

Maintaining LibreOffice Math



Takeshi Abe

2016-12-10

LibreOffice Kaigi 2016.12

tdf# 84650 Vertical dots misplaced when used before \sum \Rightarrow patch

自己紹介

- LibreOffice開発者; committer (since 2011)
- LibreOffice日本語チームのメンバー
- A member of The Document Foundation



FUN PROJECT

LIBREOFFICE IS ONE OF THE FRIENDLIEST AND FASTEST-GROWING PROJECTS IN THE FREE AND OPEN SOURCE SOFTWARE WORLD.

[DOWNLOAD NOW](#)



LibreOffice Mathとは

- LibreOfficeで作成する文書に整形された数式を書くためのアプリケーション
 - 文書中に数式を埋めこめる
 - 単独でも使える
- 要するにWordのEquation Editorみたいなもの

どんな感じに書けるの？

$$d\vec{X} = \vec{a}(t, \vec{X}) dt + \vec{b}(t, \vec{X}) dW$$

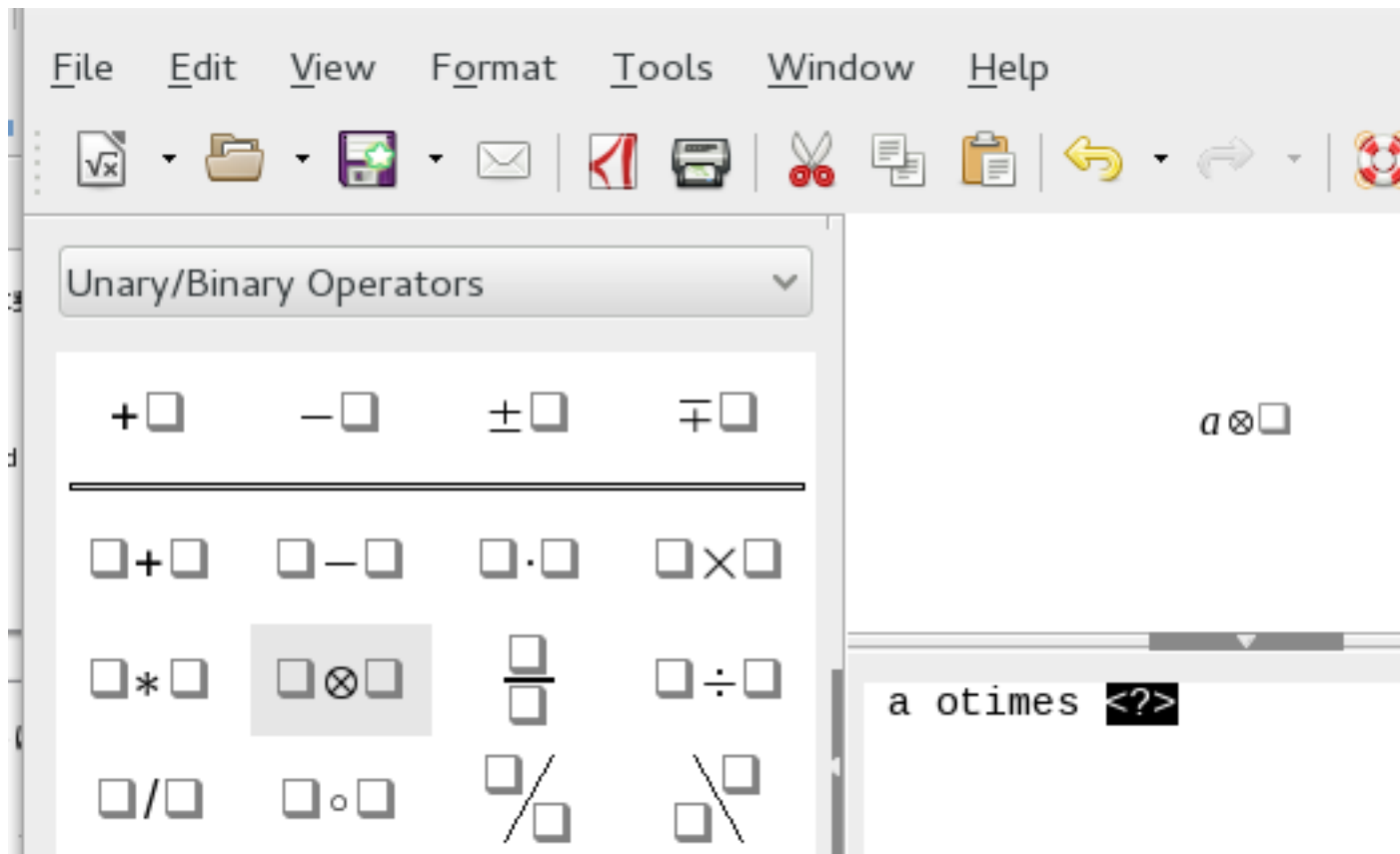
$$f(\mathbf{x}) = \alpha \sum_{i=0}^{\infty} \frac{f^{(i)}(\mathbf{0})}{i!} \underline{\mathbf{x}}^i$$

入力はStarMath記法

- 簡単

$e + z = \text{fact}\{\text{"easy"}\}$ \longrightarrow $e + z = \text{easy}!$

- メニューやダイアログから選んで入力もできる



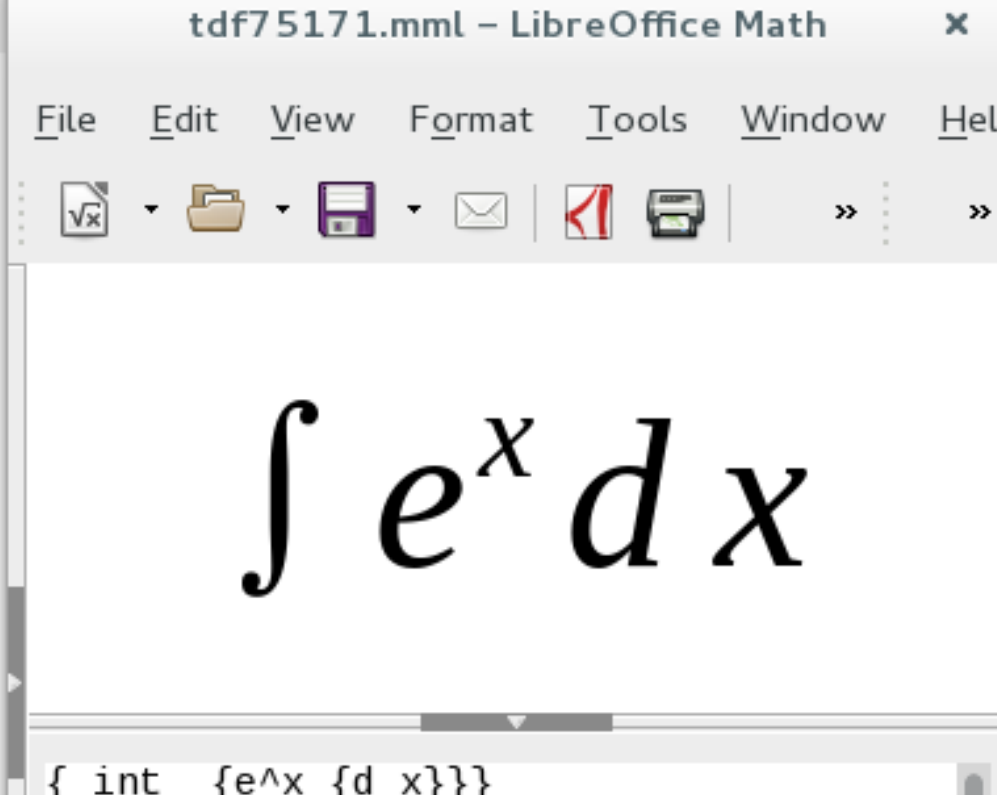
ファイル形式は国際標準に準拠

- MathML in OpenDocument Format (ODF)
 - 数式はpresentation markupの形に
 - StarMath記法での記述は<annotation>として保存

```
<?xml version="1.0" encoding="UTF-8"?>  
<math xmlns="http://www.w3.org/1998/Math/MathML">  
  <semantics>  
    <mrow>  
      <mrow>  
        <mo stretchy="false">∫</mo>  
        <mrow>  
          <msup>  
            <mi mathvariant="normal">e</mi>  
            <mi>x</mi>  
          </msup>  
          <mrow>  
            <mstyle mathvariant="normal">  
              <mrow>  
                <mi mathvariant="normal">d</mi>  
              </mrow>  
            </mstyle>  
            <mi>x</mi>  
          </mrow>  
        </mrow>  
      </mrow>  
    </semantics>  
  </math>
```

tdf75171.mml - LibreOffice Math

File Edit View Format Tools Window Help



The image shows the LibreOffice Math application window. The title bar reads "tdf75171.mml - LibreOffice Math". Below the title bar is a menu bar with "File", "Edit", "View", "Format", "Tools", "Window", and "Help". Underneath the menu bar is a toolbar with icons for opening a file, saving, printing, and other functions. The main area of the window displays the mathematical expression $\int e^x dx$ in a large, black, serif font. At the bottom of the window, there is a text input field containing the LaTeX code `{ int {e^x {d x}}}`.

「LaTeXでいいじゃん」

- LaTeXとは異なる用途
 - 汎用の組版目的ではない
 - 数式の美しさで勝負していない
- Mathの強み
 - インストールが容易
 - 文書自体を渡せば、相手も編集できる
 - 編集すると即表示、WYSIWYG
 - やる気がいらない
 - 「数式なんてどうせ読み飛ばすでしょ!」
→ もっと大事なところに集中できる

とはいえ、Mathの残念な点

- カーニングの概念がない
 - 個別に文字間隔を調整できない
- StarMath記法のUnicode対応が不十分
 - BMP外の文字に未対応(サロゲートペアが扱えない)
- StarMath記法において演算子は特定の数の引数をとる
 - 例: 二項演算子+は必ず2つの引数が必要
 - 数が合わなければエラー

しかし、逆に言うと

- 使える、けれどまだ未完成
 - 開発の余地あり
 - これからアイデアを盛り込める
- 今のStarMath記法がrestrictiveなのだから
 - 構文を後方互換に改善することが可能

Mathの沿革

- StarOffice 5.0(1998)から5.2の間で、ほぼ今の形に
 - それから約18年間、ほとんど変わっていない
- リグレッションもほとんどない
 - 今年に入って最大4つあったのを1つにした

主要な貢献者

- Thomas Lange
 - original developer (~ 2010)
- Caolán McNamara
 - Red Hatの誇るスーパーハッカー
 - 「Caolánブルバキ説」: 数人の開発者グループの名なのでは、と言われるほど
 - MathML/MathType import/export (2000 ~)
- Miklos Vajna
 - Writerとのインターフェイス
- Regina Henschel
 - QAとユーザーの視点
- Marcos Paulo de Souza, Frédéric Wang, Ivan Timofeev, Luke Dixon, Jonas Finnemann Jensen, ...
- me

ちなみに、LibOでは女性開発者も活躍

- Katarina Behrens
- Sophie Gautier
- Marina Latini
 - TDF議長
- Gülşah Köse

…他にももっといる

ある日のやりとり

tdf#97164: .docxをWriterで開いたらパーセント記号(%)が出てない!

Gülşah: 直ったみたい、パッチをレビューしてね。

Miklos: これMathのところだから、Takeshi見てくれる?

me: お、おう。。。

…無事masterにマージされ修正

そもそもなぜLibOをハックするのか

- LibreOfficeプロジェクト発足
 - 「このプロジェクトはきっと成功する」
 - そこら中に潜在的ユーザーがいる
- 楽しいし、学べる
 - C++
 - 理想論でないガチの移植可能性(ポータビリティ)
 - 去年の日本UNIXユーザー会の研究会での発表
 - リバースエンジニアリング
 - オープンなコラボレーションと議論

2011-2014

「とりあえずコードリーディングから始めるか」

- EasyHacksもやりながら

...

...

...

...

...

それだけで4年が経ちました。

ある日のやりとり2

me: (コードリーディング中)あれ、これnewしてる型とdeleteしている型が一致してないな。

me: (パッチ投げる)

Caolán: OK、ちなみに何のツールを使ってこの問題点見つけたの？

me: いや、ただ読んでたらたまたま見つかった。

Caolán: えっ

me: えっ

次のステップ: 目的を定める

- 昔からLaTeXユーザー
- dvipng経由でlibgdに関わる
 - ユーザーもプログラムを解する人であることが多い

…自分にとってMathをやるのが近道では

- 2015年からMathに集中するように

LibOのアプリケーションとして最小

```
$ find sc -type f -regex '.*\.(h|c)\(xx\)?' -print | wc -l # Calc
1765
$ find sd -type f -regex '.*\.(h|c)\(xx\)?' -print | wc -l # Draw/Impress
890
$ find sw -type f -regex '.*\.(h|c)\(xx\)?' -print | wc -l # Writer
1792
$ find starmath -type f -regex '.*\.(h|c)\(xx\)?' -print | wc -l # Math
86
$ find dbaccess -type f -regex '.*\.(h|c)\(xx\)?' -print | wc -l # Base
649
$
```

- 1桁違う
 - 規模が小さいぶん、理解し易い

小,,,,,j, ---一、 ` ---□、_ l,,,,;

{,,,,;> Tじi7 i fじjア !i,,,,; Mathは理解し易い...

ゞ,,,,ハ ノ ...!!リ;;r`

`Z;i <.,_.... ノ,,,,;> そんなふうに考えていた時期が

,,えハ、 `、_、-、_、'、 ,f` : Y,,f. 俺にもありました

~"戈、 `、_、-、_、'、 r' : ... `!

深み

- starmathモジュール外にある下部レイヤーが深刻に複雑
 - バグを直そうとすると結局そちらに手を入れることになりがち
- 特にvcl
 - Visual Components Library
 - ウィジェット、フォント、印刷、等

深み2: OpenSymbolフォント

- かつてStarSymbolと呼ばれていた
- LibreOfficeに同梱されている
- UnicodeのPUA(Private Use Area)にMathで使うような数学記号を割り当てている
 - 他のフォントに切り替えると変換が必要
 - エクスポートするにしても変換が必要

深み3: MathMLインポート

- 処理の順を追うと:
 - XMLフィルター
 - StarMath記法に変換
 - それを再読み込み
 - レイアウト対象のツリー構造が得られる

- …無駄がある上に、StarMath記法で表現できない要素が失われる

そして2016年になって

- ある日ふとbugzillaの"Formula Editor"コンポーネントのバグ一覧を眺めると、
 - 「あれ、これは直せるんじゃないか」
 - いつの間にかバグの関係しそうなコード部分が見当つくようになっていた

保守的なアプローチ

- リグレッションを避けるスタイルで進める
 - 単体テストの拡充
 - 約1,200 commitsのうち、直接のrevertは3回のみ
 - 1つはSolaris対応のコード周り
 - もう1つはassertionの間違い
 - 最後の1つはAPIの関数名に含まれるtypoを直してしまった
- もちろん革新的(「当たって砕けろ」)アプローチの開発者もいる
 - リグレッションを恐れない
 - 実際、何か壊したら誰かが気付くことが多い

LibreOffice 5.3でのMath

- LibreOffice 5.3で刷新されたレイアウトエンジン
 - HarfBuzzベース
 - brought by Khaled Hosny
 - クロスプラットフォームの強化
- Mathでは若干の課題
 - ルート記号の表示が崩れる不具合
- 一方、OOo由来のバグがいくつか直っている

結論

- LibreOffice Mathは文書に手軽に整形された数式を埋め込むのに重宝
 - WYSWYGでとっつきやすい
 - 標準文書形式(MathML)で保存でき、長期に渡る編集保管に向いている
- Mathの開発は地道に続けられている
 - バグ修正とともにコードベースの理解と改善が進んでいる
 - 新しいアイデアも実装待ち
- ぜひLibreOfficeをハックしよう

Thank you.

