

E-Commerce Unleashed

January 22, 2001
By Peter Rysavy

Dot-com businesses may be falling by the wayside, but the percentage of commerce happening via the Web continues to grow. Meanwhile, hundreds of millions of cellular subscribers worldwide are gaining access to new wireless data choices, including Web-enabled cell phones, handheld computers and PDAs. Combine these two facts, and you get an industry with an unbelievable potential called mobile commerce, or m-commerce.

Not convinced? How about this: A woman is looking at a DVD player in a store and is ready to buy, but she isn't sure if the store is offering the best price. Using the microbrowser on her cell phone, she quickly determines that your online enterprise is selling the player for 10 percent less; she purchases it from you right then for delivery the next day. Or, you send your subscriber an alert on his Palm to let him know tickets have just gone on sale for one of his favorite performers, before the subscriber even knows the singer is coming to town. Within minutes, the customer selects the best seats in his price range and purchases tickets from you.

Whether this convinces you or not, International Data Corp. forecasts that \$21 billion worth of mobile commerce will take place in 2004, while Gartner Group predicts that 40 percent of B2C (business to consumer) e-commerce in that same year will occur over wireless connections. Already, the big e-commerce players, such as Amazon.com and Yahoo, are offering mobile options. What's not to love? Actually, plenty.

Content and applications must still be developed in multiple mobile formats. Security questions continue to dog the industry. And evolving payment systems and networks are barely up to the task. In fact, the wide range of wireless networks, markup languages and devices has made developing mobile-commerce applications so complex that most organizations seeking to mobilize applications need help.

Fortunately, help is available from a rapidly expanding array of middleware solutions, wireless portals and wireless ASPs (application service providers). As part of our investigation of this industry, we sent out a detailed RFI (request for information) to middleware vendors and ASPs to learn about their approaches for enabling mobile-commerce applications.

Evolution of an Industry

The first big wireless applications, deployed in the early 1990s, were field service and dispatch. Both applications facilitate commerce but do not involve transactions. We focused on mobile-commerce applications that involve actual transactions, in which a user securely purchases or sells goods or services. Current choices are a tiny subset of what will become possible with new location technology, financial settlement

systems, devices and networks.

Here's how we see the market evolving:

- This year, companies will extend their e-commerce applications to mobile devices but will rely on existing settlement systems, such as charging a user's credit-card account. Examples of such applications are financial trading, buying tickets, ordering from restaurants, updating financial portfolios, conducting banking transactions (such as transferring funds between accounts) and comparison shopping. Financial-trading applications are leading the way, as customers can derive clear and immediate benefits.

- Now available in limited form, electronic wallets will facilitate transactions by providing a centralized way for users to maintain account and shipping information. Electronic wallets, whether hosted by portals, ASPs, banks or carriers, will play an increasingly important role in mobile commerce. Major sites, such as Yahoo, already support wallet mechanisms. Credit-card companies also are actively involved in this area. Expect dozens of such systems to crop up by the end of the year, which might cause some confusion among application developers and customers.
- Around 2002, new location technology will enable mobile-commerce applications that take a user's location into account. Based on user preferences that address privacy issues, these applications will give users access to localized and personalized information. For example, after dinner at a restaurant, someone could use his or her PDA or cell phone to get a list of movies playing at the closest theater, obtain show times and purchase tickets.
- Beginning around 2002 but evolving over the decade, new financial settlement systems will allow secure transmission of electronic cash on a wide- or local-area basis. ECash Technologies is one of the pioneers in this area. To buy a Coke from a vending machine, for instance, a user will be able to press a couple of keys on a mobile device to transfer electronic tokens -- the equivalent of virtual coins -- to the machine over a local wireless connection such as Bluetooth. This capability will apply to in-store purchases as well. In effect, mobile devices could begin to replace cash, checks and credit cards. The implications for everyone -- merchants, consumers and banks

alike -- are huge.

What devices will people be using? Initially they'll use smart phones with microbrowsers -- typically based on HDML (Handheld Device Markup Language) and WAP (Wireless Application Protocol) -- and handheld computers with wireless modems. But these categories are blurring, with cell phone modules available for handheld computers and OSes being built into cell phones. In fact, the multiplicity of devices -- not to mention physical differences, such as screen sizes -- makes developing content a big challenge.

What will drive mobile commerce will be a billion users of mobile networks within several years, with every one of those users interested in buying products and services more conveniently. But what's convenient to users spells challenge and complexity to merchants and integrators. And we mean technical complexity. We're not even going to touch the business complexities of developing profitable m-commerce applications.

Architecture and Protocols

A mobile-commerce application can be broken down into three parts: the user device, the Web site hosting the application and the payment mechanism. In addition, various gateways come into play. In the simplest model, a Web site would deliver HTML directly to a mobile device, and a protocol such as SSL (Secure Sockets Layer) or TLS (Transport Layer Security) would secure communications. To settle transactions, the Web site would communicate with a financial network either directly or via a third party. This structure merely replaces a standard desktop browser with one operating on a mobile device and a wired link with a wireless link; in fact, HTML browsers are available for Palm and Pocket PC devices. But most deployments are more complicated than this because of slow wireless-link speeds and multiple network and device types.

Most mobile-commerce application developers want to support as many mobile devices as possible. One approach might be to format the content in WML (Wireless Markup Language), which would make the application available to mobile devices with WAP browsers. And WAP browsers are also available for handhelds. What's wrong with this picture? Plenty. Supporting all the different types of available mobile devices requires either an extraordinary amount of development effort or a third-party solution (see "The Trouble With WAP," below).

Sorting through all the third-party solutions is a project in itself, as we learned through our RFI, but with due diligence, you can find excellent tools that vastly simplify the process of developing mobile applications. Just don't underestimate the amount of work this requires. Installing and evaluating these platforms are comparable in time and difficulty to working with a new computer operating system.

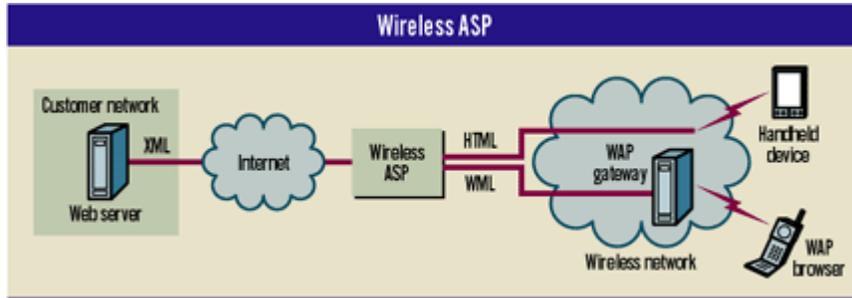
These third-party solutions are middleware platforms that typically translate from XML/HTML format located at the customer site to the appropriate wireless format, whether HDML (a precursor to WAP), HTML, WML or a format understood by interactive pagers. You can deploy these platforms yourself by purchasing a server license, or the vendor can host the platform as an ASP. The platform can also be hosted by a wireless ISP, such as GoAmerica or OmniSky Corp. (The "Wireless ASP" diagram, below, shows the ASP scenario.)

Unfortunately, the conversion from XML or HTML to WML is far from automatic. With nearly all these third-party solutions, the XML or HTML content must be created from the ground up and adhere to strict rules and templates, often in conjunction with SDKs and tools provided by the vendor. Because of a lack of standardization, the rules and templates are specific to each ASP. Thus, if you design using OracleMobile's Portal-to-Go XML, you'll be limited to OracleMobile's platform to translate the content to the various wireless formats.

Converting from XML to mobile formats will be much more reliable and flexible than starting with HTML. HTML conversions essentially involve "screen scraping," in which the middleware platform captures portions of the Web page to create mobile content. This approach is highly vulnerable to changes such as new page layouts. With XML, application developers can use specific tags to create mobile content that the middleware can interpret in an unambiguous, consistent fashion.

The middleware can perform other functions as well, such as delivering mobile content using transport protocols better suited to wireless. ("Typical Protocols Involved With a Wireless ASP or Middleware Platform," below, shows the protocols involved.)

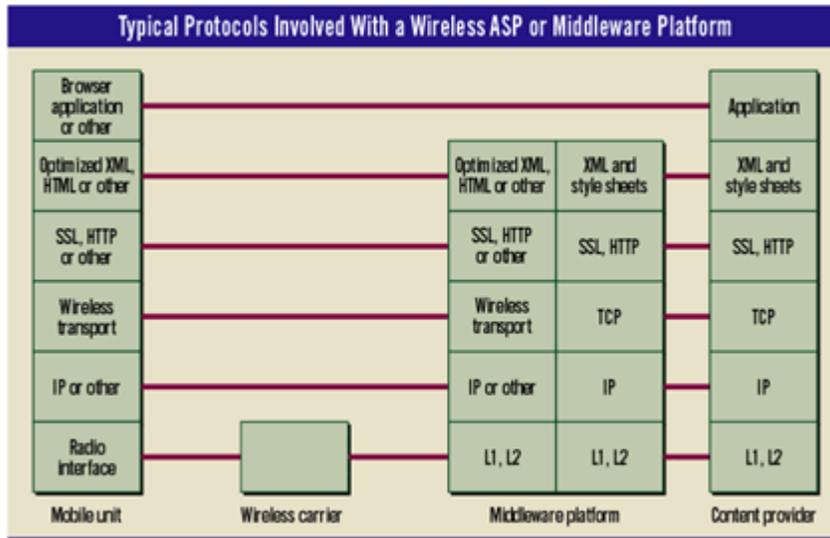
This architecture is technically efficient, as it enables standard Internet communications between the middleware platform and the content/application provider. Application developers can deploy their applications on existing Web platforms. The architecture also allows the middleware platform to deliver optimized content to the user with the most appropriate protocols for the connection. However, the solutions are vendor-specific and require a considerable commitment to a middleware platform or ASP. Many of these ASPs are new, and their long-term viability is unproven. In fact, many wireless middleware companies have come and gone over the past five years.



Increasingly, major software vendors are adding capabilities to their Web and database platforms to support mobile formats. This could eventually obviate third-party solutions, though such solutions are likely to have broader network and device support for quite some time.

Watch also for the role that Internet portals will play. These portals are in a strong position to integrate, and eventually subsume, the ASP function. Furthermore, they can provide a place for merchants to situate themselves. America Online, InfoSpace, MSN and Yahoo are ramping up their mobile emphasis and already command intense customer loyalty. For many users, the mobile environment will become an extension of their existing Internet environment. Some wireless portals, such as GoAmerica and OmniSky, are focusing on the mobile environment exclusively. It makes sense for these trailblazers to start providing financial services such as e-wallets or gateways to e-wallets.

Finally, various industry organizations are seeking to standardize mobile-commerce methods and architecture; examples include Fundamo, the Global Mobile Commerce Forum, the Mobey Forum, the Mobile Electronics Transactions initiative, Radicchio and the Wireless Data Forum. So far, these organizations are long on stating the problems and issues involved, and short on delivering actual solutions.



Security: A Pain in the ASP

The wireless model we've described raises a serious security concern, since the ASP might decrypt messages received from the mobile unit. Some ASPs can pass on messages without decrypting them; you should investigate this aspect with vendors of interest. As for the security protocols used between the mobile unit and the wireless ASP or middleware platform, typical protocols include SSL with 3DES encryption. However, many Internet authentication and encryption methods are computationally intensive. Companies such as Certicom Corp., Chrysalis-ITS and Diversinet are developing security solutions based on new methods -- elliptic-curve cryptography, for example -- that are better suited to small devices. The wireless ASP or middleware platform can then translate from these mobile-specific security protocols and conventional Internet security protocols.

Isn't the radio link also vulnerable? A sophisticated eavesdropper could access the wireless data stream, though doing so wouldn't be easy. The security protocols employed above the transport layer, such as SSL and WTLS (Wireless TLS), provide most of the protection. In addition, some wireless networks, such as CDPD (Cellular Digital Packet Data) and GPRS (General Packet Radio Service), encrypt the data stream below the network layer.

In the case of WAP and HDML, an additional tier is added to the architecture, namely a gateway operated by the carrier, as shown in "Wireless ASP Supporting Content Through a WAP Gateway," above. This tier raises additional security questions.

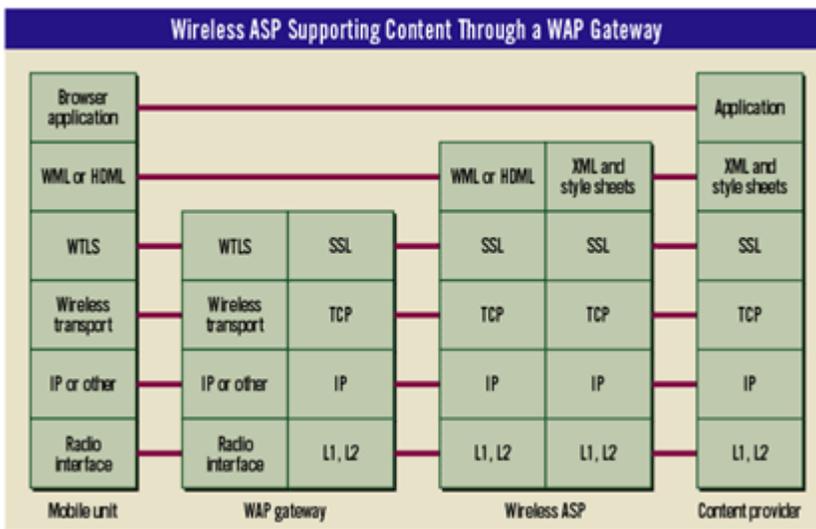
This gateway translates between SSL (or TLS) protocols and the WAP-specific WTLS protocol, resulting in all communications being decrypted and re-encrypted. Future versions of WAP will offer end-to-end security, but for now, customers must decide if they trust the wireless carrier. The gateway provides a set of wireless-optimized transport protocols as well.

You'll also need to think about user authentication. Certainly, users can enter passwords when prompted. SIM (subscriber identity module) cards, now used in GSM (Global System for Mobile Communications) cellular networks and planned for other networks, are another method. These cards uniquely map subscribers to physical cards, which the users can insert into various devices. You can then augment this form of hardware authentication with password authentication. And with electronic wallets, users need never transmit sensitive information. The bottom line is, ample mechanisms are available to secure m-commerce applications, but you must carefully design such mechanisms into your application, and you must understand your partners' security operations.

Settlement and Payments

Most of today's mobile-commerce applications are extensions of Web-based e-commerce solutions, which typically use credit cards for payment. However, as these applications mature and the variety of mobile-commerce transactions broadens, settlement systems will have to change. Simply put, we need generally available, secure electronic-wallet and electronic-cash mechanisms. This will take years, but many approaches have already been introduced.

To make purchases using many of today's Web-based commerce applications, customers must enter their credit-card information -- clearly an unwieldy approach with a mobile device and one that raises security concerns. If the credit card is already on file with a merchant, all the user has to do is authenticate himself or herself. But users don't want to have to register their credit cards with numerous merchants.

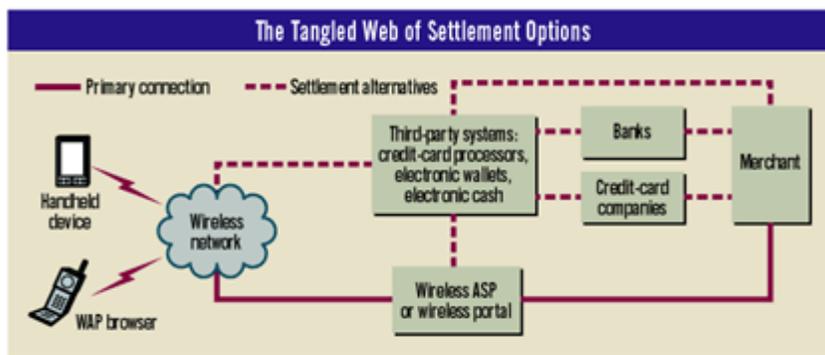


Credit-card companies and other organizations are working to solve this problem by developing electronic-wallet mechanisms; these hold users' financial information (such as credit-card accounts) and perhaps shipping info, and may contain funds as well. By supporting a particular

wallet system, you can give your customers a convenient way to pay for purchases. Some wireless ASPs, such as Snaz Commerce Solutions, host such a wallet mechanism and provide a mobile shopping portal for merchants to plug into. "The Tangled Web of Settlement Options," below, shows the various possible interconnections among wireless ASPs, wallet systems, banks and customer networks.

Wireless carriers also are in a position to play a strong role. They already have billing relationships with customers and could handle billing for merchants. NTT DoCoMo in Japan does this today with its i-mode service, for which the carrier collects 9 percent from merchants. On the other hand, this role is beyond many carriers' comfort level, since it exposes them to credit risks better handled by banks.

One reason for the initial popularity of financially oriented m-commerce applications is that the financial institution already has users' account information on file and is simply allowing them to manipulate their accounts -- letting them buy or sell stocks, for example, or transfer funds from one account to another. The challenge is to allow a user to easily purchase goods or services from any merchant or institution in a secure and cost-effective manner.



Internet economics demands a reduction in the number of middlemen involved. Today's payment systems -- credit-card companies, for instance -- involve a number of intermediaries and will not scale to high volumes of smaller transactions. This is where companies such as ECash will play a role. ECash's payment architecture lets banks offer electronic-cash services to customers, who can then spend that cash with participating merchants.

Although in their infancy, such payment infrastructures could truly revolutionize mobile commerce and, in the process, massively disrupt existing financial systems -- even making ATMs obsolete. However, this will all take time. Fraud levels in e-commerce are 12 to 18 times higher than in conventional commerce, according to Gartner Group, so banks will proceed cautiously. But if they don't proceed, they will be relegated to irrelevance.

Another significant development that could bypass carriers entirely is local-area wireless connections. In this scenario, customers establish direct connections from their mobile devices to the merchants' point-of-sale systems at the merchants' locations. Palm and Hewlett-Packard Co.'s VeriFone recently agreed to launch exactly such an initiative. The infrared link simply replaces the credit-card swipe, and the merchant uses its existing back-end credit-card payment system. This approach completely sidesteps the architectures discussed above.

Speed Bumps

Despite this market's potential, numerous speed bumps exist. Most important, no magic elixir can convert existing Web sites to a mobile format. User connections are slow and screen real estate is limited, but that's just the physical problems. Users' mobile context will force the logic of applications to change. For example, the types of airline information users want at the airport and the travel transactions they wish to conduct are inherently different from those they'll seek when they're planning a vacation from their home two months before a trip. At the airport, a user will want immediate access to real-time alternatives if a flight is canceled, the cost to jump onto another airline and the ability to instantly purchase the new ticket.

And the problems don't end with the applications. Wireless networks are barely up to the task of mobile commerce. To harness this category's full potential, wireless networks need to be packet-based, to provide permanent virtual connections that allow faster transactions and the ability to push alerts. Higher speeds will also improve session quality. But with the exception of CDPD and early-stage GPRS rollouts, it will be at least two years before all of today's cellular networks are upgraded to faster packet data services.

Should you wait? No -- today's networks are suitable for many applications. But stick to applications in which information can be conveyed in small chunks and that don't require a lot of interaction. And keep them text based.

Although we believe location technology will be available by 2002, technical hurdles and privacy issues are associated with deploying a technology that lets organizations know users' constant whereabouts. At this point, don't worry about integrating user location into your application, but start thinking about how to take advantage of this information perhaps two years from now.

Will these roadblocks completely stall the industry? No, because the target markets are potentially huge, and momentum is building. Companies will travel the bumpy road to offer mobile-commerce services. But the net effect will be a narrower initial scope of applications and delays in the widespread adoption of mobile commerce. While industry enthusiasts predict large-scale adoption within three years, realistically this will likely be two years beyond that. If ever there were an opportunity to be a pioneer, this is it. The rewards are sure to outweigh the risks.

Executive Summary

Wireless Commerce

No longer must a customer be tied to a land line to buy stocks or change an airline reservation. Cell phones, PDAs and devices we haven't yet imagined are getting unwired for commerce. But getting a piece of this growing market is elusive. Obstacles include multiple platforms, lack of payment systems, sketchy security and the need for specialized infrastructure.

Because mobile commerce is still fraught with technical problems, most companies would be wise to seek a service provider. The list of middleware vendors, ASPs and wireless portals is growing rapidly. Ten such companies replied to our RFI (page 71), with solutions for a ticketing scenario. These companies' biggest shortcoming is their lack of adequate settlement systems. No one company did everything so well that we could pick a winner.

A big issue here is security. Although standards are evolving, one--WTLS (Wireless Transport Layer Security)--has taken the lead.

If you think this category is pie in the sky, look at E-Trade. This company has been at the forefront of wireless development, fostering relationships with carriers, developers and ASPs to provide real-time trading to its 3.5 million customers. That's not just a dream. It's a glimpse of the wireless future.

The Trouble With WAP

WAP is getting a bad rap. Originally touted as a panacea for the wireless industry, the Wireless Application Protocol is now getting bad press because of interoperability problems and minuscule market adoption where it's been deployed.

WAP does indeed have interoperability issues. For example, vendors' browser and gateway implementations are inconsistent. But this is being addressed by new interoperability testing programs. Furthermore, some critics dislike having to learn a new markup language, WML (Wireless Markup Language), to work with WAP. But that's exactly what XML (Extensible Markup Language) is all about. WML is based on XML, the very nature of which is extensibility, to allow new markups for particular applications. WAP also delivers on the promise of a network-independent solution. A WAP application does not have to take into account the kind of cellular technology in use nor the underlying network protocols, so long as the network supports WAP.

No, the problem with WAP is that it does not deliver on its fundamental promise: to let application developers develop their content in one format

for all wireless networks. Perhaps this promise stemmed from the hubris of the companies that formed the WAP Forum in the first place; they thought they could impose one standard on the entire industry. The result has been far different.

WAP is real, and many operators, particularly in Europe, will support it. But in the United States, most of the big operators, including AT&T Wireless, Sprint PCS and Verizon Wireless, are still using HDML (Handheld Device Markup Language), WAP's predecessor. Why? Because they started working with HDML before WAP was standardized. Also, HDML offers more features and, ironically, better interoperability than WAP, since wireless browsers and gateways come from one company, Openwave Systems. U.S. operators do say they'll eventually move to WAP, but only when a newer version becomes available.

Meanwhile, interactive pagers and handheld devices favor HTML approaches, not WML or HDML. NTT DoCoMo, hugely successful in Japan with its i-mode service (which is based on a subset of HTML), is now trying to export i-mode to the rest of the world. The net result: WAP delivers only to a small slice of the total wireless market. And that's the real trouble with WAP.