

# Presentation with/of the `latex-maven-plugin`

E. Reissner

November 8, 2023

## Outline

## Contents

<b>1</b>	<b>Introduction and Purpose</b>	<b>1</b>
<b>2</b>	<b>Features</b>	<b>2</b>
2.1	Realized Features . . . . .	2
2.2	Planned Features . . . . .	2
<b>3</b>	<b>Goals</b>	<b>3</b>

## 1 Introduction and Purpose

This is some additional text

### Purpose of these documents

The purpose of this document is twofold:

- Give an overview over the plugin. *But:*
  - The single official description of the plugin is the manual [?].
  - The [project site](#) gives already an overview.
  - The content of this presentation is updated only by need.
- Demonstrate that the plugin can compile a presentation using the `beamer` class and a handout written as `beamerarticle`.
  - In fact, when citing this presentation as [?], this refers to both.
  - The presentation is about the plugin
  - The handout adds information on how to write a beamer presentation.
  - Both documents turn `beamer` class and package `beamerarticle` into preferred usage ensuring tests.

## 2 Features

### 2.1 Realized Features

#### Ant and Maven

Automatically creates documents from LaTeX sources during the Maven `site` phase and in an ant run.

This comprises running basic tools like `lualatex` and `bibtex` and rerunning them by need.

#### Supported IO

Supports

- many output formats like PDF, DVI, HTML, DOCX, RTF, TXT and others
- many graphical input formats like PNG, MP, FIG, gnuplot; also provides a separate goal creating them, `grp`
- bibliography, index, glossary and embedded code; in particular split index

#### Checks

Check

- sources with `chktex` and logs the results in target and goal `chk`
- versions of used tools via goal `vrs`
- log files detecting errors and warnings
- whether a document could have been reproduced, by demand

#### Orchestration and document development

Orchestration of various tools detecting need for execution e.g. of `bibtex` including `rerunfilecheck` for `lualatex` and friends.

Supports document development, mainly by cooperating with editor, viewer and with other tools in the build chain.

- Offers installation script for extensions of VS Code.
- Offers configuration file `.chktexrc` for `chktex`.
- Can create configuration file `.latexmkrc` for `latexmk` synchronized with the configuration.
- Offers a common header file `header.tex` to unify packages loaded by latex main files.

### 2.2 Planned Features

#### Planned Features

- Support `biber` replacing `bibtex` as preferred tool
- Support `xindy` replacing `makeindex` as preferred tool
- Support `bib2gls` replacing `makeglossary` as preferred tool

- Execute `glosstex` if needed
- Usage of the `multibib` macros
- ...

## 3 Goals

### Goals

An overview is on the [goals page](#).

Besides goals referring to creating output of a given type like *pdf*, *odt* and *html*, there is a goal *cfg*, which allows configuring various output formats as **targets**.

One further goal, *chk* to perform a check with `chktex` is allowed among the **targets**.

The other goals cannot serve as targets:

**vrs** to check versions of tools,

**grp** to create graphic files speeding up use of `latexmk` and

**inj** to inject useful files like headers or config files like `.latexmkrc`.