

# org-special-block-extras

Musa Al-hassy

April 16, 2020

## Contents

1	Example Use	1
2	Core Utility	2
3	Colours	3

## 1 Example Use

User type something along the lines of the following.

```
#+begin_red org
/This/
    *text*
        _is_
            red!
#+end_red
```

Which generates *red* text when exported to HTML and L<sup>A</sup>T<sub>E</sub>X, **while supporting Org markup**.

This article may be read as a PDF or as HTML or as pure Org!



Figure 1: Write Org-markup once, generate for many backends ^\_^

This text is black!  
 This text is blue!  
 This text is brown!  
 This text is cyan!  
 This text is darkgray!  
 This text is gray!  
 This text is green!  
 This text is lightgray!  
 This text is lime!  
 This text is magenta!  
 This text is olive!  
 This text is orange!  
 This text is pink!  
 This text is purple!  
 This text is red!  
 This text is teal!  
 This text is violet!

This text is yellow!

The remaining sections are implementation matter.

## 2 Core Utility

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; Core utility

```

```
(defun org-special-block-extras--advice (backend blk contents info)
  "Invoke the appropriate custom block handler, if any."
```

A given custom block BLK has a TYPE extracted from it, then we send the block CONTENTS along with the current export BACKEND to the formatting function ORG-SPECIAL-BLOCK-EXTRAS/TYPE if it is defined, otherwise, we leave the CONTENTS of the block as is."

```
(let* ((type      (nth 1 (nth 1 blk)))
      (handler (intern (format "org-special-block-extras/%s" type))))
  (ignore-errors (apply handler backend contents nil)))

(advice-add #'org-html-special-block :before-until
  (-partial #'org-special-block-extras--advice 'html))

(advice-add #'org-latex-special-block :before-until
  (-partial #'org-special-block-extras--advice 'latex))
```

### 3 Colours

```
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;;
;; Load support for 19 colour custom blocks
```

```
(defvar org-special-block-extras--colors
  '(black blue brown cyan darkgray gray green lightgray lime
    magenta olive orange pink purple red teal violet white
    yellow)
  "Colours that should be available on all systems.")

(loop for colour in org-special-block-extras--colors
  do (eval (read (format
    "(defun org-special-block-extras--%s (backend contents)
      (format (pcase backend
        ('latex  \"\\\\\\\\begingroup\\\\\\\\color{%s}%%s\\\\\\\\endgroup\")
        ('html   \"<div style=\\\"\\\"color:%s;\\\"\\\">%%s</div>\\\")
        (t       \"org-special-block-extras: Unsupported backend\\\")
        contents))"
    colour colour colour))))))
```