spotxcolor Package

Munehiro Yamamoto

2026/03/14 v1.0

Abstract

This package provides robust spot color (e.g., DIC, PANTONE) support for the xcolor package across all major TEX engines. It resolves structural PDF issues found in legacy packages and provides explicit fallback mechanisms for various drivers.

1 Motivation & Features

While the legacy spotcolor package has been used for years, it suffers from structural PDF compatibility issues with modern expl3 and xcolor updates. The colorspace package is an excellent modern alternative, but it lacks full support for the `dvipdfmx` driver.

To address these gaps, the spotxcolor package natively hooks into the xcolor package to provide a universal, print-safe solution. Its core features include:

- **Native xcolor Integration:** Use spot colors exactly like standard colors (e.g., `\textcolor`, `\pagecolor`), generating perfect PDF structures for `pdftex` and `luatex`.
- **Universal Engine Support & Safe Fallbacks:** Ensures safe CMYK fallback for `dvipdfmx` and `xetex` when using standard macros, while providing explicit injection commands (`\SpotColor`) for true spot color output in `dvipdfmx`.
- **Advanced Graphics (TikZ/PGF) Support:** Internally patches PGF to ensure print safety. It fully supports fill/stroke separation, fadings, shadings, blend modes, and safely forces uncolored patterns (including `patterns.meta`) to CMYK instead of hardcoded RGB.
- **Seamless Decorations:** Works flawlessly with colortbl (zebra-striped tables) and tcolorbox (frames, backgrounds, shadows).
- **Complete Backward Compatibility:** Perfectly emulates user commands from both spotcolor and colorspace packages.

2 Requirements

This package requires a modern TeX Live environment (with an up-to-date expl3/l3kernel), alongside the xcolor and iftex packages.

3 Loading the spotxcolor Package

Load the spotxcolor package in your preamble. It automatically detects your engine via the iftex package.

```
\usepackage{spotxcolor}
```

4 Usage

4.1 Defining and Using Spot Colors

Use `\definespotcolor` to register a spot color. The arguments are: LaTeX name, PDF name, and CMYK values. Once registered, you can use standard xcolor commands:

```
\definespotcolor{DIC161}{DIC 161s*}{0, 0.64, 1, 0}

\textcolor{DIC161}{This text is DIC 161.}
```

4.2 Dvipdfmx Explicit Spot Colors

Because the `dvipdfmx` backend strictly falls back to a CMYK representation when using standard `\textcolor`, you can force true spot color output (creating an independent separation plate in the PDF) using the `\SpotColor` command:

```
\SpotColor{DIC161}{1.0}
This text is 100\% DIC 161.

\SpotColor{DIC161}{0.5}
This text is 50\% DIC 161.
```

4.3 Note for (u)pLaTeX Users

If you compile your document with `platex` or `uplatex` and generate a PDF via `dvipdfmx`, you **must** specify the `dvipdfmx` driver option globally in your document class:

```
\documentclass[dvipdfmx]{article}
```

Without this option, modern expl3 and xcolor will default to the `dvips` driver, and the PDF objects for spot colors will not be generated correctly.

4.4 Backward Compatibility

You can safely reuse your legacy code and dictionaries written for the spotcolor package. They are automatically mapped to modern spotxcolor implementations:

```
\NewSpotColorSpace{DIC}
\AddSpotColor{DIC}{DIC161}{DIC\SpotSpace 161s*}{0 0.64 1 0}
\SetPageColorSpace{DIC}
```