Linguistics with LinguisTiX

निरंजन (Niranjan)

1 Introduction

We use LaTeX[1] extensively in many sub-disciplines of linguistics. There are dedicated packages and libraries that we need in most linguistic articles. I started using some of them at the time of my masters and was happy to see that things were getting done easily, but with more experience in LaTeX, I started craving for perfection and better features. Unfortunately, I realised that most of the packages that I used were either old or had a rigid design which made it difficult to customise or alter them according to my needs.

This is why I started thinking about a 'standard' package which will take care of most of the things needed in linguistics. Thus LinguisTiX was conceived as an idea. I started working on it in 2022, but then, I was stuck on one feature and hence couldn't release it. In 2025, I have finally managed to make a public release and get it going. This paper talks about the currently released packages from the LinguisTiX bundle.

2 About the logo of 'LinguisTiX'

It is a stylisation of the word LinguisTiX. The 'g' and the 'T' have a peculiar design in this font. The *'L'*, *'T'* and the *'X'* are stylised in capitals intentionally to match the ones seen in the LaTeX logo. I prefer to write it as LinguisTiX even with the normal font. The *'X'* is coloured differently as an ode to the omnipresent *'X'* in the free software world. The purple colour in the logo[2] is chosen as an homage to the gender activism. The New Computer Modern Uncial font is used for printing the logo.

3 The 'abc'-s of linguistics

Linguistics is a scientific discipline which deals with the phenomenon of human language. As languages are at the heart of the discourse of linguistics, most curricula start with a method of transcribing speech sounds, so that students can read the data from any language. The International Phonetic Alphabet (IPA) is one such very commonly used method. Although there are other such methods of transcription, IPA is the most robust one amongst them. Since the range of its characters is beyond ASCII, it needs an ad-hoc input method in plain LaTeX. The `tipa` package had filled this niche for many years. According to its README file [4], `tipa` was first released in 1996. It became a lingua franca of linguistics because of the shorthand method it provided for typing IPA. E.g., in order to print the symbol [ʈ],[3] one could write something like `\textipa{\:t}` with `tipa`.

Despite `tipa` being such a handy tool for input, it has some drawbacks. The major downside of it is that it is not Unicode. It was developed when Unicode was not that common. But, over time, things have changed. Unicode has become the de facto standard for data. The obvious advantages of Unicode (that do not apply to `tipa`) are that:

- it can be copy-pasted,
- it can be typed directly (as opposed to the indirect method of `tipa`),
- it can be searched,
- various Unicode fonts are available that provide several designs. Weights (e.g., light, black, medium, regular), series (e.g., bold, medium), shapes (e.g., italic, small caps) are available with them.

In order to avail ourselves of these advantages, we must use Unicode characters in the input. This difficulty was partly solved by the `tipauni` package [9]. It converts the `tipa` commands to Unicode characters internally so that the output is richer and the above-mentioned advantages are attained.

However, this is not the end. What about the source? If the source contains `\textipa{\:t}` and the PDF output contains [ʈ], it isn't as reproducible as it could have been. Even the source should (ideally) be as readable as the PDF. Of course, mathematics might be brought up into the discussion and it can be argued that even mathematicians input `\alpha` to get α.

So why not linguistic shorthands? My argument against it is that mathematical symbols are vastly more numerous than IPA. To contrast them numerically, let's look at the number of commands from the two packages that convert the TeX shorthands to Unicode in both the disciplines. The `tipauni` package provides 100 commands [9] and the `unicode-math` package provides 2448 commands [5]. Additionally, keyboards are designed to type languages, not maths. There are popular layouts like 'X-Sampa' [7] or websites like TypeIt [6] for typing IPA. IPA symbols can

[1] This is a slightly modified LaTeX logo, as suggested by Antonis Tsolomitis.

[2] Colours are only visible in the digital version of this article. Ignore if you are reading the print version.

[3] Speech sounds are of two types, i.e., phones and phonemes. Without going into the theoretical details, I will just mention that the former are denoted in brackets, e.g., [ʈ]; whereas the latter are enclosed in slashes, e.g., /ʈ/.

doi.org/10.47397/tb/46-2/tb143niranjan-linguistix

be easily mapped to the respective languages, e.g., Indic IPA [8] for Indian languages.

Considering the points mentioned above, it can be established that typing IPA in Unicode is the richest and most recommended solution for linguistics. The LinguisTiX bundle promotes this practice. One may load the bundle with:

```
\usepackage{linguistix}
```

The New Computer Modern font family, in its book weight, is then set as default for the text and the IPA. (This article is an example.)

4 Text fonts

My idea is to separate the IPA from the text exactly like maths is separated from it. The bundle has adopted a decentralised approach, allowing packages to be loaded separately. If you choose to load only the necessary packages, LinguisTiX-fonts and LinguisTiX-ipa are the ones that will produce a PDF with text and IPA in New Computer Modern Book fonts. Note that the fonts are loaded for all the families, i.e., serif, sans serif, monospaced (even for the IPA). E.g., try the following with (at least) a fully updated TeX Live 2025:

```
\documentclass{article}
\usepackage{linguistix}

\begin{document}
hello world {\lngxipa heləʊ wɜːld}

\sffamily
hello world {\lngxipa heləʊ wɜːld}

\ttfamily
hello world {\lngxipa heləʊ wɜːld}
\end{document}
```

Output:

hello world heləʊ wɜːld
hello world heləʊ wɜːld
hello world heləʊ wɜːld

Every font loaded by the package can be controlled using the keys dedicated for the same. Separate keys are provided for the text fonts and the IPA fonts.

Note that LinguisTiX changes the default lining numerals to old style ones. Lining numerals don't have ascenders and descenders that are there in the normal text. Hence, scholars like Bringhurst [1] recommend using lining numerals only when they are combined with uppercase-only text. Otherwise, old style numerals are recommended for better readability. I have made it the default with LinguisTiX-fonts. This change can be reversed by using the option `old style numbers=false` value. Note that these numerals are controlled using the `Numerals={OldStyle}` feature of OpenType fonts. Since it is enabled by default, the old style numerals might be seen despite changing the font family from New Computer Modern to another.

5 The auxiliaries of LinguisTiX

Not the ones you are thinking of, dear linguists ;-)

Although the packages inside LinguisTiX are decentralised, their keys are parsed using a single parser command, i.e., `\linguistix`. This command is provided by the LinguisTiX-base package. The logos of the bundle are printed with the LinguisTiX-Logos. Neither of these have any significant use for the general public. The former is loaded by all the packages of the bundle, in order to offer a singleton parser. The latter is a lightweight, optional package that is needed only if one wants to print the logo of LinguisTiX. Out of the ones we have discussed so far, only LinguisTiX-fonts and LinguisTiX-ipa are the public-facing packages.

6 The *normal* `\normalfont`

The last one from the bundle is LinguisTiX-nfss. This package was not a part of the original plan. However, it emerged out of a need to switch between what the package calls *'super font families'* without changing the active font attributes. Later, I felt that this can be of general use to people who need to work with the aforementioned *'super font families'*. Essentially, it means to create a group of fonts that may contain serif, sans serif and typewriter families. Thanks to Enrico Gregorio's crystal ball [2], which helped me reach the implementation of this.

LinguisTiX-nfss stores the font names of the super-families one sets. The default fonts (also consider the ones set with `\setxxxxfont`, where `xxxx` may stand for `main`, `sans` or `mono`) are also stored in some variables. Usually, after creating a super font family with Enrico's solution, issuing a `\normalfont` command to go back to the default font has no effect as the steps involve resetting `\xxdefault` (where `xx` may be `rm`, `sf` or `tt`). Thus, there is no way to go back to the default values as they are not stored anywhere else. The LinguisTiX-nfss package stores all the necessary settings, so that they can be retrieved at any given point. It provides a set of commands which changes the font to the default or the user-mentioned super font family. It provides conditionals to check the value of a particular attribute, e.g., `rm`, `sf` or `tt` family; `bf` or `md` series, etc.

In a discussion with Frank Mittelbach during the breaks of TUG annual conference 2025, some issues with the current implementation are seen. Frank suggested some alternative ways of achieving the same desired effect. I plan to work on them in future releases.

I have released a fresh package that fixes the incompatibility issue between the unicode-math and expex. Since I want to promote Unicode everywhere, I am using unicode-math by default, but this causes issues if expex is loaded too. The LinguisTiX-fixpex package handles that issue. Now the concerned packages can safely be compiled with the LinguisTiX bundle as well as other codes that involve the aforementioned combination.

7 Planned

I released this package on 18th May 2025, and had appealed to the community for supporting the development of this bundle. Hardly within a week or so, the TeX development fund was awarded to this project. The first planned addition to the bundle is the one proposed for the award, LinguisTiX-glossing.[4] It is supposed to be a modern re-implementation of the leipzig package. It will provide a simpler user-side syntax for interlinear glossing and more customisation.

7.1 *Exempli gratia*

There are many packages for typesetting linguistic examples. My personal favourite among all of the available ones is expex. It provides the richest set of features. Despite it being a very nice package, it has a very old design. It has a TeX-style syntax which is not standard in the LaTeX ethos. Consider the example that follows.

It produces a warning about substituted font shapes. The warning is completely unrelated to the user code. This is because of a bug in the leipzig package. The camel-cased macros, e.g., `\First`, `\Ipfv` are counter-intuitive as generally commands are in lowercase. As can be seen in the example, the syntax of expex is unusual as well. Sometimes, debugging and customising expex is a nightmare because of its nitty gritties (well, not so much when Enrico and Alan Munn are there to help you out [3]). I plan to solve these practical difficulties and standardise this in the LinguisTiX bundle. I plan to develop these things in a tagging-aware manner.

[4] Thanks to Karl for suggesting the name (p.c. 2025-05-25). It's not printed in the logo font deliberately to signify that it is unreleased.

```
\documentclass{article}
\usepackage{leipzig}
\usepackage{expex}
\usepackage{linguistix-ipa}
\usepackage[provide*=*]{babel}
\babelprovide[%
  onchar          = {ids fonts}%
]{marathi}
\babelfont[marathi]%
          {rm}%
          {Mukta-Light.ttf}

\begin{document}
\ex
\begingl[everygla = {\lngxipa}]
\glpreamble \rightcomment{%
              [मराठी, Marathi]%
            }
            मी गाते.//
\gla        mi ga-t̪-e//
\glb        \First{} sing-\Ipfv-\F//
\glft       I (\F) sing.//
\endgl
\xe
\end{document}
```

Output:

(1) मी गाते. [मराठी, Marathi]

m-i ga-t̪-e
1 sing-IPFV-F

I (F) sing.

7.2 अ, آ, ஐ, ڑ and more

Linguistics is a discipline for languages. Languages depend on scripts. Multilingual and multiscript documents remain difficult in LaTeX. The babel package and LuaLaTeX have improved the situation, but it's still not the easiest. After a detailed dialogue with the maintainer of babel, I learnt that we have different and mutually incompatible approaches to tackle this problem. Despite believing that babel is a robust and strong package for handling languages, because of some design-related disagreements, I plan to develop a wrapper package for multilingual documents. Underneath, it will still be babel, but with a more user-friendly syntax and some ready-to-use solutions for novice users.

8 Comments, discussions, feedback?

I am very much interested in getting feedback on these developments. I can be reached at the email ID given at the end or I have opened up a public discussion platform for LinguisTiX on Matrix which can be accessed at: `matrix.to/#/#linguistix:matrix.org`. Bug reports and patches are welcome at the repository of the package: `puszcza.gnu.org.ua/projects/linguistix`.

9 Acknowledgements

I would like to thank Meenakshi, Salil and Antonis for reviewing the draft of this paper and suggesting corrections; Karl for his persistent guidance and the TEX Users Group for financially supporting the development of LinguisTiX.

References

[1] R. Bringhurst. The Elements of Typographic Style (4th edition). Point Roberts, WA: Hartley & Marks, Publishers, 2013.

[2] E. Gregorio. How to set sans and mono fonts with `\newfontfamily`? Oct. 30, 2024. `tex.stackexchange.com/a/729805`

[3] A. Munn, E. Gregorio. ExPex fails with unicode-math. How to avoid the clash? Dec. 5, 2023. `tex.stackexchange.com/q/703094`

[4] F. Rei. The tipa package. Version 1.3. 2002. `ctan.org/pkg/tipa`

[5] W. Robertson et al. The unicode-math package. Version 0.8r. Aug. 13, 2023. `ctan.org/pkg/unicode-math`

[6] T. P. Szynalski. The TypeIt website. `ipa.typeit.org/full`

[7] J. Wells. Computer-coding the IPA: a proposed extension of SAMPA. Department of Phonetics and Linguistics, University College London. `www.phon.ucl.ac.uk/home/sampa/x-sampa.htm`

[8] निरंजन. The Indic IPA layout. Version 2.3. June 2020. `gitlab.freedesktop.org/xkeyboard-config/xkeyboard-config/-/blob/master/symbols/in`

[9] निरंजन. The tipauni package. Version 0.7a. Feb. 13, 2023. `ctan.org/pkg/tipauni`

◇ निरंजन (Niranjan)
Nagamandalam, Malayinkeezhu,
Thiruvananthapuram, Kerala,
India 695571
`hi dot niranjan at pm dot me`