The Emergence of Middleware in Home Telecommunication Equipment

So Vang, CableLabs®
Jean-Pol L. Zundel, Comcast

ABSTRACT

Cable will soon offer interactive television services. These will be joined in home networks by telephones, Internet appliances, and consumer electronic equipment. Middleware makes it possible to develop a large set of applications and to deliver them on many different hardware and software platforms. Middleware does this by providing such Web standards as HTML, ECMAScript, and Java to develop and deliver applications.

THE EMERGENCE OF MIDDLEWARE IN HOME TELECOMMUNICATION EQUIPMENT

It's been a long Friday. Peter walks through the front door and into the living room, reaches for the remote, turns the TV on, and says to himself "Time to check the mail."

Lately he's found himself using the TV for the important messages, that is, friends and family. It's much faster than turning the computer on and waiting for it to boot.

He had mail: a message from Amy, his sister, with a picture of the proud and toothless grin of her 6-year-old Jimmy. As he laughed, the day and the commute lifted from his shoulders.

The next message held a pleasant surprise: he won the auction bids on two tickets for the Springsteen concert tomorrow. Logon to the Spectrum Web site, insert the smart credit card in the set-top box, and confirm the purchase. Oh, yes, what's the weather ticker saying about tomorrow night in downtown Philadelphia?

FACT OR FICTION

Peter's story is not science fiction. It is real for a number of subscribers participating in trials throughout the United States. These trials deliver interactive services over a television network connecting set-top boxes and servers. Where previous trials over the past years have used special equipment, these trials use current or about to be deployed set-top boxes. It is also real for users in Europe in actual large-scale deployments (Fig 1).

The interactive services typically include an electronic program guide (EPG), video on demand (VoD), e-mail, games, and various degrees of Web access ranging from a limited number of sites specifically developed for television, known as a "walled garden," to unrestricted Internet access.

The two-way network is either cable television with a built-in interactive data path, such as Data Over Cable Service Interface Specification (DOCSIS™), or satellite television with a telephone dialup return data path.

The set-top box and servers deliver interactive services to the subscriber in one of two ways: thick or thin client. In either case the set-top box acts as the client, capturing the subscriber input with an infrared remote control or remote keyboard.

In the thin client model, the set-top box does little but display graphics and capture input. Most of the application logic is executed at the headend on a server. The thinnest of clients are set-top boxes with less than 30 MIPS processors, 2–4 Mb of memory, and limited graphics capabilities. With these, even graphics rendering is done at the head-end with the resulting screens sent for display by the set-top box as stills over a digital TV channel.

At the other end of the range of capabilities, thick clients are represented by the latest set-top boxes with 250 MIPS processors, 32 Mbytes of memory or more, and a graphics processor equivalent to current PCs. Applications are downloaded and executed completely in these thick clients.

HOW DID WE GET HERE?

From the 1950s to the 1980s, while operators in the United States regularly expanded their offering of television channels through terrestrial broadcast and later cable, European coun-

0163-6804/01 $10.00 © 2001 IEEE

tries limited the number of channels to one or a few state-supported channels. By the time the U.S. cable subscriber had about 50 analog channels at his/her disposal, the European countries resolved to open television to competitive private operators, in the late '80s and early '90s. These competing operators, motivated to conquer the market quickly but not equipped with a cable infrastructure, decided to leapfrog directly to digital television over satellite. Digital television consists of digitally encoding television images, and compressing and transporting them as bitstreams according to the MPEG-2 standard. This first entry in digital television was developed under a number of objectives and constraints:

- A desire to offer attractive services with at least an EPG and the possibility to add interactive services
- A limited capacity in processing speed and memory size in the set-top box
- A desire to keep costs low by obtaining multiple sources for the set-top boxes

These conditions led to the following software architecture:

- A set-top box with an interpreter creating a complete but proprietary environment, such as Pantalk by Canal+ and Ocode by Open TV
- Television-specific applications developed in the dedicated tools supplied with the interpreter

Interpreters provide a controlled environment with all necessary functions built in. This allows rapid development of safe applications with a very small memory footprint. The applications are portable in the sense that the operator can broadcast one copy of the byte code for the application, and it will load on any and all set-top boxes. Oak, the ancestor of Java, was developed for television set-top boxes by Sun at the same time and for the same reasons.

The trade-offs at that time of limited processing power and memory were:

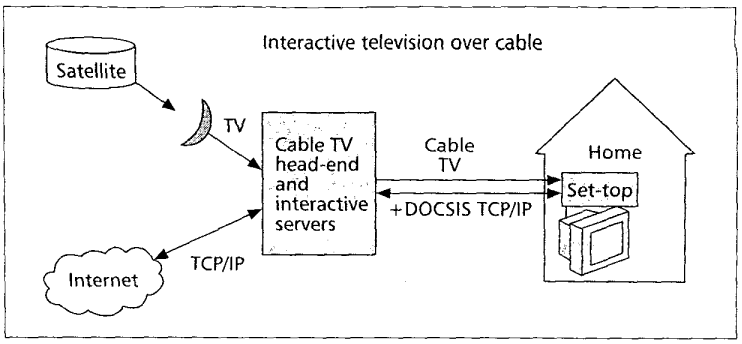
- The need for good programmers to create optimized code in a proprietary environment
- The need for careful use of graphic resources

Skipping to the second half of the '90s and to the United States, the immense success of the Internet created another school of thought based on Web technology, embodied by WebTV, now part of Microsoft, Liberate, and Power TV. Critics of this approach have properly pointed out that most of the Web does not look good on a television screen with its limited resolution and that it is a static data medium while television is defined by movement. They miss two points:

- Television can adopt the tools of the Web and yet develop applications more oriented to its media characteristics.
- Web page design can expand from static resources such as text with GIF and JPEG pictures to animated resources such as Macromedia Flash and streaming video and audio.

The tools of the Web, HTML and JavaScript, with television extensions added by each of these vendors, provide an interpreted environment which satisfies the same objectives as the European proprietary interpreter approach. However, the trade-offs are quite different:

- There is a very large base of developers for Web tools. These developers do not have to be computer programming specialists.
- The liberal use of graphics resources and the interpretation of HTML and JavaScript require generous processing power and memory capacities. We can thank the combination of Moore's law and time for making this economically possible today.



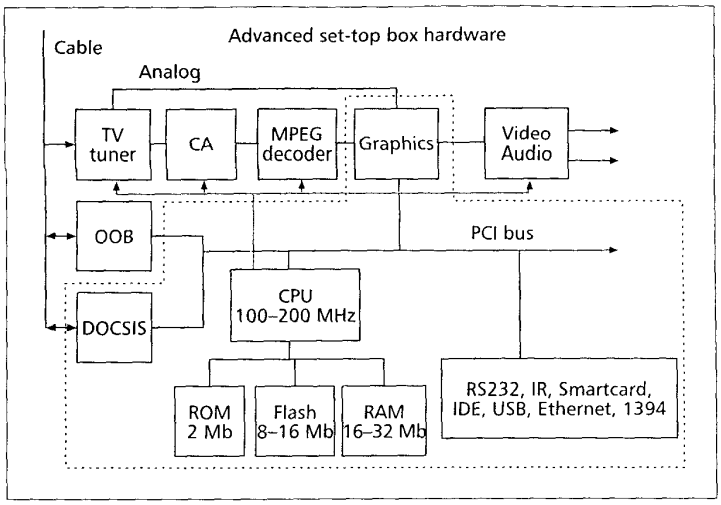
■ **Figure 1.** *Middleware emerging in the home environment.*

The Business Model

The European decision to have competing equipment suppliers for digital television set-top boxes led to middleware in the form of interpreters. This decision led to a business model that is very different from that on the North American side of the Atlantic.

When cable television went digital in the United States in the late '90s, it was and continues to be dominated by two vendors of analog television equipment: General Instrument (GI) and Scientific-Atlanta (SA). These companies provide highly integrated but proprietary environments. Each offers head-end equipment with its proprietary conditional access and encryption; each offers its set-top boxes which work only with its head-end with few applications, all coded in C to a proprietary operating system (OS) environment. Cable operators have to choose one of these two vendors and tend to populate their networks with one offering. In a cable network supplied by one of these suppliers there are no alternative sources for set-top boxes deployed yet. The life of a specific set-top hardware design is quite long. The SA Explorer 2000 and GI DCT 2000 have been selling essentially unchanged for more than three years.

In contrast, the European digital equipment market, whether satellite or cable, is not vertically integrated. It relies on multiple vendors for each component. For set-top boxes, television operators first develop the software on a reference model and then issue tenders for hardware to multiple vendors. Thus, software is common across multiple hardware platforms. Once the hardware functions are defined, cost reductions come with higher silicon integration, new designs come out every year, and prices fall quickly.



■ **Figure 2.** *Computer functionality in cable TV.*

SOFTWARE STANDARDS: DVB MHP AND OCAP

Thus, the main concept of middleware, as a dedicated environment with built-in functions available through an interpreter to develop safe and portable applications, originated from the beginning of digital television.

After the initial development and deployment of diverse middleware by suppliers, a second concept, a standard application programming interface (API) for interactive applications, emerged. This not only gives portability across boxes for one implementation of middleware, but also portability across multiple middleware providers.

The hardware vendors that had to port multiple proprietary middleware environments to their boxes, and the middleware developers that wanted a broad base of suppliers for interactive applications, decided to cooperate in the Digital Video Broadcasting (DVB) Multimedia Home Platform (DVB MHP) effort. The purpose is to define one standard environment in which to write interactive applications. DVB MHP initially settled on a Java environment.

In the United States, the growth of competition from satellite television gave the cable operators their opportunity to leapfrog from proprietary hard-coded systems to the specification of an Open Cable Application Platform (OCAP). OCAP is being defined by CableLabs®, the cable television operators consortium, in concert with middleware providers Canal+, Liberate, Microsoft, OpenTV, Philips, Power TV and Sun. OCAP is an open middleware combining a presentation engine with HTML and JavaScript and an execution engine with a Java virtual machine that aligns itself with DVB MHP.

After defining Java-only profiles for set-top boxes with little or no return path, DVB has worked on its interactive profile with HTML and European Computer Manufacturers Association standard for JavaScript (ECMAScript).

Today, there is the strong possibility of a worldwide standard for middleware combining the best Web tools with the power of a portable full programming environment.

To illustrate in more detail the components of representative middleware standards, the authors will use OCAP, with which we are most familiar.

THE OCAP ENVIRONMENT

An advanced set-top box essentially contains two sets of components:

- A digital television decoder with its pipeline made of TV tuner, conditional access to decrypt the scrambled MPEG stream, an MPEG decoder to decompress the video from the MPEG stream, and the final audio and video stages which condition the output for the television receiver. A graphics processor overlays graphics from the computer with the decoded video. Images from classic analog channels go directly from the tuner to the graphics overlay stage. The out-of-band (OOB) channel is a bidirectional channel between the cable operator headend and the set-top box, and is used to convey service information such as which programs are on which channels and their conditional access information.
- All the essential components of a computer (within the dotted line in Fig. 2): CPU, memory, peripheral bus, and a graphics processor to overlay the interactive graphics with the television image. The DOCSIS modem provides a bidirectional TCP/IP channel between the set-top box and the world over the cable. The computer controls the television decoder pipeline and provides interactive services (red line in Fig. 2).

The software diagram represents how multiple applications are served by the middleware and how the middleware controls the hardware functions through software drivers.

Figure 3 illustrates the typical applications expected to be present:

- An electronic program guide (EPG)
- VoD
- A mail application
- A chat or instant messaging application
- A browser to access a large number of applications written using the tools of the Web

The middleware allows applications to be written in either HTML/ECMAScript or Java. It is up to the application designer to choose the tools he deems best for the purpose: today there are EPGs written in HTML/ECMAScript and others in Java.

For the purpose of clear presentation, these tools are grouped in two classes: those of the presentation engine and those of the execution engine. But the division is only logical, since it is also possible to write applications combining all of these tools.

THE PRESENTATION ENGINE

The presentation engine is a familiar browser environment based on HTML for page layout and ECMAScript for logic. The two are linked by the Document Object Model (DOM) which is

a tree representation of the HTML document to allow the ECMAScript to access it and change it dynamically. This combination of HTML and ECMAScript is often referred to as DHTML, for dynamic HTML.

Additional standard browser tools are also available:

- CSS
- XML
- Flash 4.0

The Execution Engine

The execution engine is a Java environment consisting of:

- Personal Java
- Java TV, including the Java Media Framework
- Additional extensions for television from the industry bodies DVB, ATSC, and CableLabs

Personal Java is essentially all of the current Java 2.0 with the addition of the Xlet construct. The Xlet controls application life cycle from within Java, as opposed to PCs or workstations where Java applications are started and controlled from a command line or calls to the OS from a non-Java program.

Java TV is the collective name for libraries dedicated to television-specific functions:

- Service information (SI) — TV channels and the programs they carry are called *services*. These functions provide information about available services.
- Service selection — Functions to tune the set-top box.
- JMF and the broadcast pipeline — Video and audio controls and functions to synchronize applications with the video.
- Broadcast data — Functions to extract data carried in the broadcast TV channels. MPEG not only allows digitized video and audio, but also data in the same transport stream.

Additional extensions from DVB, ATSC, and OCAP cover television practice in the specific U.S. cable environment:

- Home appliance virtual interface (HAVI) — A number of graphic widgets to control typical audio-visual equipment.
- Conditional access — Functions to control access to services based on the services subscribed to by the user.
- User preferences — Functions to allow multiple users to set and save their preferences. These include favorite channels, parental controls, e-mail accounts, and cookies.
- Applications life cycle — Functions to create, start, stop, and monitor the state of applications in the set-top box.
- Monitor application — An application which always runs on the set-top box to enable navigation to other applications.
- Security

Security and Privacy

For interactive applications to be accepted by a cable customer, the subscriber must have confidence in the security and privacy of the system. This can be achieved through the following:

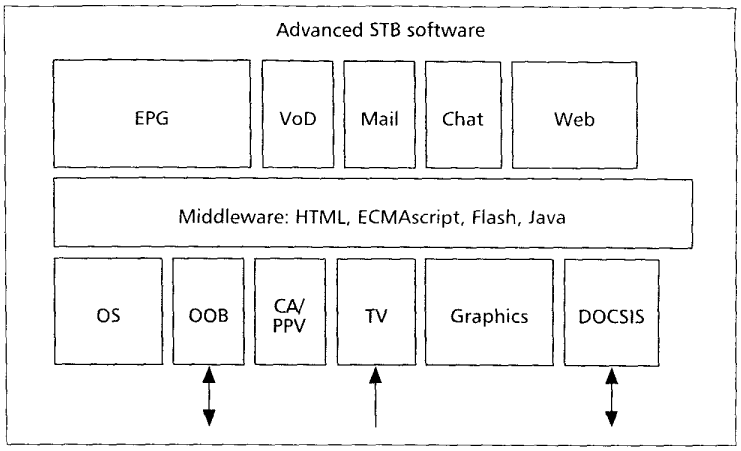


Figure 3. *The applications expected to be typical in OCAP.*

- Authentication, to know that the application downloaded from the network is the one the user asked for and has not been tampered with
- Permissions, to enforce boundaries an application cannot cross once it is running
- Encryption, to protect the private data the user allows the applications to use and transmit
- Privacy rules, to restrict private data the user does not allow to be shared from being transmitted or used

Only the most essential applications are resident on the set-top box. Most are downloaded when used from the operator's head-end or from remote sites through the Internet. Authentication ensures that the application code coming through the network is really that expected, not a Trojan horse injected on the network by a rogue party.

Authentication of code modules is done by two mechanisms:

- Signatures, where each file in each directory that makes up the application is checksummed into a digest file stored in the directory. This is done recursively up the directory tree to the root, where a signature file contains the encrypted checksum for the whole tree and the public key for it. The receiver can decrypt the checksum with the public key, recompute the useful directory and file checksums, and these will match only if there has been no tampering.
- Certificates, where a certificate file at the root of the application tree provides a continuous chain of secure certificates linking the application to the service operator. For example, the certificate file says that this application was designed by "Mixed Signals" working for "NBC Studios" who is registered at "VeriSign," a secure certificate authority. We, the cable operator, know "VeriSign," because we are registered there too, so we have a chain of trust.

Permissions allow the operator to modulate the access to resources given to any applications. A large number of applications, such as games and interactive commercials, will need to run only for a limited time, accepting input from the user while displaying on the screen and commu-

The traditional methods of tracking viewer ratings, such as Nielsen, pale in comparison with digital set-top boxes: every click, whether changing TV channel or clicking to an Internet site, can be recorded, stored, and analyzed for all users at all times.

nicating with their servers, without access to any other resource. When limited to this minimum access, these are called *unsigned applications* and do not require authentication as defined above. Thus, a large number of applications can be safely distributed with a low administrative cost. At the other end of the spectrum, applications that need to start other applications, access persistent storage, and other such privileged resources need to be authenticated and to have a permission file.

Encryption protects the data the user has entrusted to an application, such as purchases. More generally users may want all of their traffic over the network to be confidential. Secure sites for e-commerce typically use Secure Socket Layer, but the two-way channel between the set-top box and the Internet servers uses the DOCSIS™ protocol with its capability to protect all traffic by encryption.

Finally, the public will ask for privacy rules to balance its need for privacy against the convenience of accessing an interactive network. The traditional methods of tracking viewer ratings, such as Nielsen, pale in comparison with digital set-top boxes: every click, whether changing TV channel or clicking to an Internet site, can be recorded, stored, and analyzed for all users at all times. Interactive television is the holy grail for viewer profiling and ad targeting. It is certain subscribers will not agree with unchecked usage of such capabilities: they will ask for privacy rules, either voluntary or legislated, so that they can selectively share or protect what they consider private data.

Fact or Fiction?

This last scenario is not deployable today, but an essentially similar one is being demonstrated by one of the major consumer electronics suppliers:

- A wireless home network connecting televisions, phones, speakers, and DVD players with a computer acting as the home interactive server
- Combined voice and remote control input to the server for all the devices individually or collectively (i.e., play the DVD through a television set of your choice)
- A user interface through a multi-assistant which can adopt suitable synthesized voices and characters (i.e., male, female, or child)
- Unified messaging between phone and e-mail with display of messages on the television, direct play of voice messages, and synthesized reading of written messages over wireless remote speakers

Beyond the development of each technology, a standard middleware is the essential ingredient in integrating these in a practical and economical solution:

- It provides a technical framework in which to integrate the new technology.
- It allows the development of test suites for conformance to the platform standard. Test suites are very complex and expensive, requiring the cooperation of many suppliers to develop one at all.
- It encourages application developers with a universal platform to write once and deploy everywhere.
- It accelerates the growth of a market for the applications.
- It drives prices down quickly by encouraging competitive implementations of the hardware and software.

Bibliography

[1] Consumer Electronics Association (CEA) Online Resource for DTV, http://www.ce.org/market_overview/market_overview_video.asp?area=21

[2] G. O'Driscoll, *The Essential Guide to Digital Set-Top Boxes and Interactive TV*, Prentice-Hall, 1999.

[3] M. Silbergleid and M. J. Pescatore, *The Guide to Digital Television*, 3rd ed., United Entertainment Media, Mar. 1, 2000; http://www.digitaltelevision.com/publish/dtvbook/toc.shtml

[4] J. C. Whitaker, *DTV: The Revolution in Digital Video*, 3rd ed., McGraw-Hill, Jan. 25, 2001.

[5] MPEG Website, http://www.cselt.it/mpeg

[6] P. D. Symes, *Video Compression*, McGraw-Hill, 1998.

[7] CableLabs Certified Cable Modem Project (formerly known as DOCSIS) Website, http://www.cablemodem.com

[8] V. C. Majeti, C. Venkata, and J. Fijole, *Cable Modems: Technologies and Applications*, IEEE Press, 2000.

[9] DVB Consortium Website, http://www.dvb.org

[10] DVB-MHP Website, http://www.mhp.org/html_index.html

[11] M. Adams, "OpenCable Architectures," Cisco Press, 2000.

[12] CableLabs OpenCable Project Website, http://www.opencable.com

[13] World Wide Web Consortium (W3C) homepage for HTML, http://www.w3.org/MarkUp

[14] ECMA Website, http://www.ecma.ch/

[15] Sun Microsystems, Inc., Java Technology Website, http://java.sun.com

[16] Sun Microsystems, Inc., Java TV Technology Website, http://java.sun.com/products/javatv

Biographies

So Vang (s.vang@cablelabs.com) is director, OpenCable™ Software, in the Advanced Platforms and Services for Cable Television Laboratories, Inc. (CableLabs™). In this position he is responsible for managing the OpenCable™ middleware effort, also known as OpenCable™ Application Platform (OCAP). Before joining CableLabs he was a senior analyst for Thomson Consumer Electronics, where he designed and developed client/server, Web-enabled, and computer telephony applications for e-commerce. Prior to that he was a programmer analyst for BMG Direct Marketing. At BMG he designed and developed automatic phone ordering systems. He also worked with a team to design and develop workflow systems and user interfaces using client/server and object-oriented technologies. He also worked as a software engineer at the University of Indiana, where he worked with a team to develop real-time infrared capturing, processing, and multimedia applications. He earned his B.S. in computer science from Purdue University with emphasis on mathematics theory and algorithm optimization.

Jean-Pol Zundel (jean-pol_zundel@cable.comcast.com) joined Comcast in his present role of chief software architect in 1998. He is a member of the Digital Television Engineering group, which evaluates interactive television offerings, sets up trials, and initiates deployment of new services. He is a member of the technical team of OCAP and chair of the SCTE Cable Application Platform Subcommittee (CAPS). He started by learning most aspects of the computer business with Sperry-Univac, later merged with Burroughs and renamed Unisys; first as a systems analyst and pre-sales manager in the field, then at the head office in Bluebell, Pennsylvania, as a director of research developing hardware and software, as program manager for mainframe software, and directing a consulting group providing transaction processing services. In 1992 he joined Trax as VP of software development. Trax is a start-up in freight payment through automated processing of EDI over private networks and from 1996 on, over the Internet. He is a graduate of the Ecole Supérieure d'Electricité in France, as an electrical engineer with a specialty in computer science.