

# \* The next generation of communications technology

*The huge investment being made in the wireless technology sector will only be profitable if a 'killer application' materialises, propelling an explosive growth in demand for devices and bandwidth.*

In the early 1990s, the internet was used predominantly for email, file transfer and accessing remote computers (telnet). The web browser, along with the HTTP protocol, was the 'killer application' that led to the explosive growth of the internet, to the proliferation of e-commerce and to the establishment of many dot.com companies.

Something similar will occur for the wireless internet, which is just emerging. However, web browsing may not be the 'killer application' needed (WAP has not been a huge success yet). So what will make the huge investment into mobile devices and wireless communication infrastructure profitable? M-commerce could be one such important application. Others could be wireless enabled entertainment systems, local information, virtual communities, alerting and dispatching.

## THE ARCHITECTURE OF M-COMMERCE SYSTEMS

Without oversimplifying matters, an m-commerce system consists of mobile end-user devices (phones, PDAs and communicators), a data centre hosting m-commerce applications, a security authority, a clearing authority (such as a bank) and suppliers of merchandise (see Figure 1).

At the heart of this architecture is the data centre. It orchestrates m-commerce transactions issued by the mobile devices, issues shipping requests to the suppliers and charges credit cards. The data centre typically consists of a cluster of server computers running the 'server side' of an m-commerce system, maintaining transaction information in a database.

There are many key requirements of m-commerce systems:

- **Device independence:** to be successful, m-commerce systems typically need to run on a variety of mobile device platforms. Examples include conventional phones (WAP or SMS), smart phones (running the Java MIDP platform), PDAs and communicators (such as Symbian Quartz and Crystal devices). A browser-based architecture such as WAP is sufficient for some but others will require the deployment of a mobile application.
- **Bearer independence:** different wireless bearers need to be supported – WAP, SMS, GPRS and possibly UMTS.
- **Security:** identification, authentication, access control and end-to-end data encryption must be supported.
- **Reliability:** what happens if a device has weak or intermittent network coverage while an m-commerce purchase is performed? Guaranteed, exactly-once execution of m-commerce transactions is important, especially when stocks or other expensive items are purchased from a mobile device. This feature requires the deployment of middleware.

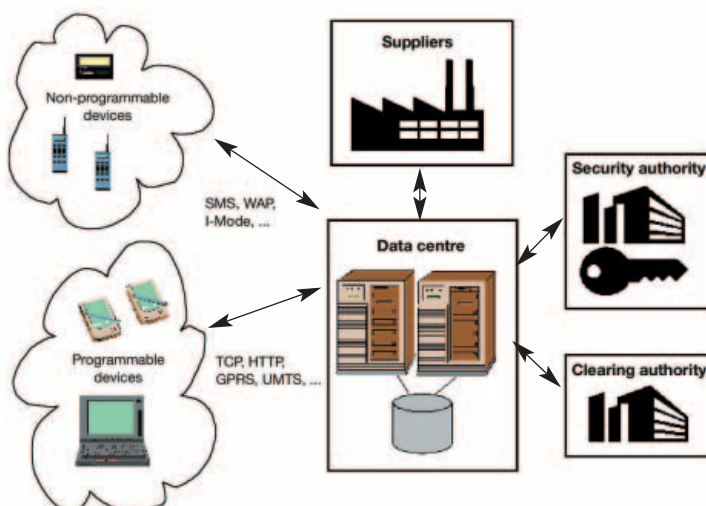


Figure 1. The architecture of m-commerce systems (arrows denote data flows)

- *Notifications*: the ability to send notification to the customer, providing updates on their transaction, is essential.

#### JAVA TECHNOLOGY ON THE DEVICES AND SERVERS

If special m-commerce software, such as a stock trading application, is to be deployed on mobile devices, Java should be used as the deployment language. By using Java, the amount of software that needs to be changed is minimised. Nowadays, viable Java execution environments are available for Palm PDAs, smart phones, Symbian, Windows CE and other platforms. Server-side Java technology, notably the Java-2 Enterprise Edition (J2EE) platform, should be used in the data centre. This allows for shorter time-to-market and avoids vendor lock-in.

#### MESSAGING MIDDLEWARE TO THE RESCUE

The Java platform or the mobile operating systems themselves do not provide the necessary tools and features for developing scalable and secure m-commerce systems in reasonable time. This is where application-to-application messaging middleware should be utilised. Messaging middleware has been used successfully for many years in financial information systems, e-commerce systems and logistics solutions.

Middleware denotes a class of 'invisible' software residing between an application and the underlying operating system environment.

Messaging middleware ensures the reliable delivery of purchase orders and other m-commerce transactions, from a mobile device to the data centre. The exactly-once delivery and execution of a transaction is guaranteed even when the user issues a 'purchase' transaction in a situation where the device might drop in and out of network coverage.

Another distinctive feature of messaging middleware is its ability to deliver information in real time, immediately after an event has occurred. For example, the data centre can send notification to a mobile device informing that an item ordered the day before has been shipped, or providing the shipping date. This type of real-time communication is useful for other mobile services as well, such as the delivery of stock quotes and news, or for enabling 'virtual communities'.

Furthermore, messaging middleware can be deployed efficiently atop a multitude of Internet standards (TCP/IP or HTTP), atop existing wireless bearers (GSM, SMS or TDMA) and atop next-generation wireless bearers (GPRS, EDGE or UMTS).

In summary, for the development of viable m-commerce systems, the Java-2 Micro Edition (J2ME) platform should be used on the mobile devices, the Java-2 Enterprise Edition (J2EE) in the data centre and messaging middleware should be the versatile 'bridge' between mobile devices and the data centre. The resulting m-commerce infrastructure is versatile, scalable, secure and based on accepted industry standards.

#### WHERE DO WE GO FROM HERE?

Messaging middleware plays a key role in m-commerce solutions – it reduces time-to-market and the risk of depending excessively on a single proprietary vendor. Messaging middleware allows the developer of m-commerce solutions to focus more on the business problem and less on issues related to deploying m-commerce software on various platforms or transmitting information reliably.

When choosing a messaging middleware vendor, it is important to ensure that:

- The middleware product runs both in the data centre and on the mobile device.
- The product fits seamlessly into the Java platform.
- The product adheres to the Java Message Service (JMS) standard as well as to internet standards such as XML and HTTP.
- The vendor can help in architecting and developing an m-commerce system.

Besides a wireless-enabled messaging middleware, the following tools and components will need to be chosen:

- A J2EE application server, which must provide high scalability and fault tolerance
- A WAP server
- A scalable database system to maintain customer account information
- Monitoring and network management software for the data centre
- Development environments and emulators for the respective mobile device platform (if customised software is deployed)

All of the components for building a scalable m-commerce system are available today, without 'locking' companies to a single vendor. They can be evaluated individually and glued together using Java messaging middleware to create an unparalleled m-commerce solution. ■

#### ABOUT THE AUTHOR

Silvano Maffei is CTO at Softwired AG. Softwired is a privately held, venture-capital-funded enterprise messaging middleware software company headquartered in Zurich, Switzerland. Softwired has developed an end-to-end wireless messaging middleware infrastructure, so that highly scalable m-commerce, wireless news distribution and location-aware services are a reality today. For more information, visit [www.softwired-inc.com](http://www.softwired-inc.com) or e-mail [info@softwired-inc.com](mailto:info@softwired-inc.com). References can be obtained from the author.