| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV PACKAGE**](http://docs.google.com/java/awt/im/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/awt/image/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/im/spi/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

## Package java.awt.im.spi

Provides interfaces that enable the development of input methods that can be used with any Java runtime environment.

**See:**

[**Description**](#3znysh7)

| **Interface Summary** | |
| --- | --- |
| [**InputMethod**](http://docs.google.com/java/awt/im/spi/InputMethod.html) | Defines the interface for an input method that supports complex text input. |
| [**InputMethodContext**](http://docs.google.com/java/awt/im/spi/InputMethodContext.html) | Provides methods that input methods can use to communicate with their client components or to request other services. |
| [**InputMethodDescriptor**](http://docs.google.com/java/awt/im/spi/InputMethodDescriptor.html) | Defines methods that provide sufficient information about an input method to enable selection and loading of that input method. |

## Package java.awt.im.spi Description

Provides interfaces that enable the development of input methods that can be used with any Java runtime environment. Input methods are software components that let the user enter text in ways other than simple typing on a keyboard. They are commonly used to enter Japanese, Chinese, or Korean - languages using thousands of different characters - on keyboards with far fewer keys. However, this package also allows the development of input methods for other languages and the use of entirely different input mechanisms, such as handwriting recognition.

## Package Specification

* [**Input Method Framework Specification**](http://docs.google.com/technotes/guides/imf/spec.html)
* [**Input Method Engine SPI Reference**](http://docs.google.com/technotes/guides/imf/spi-reference.html)

#### Packaging Input Methods

Input methods are packaged as installed extensions, as specified by the [Extension Mechanism](http://docs.google.com/technotes/guides/extensions/index.html). The main JAR file of an input method must contain the file:

META-INF/services/java.awt.im.spi.InputMethodDescriptor

The file should contain a list of fully-qualified class names, one per line, of classes implementing the java.awt.im.spi.InputMethodDescriptor interface. Space and tab characters surrounding each name, as well as blank lines, are ignored. The comment character is '#' (\u0023); on each line all characters following the first comment character are ignored. The file must be encoded in UTF-8.

For example, if the fully-qualified name of the class that implements java.awt.im.spi.InputMethodDesciptor for the *Foo* input method is com.sun.ime.FooInputMethodDescriptor, the file META-INF/services/java.awt.im.spi.InputMethodDescriptor contains a line:

com.sun.ime.FooInputMethodDescriptor

The input method must also provide at least two classes: one class implementing the java.awt.im.spi.InputMethodDescriptor interface, one class implementing the java.awt.im.spi.InputMethod interface. The input method should separate the implementations for these interfaces, so that loading of the class implementing InputMethod can be deferred until actually needed.

#### Loading Input Methods

The input method framework will usually defer loading of input method classes until they are absolutely needed. It loads only the InputMethodDescriptor implementations during AWT initialization. It loads an InputMethod implementation when the input method has been selected.

#### Java Input Methods and Peered Text Components

The Java input method framework intends to support all combinations of input methods (host input methods and Java input methods) and components (peered and lightweight). However, because of limitations in the underlying platform, it may not always be possible to enable the communication between Java input methods and peered AWT components. Support for this specific combination is therefore platform dependent. In Sun's Java SE Runtime Environments, this combination is supported on Windows, but not on Solaris.

## Related Documentation

For overviews, tutorials, examples, guides, and tool documentation, please see:

* [**Input Method Framework Overview**](http://docs.google.com/technotes/guides/imf/overview.html)
* [**Input Method Engine SPI Tutorial**](http://docs.google.com/technotes/guides/imf/spi-tutorial.html)

**Since:** JDK1.3

| | [**Overview**](http://docs.google.com/overview-summary.html) | **Package** | Class | [**Use**](http://docs.google.com/package-use.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV PACKAGE**](http://docs.google.com/java/awt/im/package-summary.html)   [**NEXT PACKAGE**](http://docs.google.com/java/awt/image/package-summary.html) | [**FRAMES**](http://docs.google.com/index.html?java/awt/im/spi/package-summary.html)    [**NO FRAMES**](http://docs.google.com/package-summary.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).