OrgMode by Example

for Literate Research

Shawn Murphy

March 16, 2024

Abstract

EMACS's `org-mode` produces literate, reproducible, publication-quality LaTeX. This document demonstrates basic formatting as well as automatic image creation using R, GraphViz and QRencode. It also illustrates the execution of Shell and Python as well as providing hints at how to install and execute the tools. A familiarity with *EMACS* is assumed.

Contents

1 Basic Techniques

1.1 Styles, Lists, Checkboxes and Links

Here is a simple list:

1. Styling: **bold**, *italic*, underlined, ~~strikethrough~~, `monospaced` or ***~~`mixtures`~~***
   - dashes
   - ☐ unchecked checkboxes
   - ☒ and checked ones
     - bullets
2. Links can be either
   - with description
   - `https://smurp.com` without description

1.2 Block Quotes

> Everything should be made as simple as possible, but not simpler
> —Albert Einstein
>
> Simplify, without excess. —smurp

1.3 Generating PDFs

Generate a PDF with the EMACS key sequence: `C-c C-e l o`

1.4 Controlling the styling more

`https://orgmode.org/worg/org-tutorials/org-latex-export.html`

1.5 Tables

URL	Description
`https://smurp.com/OrgModeByEx.pdf`	as readble PDF
`https://smurp.com/OrgModeByEx.org`	as .org source

2 Installation

2.1 APT-based, ie `debian`, `ubuntu` and friends

```
sudo apt install textlive texlive-latex-extra
```

2.2 Add this section to your `.emacs`

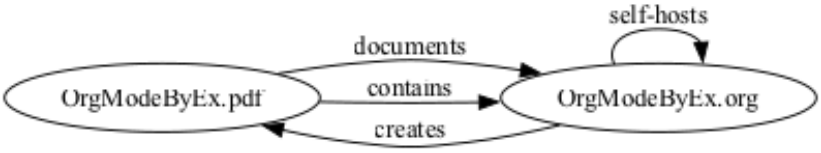
Be sure to include the languages you need

```
(org-babel-do-load-languages
 'org-babel-load-languages '((dot . t)
                             (python . t)
                             (R . t)
                             (shell . t)
                             (emacs-lisp . t)))
```

3 Executing Code

`https://orgmode.org/worg/org-contrib/babel/` for all the details.

3.1 Creating GraphViz Diagrams

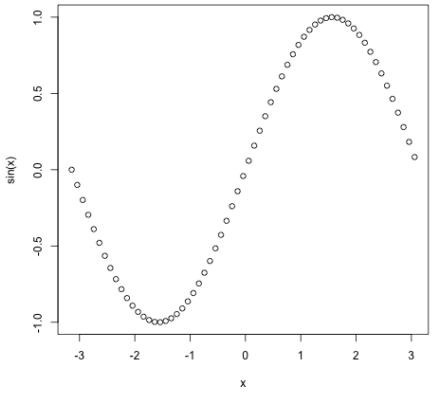


./OrgModeByExGraphViz.png

3.2 Creating R Diagrams

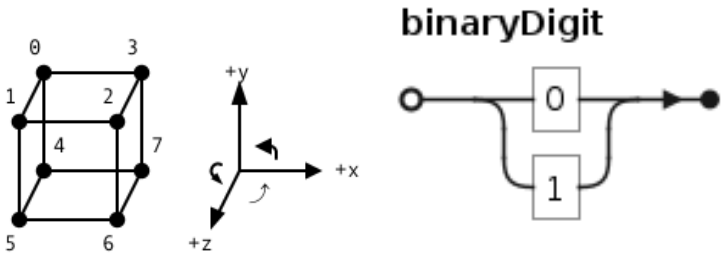
Within *EMACS* install the *ESS* package with: `M-x package-install ess RET`

- Some resources on how to use R in org-mode:
  - `https://orgmode.org/worg/org-tutorials/org-R/org-R.html`
  - `https://stackoverflow.com/questions/32440128/` (Venn diagrams)



3.3 Creating Kroki Diagrams

Requires `ob-kroki.el` from `https://github.com/smurp/ob-kroki`



3.4 Displaying the Results of Calls in Various Languages

```
Python says 'Hello World!'
   AWK says 'Hello World!'
```

3.5 Executing Shell Commands and Displaying Results

```
ls -l OrgModeByEx.{org,pdf} | \
   awk '{print "((date `" $7,$8 ") (fname `" $9 ") (size " $5 "))"}'

((date `Mar 19:10) (fname `OrgModeByEx.org) (size 5317))
((date `Mar 19:09) (fname `OrgModeByEx.pdf) (size 377101))
```

4 The source for this document

```
# -*- org-confirm-babel-evaluate: nil -*- Suppress Babel prompts during export
#+TITLE: OrgMode by Example
#+SUBTITLE: for Literate Research
#+AUTHOR: Shawn Murphy

#+OPTIONS: toc:nil
#+MACRO: SMURP https://smurp.com/
#+MACRO: SELF https://smurp.com/OrgModeByEx$1
#+LATEX_CLASS: article
#+LATEX_CLASS_OPTIONS: [a4paper]
#+LaTeX_HEADER: \usepackage{paralist}
#+LaTeX_HEADER: \let\itemize\compactitem
#+LaTeX_HEADER: \let\description\compactdesc
#+LaTeX_HEADER: \let\enumerate\compactenum

#+BEGIN_SRC shell :exports results
# apt install qrencode # on linux
# brew install qrencode # on OSX
  qrencode -o OrgModeByExQR.png "https://smurp.com/OrgModeByEx.pdf"
#+END_SRC

#+RESULTS:

#+ATTR_LATEX: :width 4cm
[[file:OrgModeByExQR.png]]

#+BEGIN_abstract
/EMACS/'s =org-mode= produces literate, reproducible, publication-
quality LaTeX.
This document demonstrates basic formatting as well as automatic image creation
using R, GraphViz and QRencode. It also illustrates the execution of Shell
and Python as well as providing hints at how to install and execute the tools.
A familiarity with /EMACS/ is assumed.
#+END_abstract
#+TOC: headlines 2
```

```
@@latex:\clearpage@@

* Basic Techniques
** Styles, Lists, Checkboxes and Links
Here is a simple list:
1. Styling: *bold*, /italic/, _underlined_, +strikethrough+, =monospaced= or /*_+=mixtures
  - dashes
  - [ ] unchecked checkboxes
  - [X] and checked ones
    * bullets
2. Links can be either
  - [[https://smurp.com][with description]]
  - https://smurp.com  without description

** Block Quotes
#+BEGIN_QUOTE
Everything should be made as simple as possible, but not simpler ---
Albert Einstein
#+END_QUOTE

#+BEGIN_QUOTE
Simplify, without excess. ---smurp
#+END_QUOTE

** Generating PDFs
Generate a PDF with the EMACS key sequence: =C-c C-e l o=

** Controlling the styling more
https://orgmode.org/worg/org-tutorials/org-latex-export.html

** Tables
| URL               | Description    |
|-------------------+----------------|
| {{{SELF(.pdf)}}} | as readble PDF |
| {{{SELF(.org)}}} | as .org source |

* Installation
** APT-based, ie =debian=, =ubuntu= and friends

#+BEGIN_SRC
sudo apt install textlive texlive-latex-extra
#+END_SRC
** Add this section to your =.emacs=
Be sure to include the languages you need

#+BEGIN_SRC
(org-babel-do-load-languages
 'org-babel-load-languages '((dot . t)
                             (python . t)
```

```
                                (R . t)
                                (shell . t)
                                (emacs-lisp . t)))
#+END_SRC

* Executing Code
https://orgmode.org/worg/org-contrib/babel/ for all the details.

** Creating GraphViz Diagrams

#+BEGIN_SRC dot :file ./OrgModeByExGraphViz.png :cmdline -Tpng -Gdpi=400
digraph mini {
  rankdir="LR"
  size="1,6"
  PDF[ label="OrgModeByEx.pdf" ];
  PDF -> ORG [ label="documents" ]; PDF -> ORG [ label="contains" ];
  ORG[ label="OrgModeByEx.org" ];
  ORG -> PDF [ label="creates" ];   ORG -> ORG [ label="self-hosts" ];
}
#+END_SRC
#+ATTR_LATEX: :width 8cm
./OrgModeByExGraphViz.png

** Creating R Diagrams
Within /EMACS/ install the /ESS/ package with: =M-x package-install ess RET=
- Some resources on how to use R in org-mode:
  - https://orgmode.org/worg/org-tutorials/org-R/org-R.html
  - https://stackoverflow.com/questions/32440128/  (Venn diagrams)

#+BEGIN_SRC R :file /tmp/xyplot.png :session :results graphics :exports none
x <- seq(-pi,pi,0.1)
plot(x, sin(x))
#+END_SRC

#+RESULTS:

#+ATTR_LATEX: :width 6cm
[[/tmp/xyplot.png]]
** Creating [[https://kroki.io/][Kroki]] Diagrams

Requires =ob-kroki.el= from [[https://github.com/smurp/ob-kroki]]

#+headers: :exports none :results none
#+begin_src kroki :file kroki-svgbob-ex.svg :cmdline --type svgbob
    0       3
     *-------*      +y
  1 /|    2 /|       ^
   *-+-----* |       |
   | |4    | |7      |
   | *-----|-*      +-----> +x
```

```
   |/      |/       /
   *-------*       v
  5       6      +z
#+end_src

# Since SVGbob can only produce SVG output and PDFLatex can use SVG
# its diagrams require conversion of the svg to png, performed by inkscape here.
#+headers: :exports none :output none
#+begin_src shell :async no
  inkscape --export-filename kroki-svgbob-ex.png kroki-svgbob-ex.svg
#+end_src

#+headers: :results none
#+begin_src kroki :file kroki-ebnf-ex.png :cmdline --type plantuml
@startebnf
binaryDigit = "0" | "1";
@endebnf
#+end_src

# https://stackoverflow.com/questions/24326290/org-mode-side-by-side-
figure-captions-for-latex-export
#+CAPTION: svgbob and ebnf examples
#+begin_center
#+ATTR_LATEX: :width 0.4\textwidth :center nil
[[file:kroki-svgbob-ex.png]]
#+ATTR_LATEX: :width 0.4\textwidth :center nil
[[file:kroki-ebnf-ex.png]]
#+end_center


** Displaying the Results of Calls in Various Languages
src_python{return("Python says 'Hello World!'")}

src_awk{BEGIN {print "AWK says 'Hello World!'"}}

src_lfe{(lfe_io:format "lfe says: hello world~n" ())}


** Executing Shell Commands and Displaying Results
#+BEGIN_SRC shell :results output :exports both
ls -l OrgModeByEx.{org,pdf} | \
   awk '{print "((date `" $7,$8 ") (fname `" $9 ") (size " $5 "))"}'
#+END_SRC

* The source for this document
#+BEGIN_SRC shell :results output :exports results
cat ./OrgModeByEx.org
#+END_SRC
```