



**JavaOne**<sup>SM</sup>

Sun's 2001 Worldwide Java Developer Conference

# New I/O APIにおけるJavaの 国際化について

風間一洋  
主任研究員  
日本電信電話株式会社

# 目的

Java 2 Platform, Standard Edition (J2SE) 1.4で追加されたNew I/O APIから、特に国際化に関連した変更・追加部分を取り上げて解説する。

# 目次

- Unicode 3.0対応
- Unicode文字の内部表現の変更
- 文字シーケンス
- 文字バッファ
- 文字集合と変換
- Unicode正規表現
- Old I/O APIの変更点

# Unicode 3.0対応

- Unicode 3.0への対応
  - 文字の追加 (ISO/IEC 10646-1 第二版と同期)
- 開発キットとUnicodeのバージョンの対応
  - JDK 1.0 = Unicode 1.1
  - JDK 1.1 ~ J2SDK 1.3 = Unicode 2.0 ~ 2.1
  - J2SDK 1.4 = Unicode 3.0

# Unicode文字の内部表現の変更

## 内部表現

- Unicode文字の内部表現形式の変更
  - UCS-2から**UTF-16**に正式に変更
  - Unicode文字が、**不定長**で表現される
- サポートされる文字
  - BMP (Basic Multilingual Plane、第0面)
    - $0000\ 0000_{16} \sim 0000\ FFFF_{16}$  (UCS-4)
  - 第1 ~ 16面
    - $00010000_{16} \sim 0010\ FFFF_{16}$  (UCS-4)
    - JIS X 0213の一部の文字が割り当て予定

# Unicode文字の内部表現の変更

## UTF-16

- U+0000 ~ U+FFFF (第0面)
  - $0000_{16} \sim FFFF_{16}$
- U+10000 ~ U+10FFFF (第1 ~ 16面)
  - 2つのUnicode文字を組み合わせて表現
  - ハイサロゲート U+D800 ~ U+DBFF
  - ローサロゲート U+DC00 ~ U+DFFF
- 例、第一面 (U+10000 ~ U+1FFFF)
  - $D800\ DC00_{16} \sim D83F\ DFFF_{16}$

# 文字シーケンス

## 文字のシーケンスに対するビューの統一

- java.lang.CharSequenceインターフェイス
- 読み出し専用の手続きの統一
  - java.lang.Stringクラス
    - 変化しない、同期なし
  - java.lang.StringBufferクラス
    - 変化する、自動拡張あり、同期あり
  - java.nio.CharBufferクラス
    - 変化する、自動拡張なし、同期なし

# 文字シーケンス

## 文字シーケンスの操作

- `charAt(int index)`
- `length()`
- `subSequence(int start, int end)`
- `toString()`

# 文字バッファ

- java.nio.CharBufferクラス
  - Unicode文字列を格納するクラス
  - java.nio.Bufferクラスのサブクラス

# 文字バッファ

## 文字の追加

- StringBufferの操作に類似している

```
new CharBuffer(10).put('a').put('b').toString();
```

# 文字集合と変換

## 文字集合

- java.nio.charset.Charsetクラス
  - byte列とchar列の間のマッピング名
  - 基本的に双方向
- 名前の分類
  1. 正準名 (canonical name)
    - 表示名 (display name)...ロケール依存
  2. エイリアス (aliases)
    - 互換名 (historical name)

# 文字集合と変換

## 正準名と推奨MIME名

- 正準名
  - MIMEを用いるアプリケーションを考慮
  - 原則としてIANAに登録済の推奨MIME名 (preferred MIME name)を用いる
  - IANAに未登録のcharset名は、“X-”で開始する
- IANAに登録されているかの検証
  - `isRegistered()`

# 文字集合と変換

## 変換関連クラス

- `java.nio.charset.CharsetDecoder`クラス
  - バイト列からUnicode文字列への変換
- `java.nio.charset.CharsetEncoder`クラス
  - Unicode文字列からバイト列への変換
- `java.nio.charset.CoderResult`クラス
  - 変換結果
- `java.nio.charset.CodingErrorAction`
  - エラー発生時の動作を定義

# 文字集合と変換

## *java.nio.charset.CharsetDecoder* クラス

- ByteBuffer CharBuffer
- 変換過程
  1. reset()
  2. decode(ByteBuffer in, CharBuffer out, **false**)を0回以上繰り返し
  3. decode(ByteBuffer in, CharBuffer out, **true**)
  4. flush()

# 文字集合と変換

*java.nio.charset.CharsetEncoder* クラス

- CharBuffer ButeBuffer
- 変換過程
  1. reset()
  2. encode(CharBuffer in, ByteBuffer out, **false**)を0回以上繰り返し
  3. encode(CharBuffer in, ByteBuffer out, **true**)
  4. flush()

# 文字集合と変換

## コンビニエンスメソッド

```
byte[] bytes =  
    Charset.forName("ISO-2022-JP").encode(chars);  
  
char[] chars =  
    Charset.forName("UTF-8").decode(bytes);  
  
// 従来のStringクラスのコンストラクタとしては追加しなかった
```

# 文字集合と変換

*java.nio.charset.CoderResult* クラス

- アンダーフロー
- オーバーフロー
- 不正な入力エラー
- マップ不可能文字エラー

# 文字集合と変換

*java.nio.charset.CodingErrorAction* クラス

- エラーに対する挙動を指定
  - 不正な入力
  - マップできない文字
- 3種類の挙動
  1. 無視 (IGNORE)
  2. 置換 (REPLACE)
  3. 報告 (REPORT)
    - `CoderResult`を返す
    - 例外の発生

# 文字集合と変換

## 文字エンコーディングの自動検出

- 自動検出の識別
  - `isAutoDetecting()`
- 検出状態の獲得
  - `isCharsetDetected()`
- 検出された文字エンコーディングの取得
  - `detectedCharset()`
- エンコード可能(双方向変換可能)かの判別
  - `canEncode()` (`java.nio.charset.Charset`)

# 文字集合と変換

## コンバータの追加

- `java.nio.charset.spi.CharsetProvider`クラス
- 拡張機能としてJARファイルをインストール
- `META-INF/services`ディレクトリ
  - `java.nio.charset.spi.CharsetProvider`ファイル
  - 一行ごとにクラス名を記述する

# Unicode正規表現

- java.util.regex.Patternクラス
  - Unicode文字を用いて正規表現を定義可能
  - Unicodeブロックのサポート

# Old I/O APIの変更点

- InputStreamReader/OutputStreamWriter
  - getEncodingメソッドの追加

# まとめ

- J2SE 1.4における国際化の拡張、変更点について解説した
- このBOFでは、以下の点について議論する
  - 現在の仕様に問題点があるか?
  - 現在の仕様が対応していない領域は?

# ポインタ

- JSR-51: New I/O API for the Java Platform
  - <http://jcp.org/jsr/detail/051.jsp>
  - [jsr-51-comments@jcp.org](mailto:jsr-51-comments@jcp.org)
- J2SE 1.4
  - <http://java.sun.com/j2se/1.4/docs/guide/nio/>
- Unicode Regular Expression Guidelines
  - <http://www.unicode.org/unicode/reports/tr18/>



**JavaOne**<sup>SM</sup>

Sun's 2001 Worldwide Java Developer Conference

**Q&A**



**JavaOne**<sup>SM</sup>

Sun's 2001 Worldwide Java Developer Conference\*