

Instructions for *ACL Proceedings

Anonymous ACL submission

Abstract

This document is a supplement to the general instructions for *ACL authors. It contains instructions for using the `trac1` Typst template for ACL conferences. The document itself conforms to its own specifications, and is therefore an example of what your manuscript should look like. These instructions should be used both for papers submitted for review and for final versions of accepted papers.

1 Introduction

These instructions are for authors submitting papers to *ACL conferences using Typst using the `trac1` style. They are not self-contained. All authors must follow the general instructions for *ACL proceedings,¹ and this document contains additional instructions for the Typst style files.

Because this document is itself written with `trac1`, it is instructive to look at [its source code](#) along with this PDF.

2 Engines

`Trac1` requires Typst 0.12. The most recent compatibility update is for Typst 0.14.

3 Preamble

You can load `trac1` into your Typst file as follows:

```
#import "@preview/trac1:0.7.1": *\n\n#show: acl.with(\n    anonymous: false,\n    title: [(your title)],\n    authors: make-authors(\n        name: "Alexander Koller",\n        affiliation: [\n            Saarland University\\
```

```
        #email("koller@coli.uni-\n        saarland.de")\n        ],\n        ),\n    )
```

You can then write the rest of your document as usual. Use the `#abstract` command to typeset your abstract.

Use `anonymous: true` to generate an anonymous version of your paper that is suitable for submission to the conference.

You can split your Typst document over multiple source files and combine them with `#include`, but the show rule with the call to `acl` should only appear once, in the main Typst source file that imports `trac1`.

4 Fonts

You may need to install a number of free fonts to make `trac1` documents conform to the ACL style. See the [README](#) for details.

The serif, sans-serif, and monospace fonts that `trac1` uses to typeset the document can be accessed in the variables `trac1-serif`, `trac1-sans`, and `trac1-mono`. Use these in your own styling if you find it useful.

5 Title and Authors

The ACL style instructions expect the title and authors to be listed at the top of the document, in a “titlebox” that is at least 5 cm high. If you have a long title or many authors, you can make the titlebox higher by passing an argument `titlebox-height: <new height>` to the `acl` show rule. The “new height” must be a [Typst length](#), and must be at least 5 cm.

¹<http://acl-org.github.io/ACLPUB/formatting.html>

070 5.1 Author lists with make-authors

071 You have two main options to format the author
072 list. The simplest is to use the `make-authors` function,
073 which will allow you to place authors and
074 affiliations side by side and on top of each other, as
075 with the `\and`, `\And`, and `\AND` macros in the LaTeX
076 style. The easiest way to understand the options is
077 to look at the example title pages at the end of this
078 document, along with their source code.

079 5.2 Free-form author lists

080 Alternatively, you can simply pass content to the
081 `authors` parameter of the `acl` function. This will
082 allow you to typeset arbitrarily complex author
083 lists. See the “Free-form authors” example at the
084 very end of this document.

085 TracI supports you in connecting authors to
086 their affiliations by offering you the `affiliation`
087 and `affil-ref` functions. You can define a
088 marker for an affiliation by calling e.g.
089 `#affiliation("uds")`, where “uds” is a unique key
090 for an affiliation. This will typeset a superscripted
091 symbol for this affiliation. You can then reference
092 affiliations after an author name by calling
093 `#affil-ref("uds")`; this will be replaced by the
094 affiliation’s symbol.

095 You can choose your own symbol for an affiliation
096 through e.g. `#affiliation("uds", symbol: sym.dagger)`. You can also pass multiple affiliation
097 strings to `affil-ref`; they will be typeset
098 connected by commas.

100 5.3 Footnotes in titleboxes

101 You can add a footnote to a titlebox using the
102 `title-footnote` function. This will generate a
103 footnote at the bottom of the page. You cannot
104 currently use the regular Typst footnote function
105 for this, because of a [known Typst limitation](#).

106 The function `title-footnote` takes two arguments.
107 The first is the footnote text, and the second
108 is a string that uniquely identifies the footnote (it
109 will be used as the name of a Typst label). You
110 can reference the same footnote from multiple
111 places in the titlebox by calling `title-footnote`
112 with the same identifier string. In this case, only
113 the first call to `title-footnote` will actually gen-
114 erate a footnote; subsequent calls will simply add
115 references to the same footnote. See the example
116 “Multiple rows of authors” at the end of this doc-
117 ument.

First column	Second column
some stuff	more stuff
second row	more second row

Table 1: An example table. Typst can simply use Unicode characters, so Table 1 from the LaTeX instructions is not needed any more.

6 Document Body

6.1 Footnotes

Footnotes are inserted with the `#footnote` command.²

6.2 Tables and figures

See [Table 1](#) for an example of a table and its caption. **Do not override the default caption sizes.**

As much as possible, fonts in figures should conform to the document fonts. See [Figure 1](#) for an example of a figure and its caption.

You can use the standard Typst `image` function to include images into your document. Typst supports PNG, JPEG, and SVG. Use SVG if you want to include a vector graphic; you can use e.g. `pdf2svg` to convert PDF files. Be aware that Typst has pretty good built-in support for generating plots (e.g. through [CeTZ-Plot](#)), so you may be able to simply generate and style your graphics within your Typst source code.

A floating element will be automatically labeled as a “Table” if the top-level element is a Typst table; otherwise Typst will call it a “Figure”. If you want a table labeled as a “Figure”, you can pass the argument `kind: image` to the `figure` call (see the [Typst documentation](#)).

By default, Typst places a `figure` within a single column. If you want a `figure` to stretch across both columns, you can pass the argument `scope: "parent"`. See the source code of [Table 2](#) for an example.

6.3 Equations

An example equation is shown below:

$$A = \pi r^2 \tag{1}$$

Labels for equation numbers, sections, subsections, figures and tables are all defined as [Typst labels](#), and cross references to them are made with `ref`.

This is an example cross-reference to [Equation 1](#).

²This is a footnote.

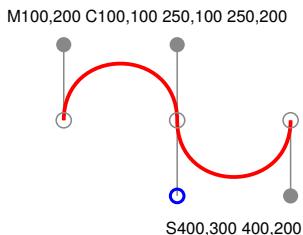


Figure 1: A figure with a caption that runs for more than one line. The example picture comes from the [openscad svg-tests](#) repository.

6.4 Lists

Typst distinguishes between lists and enums with tight and non-tight spacing. Lists and enums with tight spacing are set with no extra space between the items:

1. This is the first item of the list.
2. Here's a second item.

Lists and enums with non-tight spacing are set with a blank line of space in between, as in the `itemize` and `enumerate` environments of the LaTeX style:

- First element
- Second element

Here's some text to illustrate the distance of the list from the subsequent paragraph: Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do.

6.5 Appendices

Supply the content of your appendix after the `#show: appendix` command to switch the section numbering over to letters. See [Appendix A](#) for an example.

7 References

7.1 BibTeX Files

You can use regular BibTeX bibliography files with Typst. You can obtain the complete ACL Anthology as a BibTeX file from <https://aclweb.org/anthology/anthology.bib.gz>.

Please ensure that BibTeX records contain DOIs or URLs when possible, and for all the ACL materials that you reference. Use the `doi` field for DOIs and the `url` field for URLs. If a BibTeX entry has a URL or DOI field, the paper title in the references section will appear as a hyperlink to the paper.

7.2 Bibliographies

Tracl uses [Pergamon](#) to typeset the bibliography, with ACL-specific customization. The structure of a typical tracl document therefore looks like this:

```
#import "@preview/tracl:0.7.1": *
#import "@preview/permagon:0.6.0": *
... your document ...
#add-bib-resource(read("custom.bib"))
#print-acl-bibliography()
```

You can call `add-bib-resource` as many times as you like to make multiple Bibtex files available to your paper. Note that you have to `read` the Bibtex file yourself before calling `add-bib-resource` because of architectural limitations of Typst.

The bibliography will be printed at the location where you call `print-acl-bibliography`. This is typically after the Limitations sections, but before the appendices.

You can tweak the formatting of the bibliography by passing named arguments to the `print-acl-bibliography` function. These arguments will be passed on to Pergamon's `format-reference` function. See the [Pergamon documentation](#) for details.

7.3 Citations

[Table 2](#) shows how to cite papers in your text. Note that we use Pergamon's `cite` function, rather than Typst's builtin `cite`. This means that you must write `#cite("paperkey")` rather than `#cite(<paperkey>)`, and you cannot just write `@paperkey`.

The functions `cite` and `citep` will generate citations in the form “(author, year)”. You can write `#citet("Gusfield:97")` to get citations of the form “author (year)”, as in [Gusfield \(1997\)](#). You can use the command `#citen("Gusfield:97")` (“cite none”) to get “author, year” citations, which is useful for using citations within parentheses. A possessive citation can be made with the `#citeg` command; this will yield e.g. “[Gusfield's \(1997\)](#)”.

For comparison with the ACL LaTeX style, `citen` corresponds to their `citealp`, and `citeg` corresponds to their `citeposs`.

The citation commands are defined by Pergamon. If you split your paper across multiple source files, you must therefore `#import` Pergamon in each of them.

Output	Citation command	LaTeX equivalent
(Gusfield, 1997)	#cite("Gusfield:97") or #citep("Gusfield:97")	citep
Gusfield (1997)	#citet("Gusfield:97")	citet
Gusfield, 1997	#citen("Gusfield:97")	citealp
Gusfield's (1997)	#citeg("Gusfield:97")	citeposs

Table 2: Citation commands supported by the style file.

240 Limitations

241 Since December 2023, a “Limitations” section
 242 has been required for all papers submitted to
 243 ACL Rolling Review (ARR). This section should
 244 be placed at the end of the paper, before the
 245 references. The “Limitations” section (along with,
 246 optionally, a section for ethical considerations)
 247 may be up to one page and will not count toward
 248 the final page limit. Note that these files may be
 249 used by venues that do not rely on ARR so it
 250 is recommended to verify the requirement of a
 251 “Limitations” section and other criteria with the
 252 venue in question.

253 Tracl currently has a number of limitations
 254 compared to the more mature LaTeX ACL style.
 255 Here are some workarounds.

- 256 • The two columns of a page will not automatically
 257 be aligned at the bottom. This is a [known](#)
 258 [limitation in Typst](#) that should be fixed at
 259 some point. For the time being, you can man-
 260 ually insert whitespace above each paragraph
 261 in the shorter column with #v.

262 References

263 Rie Kubota Ando and Tong Zhang. 2005. A Framework
 264 for Learning Predictive Structures from Multiple
 265 Tasks and Unlabeled Data. *Journal of Machine
 266 Learning Research* 6:1817–1853.

267 Galen Andrew and Jianfeng Gao. 2007. Scalable
 268 training of L1-regularized log-linear models. In *Pro-
 269 ceedings of the 24th International Conference on
 270 Machine Learning*, pages 33–40.

271 Dan Gusfield. 1997. *Algorithms on Strings, Trees and
 272 Sequences*. Cambridge, UK: Cambridge University
 273 Press.

274 Mohammad Sadegh Rasooli and Joel R. Tetreault.
 275 2015. [Yara Parser: A Fast and Accurate De-](#)
 276 [pendency Parser](#). *Computing Research Repository*
 277 arXiv:1503.06733.

278

A Example Appendix

279

This is an appendix.

Three authors from the same institution

Michael Sullivan Mareike Hartmann Alexander Koller
Department of Language Science and Technology
Saarland Informatics Campus
Saarland University, Saarbrücken, Germany
`{msullivan, mareikeh, koller}@coli.uni-saarland.de`

```
#acl(  
    title: [Three authors from the same institution],  
    authors: make-authors(  
        name: ("Michael Sullivan", "Mareike Hartmann", "Alexander Koller"),  
        affiliation: [Department of Language Science and Technology\  
            Saarland Informatics Campus\  
            Saarland University, Saarbrücken, Germany\  
            #email("{msullivan, mareikeh, koller}@coli.uni-saarland.de")  
        ]  
    )
```

280

Abstract

281
282
283
284
285
286
287

In the simplest case, all authors are from the same institution. Then you can use `make-authors` to typeset a single block containing all authors and this single institution. A block is defined by a dictionary with keys `name` and `affiliation`, where `name` can either be a single author name (string or content) or an array of author names.

288
289
290

Multiple author names within the same block are separated by an amount of whitespace that is determined by the `author-spacing` parameter.

Three authors from different institutions

Alex Duchnowski
Saarland University
alex@lst.uni-sb.de

Ellie Pavlick
Brown University
ellie_pavlick@brown.edu

Alexander Koller
Saarland University
koller@lst.uni-sb.de

```
#acl(
    title: [Three authors from different institutions],
    authors: make-authors(
        (
            name: "Alex Duchnowski",
            affiliation: [Saarland University\ #email("alex@lst.uni-sb.de")]
        ), (
            name: "Ellie Pavlick",
            affiliation: [Brown University\ #email("ellie_pavlick@brown.edu")]
        ), (
            name: "Alexander Koller",
            affiliation: [Saarland University\ #email("koller@lst.uni-sb.de")]
        ),
    )
)
```

Abstract

If you want to show a separate institution for each author, you can pass an array of multiple blocks to create a row. Blocks are separated by an amount of whitespace that is determined by the block-spacing parameter.

291

292
293
294
295
296

Mix of same and different institutions

Alex Duchnowski

Saarland University

{alex|koller}@lst.uni-sb.de

Alexander Koller

Ellie Pavlick

Brown University

ellie_pavlick@brown.edu

```
#acl(
    title: [Mix of same and different institutions],
    authors: make-authors(
        (
            name: ("Alex Duchnowski", "Alexander Koller"),
            affiliation: [Saarland University\ #email("{alex|koller}@lst.uni-sb.de")]
        ), (
            name: "Ellie Pavlick",
            affiliation: [Brown University\ #email("ellie_pavlick@brown.edu")]
        ),
    )
)
```

Abstract

Within each row, you can mix blocks with single authors and blocks with multiple authors. This is useful if some authors come from the same institution and some come from different ones.

297

298

299

300

301

Multiple rows of authors

Yuekun Yao Saarland University ykyao@coli.uni-saarland.de	Yupei Du Utrecht University y.du@uu.nl	Dawei Zhu Saarland University dzhu@lsv.uni-saarland.de
Michael Hahn* Saarland University mhahn@lst.uni-saarland.de	Alexander Koller* Saarland University koller@coli.uni-saarland.de	

```
#acl(
    title: [Multiple rows of authors],
    titlebox-height: 5.5cm,
    authors: make-authors(
        ( // row 1
        (
            name: [Yuekun Yao],
            affiliation: [Saarland University\ #email("ykyao@coli.uni-saarland.de")]
        ), (
            name: [Yupei Du],
            affiliation: [Utrecht University\ #email("y.du@uu.nl")],
        ), (
            name: [Dawei Zhu],
            affiliation: [Saarland University\ #email("dzhu@lsv.uni-saarland.de")],
        )
    ),
    ( // row 2
    (
        name: [Michael Hahn#title-footnote([Joint senior authors.], "fn:senior")],
        affiliation: [Saarland University\ #email("mhahn@lst.uni-saarland.de")],
    ), (
        name: [Alexander Koller#title-footnote(none, "fn:senior")],
        affiliation: [Saarland University\ #email("koller@coli.uni-saarland.de")]
    )
)
))
```

302

303

Abstract

304 If your authors don't fit into a single row, you can
305 pass an array of rows to `make-authors`. This will
306 set the rows on top of each other, separated by an
307 amount of whitespace that is determined by the
308 `row-spacing` argument.

309 Note the use of `title-footnote` to generate
310 the “joint senior authors” footnote, which is refer-
311 enced from two different authors. Note also the
312 use of `titlebox-height` to increase the height of
313 the titlebox.

*Joint senior authors.

Free-form authors

Alex Duchnowski^{†, 1}

[†] Saarland University

{alex|koller}@lst.uni-saarland.de

Ellie Pavlick¹

¹ Brown University

ellie_pavlick@brown.edu

```
#acl(
    title: [Free-form authors],
    authors: [
        *Alex Duchnowski*#affil-ref("uds", "brown") #h(2em)
        *Ellie Pavlick*#affil-ref("brown") #h(2em)
        *Alexander Koller*#affil-ref("uds")

        #affiliation("uds", symbol: sym.dagger) Saarland University #h(3em)
        #affiliation("brown") Brown University

        #email("{alex|koller}@lst.uni-saarland.de") #h(2em)
        #email("ellie_pavlick@brown.edu")
    ]
)
```

Abstract

You can ignore the `make-authors` function and simply arrange the authors in the titlebox yourself. TracI offers the functions `affiliation` and `affil-ref` to facilitate the mapping of authors to affiliations.