Mobile Media

Making it a Reality

Two prototype apps reveal the challenges in delivering mobile media services.

Many future mobile applications are predicated on the existence of rich, interactive media services. The promise and challenge of such services is to provide applications under the most hostile conditions—and at low cost to a user community that has high expectations. Context-aware services require information about who, where, when, and what a user is doing and must be delivered in a timely manner with minimum latency. This article reveals some of the current state-of-the-art “magic” and the research challenges.

In our research we are combining media systems and mobility systems to support the creation, distribution, and consumption of rich media to mobile or nomadic users. In this article we present two visions of future applications. The first is an example of personal context-aware mobile services that leverage RFID technology. The second is a highly collaborative interactive mobile gaming system. This example in particular highlights some of the key technical challenges we encountered while developing a prototype. We then focus on some of the broader system and software issues that need to be addressed to fully realize commercially viable mobile media systems. In particular, we summarize the complexities of getting media anywhere, creating media anywhere, and securely delivering media anywhere.

FRED KITSON, HP LABORATORIES

Mobile Media

Making it a Reality

THE WORLD AT YOUR FINGERTIPS

Many of us have imagined a world where objects are content-rich, interactive, and provide personalized information. But problems abound: Where does the information about the objects reside? How do we get content to said objects? How do you make sure the content is relevant to the consumer (e.g., me vs. my grandmother)? How do you empower users so that they can decide what content they wish to experience?

Consider the following example: You enter a theater to watch a film. As you walk through the lobby you notice a poster for an upcoming feature that you want to know more about. You walk up to the poster and notice it has a "Touch Here" label. You touch your cellphone to the label and immediately receive a video preview for the film and the opportunity to pre-order tickets for opening night.

This scenario isn't that distant a reality. Three current technological developments make this possible:

- Pervasive wireless networking (e.g., Wi-Fi hotspots, cellular 3G/GPRS)
- Broad availability of "smart" wireless devices (either cellphones or PDAs)
- Inexpensive, embeddable wireless RFID tags

The last of these technological developments, embeddable tags, is the key to making this vision possible. RFID

tags are quickly gaining broad industry acceptance.¹ A simple version of RFID has been used for years in retail environments to limit "shrinkage" (read: shoplifting). As the technology has advanced, a major push has begun to use RFID tags in supply chain management.² Both of these applications use far-field RFID technology, where the tag can be read from a distance of two to four meters.

Another variant of RFID technology is near-field, where the tag and reader must be very close, generally two centimeters or less. The tag is much smaller as well. The most common implementation of near-field RFID tags is the "card key," used by many organizations to control access to their facilities.

ENTER INHAND

At Hewlett-Packard, we wanted to leverage RFID and other wireless technologies in developing our own platform to deliver a rich media experience. Enter the inHand platform. The consumer end of inHand consists of a mobile device (iPAQ PDA) enhanced with a near-field RFID reader/writer and a small application that ties the reader/writer to Web browsers and other applications that display content on the PDA. This provides an intuitive touch-to-use interface displaying virtual information on physical word-tagged objects.

The heavy lifting occurs in the back end, which consists of three major components, as shown in figure 1: a *resolver* that maps tag IDs to an URL; a *tag application manager* that determines what content to provide and its customization; and a *user profile* database.

Returning to our interactive movie poster example, the "magic" behind the "Touch Here" label is an RFID tag. Putting the mobile device in proximity

to the RFID tag permits the reader in the mobile device to read the tag's contents. Typically with limited memory, the tag would be programmed with a product manufacturer code, a product code, and a serial number.

The client then sends the tag data *and* the mobile device identification to the specified server. This latter data is crucial, because the identification of *who* is reading the tag is critical to providing customized content.

The client sends the tag ID and a user ID to the back-end server for mapping the IDs to a product or service customized for this particular user. The back-end server's hostname is configured into the client. A GPRS or 802.11 wireless data connection connects the client to the back-end server.

The resolver looks up the tag ID in the database to locate the brand and appropriate content. The user ID is used to locate the user profile. Combining the two results in personalized content. The personalization could be a simple greeting text substitution ("Hi, Mike"), regionalized content, or a unique message tailored for a unique user. The content can be flash, HTML, video, music, audio, or any other digital content. The interaction of the user with the system is also recorded (if permission is given) so that information about a consumer can be used for customization of content.

In the movie-poster example, the tag application manager responds by immediately sending a streaming video preview of the movie. You could be rewarded for touching several different tags (on the same or other movie posters) or for reading the poster with a special members-only preview or coupon for a concessions purchase.

Within the back end, the raw event (a tag swipe and a device/user identifier) is used both to select the appropriate content template and to customize it. This selection can be as simple as a direct lookup or may involve applying business logic to vary the template selection based on past user interactions. To support this selection process, previous interactions are stored within the database and may be queried to determine repeat or related tag swipes, as well as past navigation patterns of this user.

Once a content template has been selected, it is instantiated with contextual information. This information again is drawn from the database and may include user personalization, related content, and previous navigation patterns. The template is rendered in the back end in a form appropriate to the display device and transport capacity, and is returned to the requesting device. Furthermore, the navigational links within the content (if any) are tailored to this particular user, the device, and the current transport mechanism. The back end is imple-

mented using a MySQL database, Apache Web server, and PHP scripts, running on a Linux server.

THE CHALLENGES OF INHAND

InHand may sound straightforward from a technical standpoint, but we met our share of challenges on other levels. Here are some of the difficulties we encountered.

Creating context-aware content for small displays is a new skill. We worked with several brands, and each time their instincts were to move their content from the Web into a handheld format. The content needs to be precise

With potentially countless tags,
managing the resolution
and association of tags
with products
is complex.

and to the point—resulting in a haiku style. Advertisers are more comfortable luring someone to a brand and less comfortable creating content geared toward specific user profiles and products.

With potentially countless tags, managing the resolution and association of tags with products is complex. In some cases the tag should be associated with the brand and other times with a line in the brand or even a particular instance of a product. The tags are not sequentially numbered, which requires a manual process to program the back end with the tags.

Determining how to use the user profile data effectively was a challenge. For each tag resolution, a small user profile engine needs to execute and determine how to customize this tag. How do you distinguish me from my grandmother? Or in marketing speak, how are users segmented by their behavior? If I pick up a pack of hot dogs, it's easy to remind me to buy buns, but what does this mean about my car-buying preferences?

MOBILE MULTIPLAYER INTERACTIVE GAMING

One of the most compelling yet technically demanding mobile media services is the support of interactive

Mobile Media

Making it a Reality

multiplayer mobile gaming. To address this new class of mobile application, we wanted to provide a platform with a multiplayer gaming services environment for third-generation mobile networks. We also wanted to design the system for context-aware games and applications.

Such an infrastructure will easily have the resources in terms of performance and scale to accommodate many less demanding mobile services that application developers will envision. Further, interactive gaming applications demand support not only for the timely distribution of game state between players, but also for services that enhance the overall gaming and community experience, such as lobbies, chat, and voice communications. It is often said of good multiplayer games that players “come for the action and stay for each other.” In essence, multiplayer games, which promote community communications, tend to engender gaming loyalty. Therefore, we felt that enabling this type of community experience was an essential requirement for our gaming platform. Few people, however, jump right into a gaming community without trying it out or watching others play first. Accordingly, we wanted to provide a means by which potential gamers could observe experienced players playing the game.

We wanted not only to develop a system that meets the challenges of delivering these mobile multiplayer and context-aware games, but also to build it in such a way that developers are isolated from the mobile network implementation specifics. This allows them to focus on the actual game design and logic, and gives them the unique mobile services required for developing rich community interactions and compelling gaming experiences. Toward this end, we also wanted to make our platform attractive to developers by complying with the interoperability standards emerging in this application development space.

THE MGSP GAMING PLATFORM

We named our gaming platform the Multiplayer Game Service Platform (MGSP). Architecturally, it is positioned within the packet network service infrastructure of 3G mobile networks. It is designed to leverage the IMS (IP

Multimedia Subsystem) infrastructure capabilities for controlling realtime and non-realtime rich data services such as video, conferencing, gaming, streaming, and messaging on the same IP transport network (figure 2). It provides a secure infrastructure for IP-based multimedia applications. As such, 3G mobile gaming applications may use the framework provided by the MGSP gateway services and APIs to interact with the 3G IMS core. This abstraction insulates the gaming service logic from the underlying network capability and protocols.

This enables the MGSP to access all the necessary underlying network services—for example, game session management functions such as registration and routing, subscriber authentication, QoS (quality of service) support, presence, and location. During game play, the service manages the appropriate QoS assignments for the rich media game elements via its QoS manager process and its interface with the core network. This ensures the correct experience for realtime and non-realtime multiplayer games within the capabilities of the network.

To address the need for mobile gaming group communications, we turned to technologies such as PTT/PoC (push-to-talk/PTT-over-cellular) and VoIP. The typical model involves a team of players initiating a group conference. Our implementation of this model consists of both the gaming service lobby and an MRF (media resource function). We use the MRF capabilities of our Opencall Media Platform, which contains softDSP media-mixing capabilities suitable for mobile audio conferencing. Moreover, combining actual gaming state and conferencing state results in an interesting context-aware conferencing model. By providing the game servers with access to conference-control functionality within the network, via the MGSP, dynamic game-related conferencing is possible. For example, as players' locations are updated within the game, dynamic conferences are created between colocated players. Players can realize additional game audio effects via the softDSP media processor.

PERVASIVE ACCESS FOR PASSIVE PLAYERS

The MGSP also provides the facility to allow any subscriber to view near-live multiplayer gaming action via

streaming media to the mobile terminal. This mode of interaction is useful not only for casual gamers and the uninitiated to familiarize themselves with the games in a passive way, but also for any subscriber equipped with only a mobile media player who can gain access to the gaming world in this way. In observer mode, the player elects to view the camera of another active player avatar. This mode is used in conjunction with a network-side client with a streaming audio/video output channel. The

streaming data is optimized, however, using the original synthetic game image data in the frame buffers.

The streaming observer mode provides a mechanism for anyone with a mobile media player to see a particular game (or follow the adventures of a specific game hero in a persistent world game) without requiring client code download. Because of the relaxed timing requirements of the streamed video, the game service platform can push the graphics rendering off to the observer view

Mobile Media

Making it a Reality

MRF (mobile rendering function), shown in figure 3. The 3D graphics renderer provides the rendered frames plus additional information, which is used to enhance final mobile-compatible video output. For example, the depth values of the 3D scene can be used to provide enhanced resolution for objects closer to the viewer. We anticipate that there may be even more “observers” than “players,” so scalability, rather than latency, is the issue.

The MGSP provides both the game clients and servers with respective APIs for multiplayer gaming session management. In this regard, alignment with appropriate standards is an important factor for the successful adoption of our platform. Within the standardization bodies, the OMA (Open Mobile Alliance) Games Services working group is tasked with the creation of interoperability specifications, APIs, and protocols for network-enabled games. As such, the OMA represents the standardization reference of the MGSP.

THE CHALLENGES OF MGSP

As with inHand, aspects of the MGSP seem perhaps easily within reach—but implementation again revealed significant challenges. In this case, the biggest challenges were a bit more technical. We found the following areas particularly worth noting.

One well-known characteristic of IP packet transport over 3G cellular networks is the long transmission delay

resulting from retransmission and interleaving performed at the link layer to combat loss in the time-varying wireless channels. Certain genres of multiplayer games are fast-action “twitch” games, and attempting to provide such games in a mobile packet network environment is currently beyond a challenge—it’s just not possible. Typically, for good-quality gaming experiences, this genre requires total ping (round trip) times of less than 200 milliseconds. When comparing this data with actual ping test data where the average latencies on UMTS (Universal Mobile Telecommunications System) networks are in the region of 350 milliseconds, the problem is evident.

This brought our attention to the issue of actual transport requirements for mobile gaming applications and the suitability of UDP and TCP. We consider UDP unsuitable as it provides no reliability. TCP is unsuitable for wireless gaming applications because of its persistent retransmission and congestion control mechanisms. Our approach was to analyze actual transport requirements for gaming applications and design a protocol that was optimized for such applications over UMTS wireless links. Furthermore, while current IP services are “best-effort,” the 3GPP (Third Generation Partnership Project) specifications provide for radio resource reservation for traffic classes, which include an error-intolerant conversational class with target latencies of less than 250 milliseconds. Such a traffic class would be suitable for near-realtime

mobile multiplayer gaming applications. It remains to be seen if operators will actually implement such QoS mechanisms.

Another major challenge is the implementation of the gaming service user interface. Specifically, these mobile devices comprise clients for services such as presence and group list. This brings up the issue of whether a person should be able to view the

gaming status of another person via the buddy list client applications or only via the gaming lobby clients. Persons within a gaming world have various identities or “handles” that do not correspond to their real-world identities as maintained by presence and group list services. Furthermore, players may assume different identities within different game instances. For the initial design we decided to use the client game lobby application as the only means of determining the status of gaming buddies. In a current project (nicknamed Wormhole), however, we plan to provide a mapping interface from the gaming-world presence and group services to the real-world presence and group list services.

A similar issue arose in relation to additional gaming services such as chat. Should such services be provided only external to the game application (for example, within a lobby application) or be integrated directly within the games? We believe the latter has the potential to enable game developers to differentiate their mobile gaming applications via built-in chat, VoIP, streaming music, etc., and offers a more compelling experience for mobile game players.

CHALLENGES FOR MEDIA CREATION, DELIVERY, AND SECURITY

Even with key back-end content delivery elements in place, many technical challenges remain from a media communications perspective. We next discuss three critical areas of technical challenges to fully realize such

mobile media systems. The first is the ability to push or pull media through a network, which requires both wired and wireless protocols that support a wide range of content in realtime and at low cost. The second concerns the sourcing of media or content from the user. Finally, we present an issue that is fundamental to the success and adoption of such services: end-to-end security.

GETTING MEDIA ANYWHERE

A key aspect to media access anywhere involves having WAN connectivity, for example, as provided by 2.5G or 3G cellular networks and supplemented by emerging 802.11 networks. While end users can see the rapid advances in cellphone technology, many of the key enablers for media streaming to cellphones are hidden and involve the design and deployment of new packet-based infrastructures for media delivery. Many challenges emerge, including ensuring interoperability between infrastructures and devices produced by different manufacturers.³ The 3GPP is specifying the set of standards and their use (e.g., MPEG compression standards and IETF transport protocols) for media delivery over 3G networks.⁴

Providing the ability to get content at many locations (e.g., at local coffee shops, airports, airplanes, trains, or other places where people may gather) is difficult. The challenges include providing a wide selection of content, fast response time, and cost. For example, the content should be cached locally; otherwise, every time a specific

piece of information is requested, a new copy of it would be shipped over the network, leading to unnecessary bandwidth costs. Another motivation for caching is that the locations where we want to provide this service may not have a sufficiently large broadband connection (compared with the requested content size), which can lead to sizable delays in transferring large media objects.

Another question that arises is: What to cache? Media caching is more difficult than Web caching, because of both the size

FIG 4

Mobile Media

Making it a Reality

of the media objects and the time dimension. Since Web objects are small (tens of kilobytes), they can easily be cached in their entirety, whereas media objects are much larger (megabytes to gigabytes) and even a small number of media objects can quickly fill up the available cache space. In addition, media objects are generally not consumed in their entirety. For example, many people view only the first 10 to 20 seconds of a number of music videos before deciding to view a single video in its entirety. Similarly, people generally want to view only the portions of a soccer game where the goals occurred, and not the entire two-hour game.

These differences in consumption patterns have significant effects on the design of media caching systems. For example, by automatically identifying and caching only the most requested segments of media content (as opposed to the entire content), we can achieve significant improvements in performance while using only a fraction

of the ability to perform realtime video capture, MPEG-4 encoding, and storing in MP4 file format, plus 802.11 wireless for sharing the captured content. The next-generation video-enabled PDAs will have enhanced video capture and encoding capabilities including higher spatial resolutions and higher frame rates, as well as both 802.11 and cellular network connections, and they will support standard-compliant streaming into the infrastructure. The infrastructure can then store the content and share it as appropriate. With this increased capability to create content comes the problems of how to index, search, and retrieve this content.

Another challenge relates to enabling conversational services such as VoIP or videophone using an 802.11-connected PDA. This is a problem since 802.11 can be afflicted by several hundred milliseconds of delay, making conversational services unacceptable. The new 802.11e standard, as well as other technologies being developed, may help alleviate these problems.

The media should be **encrypted at the sender** and decrypted only at the receiver—and remain encrypted at all points in-between.

SECURE END-TO-END MEDIA DELIVERY

A media delivery system may need to deliver a media stream to different clients with diverse device capabilities (e.g., screen display size, computation, or memory resources) and connection qualities (e.g., available bandwidth). This may require mid-network nodes, or *proxies*, to perform stream adaptation, or *transcoding*, to adapt streams for downstream client capabilities and time-varying network conditions.

Another important property is security to protect content from eavesdroppers. This makes it necessary to transport streams in encrypted form. The problem of adapting to downstream conditions while also providing security is particularly acute for mobile, wireless clients, since the available bandwidth can be highly dynamic and the wireless communication makes the transmission highly susceptible to eavesdroppers.

The conventional approach to this problem is to decrypt the stream, transcode the decrypted stream, and re-encrypt the result. This poses a serious security threat, however, since it requires giving the transcoder the key and then decrypting the content. The challenge that

of the storage space otherwise required. This conceptually straightforward idea is practically challenging since the manner in which compressed video and audio are stored in a standard file format (e.g., MP4 or 3GPP) is quite complicated as compared with the natural temporal order of media content.

CREATING MEDIA ANYWHERE

Today's technologies allow the average person to create content and share it with others. For example, the latest PDAs include integrated image/video cameras and have

arises is how to transcode in the middle while preserving end-to-end security. The media should be encrypted at the sender and decrypted only at the receiver—and remain encrypted at all points in-between. We refer to this capability as *secure transcoding*, to stress that the transcoding is performed without requiring decryption and thereby preserving end-to-end security.

The desired capability of secure transcoding can be achieved by co-designing the compression, encryption, and packetization using a framework referred to as SSS (secure scalable streaming).⁵ Figure 4 shows a scenario where a sender transmits encrypted content to a mid-network node or proxy, which performs a secure transcoding operation to adapt the received protected data for each of three clients: one with low, one with medium, and one with high bandwidth. The mid-network node performs secure transcoding (transcoding without decryption) and therefore preserves end-to-end security. In effect, the transcoding is performed in the encrypted domain. Note that the encryption keys are available only to the sender and the receiving clients, and not to the mid-network node that is performing the secure transcoding operation.

Conceptually, secure transcoding works by enabling the mid-network node to intelligently discard less important information while still enabling the remaining more important information to be decrypted and decoded by the receiver. This capability is achieved by the co-designed coding, encryption, and packetization. Since transcoding is performed by discarding information, without requiring decryption, end-to-end security is preserved. For simplicity this discussion has focused on the security service of confidentiality (provided by encryption); additional security services such as authentication and digital signatures are also supported by the SSS framework. This technology is being incorporated into the JPEG-2000 Security (JPSEC) standard.⁶

FUN ON THE ROAD

We have presented some examples of the new mobile media frontier and the research systems and technologies that form the substrate for our vision. We believe the future includes “fun on the road” based on the notion that users will require simple natural interfaces that fit a rather restricted form factor. Such services may also be delivered to the mobile user through non-mobile interfaces such as kiosks or plasma displays but will still require a comprehensible model of use for people on the go. As such, given that users typically carry their “interface tools”—such as eyes, ears, and mouth—with them and have a passion for communication and entertain-

ment, it seems a safe bet that mobile media services will be essential. The applications illustrated in this article—namely, interactive games, video and audio communications, and personalized context-aware activities such as shopping or theater-going—will all be enhanced and made more enjoyable by these mobile services. Q

LOVE IT, HATE IT? LET US KNOW

feedback@acmqueue.com or www.acmqueue.com/forums

REFERENCES

1. Want, R. 2004. The magic of RFID. *ACM Queue* 2(7): 41-48.
2. Beckett, J. 2004. RFID and sensing in the supply chain: challenges and opportunities. <http://www.hpl.hp.com/news/2004/apr-jun/rfid.html>.
3. Wee, S. J., Apostolopoulos, J. G., Roy, S., and Tan, W. 2003. Research and design of a mobile streaming media content delivery network (MSM-CDN). *IEEE ICME* (July). <http://www.hpl.hp.com/techreports/2003/HPL-2003-77.html>.
4. www.3gpp.org.
5. Wee, S. J., and Apostolopoulos, J. G. 2001. Secure scalable streaming enabling transcoding without decryption. *IEEE ICIP* (October).
6. ISO/IEC JPEG-2000 Security (JPSEC), Final Committee Draft. November 2004.

ACKNOWLEDGMENTS

This article was prepared with contributions from John Apostolopoulos, Nina Bhatti, Nic Lyons, Shinya Nakagawa, John Schettino, and Michael Sweeney, all of HP Labs.

FREDERICK LEE KITSON is a senior director at HP's corporate research labs, where he is responsible for research in mobile and media systems worldwide. His technical areas of expertise include mobile systems, computer systems, consumer appliances, and specific technologies such as multimedia digital signal processing, communications, and computer graphics. He is also responsible for HP's participation in MIT's "Oxygen" Ubiquitous Computing Project, as well as several research collaborations, particularly in Asia, such as one with NTT DoCoMo on fourth-generation mobile systems. Kitson received his B.S. in electrical engineering from the University of Delaware, an M.S.E.E. from the Georgia Institute of Technology, and a Ph.D. in electrical and computer engineering from the University of Colorado. He is an adjunct faculty member at the Georgia Institute of Technology and has taught at U.C. Berkeley and Colorado State University.

© 2005 ACM 1542-7730/05/0500 \$5.00