

# これからの SAT<sub>Y</sub>SF<sub>I</sub> に望むこと

@Nmatician

2021 年 6 月 26 日

はじめに

# 自己紹介

- Twitter: @Nmatician
- Github: enunun
- 材料系修士卒（非情報系かつ非プログラマ）
- ソフトウェア開発に関しては素人
  - Github のページにはろくなものはない
- （今のところ）SAT<sub>Y</sub>SF<sub>I</sub> のエンドユーザー
  - 本格的に触りだしておよそ 1 ヶ月
- L<sup>A</sup>T<sub>E</sub>X と SAT<sub>Y</sub>SF<sub>I</sub> を反復横跳び
  - L<sup>A</sup>T<sub>E</sub>X とはそれなりに長い付き合い

# L<sup>A</sup>T<sub>E</sub>X と SAT<sub>Y</sub>SF<sub>I</sub>

- SAT<sub>Y</sub>SF<sub>I</sub> は T<sub>E</sub>X/L<sup>A</sup>T<sub>E</sub>X と比較して優位な点も多い
  - 事前の型検査によるエラー報告の精密さ
  - ライセンスがめんどくさくない
- 特にパッケージ開発のしやすさはトッテモスバラシイ
  - 名前空間の分離
  - 「第 0 引数」による周辺の文脈の利用
  - 便利なローカル変数
- 「巨人」たる L<sup>A</sup>T<sub>E</sub>X を参考にした部分が多い
  - これは変えたほうがいいのでは？と思う部分もそれなりに

開発側に望むこと

# 文書構造の記述方法

- 文書構造は見出しの名前で記述
  - +chapter, +section, +p 等
- 文書構造だけではなく「そのレベルの呼び名」も含む
- 各レベルの「呼び名」は文書構造の記述には不要では？
  - 従属関係のみが本質的なはず
- Markdown では「#」の数で表現
  - じゃあ Markdown 使えば？ → 表現能力に限界
- あと +p するのがめんどくさい（本音）
- パッケージ製作者にも多大な影響
  - v0.0.x の今のうちに

# 相互参照における名前空間の分離

- 相互参照はキーと番号の対応を読み取ってなされる
  - .satysfi-aux ファイルにキーと番号の対応が記録
- 「図」や「定理」等の型は記録されず
  - 自動補完させたいときに非常に面倒
- L<sup>A</sup>T<sub>E</sub>X では cleveref パッケージが有名
  - 読み込み時は \label コマンドにオプション引数が追加
  - 識別子の名前空間の分離が可能
- 要するに cleveref パッケージ相当の機能が欲しい
  - プリミティブでインターフェース提供？

コミュニティに望むこと



# 開発ノウハウの共有

ドキュメントクラスの役割は周知されているとはいいたい

- L<sup>A</sup>T<sub>E</sub>X における悲劇その 1: titlesec パッケージ
  - 見出しの体裁を変える **パッケージ**
  - 見出しの体裁はドキュメントクラス担当のハズ . . .
- L<sup>A</sup>T<sub>E</sub>X における悲劇その 2: authblk パッケージ
  - 複数著者や所属の記述に対応する **パッケージ**
  - hyperref パッケージのオプションと衝突
- ドキュメントクラスの該当部分を \renewcommand で解決 (?)
- 特に節見出しは需要に比べてあまり知られていないらしい (?)
- ドキュメントクラスに関する知識があれば **自力** で解決可能
  - 「知の高速道路」の重要性

# ソースの軽率な公開

- SAT<sub>Y</sub>SF<sub>I</sub> はまだパッケージが少ない
  - 欲しい機能は自分で実装する必要
  - しかしどうやればいいのかわからない . . .
  - ソースがなければ解決した人がいてもパクれない
- Github にあるのはパッケージとそのドキュメントが中心
  - もっと実践的な文書作成例が必要
- お前ら PDF だけ挙げるなソースも挙げろ L<sup>A</sup>T<sub>E</sub>X でも同様だぞ
  - 例：商集合の解説 (<https://github.com/enunun/quoset>)
  - このスライドも (<https://github.com/enunun/satyconf2021>)

# ソースの公開先

- Github がおすすめ
  - 個人で文書を書くだけなので add, commit, push だけで OK
- SATySFj は現在 linguist のサポート外
  - ユニークなリポジトリ数が不足
  - 怪文書を作ってリポジトリを作るだけでコミュニティに貢献！
  - このスライドも貢献にカウント（たぶん）
- 公開するときはライセンスをきちんと設定しよう
  - MIT ライセンスがおすすめ
  - コードをコピペしたときはコピペ元のライセンスに注意

Let's SAT<sub>Y</sub>SF<sub>I</sub>!!