

天文部誌
SuperNova
サンプル特別号

InkscapeとLaTeXで
部誌を作る

電気通信大学天文部
20 森山陽介

ごあいさつ

A horizontal row of 40 small circles, divided into two groups by a vertical line. The top group contains 20 circles, and the bottom group contains 20 circles.

A horizontal row of 40 small circles, with the last 10 circles in the second row.

A horizontal sequence of 30 small circles arranged in two rows of 15. The top row has 15 circles, and the bottom row has 15 circles directly beneath the top row.

夜は星を見ナイト！

天文部長〇年〇〇〇〇 UEC 天文部

目次

ごあいさつ	i
第1章 部誌を作る	1
1.1 L ^A T _E X 環境構築とその準備	1
1.2 Inkscape で表紙を作る	3
付録 A 部誌テンプレートの機能紹介	5
A.1 ディレクトリ構造	5
A.2 部誌の構造	5
A.3 部誌の書式設定	7
A.4 さいごに	8
編集後記	9

第1章 部誌を作る

4年 森山 陽介

部誌を作るためには何が必要か。

それは PDF を結合、分解、編集できることです。

そのためなら、究極的には何でもよくて、Adobe Acrobat や Adobe InDesign なんかを使ってもいいでしょう。ただ、有料だし、一時的に担当する部誌のために自腹を切るのも、部費を使うのも割にあわないでしょう。どうせなら、今後の学生生活にも役立つようなスキルを身に着けられるような作業にもしたいものです。

そこで、部誌の作成に当たっては \LaTeX を使用することにしました。特に、この部誌で使用する \Lualatex は、フォントを変更するのが容易ですし、(u)p \LaTeX で必要になるような dvipdfmx やそのオプションを必要とせずに直接 PDF 化でき、今後主流となるようなモダンなエンジンと言われています。¹⁾ 記事も \LaTeX で書けるので、そうすれば PDF 化も 1 回で済むし、体裁は勝手にやってくれるので作業量を減らすことができます。²⁾

一方で、表紙や裏表紙など、グラフィックなデザインは \LaTeX くんの得意とするところではないので、人間が作る必要があります。こういったときに使用するのは、ベクターグラフィックス³⁾を扱えるソフトです。一般的に有名なのは Adobe Illustrator でしょうが、これまた有料だし、Illustrator で作ったファイルは PDF 化しない限り Illustrator でしか開けないので、引き継ぎやすさに難があります。ここでは、フリーで Windows, Mac, Linux 全てに対応している Inkscape というソフトを用いて、グラフィカルなページのデザインを行います。

1.1 \LaTeX 環境構築とその準備

\LaTeX 環境を構築しましょう。おそらくほとんどの電通大生なら自分のマシンにインストールなりされてると思いますが、それぞれマシンも違えば、使用してる \TeX エンジンも異なるでしょう。違う環境で同じファイルを扱うとどこかで齟齬が生じるので、Docker を使って、仮想環境を用意して、そこで部誌を編集することにしてみました。基本的には <https://zenn.dev/being/articles/how-to-use-my-latex> の内容を基に環境構築したので、この通りにすればできるはず。一応、必要なことをサンプルがてら書いておきます。

1.1.1 必要なもの

- VScode
- Docker Desktop
- Git
- GitHub アカウント
- (WSL)

1.1.2 Docker Desktop のインストール

<https://www.docker.com/products/docker-desktop/> にアクセスして、Docker Desktop をインストールして、アカウント作成（飛ばしてもよい）など設定して起動させておきます。

1.1.3 git の準備

git をインストールするなりして、github と連携できるようにします。⁴⁾ git のインストールは Windows だと面倒なので、Windows の場合は WSL(Windows Subsystem for Linux) を利用します。Windows 上に

¹⁾ ここは単に思想というか、こだわりに近い

²⁾ \LaTeX 自体の体裁がなってない原稿を手直しする手間やエラー対応はあるけどね。

³⁾ コンリテで習いましたよね？

⁴⁾ 正直まだ良くわかってない部分が多くてちゃんと説明できる自信はない

1 部誌を作る

Linux の Ubuntu⁵⁾を構築して操作するものです。初めての人は、Windows の設定から「アプリ」→「オプション機能」と進んで、一番下までスクロールしたところにある「Windows のその他の機能」を選択します。そこに「Linux 用 Windows サブシステム」の項目があるので、チェックボックスにチェックが入っているか確認してください。(図 1.1) 入ってなかったら入れましょう。

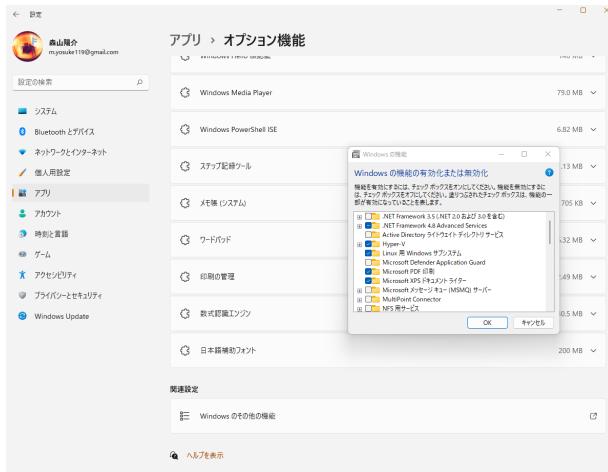


図 1.1 WSL をインストールする準備

次に、PowerShell を管理者として開いて

```
1 wsl --install
```

を入力して実行します。

すると、WSL がインストールされるので、WSL のターミナル画面を開いて、アカウント設定なり済ませて、

```
1 sudo apt update  
2 sudo apt upgrade
```

など済ませれば、git は WSL 内では使えるようになってるはず。なってなかったら

```
1 sudo apt install git
```

とかすれば大丈夫だと思います。多分。

そしたら、GitHub(<https://github.com/>) のアカウントを用意して、その情報を git に設定しておきます。これを「git の初期設定」と呼ばれているのをよく目にして最初は難のことやらさっぱりでした。

```
1 git config --global user.name "GitHubアカウント名"  
2 git config --global user.email GitHubに登録したメアド  
3 git config --global code.editor 'code --wait' ← VScodeの場合
```

1.1.4 VScode の準備

VScode の DevContainer という機能を使って行うのが便利なので、それを推奨します。VScode をインストールして、「Remote Development」という 4 つの拡張機能のパックと Docker の拡張機能をインストール。あと GitHub のアカウントとの連携とかしておきましょう。

⁵⁾好みの Linux がある人はこの限りでない

1.1.5 \LaTeX 環境構築

WSL のターミナル上で、

```
1 docker pull gher.io/being24/latex-docker:latest
```

を実行したら、天文部 GitHub にある (はず) の部誌用のファイル群をルートディレクトリごと clone(=全部ダウンロード) てしまえば 8 割方完了です。

あとはそのディレクトリを (仮に WORKDIR として)

```
1 code WORKDIR
```

といったふうに VScode で開いて、画面右下に出てくる「コンテナーで再度開く」を OK すれば、自動で環境構築が始まります。あとは画面の指示にイエスマンで従っていけば、完成です。多分。

1.2 Inkscape で表紙を作る

Inkscape(<https://inkscape.org/ja/>) をダウンロードして、新規作成で A4 用紙を選択します。そして、表紙にしたい画像を選んで配置します。

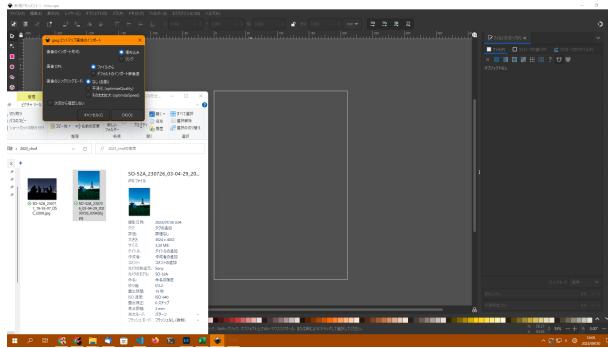


図 1.2 表紙にしたい画像をドラッグ・アンド・ドロップ

画像の縦横比をロックした上で、幅を 210 にするか、高さを 298 にするかして、A4 用紙にフィットさせます。そしたら、部誌タイトルの素材を同じくドラッグ・アンド・ドロップしていい感じの大きさと場所に配置します。⁶⁾このとき、「ドキュメントのプロパティ」からグリッドを作成しておくと、配置するときの目印になって便利です。

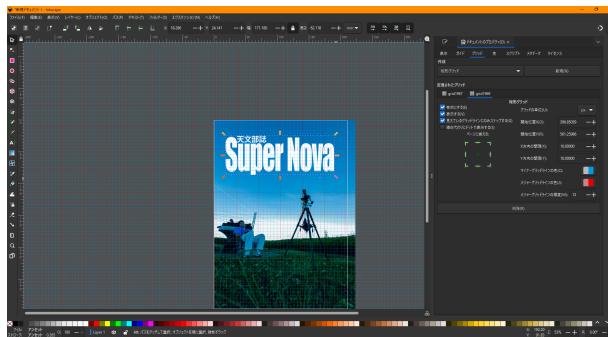


図 1.3 タイトルの配置

あとは、目次に沿って注目タイトルをテキスト入れとかクレジットを入れるとかして、表紙っぽくします。一通りできたら、「名前をつけて保存」で SVG ファイルとして保存します。

微調整など繰り返して、ようやく表紙として完成したら、上書き保存はもちろんとして、「コピーを保存」から PDF を選択して PDF として出力します。

⁶⁾ 大きさと座標の数値を決められると自動化できそう

1 部誌を作る

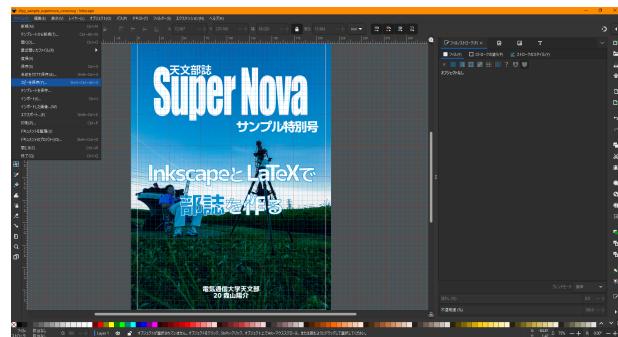


図 1.4 コピーを保存

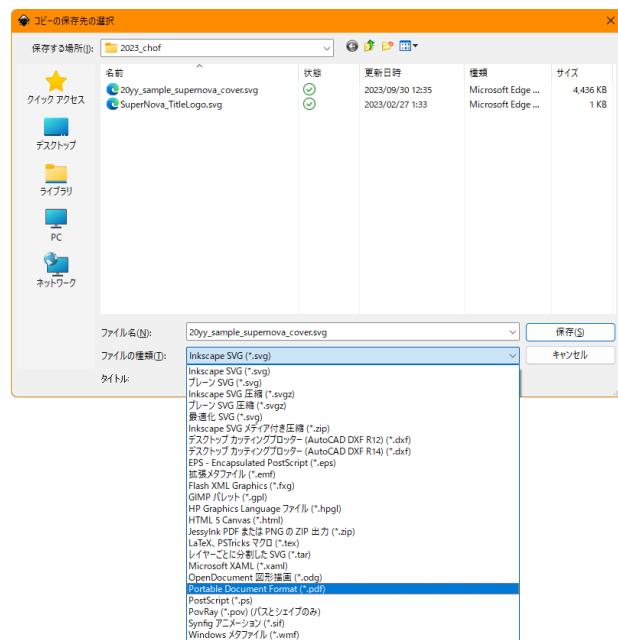


図 1.5 Portable Document Format(PDF)を選択

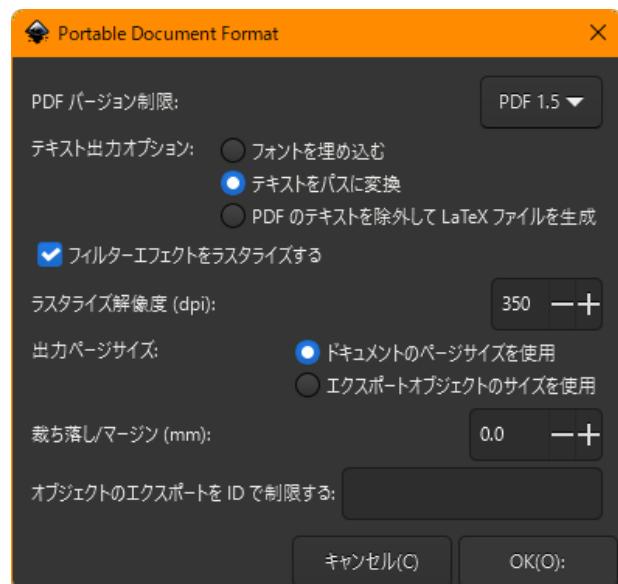


図 1.6 PDF の出力設定

裏表紙も、同じようにして作ってしまえば OK！簡単でしょ？

付録 A 部誌テンプレートの機能紹介

4年 森山 陽介

A.1 ディレクトリ構造

```
+ .devcontainer
|- devcontainer.json
|- setup.sh
+ .github
+ .vscode
|- extensions.json
|- latex.code-snippets
|- setting.json
+ bin
+ classes
:
|- supernova.cls
+ cover
+ figures
+ font
+ out
+ sections
.gitignore
.latexmkrc
:
preamble.tex
:
super_nova_20yy.tex
```

.devcontainer は \LaTeX 環境を構築する Devcontainer の設定ファイルが格納されている。

.vscode には、 \LaTeX を書く上で便利なスニペット（ショートカット）や設定が格納されている。

基本的に、super_nova_20yy.tex を main として、その中身の内容は sections ディレクトリに章ごと（人ごと）に tex ファイルを分割して読み込む形になっている。sections ディレクトリに、投稿された記事ごとのサブディレクトリを作って、その中に原稿ファイルと画像ファイルと一緒に保存する（画像ファイルはまた別に figure ディレクトリに保管するのもアリ）

A.2 部誌の構造

以下が、部誌の根幹部分になる super_nova_20yy.tex のソースコードである。

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 %
3 % 天文部誌「Super Nova」 使用エンジン：LuaLaTeX
4 %
5 % updated 09 Apr, 2021
6 %
7 % (c) Yosuke MORIYAMA
8 % 上記の行を残してつかうこと。2次配布可。ご利用は計画的に。
```

A 部誌テンプレートの機能紹介

```
9 %
10 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11
12 \documentclass{classes/supernova}% 天文部誌用のプリアンブル・マクロの読み込み
13 \%usepackage[backend=biber,style=ieee]{biblatex}
14 \%bibliography{references.bib}
15 \%include{preamble} % 投稿された記事の中に追加パッケージが必要な場合にまとめとく
16
17 %%%%%%%%% 表紙・ごあいさつ・目次 %%%%%%%%%%
18 \begin{document}
19 \gtfamily\sffamily % 本文をゴシック体サンセリフ体に統一する
20 \includepdf[noautoscale=true, fitpaper]{cover/20yy_sample_supernova_cover.pdf} %表紙
21 \frontmatter
22 \setcounter{page}{1} %ページカウンタのセット
23 \subfile{sections/goaisatu} % ご挨拶ページ
24 \setcounter{tocdepth}{1} % 目次に表示する見出しの深さ指定 0:chapter 1:section 2:subsection 3:
    subsubsection
25 \tableofcontents % 目次
26 \cleartooddpage% 奇数ページまでジャンプ
27
28 \mainmatter
29 %%%%%%%%%%%% ここから本編 %%%%%%%%%%%
30 % \subfile{hogehoge} % hogehoge.texを読み込む
31 \subfile{sections/contents}
32
33 %%%%%%%%%%%% 付録・編集後記 %%%%%%%%%%
34 \appendix
35 \subfile{sections/appendix} % 付録
36
37 \backmatter % あとがき、編集後記はこの下に
38 \subfile{sections/kouki} % 編集後記
39 \%printbibliography[title=参考文献]
40 \%addcontentsline{toc}{chapter}{参考文献}
41 % \includepdf[noautoscale=true, fitpaper]{cover/20yy_xxxx_atogaki.pdf}
42 \cleardoublepage
43 \cleartoevenpage % 奇数ページまでジャンプ
44 \includepdf[noautoscale=true, fitpaper]{cover/20yy_sample_supernova_cover_back.pdf} % 裏表紙
45 \end{document}
```

オリジナルのクラスファイル `classes/supernova.cls` を読み込んで、原稿は`\subfile{hogehoge}`の形式で読み込んでいる。この `hogehoge` の部分には、読み込みたい `tex` ファイルのパスを指定してあげれば良い。

A.2.1 原稿記事の書式

`\subfile{hogehoge}`で読み込む記事のテンプレートがこちら

```
1 %! TEX root = ../supernova_20yy.tex
2 \documentclass[../super_nova_20yy]{subfiles}
3
4 \begin{document}
5 \chapter{タイトル} % タイトル
6 \rightline{5年 電通 太郎} % 学年と名前（ハンドルネームでも可）
```

```

7
8
9 \end{document}

```

\documentclass[../super_nova_20yy]{subfiles}とすることで、親ファイルのクラス設定が引き継がれてサブファイル単体でもコンパイルすることができる。¹⁾

そして、部誌編集担当の主な仕事は投稿された原稿をこのサブファイルの書式に直していくことになる。といっても、プリアンブルに追加パッケージがないかだけ確認して（あつたら親ファイルのプリアンブルに追記）、サブファイルの書式に書き換えるだけ。

ひとつ、注意したいのが図の読み込みなどに関してで、サブファイルからの\includegraphicsなどでは、

```

1 \begin{figure}[H]
2   \centering
3   \includegraphics[width=.5\textwidth]{figures/ss265.png}
4   \caption{表紙にしたい画像をドラッグ・アンド・ドロップ}
5   \label{fig:ss265}
6 \end{figure}

```

と言ったように、サブファイルからみた相対パスではなく、親ファイル（ルート）からみた相対パスを書き込まなければならないことに注意

A.3 部誌の書式設定

詳しくは、supernova.cls を参照してみてほしいが、

```

1 \NewPageStyle{astrobook}{ % 生TeX原稿での柱とノンブルの設定
2   yoko,
3   odd_running_head=_section,
4   even_running_head=_chapter,
5   running_head_position=top-fore-edge,
6   nombre_position=bottom-fore-edge
7 }
8 \pagestyle{astrobook} % 上の設定を適用

```

では、ページ上部のページ説明欄（柱）とページ下部のページ数表記（ノンブル）の設定を行っている。tex ファイルで書かれた原稿には、これらの設定が反映されるが、PDF で入稿されたページを普通に\includepdf などで読み込むとこの設定は反映されず、柱は良いとしてもページ数が飛んでしまうなどでして統一感を損ねる。そこで、PDF 原稿の読み込みには

```

1 \newcommand{\article}[4][-]{ % PDF原稿の読み込みマクロの設定
2   \markboth{\#2}{\#3}
3   \phantomsection % hyperrefおよびPDFのブックマークに表示させる
4   \addcontentsline{toc}{chapter}{\#2} % 各々のタイトルなどは目次ではchapter扱い
5   \includepdf[pages=\#1, noautoscale=true, fitpaper, pagecommand={}]{\#4}
6 }

```

で定義したコマンドによって、\article{題名}{著者}{ファイル名}のようにして読み込む。このとき、\article[1]{題名}{著者}{ファイル名}というようにすれば、読みみたい PDF 原稿の 1 ページ目だけが読み込まれることになる。[1-3] というようにすれば 1 ページ目から 3 ページ目まで読み込むことになる。詳しくは、\includepdf の pages オプションについて参照してほしい。

¹⁾何故かできるやつとできずにエラー吐かれるのがあるけど、よくわかってない

A.3.1 スニペットと図表参照の簡便化

クラスファイルの末尾に

```

1 % 図表番号を参照するときに図\ref{hoge}とかしなくて言いようとするマクロ
2 \newcommand{\figref}[1]{図\ref{fig:#1}}
3 \newcommand{\tabref}[1]{表\ref{tab:#1}}
4 \%newcommand{\eqref}[1]{式(\ref{eq:#1})}
5 \renewcommand{\eqref}[1]{式(\ref{eq:#1})}

```

を追加している。これによって、図表の参照が楽になる。

ただし、図表の\labelを宣言するときに\label{fig:hoge hoge}といった具合に、図なら[fig:], 表なら[tab:], 数式なら[eq:]を書いておく必要がある。²⁾そして、めんどくさいのが、図や表を書くときのコマンド群を羅列することだが、スニペットを設定してあるので、"figure"と打ってtabを押すなどして変換候補を確定すれば

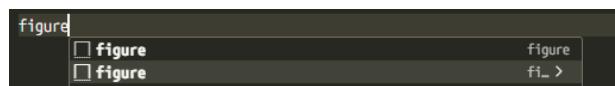


図 A.1 figure と入力して変換候補を確定

```

1 \begin{figure}
2   \centering
3   \includegraphics[width=.5\textwidth]{}
4   \caption{}
5   \label{fig:}
6 \end{figure}

```

が出てくる。同じように"table"と打てば

```

1 \begin{table}
2   \centering
3   \caption{}
4   \label{tab:}
5   \begin{tabular}{}
6     \hline
7   \end{tabular}
8 \end{table}

```

が出てくるし、"doublefigure" や"doubletable"と打てば横に並んだ 2 つの図や表のテンプレートも出てくる。呼び出す名前は.vscode/latex.code-snippets のファイル内の"prefix"に該当する部分の名前を入力すれば良い。

A.4 さいごに

ここまでが大体の流れになる。これを基に使いやすいうように作り替えてくれて構わないし、これを基にしてくれなくとも構わない。他にもいろんな機能をつけたり、統一感を出すために泥臭い工夫した記憶もあるけれど、書ききれないで、過去のファイルをドライブから漁って参考にしてみてほしい。良い部誌ができることを期待している。

²⁾え、めんどいって？いや、これがあるおかげで助かるときもあるんだよ

編集後記

部誌編集長〇年〇〇〇〇



調布で天体観測



部員募集中とかなんとか

天文部に対する熱い想いの丈をここにぶつけるとか
いい感じの写真を貼つけておくだけでもOK
ま、裏表紙にそこまでこだわる必要も無し
ただ、いい感じにカッコつけられると満足感あるよね
表紙・裏表紙は自分の好きな雑誌とか見て
"インスピレーション"受けて作ってみただけ
これを作り替えてもヨシ、イチから作ってもヨシ



電気通信大学 天文部

部誌 20YY-DD



@uec_astro



www.astro.club.uec.ac.jp/

Printed at Pict-Lab

定価 お気持ち (本体 活動内容+青春10%)
時 次号発刊まで