Volume 07 Issue 01 Published, February 19, 2003 ISSN 1535-766X

Intel® Technology Journal

Managed Runtime Technologies

Managed Runtime Environments for Next-Generation Mobile Devices

A compiled version of all papers from this issue of the Intel Technology Journal can be found at:
http://developer.intel.com/technology/itj/index.htm

Managed Runtime Environments for Next-Generation Mobile Devices

Paul Drews, Intel R&D, Intel Corporation
Doug Sommer, Intel R&D, Intel Corporation
Roger Chandler, Intel R&D, Intel Corporation
Terry Smith, Intel R&D, Intel Corporation

Index words: Mobile, Handset, Wireless, Services, Mobile Data, Operator, Mobile Applications

ABSTRACT

The adoption of cellular communications has been one of the fastest growing technology trends in history. Many analysts predict that the demands of our increasingly wireless world will result in the rapid convergence of cellular communications and powerful, handheld computing devices, enabling a wide array of exciting new user experiences. By 2006, the analyst firm Instat/MDR predicts there will be over 760M Internet-enabled mobile devices in use (1), and the ARC Group predicts that by 2007, over 1.7 billion users will utilize wireless data services (2). Intel is a leading building block supplier to this new converged device industry with our Flash memory products, our high-performance Intel® XScale™ mobile application processor family, and our Personal Internet Client Architecture (PCA) for mobile computing devices and handsets. Intel is also developing key technologies that will accelerate the adoption of managed runtime environments (MRTEs) for mobile devices. The industry predicts that the majority of converged devices will include MRTEs (primarily in the form of J2ME and .NET*) and because of this, MRTEs are a key component of Intel's overall mobile industry strategy.

This paper describes how MRTEs are important enabling technologies for the future of wireless computing and how they are contributing to the fast delivery of wireless data services. The term "managed runtime environment" as used in this paper refers to new functionality and technologies that extend the capabilities of the first generation of runtime environments, notably, the Java* Virtual Machine and the Microsoft .NET Framework. These first-generation MRTEs need to evolve further to serve the demands of the mobile data market. We also illustrate how PCA is the ideal platform to take full advantage of MRTEs, and we conclude with descriptions of Intel's R&D to enable the next generation of MRTEs for mobile devices

INTRODUCTION

The launch of cellular communications several years ago had one purpose in mind: mobile voice communication. Analog-based voice communication was the sole purpose of cellular handsets for many years until the transition to digitized voice data came about in the 1990s. Regardless of the type, cellular handsets and networks were architected and deployed to accommodate voice traffic. It was not until the introduction and ensuing popularity of Short Messaging Services (SMS) that cellular operators and handset manufacturers began to explore the possibilities and ramifications of building systems capable of handling a wide range of digital datatypes. The promise of 2.5G and 3G networks, widely deployed wireless LANs, in combination with the Internet explosion, has fueled a global demand for cellular handsets that deliver both voice communications and general-purpose computing capabilities.

Most cellular network operators and handset manufacturers have either announced or begun deployment of first-generation data applications and services for cellular handset users. These initial applications include 2D games, electronic mail, multi-media messaging, personal information management, and a host of other applications once found only on personal computers. As more data-intensive mobile applications are deployed, developers and service providers are encountering troublesome issues stemming from the heterogeneity of today's mobile device platforms. In order to successfully deploy a wide range of mobile data applications and services, the device platform itself needs a consistent software interface layer that developers can rely upon when developing and deploying applications. This will serve to not only insulate the mobile application developer from the underlying hardware and software variables of the device, but also create a "common platform" necessary to jump-start a new wave of mobile applications. This interface layer, so to speak, must also protect the integrity of the device's core capabilities and enable the service provider to more effectively install, manage, and maintain the applications and services on the device. This layer is a managed runtime environment (MRTE).

BENEFITS OF MANAGED RUNTIME ENVIRONMENTS

The term "managed runtime environment" as used here refers to new functionality and technologies that extend the capabilities of the first generation of runtime environments, notably the Java Virtual Machine and the Microsoft .NET* Framework. MRTEs provide an exciting array of benefits to mobile application developers, operators, and end users, as follows:

- A platform-independent programming environment that makes it far easier (than native code) to move applications between platforms.
- A sandbox runtime environment that prevents rogue programs from disrupting the platform.
- Garbage-collection-style memory management and incorrect-reference (pointer) protection that together nearly eliminate a major source of programming errors.
- A dynamic code-loading mechanism that makes it easier to extend platform capabilities with new applications and class libraries.

These benefits are so substantial that MRTEs should be considered an essential design element of all new mobile device designs. Figure 1 below represents the architectural framework for an MRTE.

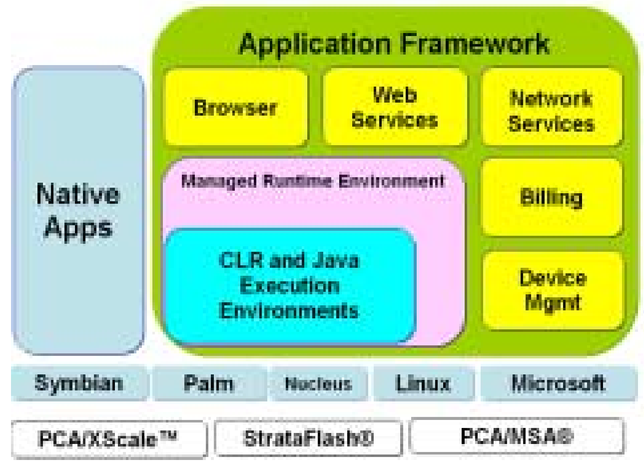

Figure 1: Architectural framework for MRTEs

Further Development of MRTEs

While MRTEs provide many benefits, today's managed execution environments need to evolve further to serve the growing demands of the mobile communications and computing industry. Increasing the intelligence and capabilities of mobile clients introduces many new applications and services to mobile devices, such as Web access, e-mail, and multimedia application processing. With these new capabilities, it is critical that the communication capabilities and integrity of the device be protected, even in the presence of unauthorized applications. Several kinds of protection need to be built into the platform to ensure highly reliable communications, including the following:

- Communications need to be isolated from applications.
- Applications need to be isolated from one another.
- Resource management and recovery need to be built into the platform.

Wireless communications also need to guard against incorrect or malicious devices in the environment.

ESSENTIAL BUILDING BLOCKS FOR THE NEXT GENERATION OF MANAGED RUNTIME ENVIRONMENTS

Through internal technology development and partnering with key technology providers in the industry, Intel is committed to ensuring future MRTEs meet the requirements of next-generation mobile applications and services. To deliver on this vision, Intel is creating building blocks that meet the following criteria:

* Other brands and names are the property of their respective owners.

- *Standardized.* The mobile software development environment demands this attribute in order for manufacturers and operators to ensure interoperability. Communities and standards groups such as the Open Mobile Alliance (OMA) and the Java Community Process (JCP) are therefore essential.
- *Open.* The best specifications are those that are developed by a wide array of contributors, thereby meeting the needs of the entire mobile industry. Intel sees tremendous value in creating building blocks whose functionality was specified by many contributors.
- *Optimized*. Intel is working diligently to ensure that MRTE building blocks run efficiently on our architectures and deliver value to application developers, original equipment manufacturers (OEMs), and carriers.
- *Scalable.* Application developers are confronted by a myriad of platform choices in targeting applications. Requiring a unique application image for each platform increases development cost and complexities for independent software vendors (ISVs) and carriers. Enabling seamless scalability across the range of mobile platforms, from low-end cellular terminals to high-end "Smart Phones," is a key design requirement.
- *Adaptable.* MRTEs and the applications that take advantage of them are evolving rapidly. In anticipation of the needs of newly emerging MRTE standards, the low-level building blocks provided by mobile platforms need to be more general-purpose and more powerful than what is offered today. This enables rapid time-to-market for system designers by allowing them to adapt the platform's foundation capabilities quickly and efficiently to the new standards.

With these criteria in mind, Intel and its partners are working hard on the next generation of MRTE building blocks, which are described next.

MRTE Building Blocks

Described below are some of the attributes of the next generation of MRTE building blocks.

- *Advanced dynamic compilers*. The initial versions of MRTEs for cellular terminals were generally reliant on interpreted execution. This was fine for simplistic data applications, but it fails to meet the performance requirements of the latest mobile applications and services. Combined with the latest advances in mobile application processor technology found in the Intel® XScale™ technology family, advanced dynamic compilers deliver superior performance within a memory-constrained environment.
- *Platform management.* One of the most resource-intensive aspects of cellular operations is the customer care requirement for deployed handsets. Diagnosing and resolving customer problems in an efficient manner can enhance the operator's bottom line. New platform management technologies will enable carrier operations to more quickly spot and fix software problems and identify hardware issues for replacement or repair.
- *Extended battery life.* A key criterion in the end-user selection process is battery life, both standby and talk-time. The addition of data services will only increase the demand for battery-saving technologies. Intel is hard at work, both at the platform level and in the MRTE environment, to deliver power management capabilities and more power-efficient components and building blocks.
- *Flash management*. The Intel Personal Internet Client Architecture (PCA) supports scaling of flash memory over a wide range of densities and mid-level building blocks for a flash file system.
- *Secure provisioning.* The industry's "best known methods" for checking the integrity and permissions of software and commands centers on verifying digital signatures against configured public keys that specify their authorized source. A secure provisioning building block provides capabilities for configuring authorization keys; verifying digital signatures of software, commands, or other data; defining sets of permissions and associating them with authorities; and checking permissions. This foundation building block can easily be adapted to implement current and emerging provisioning standards.
- *Resource monitor and recovery*. These functions provide a generic mechanism for the managed runtime environment to track the allocation of system resources, such as peripherals, and to recover the resources in the event of an unexpected application termination or failure to return resources.

Intel® XScale™ is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

- *User-data management*. These interfaces provide a framework for applications to separate data created by the user from application installation data, giving the ability to backup, remove, and restore user data and application data independent of one another.

- *Data location management*. This building block supports an evolving industry trend toward flexibility and transparency regarding the actual location (local or remote) of a persistent data object to promote "go-anywhere, access-anytime" availability of data.

DELIVERING A MOBILE DEVICE PLATFORM OPTIMIZED FOR MANAGED RUNTIME ENVIRONMENTS

Intel has a long history of delivering building block components to the computing industry. Delivering high-performance platform components for the converged communications and computing industry is a top strategic priority for Intel. Achieving the best-possible performance often involves a particular challenge: on the one hand, a performance improvement frequently takes advantage of a particular hardware feature; while on the other hand, it is important to keep applications free from particular hardware dependencies so that they can scale to a wide variety of devices.

A highly effective technique for meeting both of these criteria is to identify areas where performance is critical, optimize them with native features at a low level, and adapt these low-level optimizations to widely used, standardized industry interfaces. In this way, applications only "see" standardized interfaces, so they may run anywhere, while still getting higher performance on optimized platforms.

Some current and future fruitful areas that Intel has found for optimization are briefly described below:

- *Intel® Integrated Performance Primitives*. These primitives provide high-speed implementations of functions used in algorithms such as multimedia codecs (encoder/decoder engines). They significantly reduce the time and effort spent on algorithm development. The Intel Performance Primitives Library is a low-level building block that may be used as the foundation for other mid-level building blocks such as media playback and graphics.

- *Real-time media playback.* A rapidly emerging use for mobile devices is in the area of playback of real-time media types such as audio and video, either from streaming or stored sources. This tends to demand high computation rates to achieve good playback quality and high compression or decompression ratios. A current R&D effort is focused on producing a high-performance optimized JSR 135 Java Mobile Media framework, including a set of audio and video players for popular formats.

- *High-performance graphics.* Mobile devices are experiencing rapidly increasing use for graphics-intensive gaming. Today's mobile games emphasize 2D animation, scrolling, and sprites. As this trend continues, we can expect such games to expand to 3D graphics. Future R&D efforts will focus on producing optimized implementations of the mobile information device profile (MIDP) 2.0 2D gaming additions and the JSR 184 3D graphics libraries.

- *Execute in place*. This library provides functions for optimizing application modules for high-speed start-up directly out of flash memory. It does not have a distinct interface directly available to applications but would instead be integrated into the loader or the optimizing compiler of a managed runtime environment (MRTE) virtual machine.

Delivering high-performance, memory-efficient, and energy-friendly MRTE solutions is a tricky balance. MRTE solutions must also conform to industry standards and scale across a range of platform capabilities. The need for optimized libraries grows along with the rapid emergence of new MRTE libraries. One company alone is not up to the optimization task. Intel has developed a number of initiatives and established a broad range of industry partnerships to enable "best-of-class" MRTE support. Some of these efforts are described below:

- *Tools.* Intel has developed or supported the development of several different optimizing tool-chains for the Intel® XScale™ software creation. Details can be found at http://www.intel.com/software/products/. In addition, Intel has released a version of its award-winning VTune optimization tool to better support software developers creating high-performance software for our architecture.

- *Guides.* The Intel XScale architecture is a tremendous advance over our previous StrongARM implementation, both in terms of processor clock frequency and in the detail of its microarchitecture. To assist the software developer in building high-

® Intel is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

Intel® XScale™ is a trademark of Intel Corporation or its subsidiaries in the United States and other countries.

performance Intel XScale applications and MRTEs, developer documentation can be found at http://www.intel.com/design/intelxscale.

- *Partnerships.* Intel has established a number of critical relationships for "best-of-class" MRTE solutions. Many of these have yet to be publicly announced. Combined with superior platform and architecture technology, these efforts will enable new classes of mobile applications and services that will drive new business opportunities for operators and manufacturers alike.

CONCLUSION

Mobile data applications represent the next big revenue opportunity for the wireless industry. New hardware and software technologies will enable mobile handsets to run Internet-aware applications and Web services, and managed runtime environments like Java* and Microsoft .NET* Framework are making it easier to quickly create, deploy, and manage mobile data applications. Managed runtime environments effectively insulate applications from the variables of the systems they run on, reducing development time and easing deployment. They also help provide secure, manageable, and connected applications to users, increasing the demand for data services on wireless networks.

Still, the technological and business challenges facing today's mobile application developer are complex, daunting, and rapidly changing. Managed runtime environments provide the device independence, software portability, speed of development, and security that today's Internet applications demand, while the Intel Personal Internet Client Architecture, coupled with Intel's software building blocks allow hardware and software developers to more easily implement these capabilities in new handheld designs. These systems can take advantage of low-level hardware features for efficiency while supporting high-level standards, thus allowing application interoperability across a wide range of devices.

REFERENCES

[1] Instat/MDR, *Mobile Internet Access Devices, 2002-2007*, April 2002.

[2] ARC Group, *Future Mobile Handsets: Worldwide Technology & Market Developments, 2002-2007*, April 2002.

* Other brands and names are the property of their respective owners.

AUTHORS' BIOGRAPHIES

Paul Drews is a senior software engineer in the Emerging Platforms Lab, currently working on performance-optimized Java* libraries for Intel® XScale™ technology. Mr. Drews has been with Intel for over 20 years, and he has been awarded eight patents in the areas of operating system performance, graphics, distributed hypertext, and security. He also received an Intel Achievement Award for his work on the WinSock 2 project. Drews received his B.A. degree from Luther College with triple majors in physics, mathematics, and computer science. His e-mail is paul.drews@intel.com.

Doug Sommer is a senior engineering manager in Intel's Emerging Platforms Lab. Doug and his group are focused on high-performance virtual machine technologies and new manageability and security capabilities for mobile platforms. Doug has worked for Intel for the last eleven years on projects such as development tools, digital video, new media, Java technology, and mobile platform building blocks. He holds a B.S. degree in Computer Science from Oregon State University. His e-mail is doug.sommer@intel.com.

Roger Chandler is a Market Development Manager in Intel's Emerging Platform Lab. While at Intel, his areas of focus have included manufacturing process analysis, high-performance 3D technologies, and home networking. He was a 2001 recipient of an Intel Achievement Award for his work in the field of Web 3D. He is currently focused on market development strategies for Intel's many wireless technology initiatives. He holds an MBA degree from the University of Georgia and a B.A. degree from the University of Tennessee. His e-mail is roger.d.chandler@intel.com.

Terry Smith is a Business Development Manager in Intel's Emerging Platform Labs, focusing on Intel's work in Managed Runtime Environments and Mobile Architectures. Previously he was responsible for Intel's Common Data Security Architecture strategy. He holds an MBA from the University of Texas-Austin and a B.S. degree in Math/CS from the University of Illinois. His e-mail is terry.a.smith@intel.com.

* Other brands and names are the property of their respective owners.

For further information visit:

developer.intel.com/technology/itj/index.htm