Java™ Technologies for Interactive Television

Technical White Paper

Please
Recycle

™

Adobe PostScript

Contents

Overview

With the rollout of digital television around the world, broadcasters and content providers have new opportunities to enhance the viewing experience. The development and adoption of interactive TV (iTV) standards, such as Digital Video Broadcasting's Multimedia Home Platform (MHP) and the Open Cable Application Platform (OCAP), has paved the way for content providers to create compelling applications and programs that are interoperable across a variety of middleware vendors and set-top boxes. Java technologies play a vital role in creating, deploying, and executing this enhanced or interactive content.

Data broadcasting (or datacasting) makes it possible to embed application data into the broadcast audio/video stream. This application data can be tightly coupled to the video for close synchronization. It can also be loosely coupled, where the data and the video are related in content, but not related in time. In addition, the data can be noncoupled, such as in running stock tickers or games.

The benefits of iTV come from its ability to use data bandwidth by:

- Enhancing the program content with either loosely coupled or tightly coupled data. This can help attract and retain viewership, increasing the program's value for advertisers.

- Providing services unrelated to video data, such as stock and news tickers, e-mail, chat, games, and so on. These value-added services help differentiate broadcasters, multiple service operators(MSOs), and content providers.

Java TV™ technologies put this data to use in an environment where applications can be executed locally — within the set-top box — while providing a robust, head-end architecture for managing and deploying data and applications.

Java™ Technology Overview

Java TV technology for digital television receivers is based on the Java™ platform, which consists of the Java virtual machine (JVM™) and several reusable libraries of code in the Java programming language. The Java TV API is a standard extension to the Java platform and provides reusable, television-specific libraries.

The JVM resides on the digital television receiver and executes Java code there. It has been designed to ensure secure execution of code. Byte-code verification guarantees the validity of instructions executed by the JVM. Class loading mechanisms enforce how code is loaded into the machine, and can provide assurance about the code's source.

The PersonalJava™ application environment has been optimized to work on resource-constrained consumer electronics devices, such as digital television receivers. Features not required in the Java platform for these devices have been removed in the PersonalJava application environment.

This environment provides much of the functionality needed by Java applications and applets, such as a user interface toolkit, input and output, networking, internationalization, security, and class loading. However, the PersonalJava environment does not include television-specific functionality — that is provided by the Java TV API.

The Java libraries resident on a digital television receiver consist of a set of core class libraries from the Java platform, optional libraries that may be included based on a specific implementation's requirements, and the Java TV API. The class libraries contain functionality that can be used by all Java applications and applets, so they can remain smaller and easier for software developers to write.

The way Java applications and applets are deployed provides a key benefit. Java code need not reside on the receiver at all times. It can be kept elsewhere and brought to the receiver only when the code is needed. When it is no longer needed, the code is automatically freed from the receiver's memory through the garbage collection process provided by the JVM.

Java-based applications are inherently cross-platform, meaning that developers need author the content only once, and it will run across any hardware device supporting the Java platform, regardless of the underlying operating system and CPU. This architecture-neutral approach is useful not only for network-based applications, but also for single-system software distribution. In contrast, when writing applications that do not use Java technology, programmers need to produce separate versions for each of the many supported RTOS architectures and processors.

What Java Technology Provides

The power and flexibility of the Java platform makes it the ideal choice for enhancing digital broadcasts. Content providers, software developers, and television system operators benefit from new features and revenue possibilities produced by Java platform-enabled functionality, while viewers enjoy additional information, convenience, and control of the content they are viewing.

Worldwide, Java technology is used on tens of millions of devices, is supported by more than three million programmers and dozens of tool suites, and has been licensed by hundreds of manufacturers and technology providers for use in devices ranging from smart credit cards to mainframe computers.

The Java platform provides the following benefits:

- Portability across platforms: Write Once, Run Anywhere™ (operating system platform-independence)
- Faster time to market (ease of code reuse)
- Lower development costs (ease of code development and reuse, ease of remote support and upgrades)
- Security (information can be protected, individual capabilities can be granted to applications)
- Support for memory-constrained devices
- Standards compatibility
- Enhanced code quality (enabled by the Java language's object orientation and security features)
- Expanded market for developers (same applet/application spans a larger number of users; consumer device users can use many existing Java applets and applications).
- Java HotSpot™ client and server virtual machines
- Improvements to startup time and memory footprint

Set-Top Box Software Technology

Two types of technology are typically found in first-generation, interactive-capable television environments. HTML-based technology, often based on the Advanced Television Enhancement Forum (ATVEF) specification, is typically supported within a browser environment and provides the dominant interface for Microsoft's WebTV. Proprietary environments from OpenTV, PowerTV, and others have also been deployed on a variety of set-top boxes. Next-generation set-top boxes will enable full interactivity based on Java TV technologies.

HTML Technology

While HTML is sufficient for many purposes, it is fundamentally not designed for broadcast television or advanced interactivity. Content developers have very limited control over how screen layout is rendered. Dynamic elements such as animation, fades, wipes, and the like are very familiar to all television viewers, but range from difficult to impossible to implement using ATVEF-style interfaces. In addition, local computation is not possible, making most types of interactivity impossible without a persistent two-way connection and a head-end infrastructure to support this interactivity. Games are exceptionally difficult to program, due to the lack of an environment that can support local calculations.

While there is a wealth of content available that was authored using HTML-based technologies on the Web, the vast majority of this is unsuitable for television use, due to the unique display requirements of television. A television screen is

significantly different from a computer monitor, with television having much more critical user interface requirements. Practical experience has shown that, to be suitable for television, most content needs to be authored from scratch.

An additional, often overlooked issue is the fact that HTML is not a true standard. There are no compliance suites or formal certification procedures for an interpreter. Therefore, the proliferation of extensions has led to an unofficial specification based on the features (and bugs) implemented by the most popular browser, resulting in a compatibility nightmare for content authors attempting to ensure compliance across a range of middleware implementations.

Proprietary Technology

Proprietary interactive TV programming environments, such as OpenTV, PowerTV, and Microsoft Windows CE, overcome many of the problems exhibited by HTML but pose additional challenges. Foremost among these is the nonstandard nature of each environment. Content authors must learn to work with multiple development environments, each with unique APIs, capabilities, and tools. In addition, content is not portable, requiring content developers and operators to deploy several versions of an application for all software environments that may be present on a given network — an enormously complex, expensive, and inefficient situation.

Java Technology

The Java platform overcomes the limitations of HTML-based technology while avoiding the shortcomings of proprietary environments. As a full programming language and application environment, Java enables advanced levels of interactivity; dynamic, broadcast-quality graphics; and local computation. This provides an unlimited range of possibilities for content authors without requiring access to a return channel and head-end infrastructure. Java applications are inherently portable across set-top boxes with different underlying hardware and operating systems, greatly simplifying the task of managing content distribution. The Java platform is also among the most secure environments known to computer science — the source code has been scrutinized by experts worldwide, so potential security holes are identified and fixed long before production code is deployed. Millions of developers are currently using the Java language with a wide range of tools, providing an inexpensive, knowledgeable base of talent from which to draw content authors.

Standards Bodies

The positive attributes described here have led to the adoption of Java technology by standards organizations around the world in specifications designed for consumer devices. Widespread industry support, robust compliance suites, and strong intellectual property and patent protection have persuaded these organizations to base their specifications on Java technology:

- Advanced Television Systems Committee (ATSC)
- Cable Television Laboratories (CableLabs)
- Digital Television Industrial Alliance (DTVIA)
- Digital Video Broadcasting (DVB)
- Home Audio/Video Interoperability (HAVi)
- Open Services Gateway Initiative (OSGi)

Specifically for digital television, CableLabs' Open Cable Applications Platform (OCAP), DVB's Multimedia Home Platform (MHP), and ATSC's Digital Television Application Software Environment (DASE) are all based on Java TV technology, making it a likely candidate as a worldwide standard for digital television content. The end result will be a much larger library of available applications and services for operator deployment. Sun has negotiated agreements with both CableLabs and DVB, allowing vendors to implement each organization's respective Java technology-based solution and utilize the appropriate specification without requiring those vendors to have a direct relationship with Sun.

Conclusion

As the core of interactive TV standards, Java technologies bring interoperability to set-top box applications. In addition, the widespread deployment of Java technology in other consumer devices, such as residential gateways, mobile phones, PDAs, game consoles, and two-way pagers makes it possible for content authors to deploy applications and services across a range of devices, while taking advantage of a common software architecture and development tools. This will become increasingly important with the advancement of high-speed, mobile data and home networks. The Java platform common to all these devices will ensure much greater content availability, shorter time to market, and lower development costs.

Sun microsystems

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303

1 (800) 786.7638
1.512.434.1511

www.sun.com