated.

The fact that the hospitality industry can be easily segmented into distinct objects for modeling appears to be a natural extension of object-oriented programming. The area of hospitality information systems may well be revolutionized by the implementation of an object-oriented environment.

Wireless Computing Removes Obstacles

The development of wireless computer systems only recently became technologically feasible. By rendering a system wireless, system designers have removed many of the obstacles to greater system flexibility and expendability. The use of radio frequency (RF) spectrums has enabled system manufacturers to transmit information without the constraints or expense associated with cabled systems. Wireless communications involve the substitution of cables with radio frequencies and/or satellite technology to carry information between computers. In exchange for providing maximum flexibility, wireless systems rely heavily upon RF transmission. The cost of cabling can be a significant hidden expense to traditional system installation and hence is an important system consideration. In addition, also important is the fact that currently wireless systems have trouble transmitting data at even half the speed of cabled systems. This is expected to change, however, in the near future.

There are at least three types of wireless transmission techniques available: infrared transmission, microwave-based, and

spread spectrum technology.¹⁰ An infrared transmission configuration employs infrared light waves to transmit signals throughout a network. Microwave-based technology relies on high-frequency radio communications at a comparably low power consumption. Spread spectrum radio transmission divides the radio signal into multiple channels to cover a broader bandwidth.

One of the primary goals of wireless communications is to make the user unaware that the device they are using is wireless. A network adapter card typically is installed into each component system device. This card emulates antenna signaling for data transmission to similarly-equipped machines. Typically, communicating units must be positioned in a clear sight line and normally cannot be placed more than 1000 feet apart. Communications occur in a frequency band just outside that reserved for cellular telephones. Radio frequency (RF) technology avoids the hassles and expense of conventional wiring by using radio wave transmission. Radio waves are characterized as the least energetic form of electromagnetic radiation, with the lowest frequency and longest wave length. It is important to note that RF devices must transmit radio waves to a receiver that interfaces to a base station. The base station, in turn, must be connected to a computer for subsequent communication and data capture to occur.

There are several advantages of wireless technology, including ease of installation (wireless networks are easier to install and are much more flexible in design than cable-dependent systems); data integrity (integrity and security are protected by the unique frequency of data transmission); cost savings (installation expenses for a wireless system will result in a dollar savings as a trade-off for cabling); and equipment mobility (since cabling is eliminated, hardware configurations are highly flexible).¹¹ From a disadvantages standpoint, wireless technology needs to overcome such factors as potential signal interference (a data transmission path cannot be obstructed, communication from sender to receiver must be in a clear line-of-sight); system expense (even when a system serves a large number of users, a wireless network can be difficult to cost justify); FCC licensing (some wireless communication networks require users to secure an FCC license to protect the system from electrical interference); and speed of transfer (a wireless network is usually not capable of data transference at rates equivalent to conventional technology).

There are a number of palm top and notebook devices with wireless communications capability. Basically these devices consist of liquid crystal display screens, a keyboard consisting of alpha and numeric keys, and a microprocessor. Inclusion of the microprocessor allows compatibility with operating systems (such as MS-DOS) as well as the capability to run programming languages. These portable devices are primarily used as data entry devices and can be enhanced to transmit data through RF technology.

Currently, data entry takes place through a keyboard or bar code scanning device and transmission results from antenna signalling. Future possibilities include portable touch screens, color screens, voice recognition and pen-based computers. These devices may offer unparalleled wireless connectivity to the hospitality industry in the near future.

Neural Networks Are Not as Fast as Brain

A neural network is an information processing system that is modeled after the human brain's interconnected system of electrically activated neurons. In essence, the term is intended to describe a computer that simulates the activities of the brain. Neural networks are characterized by the fact they are capable of accurately learning. The learning function is an outgrowth of pattern recognition and sorting as well as trial and error as opposed to the programming of decision rules in an expert system. Neural networks have three characteristics: they rely on artificial neurons (nerve cells); each neuron connects to at least one other neuron; and they are based upon a learning model.

Similar to how a computer may use bits to accomplish the processing of large amounts of data, the brain uses neurons to perform equivalent functions. Both the brain and a computer operate using the binary logic of two states (on/off). Unlike normal computer programs, but like the human brain, neurocomputers establish a problem-solving algorithm when given the examples of inputs and required outputs.

Neural networks learn using two methodologies: supervised learning (also referred to as back propagation) and unsupervised learning. Supervised learning involves training neurocomputers by having it compare its answer with answers already programmed into the computer. The computer then strives to change its data processing until its results correspond to those of the computer. Programming a computer using structured learning requires the input of the characteristics of a problem using case studies (empirical) and then letting the computer learn from the cases. There needs to be an external teacher who possesses the correct answer.

In an unsupervised learning mode, a neurocomputer may be used to analyze inputs and store them in a pattern which then must be assigned a value (by the user). The motivation for unsupervised learning is that a lot of this type of learning takes place in the market. While neurocomputers are capable of learning, they are unable to learn everything or process information as quickly as the human brain. Unstructured learning is especially valuable in cases where the outcome is not predetermined or known.

The strength of neural networks is pattern recognition. Their weaknesses are deductive reasoning and mathematical computations easily performed by conventional computers. Neural networks,

good at determining the financial indicators in a pile of stock market data, are unable to accurately balance a company's books. Neural networks may well replace the expert systems that hospitality information system specialists currently struggle to develop and refine. For example, a reservation system is built upon a series of known procedural steps, most of which are objective in nature. A neural computer process could be used to monitor the equation and allocation of available accommodations with requested space. By automating a significant portion of the process, a much more productive and accurate conclusion may be determined.

The wide variety and acceptance of advanced user interfaces provide a basis for prognosticating that the trend toward making computers more people literate so people won't have to become more computer literate is expected to continue. Computers are simply becoming more intuitive and therefore more readily accessible to a broader array of users.

As technology continues to improve and applications take advantage of new interfaces, the hospitality industry will be able to use these advances to improve both corporate and unit-level operations. From having guests use voice commands to regulate room conditions, to storing digitized guest signatures, to having touchscreen kiosks act as self-service concierges, the options for hospitality applications will be significantly enhanced. Hospitality companies that employ advanced user interfaces may well benefit from superior guest services, increased operational control, and reductions in personnel.

Object-oriented applications together with the development of sophisticated wireless computer communication systems may greatly enhance hospitality information system computing. While object-orientation environments enable the construction of more responsive and variable sensitive computing, they also render less complicated software schemes. A major advantage of wireless technology is that it provides users with maximum hardware configuration flexibility.

Future computer system applications are likely to be more intuitive-based, object-oriented, and wireless. As a futuristic alternative, neural networks may replace operating system offerings and significantly impact on hospitality technology.

References

¹M. L. Kasavana and J. S. David, "Making Computers More People Literate," *Cornell Quarterly*, (April 1993), pp. 69-76.

²"World of Windows," *PC Novice Premium Issue*, (Lincoln, NE: Peed Corporation, 1992), pp. 5-6.

³Howard Eglowstein, "Keyboards Without Keys," *Byte Magazine*, (July 1992), pp. 287-290.

⁴M. L. Kasavana and J. J. Cahill, *Managing Computers in the Hospitality*

Industry, 2nd ed., Educational Institute of AH & MA, (December 1992), pp. 161-163.

⁵M. L. Kasavana, "Insights Into Bar Coding and How It Works," *Computers, Foodservice and You*, (November/December 1991), p. 3.

⁶Wes Thomes, "The Future of Computing," *Desktop Communications*, (July-August 1992), pp. 29-32.

⁷Nicholas Baron, "The Future of Pen Computing," *Byte Magazine*, (March 1992), p. 115.

⁸Hon Udell, "Windows Meets the Pen," *Byte Magazine*, (June 1992), pp. 159-168.

⁹"The Software Challenge," *Understanding Computers*, (Alexandria, VA: Time-Life Books, 1991), pp. 66-79.

¹⁰M. L. Kasavana, "Computer Link: Radio Days," *Restaurant Business Magazine*, (June 10, 1991), p. 82.

¹¹*Ibid*, p. 82.

Michael L. Kasavana is a professor in the School of Hotel, Restaurant, and Institutional Management at Michigan State University.