

NIO-CHARSET

Overview

NIO-Charset - [java.nio.charset](#), [java.nio.charset.spi](#)

Status

Check the component development status page for the status of other [ClassLibrary](#) modules.

J2SE 1.4.2

Current code in classlib is assumed compliant with J2SE 1.4.2 specification.

Java SE 5.0

Paulex and Richard are working on this module.

New Methods in JSE 5.0

There are two new methods added since Java 5.0

1. `public static Charset defaultCharset()`
2. `public final int compareTo(Charset that)`

Status: Done.

References: [HARMONY-96](#) Two new methods need to be implemented by `java.nio.charset.Charset`

Bug fixing

Here are some bugs reported again nio-charset

[HARMONY-45](#) `Charset.isRegistered()` method always return true

[HARMONY-64](#) `java.nio.charset.Charset.forName(String name)` does not throw [UnsupportedCharsetException](#) if invalid name starts with "x-"

[HARMONY-67](#) `java.nio.charset.Charset.decode(ByteBuffer)` throws unexpected [BufferOverflowException](#) for UTF-16BE, UTF-16LE, UTF-16 charsets.

[HARMONY-68](#) `java.nio.charset.Charset.isSupported(String charsetName)` does not throw [IllegalCharsetNameException](#) for spoiled standard charset name

[HARMONY-99](#) `java.nio.charset.CharsetDecoder.decode(ByteBuffer in)` does not throw [MalformedInputException](#) when buffer's current position is not legal

[HARMONY-113](#) `java.nio.charset.Charset` should regard empty charset name properly as illegal [CharsetName](#)

[HARMONY-124](#) `java.nio.charset.CharsetDecoder` doesn't throw [CoderMalfunctionError](#) exception when `decodeLoop` threw unexpected exception.

[HARMONY-137](#) [CharsetDecoder](#) should replace undefined bytes with replacement string

[HARMONY-141](#) Constructors of `java.nio.charset.CharsetEncoder` do not validate arguments

[HARMONY-142](#) `java.nio.charset.CharsetDecoder` constructor doesn't throw [IllegalArgumentException](#) when `averageCharsPerByte` is greater than `maxCharsPerByte`.

[HARMONY-148](#) `java.nio.charset.CharsetDecoder: decode(in,out,endOfInput)` method doesn't preserve replace string for successive decode invocation while "out" doesn't have enough space for replace string.

[HARMONY-149](#) `java.nio.charset.Charset.forName(name)` return different reference value when requiring the same `Charset`.

[HARMONY-150](#) `java.nio.charset.Charset.decode(in)` doesn't use the same cached decoder.

[HARMONY-170](#) `java.nio.charset.CharsetEncoder.encode` method does not handle the unexpected exception

[HARMONY-173](#) `java.nio.charset.CharsetEncoder.encode` method does not reserve remaining bytes for next invocation.

[HARMONY-182](#) `java.nio.charset.Charset.encode(in)` doesn't use the same cached encoder.

Refector

[HARMONY-123](#) Refactor: `java.nio.charset.CharsetDecoder.decode`