

Design notes

by Jeremias Märki

Table of contents

- 1 Requirements..... 2
- 2 Package overview..... 2
 - 2.1 org.krysalis.barcode4j..... 2
 - 2.2 org.krysalis.barcode4j.impl..... 2
 - 2.3 org.krysalis.barcode4j.output (and subpackages)..... 2
 - 2.4 org.krysalis.barcode4j.tools..... 2
 - 2.5 org.krysalis.barcode4j.cli..... 2
 - 2.6 org.krysalis.barcode4j.servlet..... 2
 - 2.7 org.krysalis.barcode4j.xalan..... 3
 - 2.8 org.krysalis.barcode4j.saxon..... 3
 - 2.9 org.krysalis.barcode4j.playground..... 3

1 Requirements

- The whole barcode package shall be flexible and extensible. New implementations (barcode type or output format) should be easy to add.
- The design shall be done in a way that different barcode types (1D vs. 2D) can be supported.
- The barcodes implementations shall be configured using XML and in a way that the whole package can easily be integrated into a bigger work.
- The barcode logic (checksums, encoding, character sets etc.) shall be separated from the rendering of the barcode so different output formats (SVG, bitmaps etc.) can easily be supported.

2 Package overview

2.1 org.krysalis.barcode4j

This package holds the primary interfaces for generating barcodes.

2.1.1 Main interfaces

- **BarcodeGenerator**: Main interface for generating barcodes.
- **BarcodeClassResolver**: Resolves symbolic names to barcode implementation classes.
- **BarcodeLogicHandler**: Receives barcode events (similar to SAX events for XML) generated by barcode logic implementations.

BarcodeUtil is an very useful helper class. Have a look!

2.2 org.krysalis.barcode4j.impl

This package is the home of the standard barcode implementations. Each implementation consist mostly of two classes, a logic implementation and a presentation implementation. The presentation part calls the logic part to generate something similar to SAX events but for barcodes (See **BarcodeLogicHandler**).

2.3 org.krysalis.barcode4j.output (and subpackages)

This is where the various output formats are implemented. To implement a new coordinate-based output format the main task is to create a new **CanvasProvider** implementation.

2.4 org.krysalis.barcode4j.tools

This package contains various helper classes used internally.

2.5 org.krysalis.barcode4j.cli

This package contains the command-line interface.

2.6 org.krysalis.barcode4j.servlet

This package contains the barcode servlet.

2.7 org.krysalis.barcode4j.xalan

This package contains the XSLT extension for [Apache Xalan](#).

2.8 org.krysalis.barcode4j.saxon

This package contains the XSLT extension for the [SAXON XSLT Processor](#) by [Michael Kay](#).

2.9 org.krysalis.barcode4j.playground

This package contains the playground GUI application. It's used to play around with barcodes. Have fun!

Note:

This is work in progress! Help wanted! Be creative!