

# Example lesson

How to teach using org-mode  
for fun and profit

Olivier Berger

2016-12-20

# Table of Contents

- Introduction
- About this slides deck
- Features
- HowTo
- How it works / Installation
- Annex

# Introduction

This is a demo document about the [codename org-teaching](#) framework.

- *You're viewing a [reveal.js](#) Web slides deck. You may press 's' to view presenter notes.*

# Org-mode powa

This framework heavily relies on [org-mode](#) (version 9 at the time of writing) and the [org-reveal](#) exporter for reveal.js.

# About this slides deck

These slides are one variant of the same teaching material,  
also available [as a handbook](#).

You may prefer to view them in your Web browser in full-  
screen (F11 for instance).

# Features

# Writing teaching material in org-mode

The goal is to be able to edit a single file (namely `lesson.org`) which will contain, in a single source, all the content of a lesson, written with org-mode syntax.

From this single source, several documents are generated :

- **slides** (as a dynamic Web document) for overhead presentation
- a **handbook** that contains the same information (or more) and can be handed to the students for work outside the classroom (a PDF file)
- another **handbook** for the teaching team, to provide additional instructions (also a PDF file)

# org-reveal slides

Pretty much all features of reveal.js supported by the org-mode reveal.js exporter [org-reveal](#), should be supported too.

An example org-reveal document is [available here](#) for inspiration (it's the export of org-reveal's README.org, actually).



# Structure of the sections / slides

I'm using the 3 levels of outlining / sectioning so that the content can be sectioned in the same way in lesson.org and appear appropriately in the slides and handbook, with these principles:

1. First level outlines define main sections of the document.
2. Second level outlines are the main "horizontal" slides that will be played with page up/down
3. Third level outlines may be used for additional content ("vertical" slides) that may be skipped for the presentation, but is still accessible with cursor keys.

# Presenter notes / content for the handbook

`org-reveal's Speaker notes` may be added to the slides (and will only appear on dual-screen presentation after having pressed 's': standard reveal.js feature).

They will be masked for the audience, but will, by default appear in the handbook given to the students.

```
# +BEGIN_NOTES  
This is a note  
# +END_NOTES
```

# Masking content for some audiences

I've implemented some easy ways to preserve some of the content of the same lesson.org source for certain outputs (using org exporter's standard EXCLUDE\_TAGS):

## ***Slides only* material**

that won't be embedded in the handbook : surprise stuff for live audience, or HTML-only hacks;

## ***Teachers only* material**

secret knowledge that only adults need to know (for instance), which won't be exported;

## ***Handbook only* material**

stuff that only fits in the handbook, and/or only exports as LaTeX and not HTML.

# Stuff only meant for presentation

Tagging a section/slide with `:slidesonly:` means it isn't exported in the handbooks.

Below is an example (or not)...

Regular slide (no tag on heading line)

There should be no "Only in the slides" after this section, in the handbooks, as it has been tagged with slidesonly.

Only in the slides

On the contrary, in the slides view, this appears, as there's a  
:slidesonly: tag on the current head line.

# Stuff only meant for teachers

Tagging a section/slide with :teachersonly: means it isn't exported in the students handbook (nor in the slides).

Below is an example...

Regular slide (no tag on heading line)

There should be no "Only for teachers" after this section, in the slides or in the students handbook, as it has been tagged with teachersonly.



# Notes only for the teachers

This slide/section contains notes, but only part of it is displayed in the presentation notes included in the handbook. Special notes and are kept only for the teachers handbook.

We use an org-mode drawer for that (additional benefit is that the content is folded by default in emacs, as it may be verbose and/or "sensitive") :

```
# +BEGIN_NOTES
```

```
This part of the note can be viewed by the students in the handbook.
```

```
:TEACHERONLY:
```

```
Not this one
```

```
:END:
```

```
# +END_NOTES
```

# Stuff only in the handbooks

Just like sections are for slides only, others can be for the handbook only, using the `handbookonly` tag. This may be useful for **Annex** sections for instance, or for stuff that the HTML exporter won't like, with inline LaTeX.

# Code colorization

Code is colorized / highlighted in the slides :-)

# Misc org-mode

## Babel powa

As you're using org-mode, its babel components are available, to embed source code in the same lesson.org source, and manage executable code and teaching material at once.

Look for *literate programing* instructions in the [org-mode docs](#) to know more.

# Missing features ?

Please try and talk to me to suggest new stuff and/or provide patches ;)

# HowTo

# Use Emacs

There's no production chain, makefiles or other authoring tools.

No new Elisp either.



# Modify only the lesson.org

**Only one file should be edited for writing the lesson's material : **lesson.org****

Only exception is modification of some configurations for title pages and other bits that shouldn't change much in time (see next section).

# Configuration

Each lesson.org needs some configuration :

- Configure org-reveal-title-slide in slides.org.
- Configure header (`\lhead{...}` and `\rhead{...}`) and footer (`\lfoot{...}` and `\rfoot{...}`) elements in the headers like `#+LaTeX_HEADER: \rhead{...}` in handbook.org and teacher-handbook.org.

# Generating final documents

We're using the standard exporters so each output format will be exported from its corresponding umbrella .org source

:

## **slides**

open slides.org, then C-c C-e R ... for org-reveal  
export (to slides.html), provided that you have loaded  
org-reveal in Emacs

## **handbook**

open handbook.org, then C-c C-e l ... for LaTeX  
export (to handbook.pdf)

## **teacher handbook**

open teacher-handbook.org, then C-c C-e l ... for  
LaTeX export (to teacher-handbook.pdf)

## Exporting slides to HTML with org-reveal

Depending on how you installed org-reveal (**Git submodules** or otherwise), org-reveal may already be available.

If not yet, load it with M-x load-file from the location of its Git submodule (elisp/org-reveal/ox-reveal.el by default).

## Printing slides

I've tested **DeckTape** using a Docker container containing PhantomJS and decktape to convert the slides to a single PDF document.

See the provided **decktape.sh** script that runs the container, bind-mounting the working dir into the container, so that input and output files can be found.

Note that I used a rebuilt Docker image, reusing the **DeckTape Dockerfile**, rebuilding with something alongside:

```
docker build -t obergixlocal/decktape .
```

# Known Issues

## Firefox issues ?

We have experienced issues with presentations made on some versions of Firefox, which are known by reveal.js maintainer... maybe best viewed in chrome.

You may prefer to have a PDF variant of the slides (see [Printing slides](#)) in case.



# How it works / Installation

# Use the source Luke

See the contents of the files... but be wary that it's sometimes messy and incrementally obtained.

Emacs is your buddy.

Git clone from <https://gitlab.com/olberger/org-teaching.git> (see the [Gitlab project](#))

# Git submodules

The repository contains Git submodules for :

- reveal.js
- org-reveal

So :

```
git submodule init  
git submodule update
```

You may prefer to install them another way (ELPA repo, CDN, etc.)

# Slides appearance

# Reveal.js settings

See the org-reveal settings set in the sources and the docs for a detailed explanation.

I'm using the following for a "frugal" look close to what powerpoint or beamer (?) could look like :

```
# +REVEAL_HLEVEL: 2
# +REVEAL_THEME: simple
# +REVEAL_TRANS: fade
# +REVEAL_SPEED: fast
# +REVEAL_MARGIN: 0.0
# +REVEAL_EXTRA_CSS: ./presentation.css
# +REVEAL_ROOT: ./reveal.js

# +OPTIONS: reveal_center:nil
```

## Section separators

The highest level sections include the following properties below the heading line, to customize the look of the slide.

```
:PROPERTIES:  
:REVEAL_EXTRA_ATTR: class="center"  
:reveal_background: #dbdbed  
:END:
```

This is intended to provide some visual sense of the transitions between sections. Please adapt and report.

Title screen picture (logos, etc.)

I'm not yet sure how much may be achieved with HTML and CSS for the title page of the slides deck, so I've relied on the embedding of a background image that will contain the logos and additional graphics.

```
# +REVEAL_TITLE_SLIDE_BACKGROUND: ./media/title-slide-background.png
```

I'm quite sure this could be improved.

# Annex



# Thanks

- All contributors to org-mode (special kudos to Carsten Dominik and Bastien Guerry)
- Yujie Wen for org-reveal
- Hakim El Hattab for reveal.js
- My colleagues at Telecom SudParis who had to teach with this tool without much rehearsal
- Our students who endured it for a brand new course (and included bugs)

# Feedback

I may be contacted from [my Web page](#) or via [the Gitlab project](#).