J2ME™ Content Handler API

Version 1.0 for the Java™ 2 Platform, Micro Edition

Developers serve an ever-growing market that comprises not only existing servers and PCs, but also diverse emerging products, including set-top boxes, embedded devices, and wireless mobile electronics such as cell phones and PDAs. While each of these new conveniences requires access over the Internet to back-end data, applications, and functionality, they also all use different technologies, protocols, and formats.

As this market for information devices continues to expand and diversify, the lack of integrated cross-platform, cross-device, content-handling capability has become increasingly apparent. Users expect, and demand, the ability to seamlessly access different applications and apply them to a variety of data and data types.

In the past, Java™ 2 Platform, Micro Edition (J2ME™) platform developers lacked a standard, nonproprietary method and application model that allowed invocation of J2ME applications from native applications, and vice versa. As a result, developers faced the prospect of writing, maintaining, and debugging unique code for every conceivable type of consumer electronics platform. Custom development proved so expensive and impractical that mobile and embedded devices effectively lay beyond the reach of content handlers and applications.

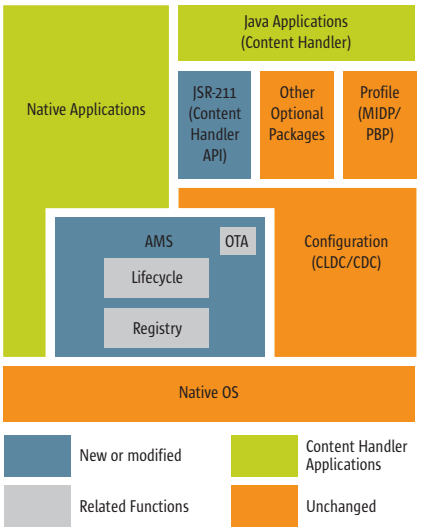
Highlights

- Provides a mechanism for J2ME™ applications to register and handle MIME types.
- Improves the user experience by seamlessly integrating applications that handle different types of content.
- Allows the creation of more compelling applications that can exchange and use data from one to another.
- Enables developers to leverage existing applications to reduce development time.

The JSR-211 Content Handler API Specification

The purpose of the J2ME Content Handler API (CHAPI) specification is to define an optional package for a content handler API and associated application model, permitting the invocation of J2ME applications from native applications, and vice versa.

The CHAPI execution model allows an application to invoke registered Java technology applications and applications based on other technologies by URL, content type, or content handler ID. The application can use actions provided by the handler on the invoked content. By enabling J2ME applications to handle multimedia and Web content, developers can provide users with a seamless, integrated user environment for mobile phones and wireless devices.



A Java and Native Runtime Environment including the Content Handler API

For example, an application can use a content handler to obtain a specific display, or it can use other content handler capabilities such as Open, Print, and Play. An application such as a Mobile Information Device Profile (MIDP) MIDlet or a Personal Profile xlet can be registered to handle the MIME type image in the Portable Network Graphics (PNG) format. Other applications can then request the handler to display the image. The handler can provide

multiple actions that control how the content is displayed, modified, or returned. The execution model leverages the device's application management software (AMS) to provide a smooth user experience, control the execution of the applications, conserve resources, and enforce the security policy of the device and its Java runtime environment.

Benefits for developers

The JSR-211 CHAPI specification allows developers to:

- Register Java technology applications to handle MIME types
- Invoke Java technology applications via URL
- Handle a URL in an invoked application
- Invoke a native application from a Java technology application and vice versa
- Integrate different applications

These and other CHAPI benefits allow developers to create new, value-added applications, and reduce time to market by using existing applications.

Benefits for end users

CHAPI applications can provide end users with seamless integration of applications handling different content types. For example:

Example #1 — A user receives a Short Message Service (SMS) note with a link to a media file (such as JPG or MPEG).

1. The media file has a specific MIME type.
2. If the media player application is not already present, a well-known provisioning server is queried with the MIME type, and the appropriate application is downloaded and installed on the device.
3. The media player application is started with the link to the media file.

Example #2 — Software update. An application, such as a game, checks to see if there is a newer version available.

1. If an update is available, the application invokes the link to the newest version, and triggers the installation.
2. When the installation is complete, the updated application is restarted.

About Sun

For years, customers have turned to Sun Microsystems to help them expand their business, lower their costs, and gain competitive advantage. Sun is a leading provider of industrial strength hardware, software, services, and technologies that make the Net work.

For more information

To learn more about the JSR 211 Content Handler API Specification, visit http://java.sun.com/products/chapi.

Learn More

Get the inside story on the trends and technologies shaping the future of computing by signing up for the Sun Inner Circle program. You'll receive a monthly newsletter packed with information on the latest innovations, plus access to a wealth of resources. Register today to join the Sun Inner Circle Program at sun.com/joinic. To receive additional information on Sun software, products, programs, and solutions, visit sun.com/software.

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 USA **Phone** 1-650-960-1300 or 1-800-555-9SUN **Web** sun.com